Universidad Carlos III de Madrid	**Lab exercise: Practical analysis of ARP operation - S12**
Communication Networks and Services	2023-2024

1.- Introduction and Objectives

The objective of this lab session is familiarizing students with the IP configuration of end systems and the operation of the ARP protocol. Students will work on the configuration of a simple scenario with end systems (PCs under Linux operating system) in the lab network.

The operative system available in each PC in the lab, has implemented the network protocol stack usually known as TCP/IP stack. This network protocol stack enables the network communications. You can find information about the implementation of this stack using the Linux IP configuration - Quick Guide in the command-line interface (eg. 'man 7 ip').

2.- Lab Exercise Rules

Read carefully the lab exercise instructions before starting with the exercise

It is recommended that the practice be done in pairs (although it is possible to do it individually). In any case, each student will deliver his/her own document with the answers to the questions posed in the practice statement, and must perform the practice from his/her own computer (the screenshots requested must correspond to his/her own computer; the text of the answers must be his/her own, even if what is explained has been understood by talking to the practice pair). The realization in pairs implies that the students of the pair can be connected online during the realization of the practice, progress together to see that they are on the right track, consult any doubt, and/or ask each other for help. It is possible, therefore, any work mechanism that allows each member of the pair to rely on the other to carry out the practice (but not that one student does the practice for the other).

The group report submitted must clearly identify the name of the student and the name of his/her group partner. Therefore, it is not expected that the reports of the two students in a pair reflect independent work, but the performance of the group cannot be done in groups of more than two students, and it is not valid that the reports of students who are not partners reflect work that was not performed independently.

In order to carry out the practice it is necessary to have a user account in the teaching laboratories of the Department of Telematics Engineering. The practice will be carried out by remote access to a computer of the Department of Telematics Engineering through https://aulavirtual.lab.it.uc3m.es (the guide [1] explains in detail how to do this). The Linux operating system is used. For some of the commands it is necessary to have superuser permissions (administrator, root).

For such commands (address configuration, routes, etc), you must prepend the sudo command (ask for the user's password). If you have problems, please ask your instructor.

☞ *The parts of the statement between these symbols raise questions that are proposed throughout the session. Each student must answer these questions in writing and deliver them in a document (practice report) uploaded through the global classroom - Aula Global.* ☜

In some parts of the practice, the traffic capture and analysis software `Wireshark` *application* will be used.

Estimated lab time: 100 minutes

3.- Work to do prior to the beginning of the laboratory session

Review in the course book the operation of the ARP address resolution protocol (section 5.4.2, including sending a datagram to a node outside the subnet) and in general the knowledge from class sessions 7-8.

Review the operation of the Wireshark program. To use the program, you can refer to the system manual pages or the online documentation (Wireshark Documentation), available at http://www.wireshark.org/docs/. It is also recommended that you carefully read the Wireshark help guide provided with the practice material for session 6 before doing the practice.

Prepare the practice by reading the statement carefully. Think about the results that are to be expected from each section of the practical. Review the theory where necessary. Study the manuals and command help pages, and have identified all the commands necessary to perform the practice. This will allow you to understand and make much better use of the practice development.

4.- Description of the practice

Connect to a lab computer indicated for the practical. All practice questions refer to the lab computer, regardless of whether you are communicating with it remotely through another computer. The operating system will be Linux, which has a wealth of network configuration and monitoring tools. The tools that we will use for this practice are:

- hostname. Returns the name of the local machine (PC).
- ifconfig. Allows to configure (and display the configuration) network interfaces (basically addresses). In MS Windows systems the equivalent command is ipconfig.
- route. Displays (and modifies) the local forwarding table (that of the machine itself).

- arp. Displays (and modifies) the cache table that associates IPv4 addresses with known MAC addresses. With the -n parameter it displays addresses instead of domain names.

- ip. The above three commands are typical in any Unix operating system (and also in some others).

However, there is now another command, **the ip command**, which combines the functions of all of them (plus a few others). The ip command is more versatile and the only one guaranteed to provide complete information in modern Linux kernels. It is recommended that you become familiar with both options but get used to using the ip command.

- ping. Verify IP connectivity between two networked machines.

For the application of these tools you can get help, through the system manual pages (man <command>) or the online help of the tools themselves, especially the ip command (ip help), which is very complete and simple.

During practice you are asked to perform actions that will require you to choose the appropriate command and use it. Sometimes there are several commands that can be used (e.g. ifconfig and ip). Try to test as many options as possible so that you can work with as many tools as possible (note that not in all real environments you will find all these tools available).

