Bachelor's Degree in Techniques for Software Development

Networks and Internet Applications

Practical exercise 1: Watch the network and you will find out what is going on there

In this practice, we will analyze some protocols that are usually present in everyday communications we make using computer networks. For this reason, it will be very useful to install a packet analyzer, which, visually, will allow us to understand what is happening at a glance. Analyzers are programs that capture packets that send and receive our computer's network card. In section **Annex: Wireshark preliminary guidelines** you have instructions for installing and testing how the packet analyzer that we will use in the subject: *Wireshark*, works.

Once you have installed the tool required to analyze what happens in the network, we need a common study scenario for us: Internet browsing. In order to solve the following exercises, you should run Wireshark and start capturing packets. Once you have done this, you have to open a browser and go to the following web site:

http://www.edu4java.com/es/web/web1.html

At this point, you can stop capturing Wireshark packets and save the capture into a file to follow with the work afterwards. To use it, you just have to open Wireshark and open the capture file stored in disk.

Qualification

This practice has four parts that should be done in sequential order (first part 1, then part 2, …). The final qualification depends on the delivered parts:

Part 1. Link and network layers	*Maximum qualification C-*
Part 2. Transport layer	*Maximum qualification C+*
Part 3. Application layer	*Maximum qualification B*
Part 4. Network security	*Maximum qualification A*

In order to obtain the aforementioned qualification, you have to do all exercises of the indicated part. **If there is any exercise or section not done or incomplete, it will mean not obtaining the corresponding qualification.**

***Note**: Upload your responses in the subject's classroom, both the **pdf file with the responses to the questions of all parts and the file captured by Wireshark**. Please do not use any compression program to submit in a single file. Name the files with your full name.*

First part (maximum qualification: C-): Link and network layers

Internet protocols stack is structured in several layers. In this first part, you have to work with the lowest levels and related to the network cards: Link and network layers. We work with MAC and IP addresses, identifying the network card and the device, respectively.

As support material to make this part of the practice, we propose you the following sections from the Computer Networking book (8th edition), especially sections concerning protocol headers:

- **6.4** Switched Local Area Networks pp. 516-517
  - **6.4.2** Ethernet
- **4.3** The Internet Protocol (IP) pp. 361-363
  - **4.3.1** IPv4 Datagram Format

1. Click over a specific frame from the capture you have done with Wireshark. Check that it includes the **Ethernet** header with data from the link layer. Show via a screenshot the Ethernet header contents. Respond to the following questions:
   a) What are source and destination addresses? Who do you think they belong to?
   b) Who assigns these addresses?
   c) What does type field mean?
   d) Which function has the CRC field? Respond in a theoretical way.
   e) What is the purpose of the preamble in an Ethernet frame?
   f) What data does this Ethernet frame carry?

2. In the previous frame, notice that the detail of an **IP** header is also included. Show via a screenshot the IP header contents. Respond to the following questions:
   a) What are source and destination addresses? Who do you think they belong to?
   b) Why the packet has an identifier?
   c) Which flags are active in this packet?
   d) Explain what the TTL value means in the analyzed packet.
   e) Why is a checksum field needed again in the IP protocol?
   f) Go to menu *Statistics* > *IPv4 statistics* > *IP protocol types*. Which is the protocol sending more packets? Why?

3. Finally, without applying any filters, go to the statistics menu *Statistics* > *Protocol Hierarchy*. Show in a screenshot the results and comment them, relating them with packet encapsulation and de-encapsulation, a fundamental pillar of network communications.

Second part (maximum qualification: C+): Transport layer

In this second part of the practice, we will deepen on the next layer of the Internet stack: Transport layer. There are two protocols offering different services at this layer: UDP (*User Datagram Protocol*) and TCP (*Transport Control Protocol*). You can work with the same packet capture you already used in the previous section.

