KTH Electrical Engineering

EP2120
Internetworking

Lab 1

ARP, IP, UDP

September 2021

Division of Network and Systems Engineering
School of Electrical Engineering and Computer Science
KTH Royal Institute of Technology
Stockholm, Sweden

Contents

1 Introduction

This syllabus contains the information about Lab 1. The topic of this lab is basic networking, ARP, IP and UDP. Please read the syllabus carefully.

1.1 Goals of the Lab

The objective of this lab is the following:

- Give you an overview of the most important network diagnosis tools, like traceroute, netstat, ping.
- Introduce you to `Wireshark`, a tool often used to capture network traffic.
- Make you familiar with the operation of the ARP table.
- Give you insight into the operation of UDP, and fragmentation for UDP
- Introduce you to common applications, like `telnet`, `ftp`, `smtp` and the system resolver.

1.2 Requirements and reporting

In order to pass the lab, you need to fulfill the following:

- Pass the preparation quiz on Canvas before the lab starts. The preparation quiz is a prerequisite for performing the lab.
- Perform the lab. You need to be present during the lab session.
- Show your solutions to the questions to one of the lab assistants.

2 Preparation for the lab

In order to prepare for the lab, read the items in the reading list and answer the questions below. The lab is limited in time, so it is necessary that preparations are made in advance. The schedule is tight.

2.1 Lab equipment and software instructions

For this lab, every group of two persons will use the following equipment:

- One laptop
- Two straight Ethernet cables

Ethernet cables will be provided in the lab, and you need to bring one laptop per group. You can either use your laptop as an end-host to perform the lab, or connect your laptop to one of the Raspberry Pis in the lab. We recommend that you use Raspberry Pis, which are well-configured and are easy to use. For detailed information about how to connect to the Raspberry Pis in the lab, please check the Raspberry Pi tutorial available in the Lab module on Canvas.

If you would like to use your own laptop as the end-host, please ensure that you have an Ethernet port (either built in, or via a USB adapter). You can then choose from 2 options:

1. If your computer is running Linux, you only need to ensure that your system supports all networking functionalities required to perform the lab. The check-list is given in Section 2.2. Please install all tools listed there.

2. If you are a Windows user (or use Mac OS X), you can set up a virtual machine running Linux from a bootable CD. For installing Linux with VirtualBox, please refer to the installation manual available in the Lab module on Canvas. (If you are already familiar with the installation procedure, you can just proceed to download and install the ISO image which we will use in the lab—it contains all the necessary tools to perform the lab. The image is available at `https://kth.box.com/s/yxzse7zcje8q13irs3w38uozwrorr9eq` (to access use the password lab).

Note also that we can not provide full troubleshooting support for Mac OS X users. As mentioned above, one host is sufficient for a group of two students. Thus, if you know who will be your lab partner, you can decide together and choose the optimal option for your group.

2.2 Software tools

Please make sure that the TCP congestion control algorithm running on your computer is Reno (instead of cubic). To check the congestion control algorithm running on your computer, please run the following command.

```
sudo sysctl net.ipv4.tcp_congestion_control
```

If your computer is not running Reno as the congestion control algorithm, you can try to switch to Reno by running the following command as the root user.

```
echo reno > /proc/sys/net/ipv4/tcp_congestion_control
```

The following software tools will be used in the lab. Manual pages of these tools are installed on all Linux computers (man command) and they can also be found on the web (for example `http://linux.die.net`). Read the manual pages of these commands. If you plan to use your own Linux computer during the lab, ensure that you have a working installation of all of the listed tools.

- `arp` - Manipulate the system ARP cache.

- `ftp` - Internet file transfer program.
- `ifconfig` - Configure a network interface.
- `netstat` - Print network connections, routing tables, interface statistics, etc.
- `ping` - Send ICMP ECHO_REQUEST to network hosts.
- `route` - Show and manipulate the IP routing table.
- `sendmail` - An electronic mail transport agent.
- `ssh` - Remote login program.
- `telnet` - User interface to the TELNET protocol, for interactive communication with another host.
- `traceroute` - Print the route packets trace to network host.
- `ttcp` - Test TCP and UDP performance.
- `wireshark` - Interactively dump and analyze network traffic. Read more about Wireshark in Section 7.

