

**«ІНСТИТУТ ПРИКЛАДНОГО СИСТЕМНОГО АНАЛІЗУ»
НАЦІОНАЛЬНОГО ТЕХНІЧНОГО УНІВЕРСИТЕТУ УКРАЇНИ «КП»
КАФЕДРА МАТЕМАТИЧНИХ МЕТОДІВ СИСТЕМНОГО АНАЛІЗУ**

**Лабораторна робота №2
з курсу «Комп'ютерні мережі»**

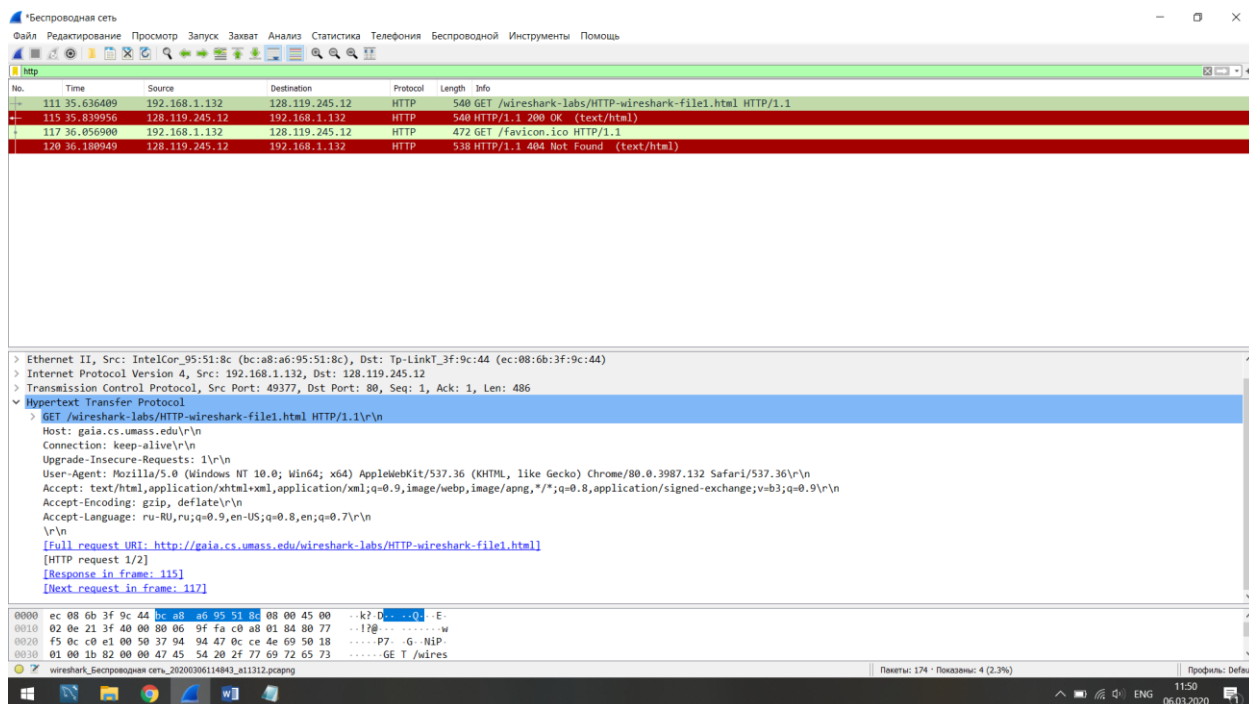
Виконала: студентка 3 курсу

групи КА 73

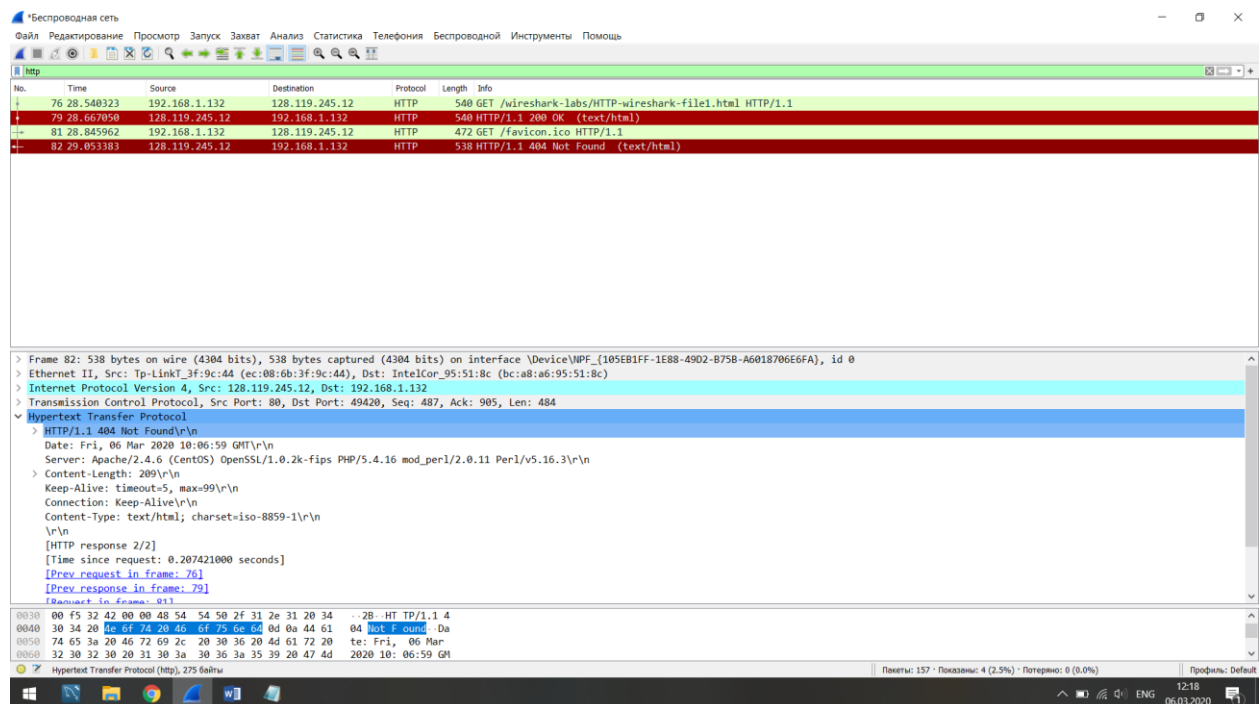
Собкович В.С.

Прийняв: Кухарєв С.О.

Київ 2020 р.



Frame 111: 540 bytes on wire (4320 bits), 540 bytes captured (4320 bits) on interface
 \Device\NPF_{105EB1FF-1E88-49D2-B75B-A6018706E6FA}, id 0
 Ethernet II, Src: IntelCor_95:51:8c (bc:a8:a6:95:51:8c), Dst: Tp-LinkT_3f:9c:44 (ec:08:6b:3f:9c:44)
 Internet Protocol Version 4, Src: 192.168.1.132, Dst: 128.119.245.12
 Transmission Control Protocol, Src Port: 49377, Dst Port: 80, Seq: 1, Ack: 1, Len: 486
 Hypertext Transfer Protocol
 GET /wireshark-labs/HTTP-wireshark-file1.html HTTP/1.1\r\n
 [Expert Info (Chat/Sequence): GET /wireshark-labs/HTTP-wireshark-file1.html HTTP/1.1\r\n]
 Request Method: GET
 Request URI: /wireshark-labs/HTTP-wireshark-file1.html
 Request Version: HTTP/1.1
 Host: gaia.cs.umass.edu\r\n
 Connection: keep-alive\r\n
 Upgrade-Insecure-Requests: 1\r\n
 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/80.0.3987.132 Safari/537.36\r\n
 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9\r\n
 Accept-Encoding: gzip, deflate\r\n
 Accept-Language: ru-RU,ru;q=0.9,en-US;q=0.8,en;q=0.7\r\n
 \r\n
 [Full request URI: http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file1.html]
 [HTTP request 1/2]
 [Response in frame: 115]
 [Next request in frame: 117]



Hypertext Transfer Protocol

HTTP/1.1 404 Not Found\r\n

Date: Fri, 06 Mar 2020 10:06:59 GMT\r\n

Server: Apache/2.4.6 (CentOS) OpenSSL/1.0.2k-fips PHP/5.4.16 mod_perl/2.0.11 Perl/v5.16.3\r\n

Content-Length: 209\r\n

Keep-Alive: timeout=5, max=99\r\n

Connection: Keep-Alive\r\n

Content-Type: text/html; charset=iso-8859-1\r\n

\r\n

[HTTP response 2/2]

[Time since request: 0.207421000 seconds]