As support material for doing this second part of the practice, we propose you the following sections of the Computer Networking book (8th edition), especially sections concerning protocol headers:

- **3.3** Connectionless Transport: UDP p. 228
  - **3.3.1** UDP Segment Structure
- **3.5** Connection-Oriented Transport: TCP pp. 260-265
  - **3.5.2** TCP Segment Structure

1. First of all, you have to analyze the UDP protocol. You can filter this protocol in Wireshark and you will only see packets from this protocol. Show via a screenshot the header fields and respond to the following questions:
   a) Which protocol/s from the application layer has/have generated these packets?
   b) Why do you think UDP is used instead of TCP?
   c) Why do values have source and destination ports? What does it mean?
   d) Why is the checksum field required?
   e) How is it calculated?
   f) What is the value of the length field? Check with hexadecimal values of the UDP packet that this is the value indeed.

2. Next, we have to check how the TCP protocol works. Go to the statistics menu *Statistics* > *Flow Graph*, and select only TCP type, checking the option *Flow type* > *TCP flow*. Show in a screenshot the phases where the connection is established and finished. Explain the process. What is the flag PSH used for?

3. Go to the main packets list. You can filter using TCP protocol to see only packets concerning this protocol. Select a packet and show in a screenshot the TCP header fields content and respond to the following questions:
   a) Which is the sequence number? What is it used for?
   b) And the ACK number?
   c) Which values have source and destination ports? What does it mean?
   d) Which flags are active in this packet?
   e) Explain what the values relative to the window mean.
   f) Which data does this TCP packet carry?

Third part (maximum qualification: B): Application layer

Following the Internet stack structure, the last layer is the one interacting with applications executed by the user, directly or indirectly: Application Layer. At this layer we can find several protocols, but in this part, we will focus on two of them: DNS (Domain Name Server*)* and HTTP (HyperText Transfer Protocol).

As support material for doing this third part of the practice, we propose you the following sections of the Computer Networking book (8th edition):

- **2.4** DNS—The Internet's Directory Service pp. 161-164
  - **2.4.3** DNS Records and Messages
- **2.2** The Web and HTTP pp. 131-138
  - **2.2.3** HTTP Message Format
  - **2.2.4** User-Server Interaction: Cookies

1. First of all, you have to analyze a **DNS** header from a request and a response, so you can filter in Wireshark by DNS protocol and you will see only packets from this protocol. Show a screenshot with the header fields and respond to the following questions:
   a) Which source and destination ports are used?
   b) What is the transaction identifier for?
   c) What do flags mean?
   d) In the request, which domain is requested?
   e) Which kind and class this domain is? What does it mean?
   f) Can there be multiple records (RR, resource records) in the response?

2. Now analyze an **HTTP** header from a GET request. Show a screenshot with the header fields and respond to the following questions:
   a) Which source and destination ports are used?
   b) Which version of HTTP protocol is?
   c) Are persistent connections used?
   d) Which language is used? Why is it provided?
   e) Which kind of content could be processed?
   f) Which coding formats are used?

3. Analyze an HTTP header of a successful GET response. Show a screenshot with the header fields and respond to the following questions:
   a) What does this response contain?
   b) Which kind of content contains?
   c) Is the content fragmented? How do we know?
   d) Which fields are related to dates? What do they mean?
   e) Are cookies used? How do they work?
   f) Describe what the fields related to caches mean.

4. Go to the Wireshark statistics menu *Statistics* > *HTTP* > *Packet counter*. Show a screenshot where the unsuccessful responses received are seen (they have value *Count* > *0*). Describe what each error code means and how many packets of each packet have been received. Afterwards, select an error code and look for it on the main packet capture screen. Show a screenshot with the header fields with a brief explanation.

Fourth part (maximum qualification: A): Network security

To do the last part of this practice, we have to focus on the interaction that happens when we write our username and password to access an online page where you are registered. You cannot use the UOC campus access page.

1. Show a screenshot of the URL and the webpage where you write your username and password.

2. Run Wireshark and start capturing packets. Access the webpage with your credentials. Which application layer protocol is used to offer security to this online page? Describe its main features.