Some Linux distributions (e.g. Fedora) come with `ttcp` installed. If this is not the case with the distribution you use, you will need to install it yourself. You can do this by downloading the source file, e.g. from www.netcore.fi/pekkas/linux/ipv6/ttcp.c. and compiling it with

```
gcc ttcp.c -o ttcp
```

Then, copy the file to your */usr/bin* folder:

```
sudo cp ttcp /usr/bin
```

Note that certain distributions may have alternative versions of `ttcp` available, e.g. `nttcp` in Ubuntu. Please do NOT use these versions, as different versions may be incompatible with one another.

Wireshark is another tool that you need to install if you do not have it. Guidelines are given here:
https://www.wireshark.org/docs/wsug_html_chunked/ChapterBuildInstall.html .
In the online Wireshark manual, read carefully the sections 2.1, 2.2, and follow the instructions from sections relating to UNIX operating systems (from Section 2.5 onwards).

2.3 Reading list

Forouzan, "TCP/IP Protocol Suite", 3rd ed., Chapters 4, 5, 6, 7, 8, 9, 11, 12.
Forouzan, "TCP/IP Protocol Suite", 4th ed., Chapters 4, 5, 6, 7, 8, 9, 13, 14, 15.

- Fragmentation
- UDP, TCP
- EP2120/IK2218 Lecture Notes: Addressing, IP, Routing, ARP, UDP, TCP

2.4 Operating system instructions

Note that these instructions apply only for those using the provided ISO image, either with VirtualBox or, booting from the provided USB sticks.
The username and password that you will use to log in to the system are

username: *user*
password: *1234*

Several commands (e.g., `ifconfig` and `route`) used during the lab can only be executed with superuser (*root*) privileges. In order to be able to execute these commands as *root* you will have to precede them with the command `sudo`. As an example, to configure an interface using `ifconfig` you have to type

```
sudo ifconfig eth1 192.168.0.25 netmask 255.255.255.0 broadcast 192.168.0.255
```

The first time that you execute the `sudo` command you will be asked to provide your password. An alternative to using `sudo` is to start a shell as `root`, and to enter the commands in this shell. You can do this by typing:

```
sudo -s
```

Since during this lab you are primarily going to use system commands, we recommend you to start a shell as `root`.

3 Configuration

3.1 Setting up the environment

During this lab session you will work with what could be a corporate network of a company with several hundreds of users. The name of the fictitious company is Acme. It has four departments: administration, production, marketing as well as research and development. The network has four backbones, one per department. The backbones are connected to a single router as shown in Figure 1. Before you start the exercises, you must *manually* configure your host's network connectivity, i.e. statically configure the IP address and possibly also DNS information. You must select an IP address from one of the four subnets, connect your laptop to the corresponding subnet and configure your IP address, netmask and broadcast address.

- At the beginning of the lab session, each group will be given a label that marks their host in the network diagram above. Based on your label, configure your host with the parameters given in the diagram. For example, if your label is MAR1, your computer belongs to the Marketing subnet (`192.168.0.8/29`) and the computer's IP address is `192.168.0.10`.
- In the patch panel make sure that your machine is connected to the switch for the subnet that contains your IP address. The socket where you plug in the network cable on the table in front of you has a number. The corresponding number on the patch panel in the rack has to be connected to the correct switch.

