***CS 421 Wireshark Assignment  
Görkem Kadir Solun 22003214***

Taking Wireshark for a Test Run

# What to hand in

## 1. List up to 10 different protocols that appear in the protocol column in the unfiltered packet-listing window in step 7 above.

ieee1905, UDP, TLSv1.3, TLSv1.2, TCP, STP, SKYPE, HTTP, DNS, BFCP, ARP

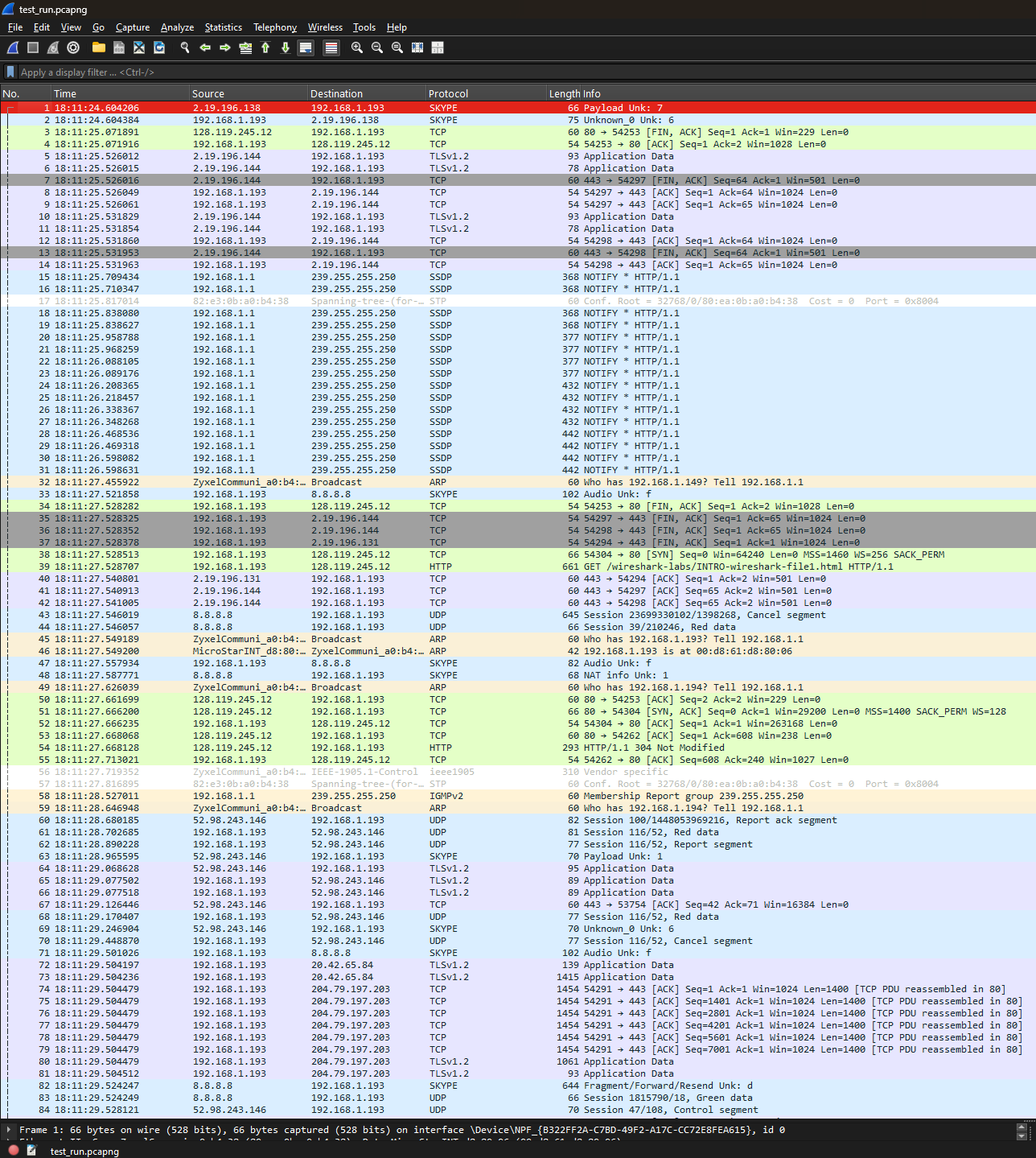


Figure 1: Some of the protocols from the first task

## 2. How long did it take from when the HTTP GET message was sent until the HTTP OK reply was received? (By default, the value of the Time column in the packet listing window is the amount of time, in seconds, since Wireshark tracing began. To display the Time field in time-of-day format, select the Wireshark View pull down menu, then select Time Display Format, then select Time-of-day.)



~0.12 seconds

## 3. What is the Internet address of gaia.cs.umass.edu (also known as www.net.cs.umass.edu)? What is the Internet address of your computer?

My computer: 192.169.1.193

gaia.cs.umass.edu: 128.119.245.12

## 4. Print the two HTTP messages displayed in step 9 above. To do so, select Print from the Wireshark File command menu, and select “Selected Packet Only” and “Print as displayed” and then click OK.

A screenshot of a computer

Description automatically generated

Figure 2: Print of HTTP OK

A screenshot of a computer

Description automatically generated

Figure 3: Print of HTTP OK

Wireshark Lab: HTTP

# 1. The Basic HTTP GET/response interaction

A screenshot of a computer

Description automatically generated

Figure 4: Downloaded the first HTML file

## A screenshot of a computer Description automatically generated

Figure 5: HTML file download GET

## A screenshot of a computer Description automatically generated

Figure 6: HTML file download OK

## 1. Is your browser running HTTP version 1.0 or 1.1? What version of HTTP is the server running?

It uses HTTP 1.1 by checking the response and request versions in the header.

## 2. What languages (if any) does your browser indicate that it can accept to the server?

“Accept-Language: en-US,en;q=0.9,tr;q=0.8\r\n” at GET request implies US English and TR Turkish.

## 3. What is the IP address of your computer? Of the gaia.cs.umass.edu server?

In the GET request, it says my IP is 192.168.1.193, and gaia.cs.umass.edu’s IP is 128.119.245.12.