3. Focus on a specific packet from this protocol. Show a screenshot with the header fields and explain what you observe.

4. Over which transport layer protocol travels the security protocol used to access the online page?

5. Finally, while Wireshark is capturing packets, connect to the web site

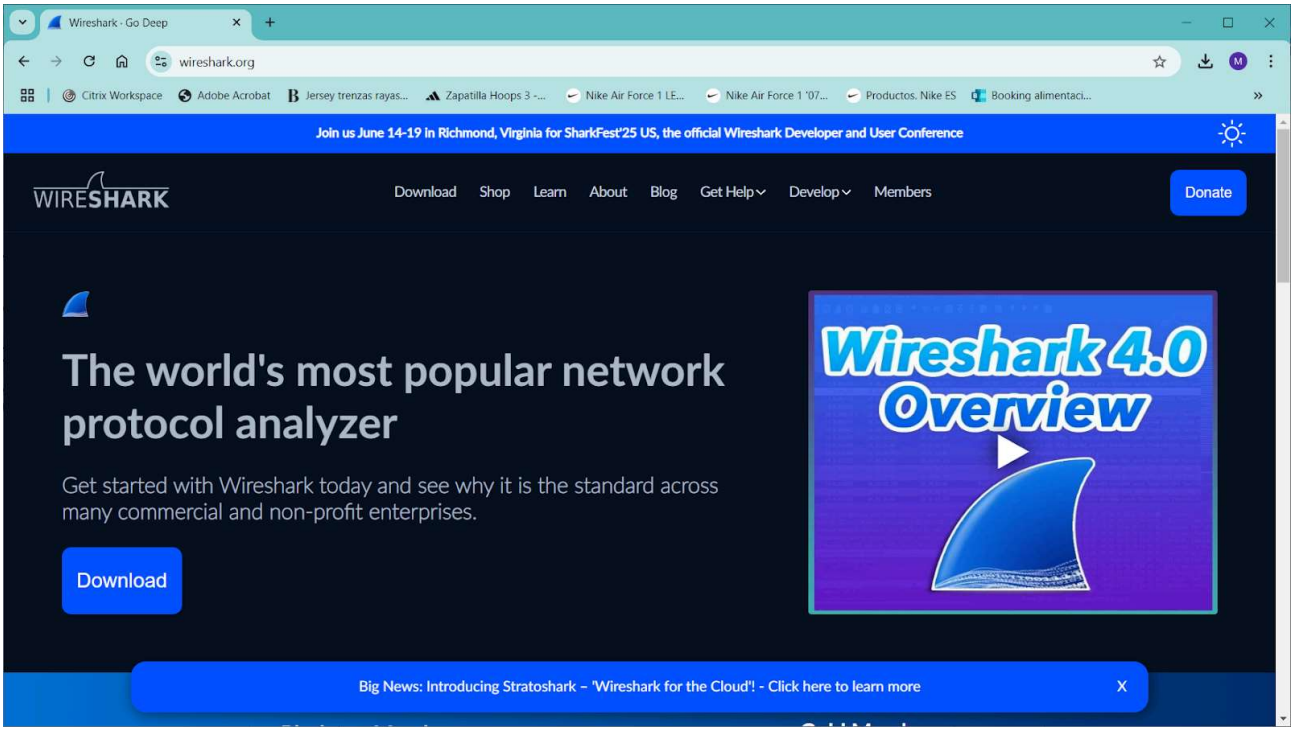
   http://gaia.cs.umass.edu/wireshark-labs/protected_pages/HTTP-wireshark-file5.html

   When the web site asks for authentication data, enter the username **wireshark-students** and password **network**. Select *http* as Wireshark filter. The username and password are visible after line *Authorization: Basic* in a HTTP GET message. Show a screenshot where this information is present.