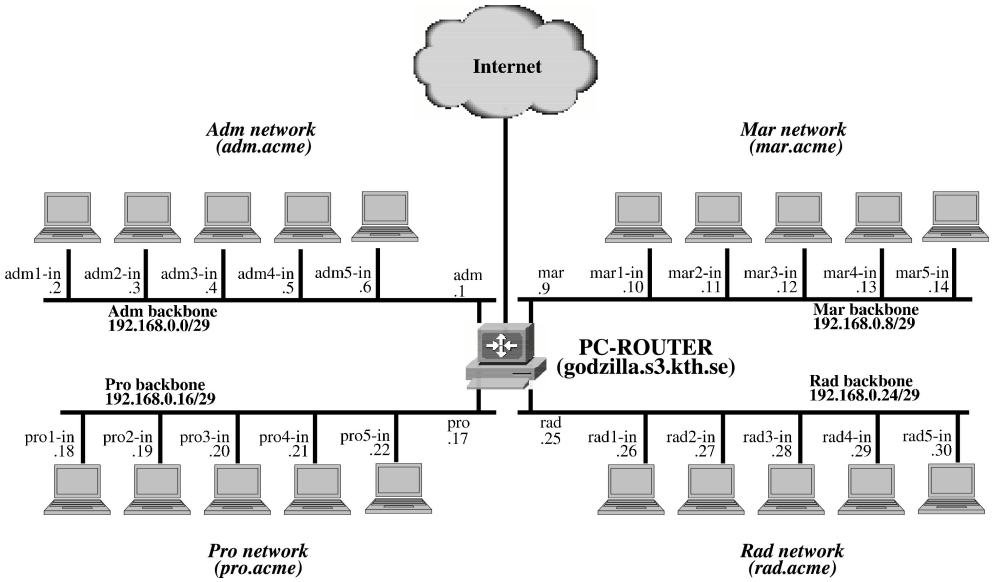

Figure 1: The network diagram.

- In the patch panel make sure that your machine is connected to the switch for the subnet that contains your IP address. The socket where you plug in the network cable on the table in front of you has a number. The corresponding number on the patch panel in the rack has to be connected to the correct switch.

- If you are performing the lab on your own Linux machine (i.e. not using VirtualBox or booting from one of the USB sticks provided during the lab) you will have to ensure that the systems network manager is disabled. In Ubuntu, this can be done with the command

  ```
  service network-manager stop
  ```

  At the end of the lab you may restart the network manager with the command

  ```
  service network-manager start
  ```

- The `ifconfig` command is used to configure network interfaces on your computer.

  You can find information about the arguments and options of the `ifconfig` command by typing `man ifconfig` in the terminal window:

  - Which interfaces are defined on your computer?
  - What IP addresses are assigned to them?
  - On the interfaces with IP addresses: What is the interfaces' broadcast address?
  - What other information can you see on the interfaces?

  Now use the `ifconfig` command to bring up the interface `eth1` and configure it with your IP parameters (address, netmask and broadcast address for the subnet).

  To bring the network interface up manually, the syntax of the command is:

```
ifconfig Interface_name up
```

The syntax of the command when you would like to assign IP parameters is

```
ifconfig <iface> <ipaddr> netmask <netmask> broadcast <bcast>
```

After configuring the interface try to ping the PC-router's interface that belongs to your subnet. Then, try to ping one of the other three interfaces of the PC-router or try to ping one of the KTH DNS servers with address 130.237.72.200.

- You will see that you are unable to reach addresses outside your subnet. If you cannot find the explanation for this, the next step will lead you to it.

- Add a default route to the gateway with the `route` command.

  After you have checked that your interface is configured properly, but you still do not have any response from ping or traceroute, it is a good time to check that the routing information in your PC is correct. You can check the content of the routing table by typing `route` in the terminal window. You will see the output of this command like the one below.

```
[lab@localhost lab]# route
Kernel IP routing table
Destination      Gateway         Genmask         Flags Metric Ref    Use Iface
10.0.213.0       *               255.255.255.0   U     0      0        0 eth1
127.0.0.0        *               255.0.0.0       U     0      0        0 lo
default          itguest-gw.gues 0.0.0.0         UG    0      0        0 eth1
```

  Or, if you execute this command with the option "-n":

```
[lab@localhost lab]# route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
10.0.213.0      *               255.255.255.0   U     0      0        0 eth1
127.0.0.0       *               255.0.0.0       U     0      0        0 lo
0.0.0.0         10.0.213.1      0.0.0.0         UG    0      0        0 eth1
```

  If your PC has one network card the routing table will consist of three records: the route to your network, the route to the 127.0.0.0 network, and the default route. When sending packets to an IP address that is inside of your own network, your PC will use the first record; for the packets whose destination is outside your network the PC will use the third record, and send them to the default gateway. Check the entry corresponding to the **default** route (the network address for default route is 0.0.0.0), it should point to the first router in your network. If you do not have this record or it does not point to the first router configure the routing table with the command

```
route add default gw router_addr
```

  Before you try this command, make sure that you can ping the router.