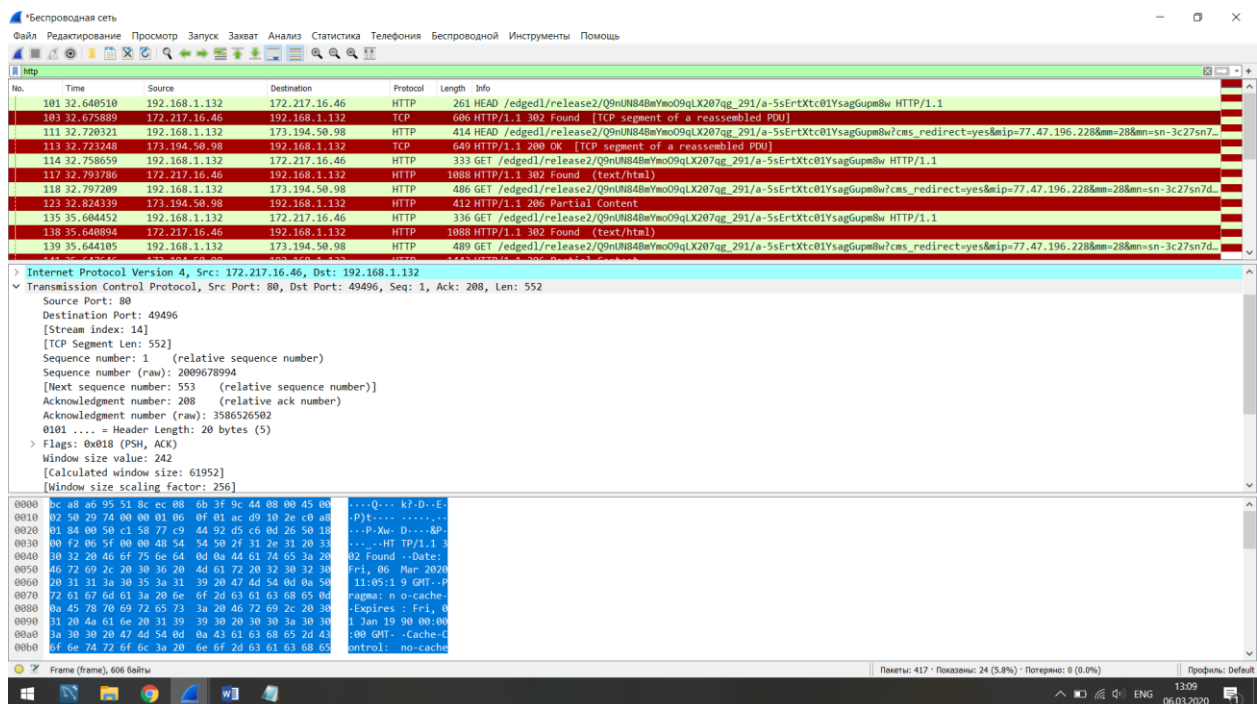
[Prev request in frame: 76]

[Prev response in frame: 79]

[Request in frame: 81]

[Request URI: http://gaia.cs.umass.edu/favicon.ico]

File Data: 209 bytes



Hypertext Transfer Protocol

GET /wireshark-labs/HTTP-wireshark-file4.html HTTP/1.1\r\n

[Expert Info (Chat/Sequence): GET /wireshark-labs/HTTP-wireshark-file4.html HTTP/1.1\r\n]

[GET /wireshark-labs/HTTP-wireshark-file4.html HTTP/1.1\r\n]

[Severity level: Chat]

[Group: Sequence]

Request Method: GET

Request URI: /wireshark-labs/HTTP-wireshark-file4.html

Request Version: HTTP/1.1

Host: gaia.cs.umass.edu\r\n

Connection: keep-alive\r\n

Upgrade-Insecure-Requests: 1\r\n

User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)

Chrome/80.0.3987.132 Safari/537.36\r\n

Accept:

text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9\r\n

Accept-Encoding: gzip, deflate\r\n

Accept-Language: ru-RU,ru;q=0.9,en-US;q=0.8,en;q=0.7\r\n

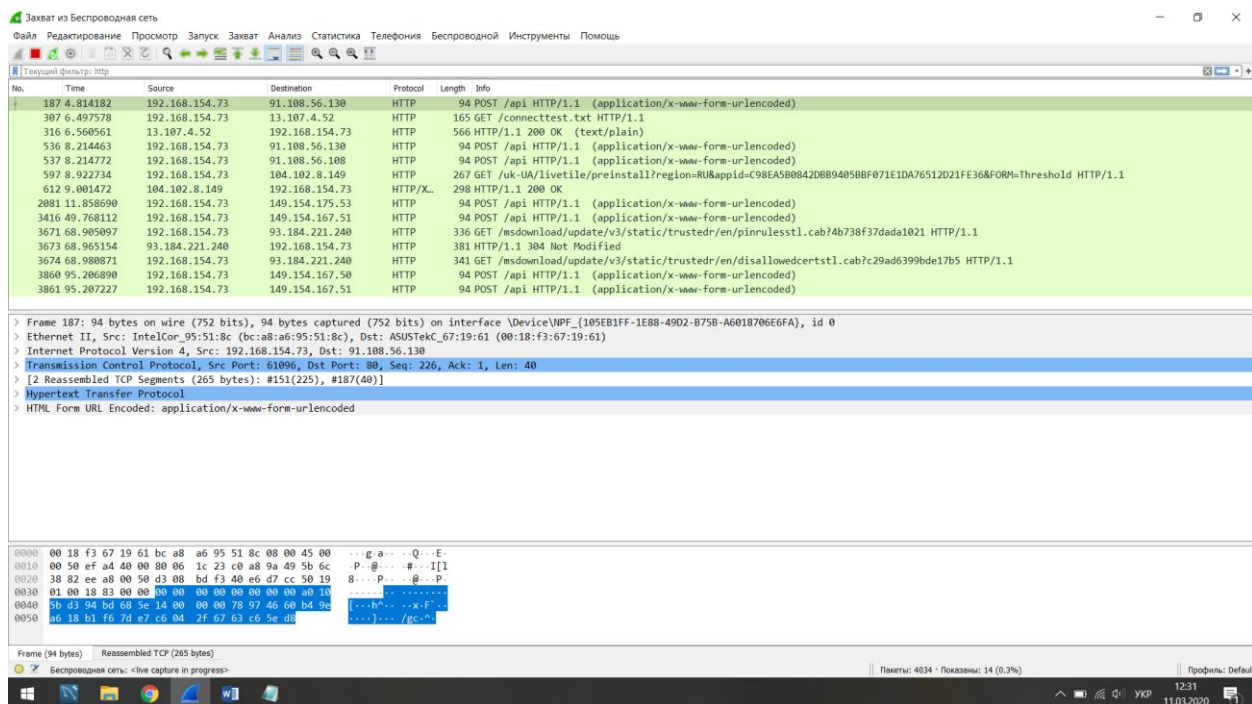
\r\n

[Full request URI: http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file4.html]

[HTTP request 1/3]

[Response in frame: 129]

[Next request in frame: 131]