Annex: Wireshark preliminary guidelines

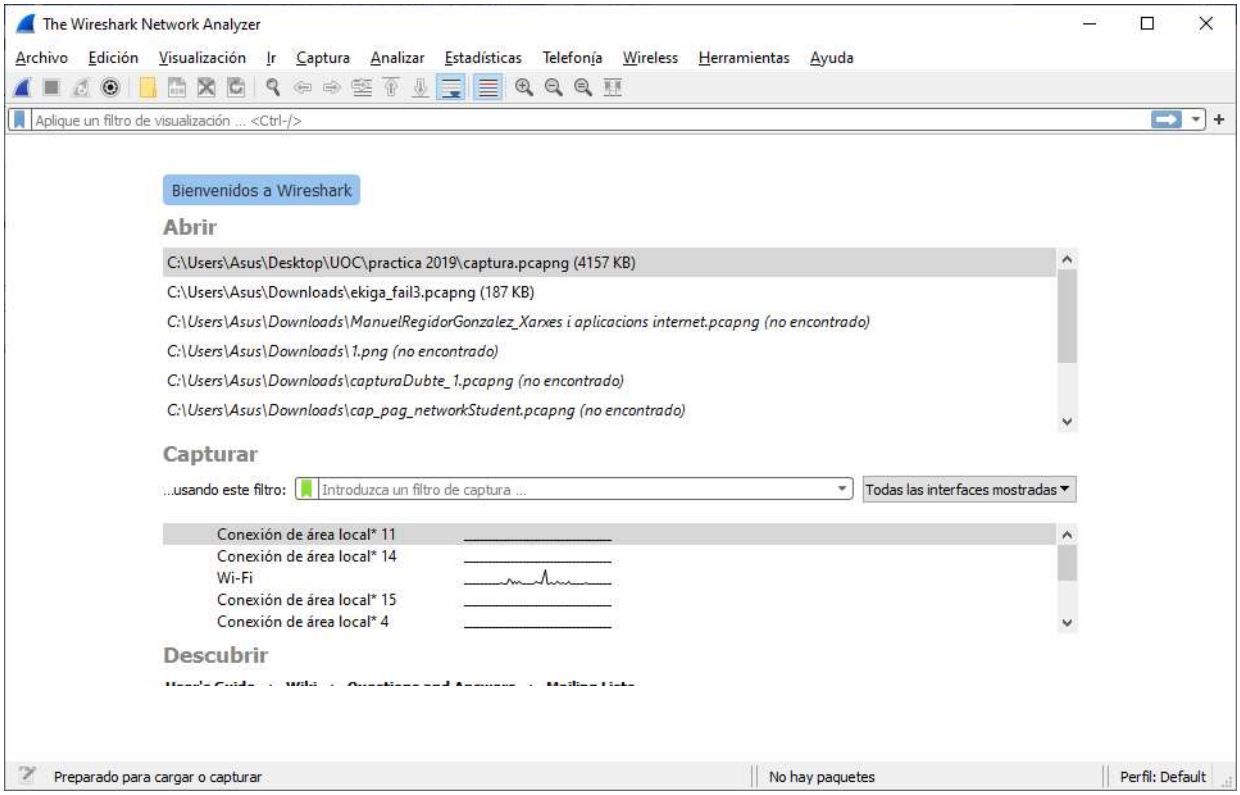
To be able to do this practice you can use a computer with GNU/Linux or Windows operating system. Your computer must be connected and configured to go to the Internet, through a LAN, ADSL, optical fiber or any other system.

Moreover, it is necessary to download the Wireshark program from https://www.wireshark.org. You can either use Windows or GNU/Linux version. In this way, once you enter in the Wireshark web site, you can click Download and select which program to download depending on your operating system. Once the corresponding files are downloaded, follow program installation instructions. You can find the complete user manual or basic tutorials at https://www.wireshark.org/docs/