- If name resolution does not work you might have to edit the file `/etc/resolv.conf` to configure the correct nameserver (DNS). Set the nameservers to `130.237.72.200` and `130.237.72.201`.

  Note: If you cannot edit the /etc/resolv.conf file manually using an editor, try typing instead:

  ```
  echo "nameserver 130.237.72.200" | sudo tee - a /etc/resolv.conf
  ```

Your host is now configured and is able to reach any other host on the Internet.

4 Netstat

Netstat is a command that shows you statistics of the TCP/IP stack, such as which ports are open in your system as well as the processes associated with the ports used.

- What TCP connections are established on your system?
- Is your system listening on any TCP ports? Which ports and what are the related processes?
- Open a second terminal window and start an FTP connection to ftp.sunet.se by typing

  ```
  ftp ftp.sunet.se
  ```

  Which TCP connections are established now? Close the connection and exit ftp with the command `bye`.
- What flags should you use so that `netstat` avoids resolving IP addresses to host names?
- Take a look at the routing table. Are routing tables really necessary on a host?

5 Ping

Ping is used for network maintenance and allows to check whether the network is well configured and there is connectivity among the hosts.

- Ping your neighbour's computer. Study the output of "ping" and be ready to describe what everything means.
- Ping `www.google.com`. How is the output different from the previous ping?
- Ping `130.237.32.51`. How is the output different from the previous pings?
- Ping `ftp.sunet.se` with TTL set to 1. What happens? Increment the TTL and retry the command until you stop getting errors. Describe what happened.

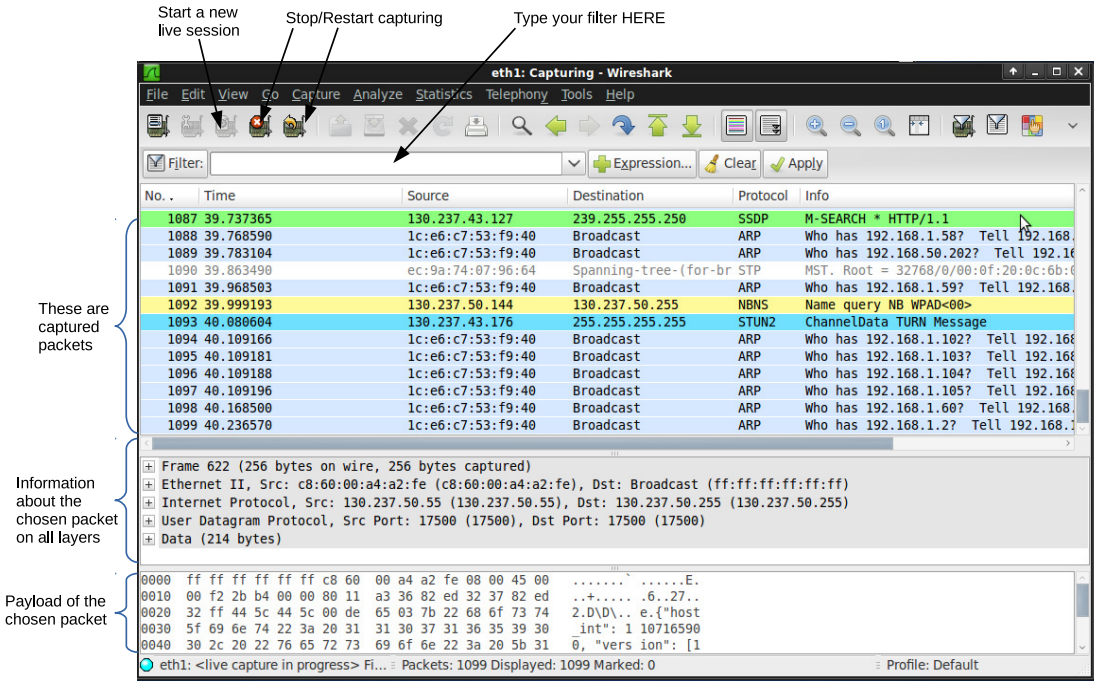

Figure 2: Main `Wireshark` window containing three frames.

6 Traceroute

The command `traceroute` is used to discover the IP level path of packets travelling between two hosts.

