Lab 2 - Session 1

Daniel Perdices

May 13, 2024

Universidad Autónoma
de Madrid

What are we going to do today?

Test P1

Intro to Lab 2

Self-assessment

Material

Test P1

How is the test?

Multiple-choice questions in moodle.

- Password will be provided. Wait for it.
- No materials can be used unless it is indicated
- You must obtain at least 50% correct answers
  - Incorrects answers do not substract
- **If you do not obtain 50%, your grade for Lab 1 is 0.**
- We start at 11:10 (or later). You cannot leave the room once it has started.

Test time

Figure 1: *Ave, Caesar, morituri te salutant*

Intro to Lab 2

What is Lab 2 about?

In Lab 2, we are going to implement an ARP resolver with cache and ARP reply.

- Implement Ethernet layer
    - Format
- Implement ARP layer
    - Format
    - Order

How to start?

0) Watch the ARP video (or pay attention to my explaination). Read the assignment.

1) Download the material for moodle

2) Read functions of ethernet.py

2.1) Which functions are already implemented?

2.2) Which functions should be implemented by you?

3) Read arp.py

3.1) How ethernet.py and arp.py interact?

3.2) Which functions are already implemented?

3.3) Which functions should be implemented by you?

ARP request

Ethernet:

- dstMAC: ?
- srcMAC: ?
- ethertype: ?

ARP:

- hwtype: ? ptype: ?
- hwlen: ? plen: ?
- opcode: ?
- s_hwaddr: ? s_paddr: ?
- t_hwaddr: ? t_paddr: ?

ARP response

Ethernet:

- dstMAC: ?
- srcMAC: ?
- ethertype: ?

ARP:

- hwtype: ? ptype: ?
- hwlen: ? plen: ?
- opcode: ?
- s_hwaddr: ? s_paddr: ?
- t_hwaddr: ? t_paddr: ?

ARP grat. request

Ethernet:

- dstMAC: ?
- srcMAC: ?
- ethertype: ?

ARP:

- hwtype: ? ptype: ?
- hwlen: ? plen: ?
- opcode: ?
- s_hwaddr: ? s_paddr: ?
- t_hwaddr: ? t_paddr: ?

Self-assessment

What you should have done by today

- Read Assignment 2
  - What are you being asked for?
  - What is the material provided?
  - How are you going to be evaluated?
  - How do you know that what you are doing is right?
- Run program `practica2.py`
- First draft of ethernet.py

What you should have done by next week

- An overview of arp.py
- A revised draft of ethernet.py
- Questions about the ARP implementation.

Material

Repited code

Functions/classes are useful to avoid code repetition. If you do sth more than once, consider creating a function/class.

- Why? Because when you have a bug, you only have to fix it in one part of the code.
- Example

```
def createARPPacket(hwtype=0x0001, ptype=0x0800,
                    hwsize=6, psize=4, #TODO
                    ):
    return bytes()
```

Repited code

Another example

```
class ARPPacket():
    HWTYPE = 0x0001
    # TODO
    def __init__(self, hwtype=HWTYPE, #TODO
                ):
        self.hwtype = hwtype

    def to_bytes(self):
        return bytes()

    def from_bytes(data):
        return ARPPacket()
```

Where are these slides?

`github.com/dperdices/redes1-1391-2022`

- Source code of this slides (in markdown)
- Slides in PDF
- Other resources

If you want a completed version or find any mistakes,

- Fork the repo
- Complete it / Fix it yourself
- Make a PR
- Wait for my approval (or comments)