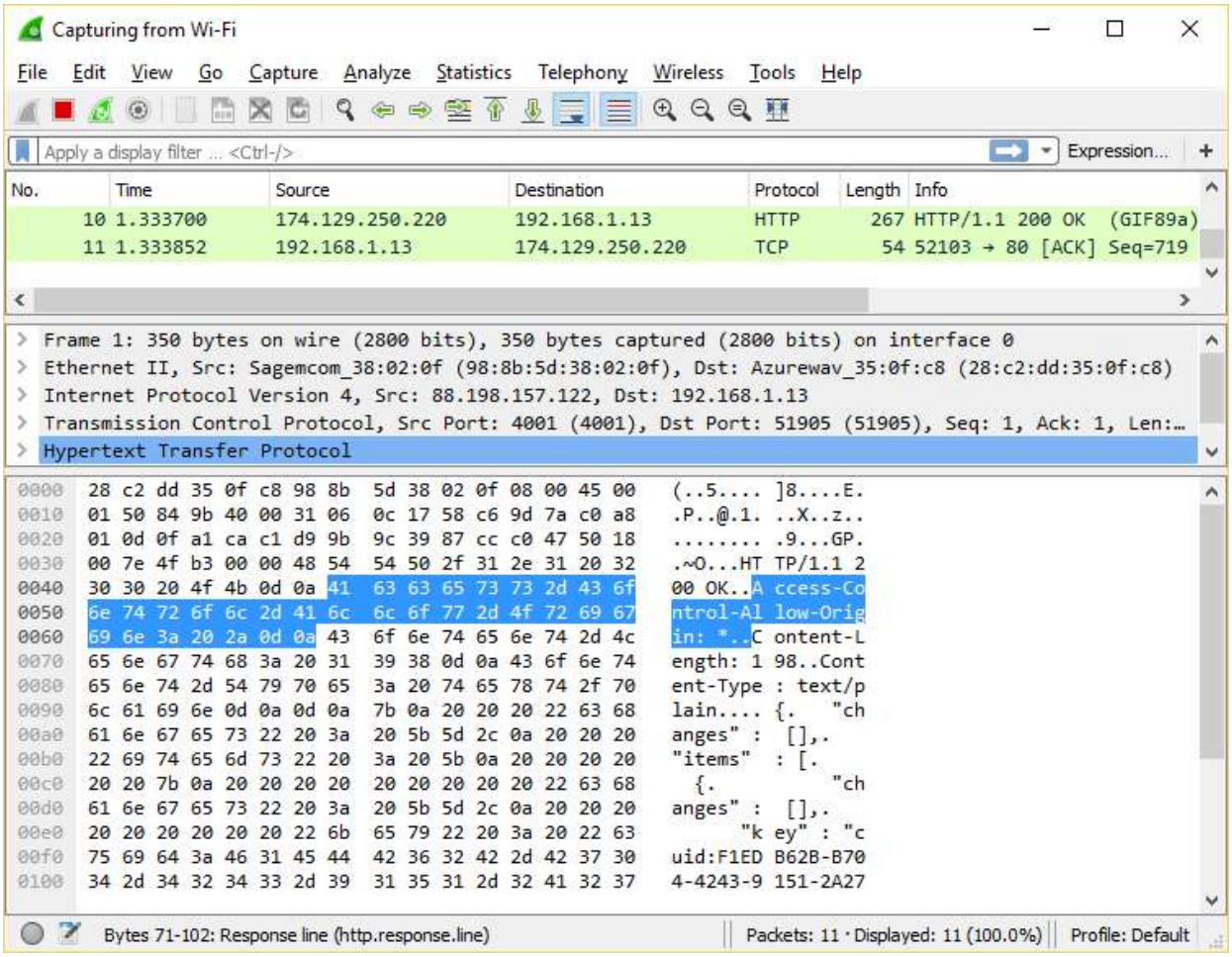


In this practice you will analyze link, network, transport and application protocols using the information provided by Wireshark, as well as standard applications both from Internet and multimedia.

To do so, you first need to select the network interface where you will capture traffic. Afterwards, before connecting through an existing Internet browser, click on the menu *Capture* > *Start*. To stop the capture, go to menu *Capture* > *Stop.*



As you can see in the next screenshot, the window is divided in three parts. To analyze a specific packet, click on a frame in the upper window. In the middle section, you can see the contents of the frame selected in the upper window, with a summary line of each header from MAC layer (Ethernet in the example), IP, TCP/UDP and Application. If you click at the symbol '>' to the left of each line, it will expand and show the details of all fields in that header. At the bottom the hexadecimal dump of the analyzed frame appears.



***Note**:* To facilitate understanding of the Wireshark capture you can select in the analyzer the filter you require (menu *Capture>Capture Filters*) and then perform, for example, an access with a browser to a web server. Or, once the capture stops, go to menu *Analyze>Display Filters* and select the required filter. You can filter by source IP address, destination IP, protocol, etc.

For more information about basic tutorials, you can refer to:
https://lifewire.com/wireshark-tutorial-4143298

Introduction to network traffic analysis with Wireshark
https://www.youtube.com/watch?v=shp42M7gbDE

Free Webinar: Wireshark
https://www.youtube.com/watch?v=kwxgHspdmag