- How many router hops are there between you and the destinations below? Also: What do the hostnames (in the `traceroute` output) mean?
  - `rtfm.mit.edu`
  - `acs.ucalgary.ca`
  - `www.yahoo.se`
  - `210.155.130.190`
- What does `traceroute` do? What kind of conclusions can you draw from the output of `traceroute`? What do the times mean?
- How can you get `traceroute` to send max two packets with TTL=10 and aim at `www.nada.kth.se`, port 80?

7 Wireshark

`Wireshark` is a packet sniffer that enables you to inspect the packets on the network. In order to start `Wireshark` do the following:

- wireshark & (make sure you do this as root user - use `sudo` or run it in a shell as `root`)
- Start a capture and set it to update list in real-time.

The wireshark window consists of three frames. A list of the captured packets is shown in the top frame. When a packet in the list is marked, the content is shown in the middle and lower frames. The middle frame shows the packet in symbolic form, while the lower frame shows the packet in raw, hexadecimal format. In the middle frame, the fields may be expanded to show a more detailed view.

A session is started by clicking on Capture»Start. This brings up a capture options window. Typically, the display option *Update list of packets in real time* and *Automatic Scrolling in live capture* should be selected. When started, the packets are shown in the main area, and a capture window is shown. To stop the capture, click on the *stop* button in this window.

Some statistics can be shown by choosing Tools»Summary.

The captured data can be saved as follows:

- Stop capturing the packets.
- Click the *File»Save As* on the menu bar.
- Choose the *Wireshark/tcpdump/...-pcap* format.
- Name the file.
- Click the *Save* button.

Exercise:

- Ping some hosts and study the output in `Wireshark`. Investigate all the fields of a ping packet. What protocol headers can you identify?

8 Arp

`Arp` is a network protocol used for the discovery of a host's physical address (MAC), when its IP address is known.

Take a look at your `arp` table.

- How many tuples does your `arp` table contain?
- How is the `arp` table built up?
- Why is an `arp` table necessary on your computer?
- Investigate the meaning of the flags in the `arp` table.

9 UDP Measurements and `ttcp`

In this lab exercise you will send data via UDP between two hosts and you will observe the traffic. You will use `ttcp` to send the data. Please consult the man page of `ttcp` (`man ttcp`) for the command line parameters of `ttcp` if you have not done so yet.

1. On your host, start `Wireshark` to capture traffic on interface eth1, add a filter so it will only capture the UDP traffic from and to host "TA-host".

2. Click Capture»Start on the menu bar to start capturing the traffic.

3. On your host: start a `ttcp` receiver to receive UDP traffic at port number 1234.

4. Login to host "TA-Host" using ssh. Start a `ttcp` sender to send 10 UDP packets with the length of 1000 bytes to your host at port number 1234.

5. When the transfer is completed, stop capturing the packets by pressing `Stop` in the "Wireshark Capture" window. You should now have a packet trace in the main window. Make sure that you recognize the Ethernet header fields, the IP header fields, and the UDP header fields in the mid section.

6. Observe the traffic and answer the following questions:

   (a) How many packets were transferred in the two directions?
   Observe that `ttcp` always transfers six extra UDP packets when testing UDP transfer. One of the six packets is at the beginning of the transmission and the other five packets are at the end of the transmission. You should always exclude these packets when doing measurements.

   (b) How many bytes were transmitted in the recorded transfer in total including the Ethernet, IP, and UDP headers as well as the application data? (Do not include the six extra UDP packets in the calculation, which constitute 360 bytes.)
   Hint: Check the statistics summary by choosing Tools»Summary.

   (c) How many bytes of user data were transmitted?
   (Calculate it or find it out by looking at the statistics from `ttcp`.)
   Make not of this calculation, since you will need it in the second lab.

   (d) Inspect the UDP header fields. Which fields do not change in the different packets? (Note that you should disregard the 6 extra packets).

   (e) What is the UDP port number used in host "TA-Host"?

10 UDP and fragmentation

In this exercise you will observe the effects of fragmentation in UDP. Fragmentation occurs in the network layer, when the transport layer sends so much data to the IP layer that the data together with the network layer header exceeds the Maximum Transmission Unit (MTU) of the underlying link.

In the following you will investigate how IP fragmentation works for UDP traffic. You will use `ttcp` to send data.