## A close-up of a number Description automatically generated4. What is the status code returned from the server to your browser?

It is “200” as written in the OK response.

## A close up of a text Description automatically generated5. When was the HTML file that you are retrieving last modified at the server?

It is “Sun, 13 Oct 2024 05:59:02” as written in the OK response.

## A black text on a white background Description automatically generated6. How many bytes of content are being returned to your browser?

This value is in the "Content-Length: 128" line of the HTTP OK response header.

## 7. By inspecting the raw data in the packet content window, do you see any headers within the data that are not displayed in the packet-listing window? If so, name one.

No additional headers are visible within the raw data that are not already displayed in the packet-listing window. All the key HTTP headers, such as Date, Server, Last-Modified, ETag, Content-Length, Keep-Alive, Connection, and Content-Type, appear fully displayed in the packet-listing window. Please check Figure 6 above.

# 2. The HTTP CONDITIONAL GET/response interaction

A screenshot of a computer

Description automatically generated

Figure 7: Not modified

A screenshot of a computer

Description automatically generated

Figure 8: The third GET request of the cached file.

A screenshot of a computer

Description automatically generated

Figure 9: Successful file retrieval with an OK response.

A screenshot of a computer

Description automatically generated

Figure 10: The first GET request of the file.

## A screen shot of a computer Description automatically generated8. Inspect the contents of the first HTTP GET request from your browser to the server. Do you see an “IF-MODIFIED-SINCE” line in the HTTP GET?

No.

## 9. Inspect the contents of the server response. Did the server explicitly return the contents of the *file? How can you tell?*

A screen shot of a computer

Description automatically generatedBy comparing Figure 9 with 7, we can see that the OK response has the file data. With this, we can tell that the file was retrieved in the OK response and not in the Not modified response.