Hypertext Transfer Protocol

POST /api HTTP/1.1\r\n

Host: 91.108.56.130:80\r\n

Content-Type: application/x-www-form-urlencoded\r\n

Content-Length: 40\r\n

Connection: Keep-Alive\r\n

Accept-Encoding: gzip, deflate\r\n

Accept-Language: ru-RU,en,*\r\n

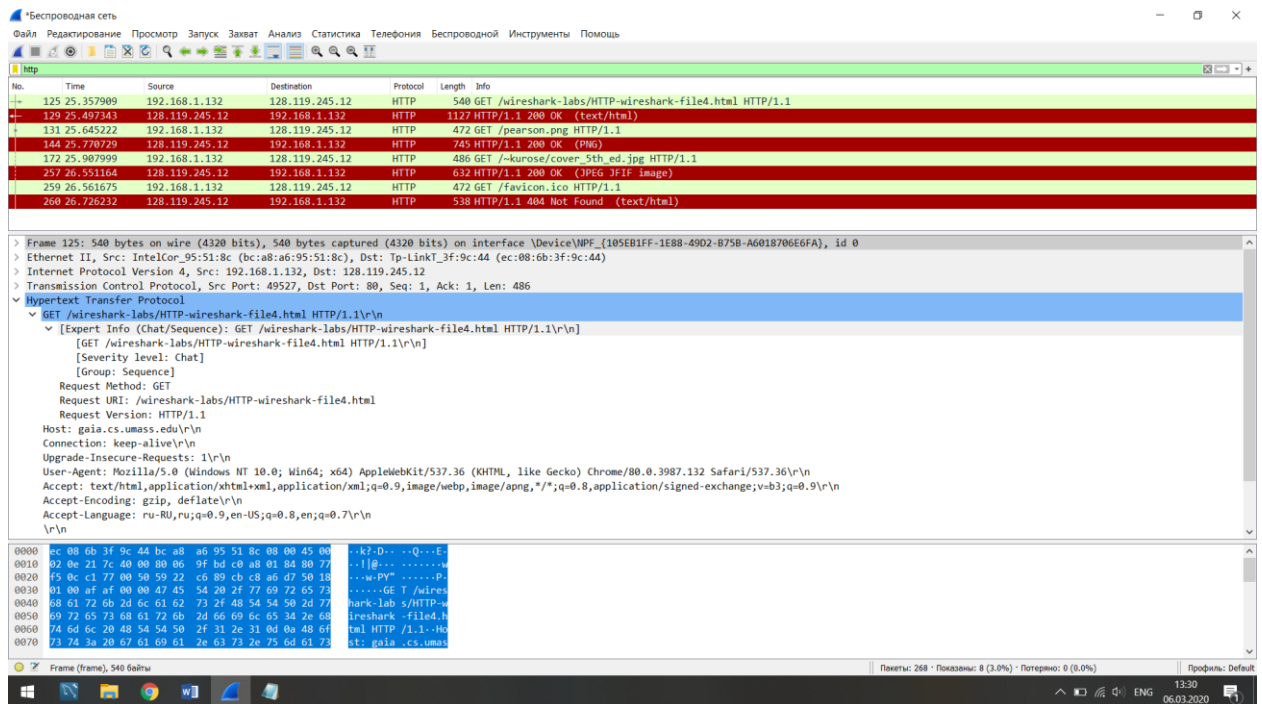
User-Agent: Mozilla/5.0\r\n

\r\n

[Full request URI: http://91.108.56.130:80/api]

[HTTP request 1/1]

File Data: 40 bytes



Контрольні запитання:

1. Яку версію протоколу HTTP використовує ваш браузер (1.0 чи 1.1)?

1.1 (GET /wireshark-labs/HTTP-wireshark-file4.html HTTP/1.1)

Яку версію протоколу використовує сервер?

1.1 (HTTP/1.1 200 OK (text/html))

2. Які мови (якщо вказано) браузер може прийняти від сервера?

Російська, американська англійська, англійська.

(Accept-Language: ru-RU,ru;q=0.9,en-US;q=0.8,en;q=0.7\r\n)

3. Які IP-адреси вашого комп'ютера та цільового веб-сервера?

192.168.1.132

128.119.245.12

4. Який статусний код сервер повернув у відповіді вашому браузеру?

200

5. Коли на сервері в останній раз був модифікований файл, який запитується браузером?

Last-Modified: Fri, 06 Mar 2020 06:59:02 GMT\r\n

6. Скільки байт контенту повертається сервером?

128

7. Переглядаючи нерозібраний байтовий потік пакету, чи бачите ви деякі заголовки в потоці, які не відображаються у вікні деталей пакету? Якщо так, назвіть один з них.

Ні

8. Перевірте вміст першого запити HTTP GET від вашого браузера до сервера. Чи є в

ньому заголовок IF-MODIFIED-SINCE?

Ні

9. Перевірте вміст першої відповіді сервера. Чи повернув сервер вміст файлу безпосередньо у відповіді?

Так

10. Перевірте вміст другого запиту HTTP GET. Чи є в ньому заголовок IF-MODIFIEDSINCE? Якщо так, яке значення йому відповідає?

Ні

11. Який код та опис статусу другої відповіді сервера? Чи повернув сервер вміст файлу безпосередньо у відповіді?

404 Not Found, ні

12. Скільки повідомлень HTTP GET було відправлено вашим браузером?

1

13. Скільки пакетів TCP було необхідно для доставки одної відповіді HTTP-сервера?

1

14. Який код та опис статусу був у відповіді сервера?

200 OK

15. Чи зустрічаються у даних пакетів-продовжень протоколу TCP стрічки з кодом та описом статусу відповіді, або ж якісь заголовки протоколу HTTP?

Ні

16. Скільки запитів HTTP GET було відправлено вашим браузером? Якими були цільові IP-адреси запитів?

4, 128.119.245.12.

17. Чи можете ви встановити, чи були ресурси отримані паралельно чи послідовно? Яким чином?

Послідовно, за часом надходження відповідей та адресами.

Висновок:

В даній роботі проаналізовано деталі роботи протоколу HTTP. HTTP — протокол прикладного рівня. Обмін повідомленнями йде за звичайною схемою «запит-відповідь». Для ідентифікації ресурсів HTTP використовує глобальні URI. На відміну від багатьох інших протоколів, HTTP не зберігає свого стану. Це означає відсутність збереження проміжного стану між парами «запит-відповідь». Компоненти, що використовують HTTP, можуть самостійно здійснювати збереження інформації про стан, пов'язаний з останніми запитами та відповідями. Браузер, котрий посилає запити, може відстежувати затримки відповідей. Сервер може зберігати IP-адреси та заголовки запитів останніх клієнтів. Проте, згідно з протоколом, клієнт та сервер не мають бути обізнаними з попередніми запитами та відповідями, у протоколі не передбачена внутрішня підтримка стану й він не ставить таких вимог до клієнта та сервера.