1. Start capturing packets with `Wireshark` on your host. For this exercise it is convenient to set Display options "Update list of packets in real time" and "Automatic scrolling in live capture".

2. Use ssh to login to Host "TA-Host" and use `ttcp` to send packets to your host. Increase the size of the UDP datagrams (the -l option) until fragmentation occurs. Note that when `ttcp` runs UDP, it is not necessary to start a receiving `ttcp`. You can work solely on the sending `ttcp` in this exercise and increase the packet size gradually. A hint is to start with sending datagrams of size 1470 bytes and to work upwards.

   (a) What is the largest UDP data length you can send without fragmentation?

   (b) Can you explain this length (given an MTU of 1500)?

   (c) What is the largest UDP data length the system can send, regardless of fragmentation?
   Hint: the largest possible IP packet is $2^{16} - 1$ bytes long. `ttcp` will return "Message too long" for over-sized packets.

   (d) What is the explanation for this length?

Please ask one of the lab assistants to check your progress before you continue the lab.

11 Telnet

`Telnet` is a network protocol used for bi-directional text-based communication between two hosts over a network.

- Use telnet to connect to the HTTP daemon (httpd) at `www.nada.kth.se`. To download the main web page (index.html), type: `GET ./`
- Why is TCP and not UDP used for HTTP traffic?
- What are the advantages/disadvantages?
- Start a TCP session to the "daytime" service on localhost.
  - First you have to make sure that the daytime service is actually running (for example with netstat).
  - If it is not running you have to start it. The configuration file for the daytime server is `/etc/xinetd.d/daytime`
  - Make sure that the field *disable* is set to *no*.
  - Restart the xinet service (which runs the daytime server and others).
  - `sudo invoke-rc.d xinetd reload`.

12 Anonymous ftp

`ftp` is a network protocol used to transfer a file from one computer to another over a network.

- How do you log onto an ftp server anonymously?
- What are the main differences between normal ftp and anonymous ftp?
- Why are both necessary?
- Start capturing packets with Wireshark and add a filter to display only FTP or TCP traffic.
- Get the file `rfc0959.txt` from `ftp://ftp.ietf.org/rfc`. Download this file using the command line interface, and not netscape etc.
- Study the captured list of packets. What source and destination ports were used to transfer the data? Based on the previous, can you tell which ftp mode was used?
- What do the following commands do in ftp?
  - user
  - image
  - ascii
  - get
  - put
  - mget/mput
  - cd
  - lcd
  - ls
  - prompt
  - hash
- Take a quick look at the file rfc0959.txt. What does it describe? What do the following commands do in the FTP protocol?
  - USER
  - CWD
  - TYPE
  - LIST

13 Mail

Check if the smtp server on your machine is running. This can be done with the command `service exim4 status` or alternatively `ps ax|grep exim4`. If it is not running you can start it with `sudo /etc/init.d/exim4 start`.

- Use the command `sendmail -v email@address.com` to send an email to your regular email account. The "-v" flag makes the tcp session verbose, so you can see what protocol messages were used. When using `sendmail`, you have to manually specify the headers of the e-mail message as well as the body. An example of execution of `sendmail` is

```
user@live:~$ sendmail user@domain.com
From:  John Doe <johnd@somedomain.com>
Reply-to:  johnd@somedomain.com
Subject:  Hello
This is a test message!

.
```

  After typing in the headers and the body of the mail message press `ctrl-d` or "." to send.

  - Which lines are sent by your computer to the mail server?
  - Which lines are returned by the remote host?
  - How did you see if the email was successfully sent?

- Find out the domain name of your smtp server using the command: `dig domain.name mx`, where *domain.name* is the domain name part of your favourite email address. If this doesn't give you an domain name, try `dig domain.name a` to get an IP address instead.

14 System resolver

Purpose: try to understand why the system resolver is needed and how it works. Start `Wireshark` and try the following:

- Ping www.sunet.se
- Ping 130.239.8.25

Explain the difference in output.

`/etc/resolv.conf` is the file that configures the system resolver. Read the manual page for `resolv.conf` and change it (You will need root access to change it).
When you name a host computer on the network can you then use IP-address and/or host-name?
What is necessary for the system-resolver to work?
How dependent is a computer-user on a working resolver?