Discover your PC configuration in the laboratory:

☞ a. What is the name of the computer on which you are doing the lab exercise? How many network interfaces does it have and what type? What is the link level address (MAC address, Ethernet, or HW address) per interface? ☜

☞ b. What are the IPv4 addresses of the computer (check them using the appropriate commands on the computer)? Include a screenshot showing information about the computer's interfaces (the result of running the appropriate command in a terminal). ☜

☞ c. For the lab IPv4 network (interface eth0) what is the network prefix? what is the network address and what is the broadcast address? how many PCs can be assigned an IPv4 address with the range of our (sub)network? (to answer these questions you will have to rely on information from the lab computer obtained by running the appropriate commands, and you will have to use your knowledge to deduce more information from the above). ☜

2. **Check the operation of the ARP protocol:**

   a. Let's call your computer my_PC. For the test you need to communicate with another computer that we will call another_PC. This other_PC will have an IPv4 address that we call dirIP_other_PC. This address will be used as the destination during the test. The name of another_PC should be calculated as follows: given the name of your computer found in point 1a (example, my_PC=it001), increment the numeric value by one (in the example, other_PC=it002)

   b. Ping another_PC to check connectivity with it and find out its IP address. If you do not have connectivity with another_PC, increment the numeric value of another_PC again (as in 2.a, in the example another_PC would become it003), and so on until you have connectivity with another_PC. If you increase the value 5 times and you do not get connectivity, or the answer you get is from a different name than the one you pinged (e.g. you ping doc035 and get an answer from jbit150), ask the teachers of the subject for the equipment to use as another_PC. .

   c. Taking into account the subnet addressing of your computer's eth0 interface that you found out in section 1.b, determine whether another_PC is on the same subnet as your computer's eth0 or on a different one. Indicate in your answer your reasoning and the address and name of another_PC.

   d. View the address resolution table (IPv4 addresses <-> MAC addresses) that your PC has in cache (ARP table). does dirIP_otherPC appear in that table? why do you think it is there? Include a screenshot of the ARP cache table....

   e. Enter the command *`ip neigh del <dirIP_otroPC> dev eth0`.* Check the manual pages to see what this command does and verify that the command has done what you expected it to do..

   f. Make sure that the entry for dirIP_otherPC does not appear in the ARP cache. Run from your computer `ping -c 1 dir_otroPC`. Check the ARP cache again. Explain what happened.

   g. Run it again, from your computer `ping -c 1 dir_otroPC`. Do you notice any difference with the previous case (paragraph f.)? Please explain. Hint: note the "time" values displayed on the screen by the executions of both pings. Include screenshots of the results of both pings.

Analysis of generated ARP traffic (destination is in the same subnet):

Now we are going to use Wireshark to analyze the traffic generated on the network when a ping command is executed. The first traffic that will be generated, if the ARP cache of the computer that wants to send the traffic does not contain the MAC of the destination in the subnet, is ARP traffic, which is the one we are going to analyze first.

h. Delete again from the ARP cache of your PC the entry corresponding to dirIP_other_PC (so that ARP traffic is generated when we send traffic to that destination). Now ping the other_other_PC capturing the exchanged traffic with wireshark. Using the appropriate display filter in wireshark , obtain the ARP frames exchanged as a result of the ping, include a screenshot showing the ARP request in wireshark with the decoding of the Ethernet headers and the ARP message itself extended (showing the value of the fields in those headers), and complete the following tables:

ARP Request:

Ethernet Header	**Specific address (eg: 12:34:32:A2:C1:22 or 138.2.17.4)**	Identify which computer it belongs to (Options: *dirMAC_miPC, dirMAC_otroPC, dirMAC_broadcast, dirMAC_router)*
MAC address source:		
MAC address destination:		
ARP fields	**Specific address (eg: 12:34:32:A2:C1:22 or 138.2.17.4)**	Identify which computer it belongs to (Options: *dirMAC_miPC, dirMAC_otroPC, dirMAC_broadcast, dirMAC_router, dirIP_miPC, dirIP_otroPC, dirIP_router, campo_vacío)*
MAC address source:		
Sender IP address:		
Target MAC address:		
Target IP address:		

ARP Reply:

Ethernet Header	**Specific address (eg: 12:34:32:A2:C1:22 or 138.2.17.4)**	Identify which computer it belongs to (Options: *dirMAC_miPC, dirMAC_otroPC, dirMAC_broadcast, dirMAC_router)*
MAC address source:		
MAC address destination:		
ARP fields	**Specific address (eg: 12:34:32:A2:C1:22 or 138.2.17.4)**	Identify which computer it belongs to (Options: *dirMAC_miPC, dirMAC_otroPC, dirMAC_broadcast, dirMAC_router, dirIP_miPC, dirIP_otroPC, dirIP_router, campo_vacío)*
Sender MAC address:		
Sender IP address:		
Target MAC Address:		
Target IP Address:		

3. IP traffic analysis (destination is in the same subnet)

In the previous section we analyzed the ARP traffic generated by the execution of the ping command in section 3a. This ping also generated IP packets containing ICMP messages (those used to perform the ping function). In this section we will analyze this traffic.

a. Set a suitable filter in *wireshark* to display the packets we want to analyze.

b. What type/s are the ICMP messages contained in the data field of the exchanged IP datagrams?

c. Complete the following table with the information of the first IP datagram sent as a result of the ping: Also include a screenshot showing in *wiresharkLab* the source and destination addresses of the MAC header, and the source and destination addresses of the IP header of this packet.

MAC Header	**Specific address (eg: 12:34:32:A2:C1:22 or 138.2.17.4)**	Identify which computer it belongs to (Options: *dirMAC_miPC, dirMAC_otroPC, dirMAC_broadcast, dirMAC_router)*
MAC address source:		
MAC address destination:		
IP Header	**Protocol, address, TTL**	Identify which computer it belongs to (Options: *dirMAC_miPC, dirMAC_otroPC, dirMAC_broadcast, dirMAC_router, dirIP_miPC, dirIP_otroPC, dirIP_router, campo_vacío)*
IP address source:		
IP address destination:		
Protocol field:		
TTL:		

4. IP traffic analysis (destination is on another subnet)

In this section we are going to analyze the traffic when the destination is outside the subnet. In this section use contrabajo.it.uc3m.es as other_PC.

a. Taking into account the addressing used in the subnet to which your computer's eth0 is connected (which you found out in section 1.b) check and reason that now my_PC is in a different subnet than another_PC.

b. Study the IPv4 forwarding table of your PC, include a screenshot of the output of the command you used to find out the forwarding table, and rewrite it in the format used in the theory classes:

PREFIX	**OUTPUT INTERFACE**	**NEXT HOP**

c. Determine the default router used by your PC. What IP address (dirIP_router) and what MAC address (dirMAC_router) does the router interface connected to the subnet where your PC is located have? Look carefully at the MAC address of the router, as it will appear later in practice and must be recognized.

d. Capturing traffic, ping dir_otherPC and fill in the following table with the information from the first IP datagram sent by your PC as a result of the ping (use the appropriate filter in wireshark to see the traffic you are interested in). Also include a screenshot showing in wireshark the source and destination addresses of the MAC header and the source and destination addresses of the IP header of this packet.

MAC Header	**Specific address (eg: 12:34:32:A2:C1:22 or 138.2.17.4)**	Identify which computer it belongs to (Options: dirMAC_myPC, dirMAC_anotherPC, dirMAC_broadcast, dirMAC_router, dirMAC_router, etc.).*)*
MAC address source:		
MAC address destination:		
IP Header	**Protocol address, TTL**	Identify which computer it belongs to (Options: dirMAC_myPC, dirMAC_anotherPC, dirMAC_broadcast, dirMAC_router, dirMAC_router, etc.).*, empty filed)*
IP address source:		
IP address destination:		
Protocol field:		
TTL:		

e. Based on what we have seen in sections 4.c and 5.d (looking at the MAC addresses), and assuming that the ARP cache of your PC is empty:

- When your PC has to send IPv4 traffic to another_PC on another subnet, what MAC address will your PC need to find out via ARP to send the traffic? what IP address will it use to ask for it?
- And if the traffic is to another_PC on your subnet?

f. According to the previous section, the PC does something different when it is going to send traffic to a destination in its own subnet than when it is to a destination in another subnet. Explain, based on the answer to section 5b, how the IP layer decides to treat the IP datagram destined to another_PC used in section 4 and how it does it for the other_PC in section 5.

Bibliography

[1] Virtual Classroom (Aula Virtual) Connection Guide. Available in Aula Global, Networks and Communication Services, Session 6.-

Supplementary bibliography

- Manual pages of GNU/Linux
- Linux Advanced Routing & Traffic Control HOWTO, http://www.lartc.org/howto [Accessed Jan 2021].
- Linux Networking HOWTO, http://www.tldp.org/HOWTO/Net-HOWTO/ [Accessed Jan 2021].