## A screenshot of a computer Description automatically generated10. Now inspect the contents of the second HTTP GET request from your browser to the server. Do you see an “IF-MODIFIED-SINCE:” line in the HTTP GET? If so, what information follows the “IF-MODIFIED-SINCE:” header?

Yes. When we first got our file.

## A screen shot of a computer program Description automatically generated11. What is the HTTP status code and phrase returned from the server in response to this second HTTP GET? Did the server explicitly return the contents of the file? Explain.

No. The server did not return the file's contents requested as the file was not modified.

# 3. Retrieving Long Documents

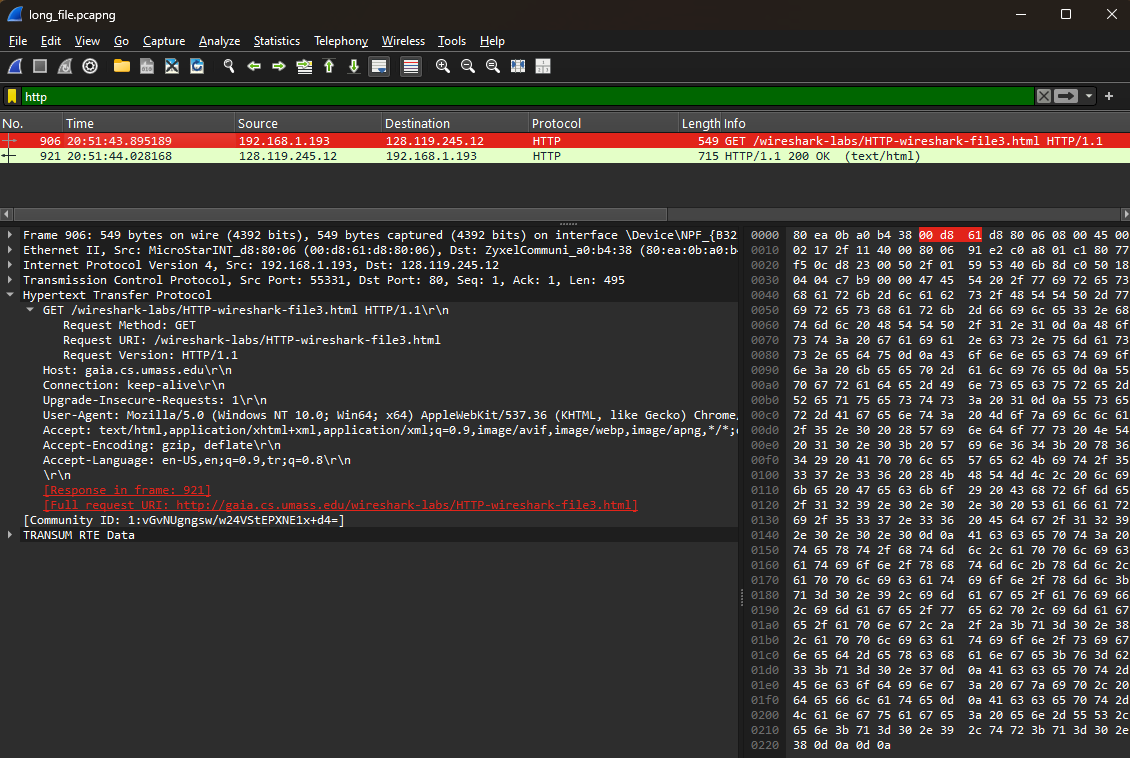


Figure 11: GET request for the long file.

A screenshot of a computer

Description automatically generated

Figure 12: Retrieved long file

## 12. How many HTTP GET request messages were sent by your browser?

One.

## A screenshot of a computer program Description automatically generated13. How many data-containing TCP segments were needed to carry the single HTTP response?

Four.

## A screen shot of a computer Description automatically generated14. What is the status code and phrase associated with the response to the HTTP GET request?

200 OK.

## 15. Are there any HTTP status lines in the transmitted data associated with a TCP induced “Continuation”?

There is no additional HTTP status line in these continuation packets—just the payload's continuation.

# 4. HTML Documents with Embedded Objects