

Санкт-Петербургский Политехнический Университет Петра Великого
Институт компьютерных наук и технологий
Кафедра компьютерных систем и программных технологий

Защита информации

Отчет по лабораторной работе №1

Исследование сетевого трафика

Работу выполнил:
Раскин Андрей
Группа: 43501/3
Преподаватель:
Новопашенный Андрей Гелиевич

Санкт-Петербург
2017

1 Цель работы

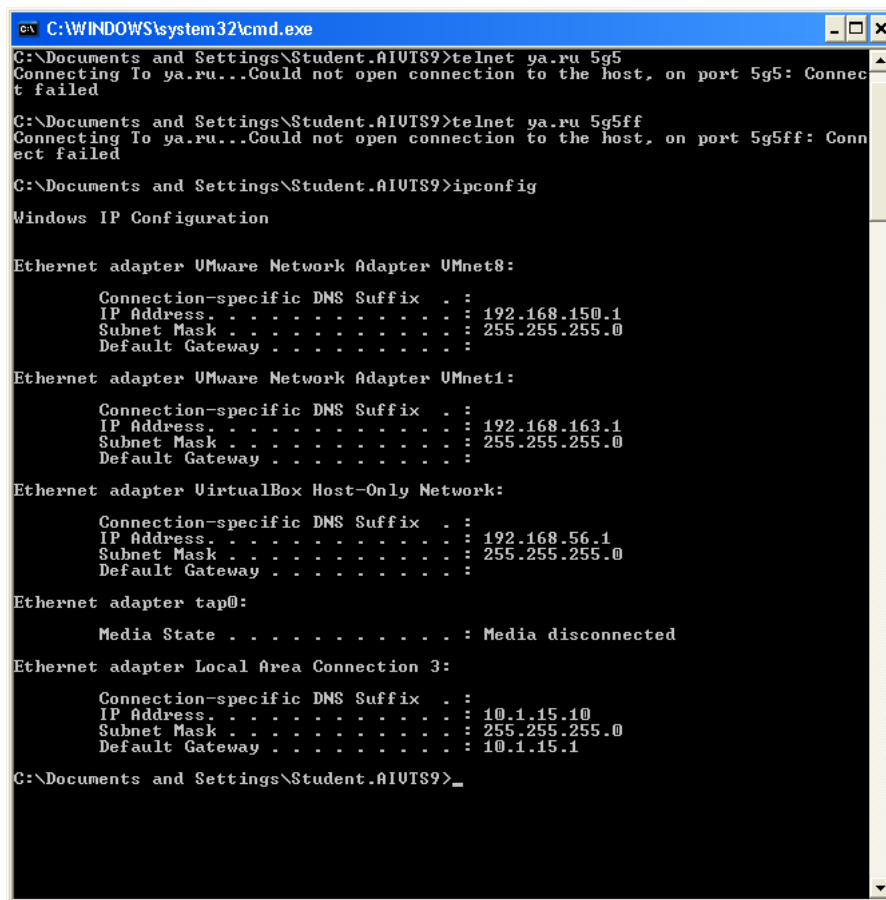
Закрепление навыков работы в программе Wireshark и знаний о некоторых сетевых протоколах.

2 Программа работы

При помощи анализатора сетевого трафика Wireshark продемонстрировать в сети:

1. Работу утилиты ping
2. Работу утилиты tracert
3. Работу ICMP-протокола в следующих ситуациях:
 - Отправка фрагментированного ping'a,
 - Получение ошибки 3.1 (Destination host unreachable)
4. Работу ARP-протокола (запрос и ответ);
5. Работу протокола TCP в следующих ситуациях:
 - Установка соединения,
 - Разрыв соединения,
 - Попытка соединения на отсутствующий порт

3 Конфигурация компьютера в сети



```
C:\WINDOWS\system32\cmd.exe
C:\Documents and Settings\Student.AIUTS9>telnet ya.ru 5g5
Connecting To ya.ru...Could not open connection to the host, on port 5g5: Connection failed
C:\Documents and Settings\Student.AIUTS9>telnet ya.ru 5g5ff
Connecting To ya.ru...Could not open connection to the host, on port 5g5ff: Connection failed
C:\Documents and Settings\Student.AIUTS9>ipconfig

Windows IP Configuration

Ethernet adapter VMware Network Adapter VMnet8:

    Connection-specific DNS Suffix  . : 
    IP Address. . . . .               : 192.168.150.1
    Subnet Mask . . . . .             : 255.255.255.0
    Default Gateway . . . . .         : 

Ethernet adapter VMware Network Adapter VMnet1:

    Connection-specific DNS Suffix  . : 
    IP Address. . . . .               : 192.168.163.1
    Subnet Mask . . . . .             : 255.255.255.0
    Default Gateway . . . . .         : 

Ethernet adapter VirtualBox Host-Only Network:

    Connection-specific DNS Suffix  . : 
    IP Address. . . . .               : 192.168.56.1
    Subnet Mask . . . . .             : 255.255.255.0
    Default Gateway . . . . .         : 

Ethernet adapter tap0:

    Media State . . . . .             : Media disconnected

Ethernet adapter Local Area Connection 3:

    Connection-specific DNS Suffix  . : 
    IP Address. . . . .               : 10.1.15.10
    Subnet Mask . . . . .             : 255.255.255.0
    Default Gateway . . . . .         : 10.1.15.1

C:\Documents and Settings\Student.AIUTS9>
```

Рис. 1: Конфигурация сети

4 Ход работы

4.1 Работы утилиты ping

Ping — утилита для проверки целостности и качества соединений в сетях на основе TCP/IP, а также обиходное наименование самого запроса. Утилита отправляет запросы (ICMP Echo-Request) протокола ICMP указанному узлу сети и фиксирует поступающие ответы (ICMP Echo-Reply). По умолчанию производится 4 попытки отправки запроса.

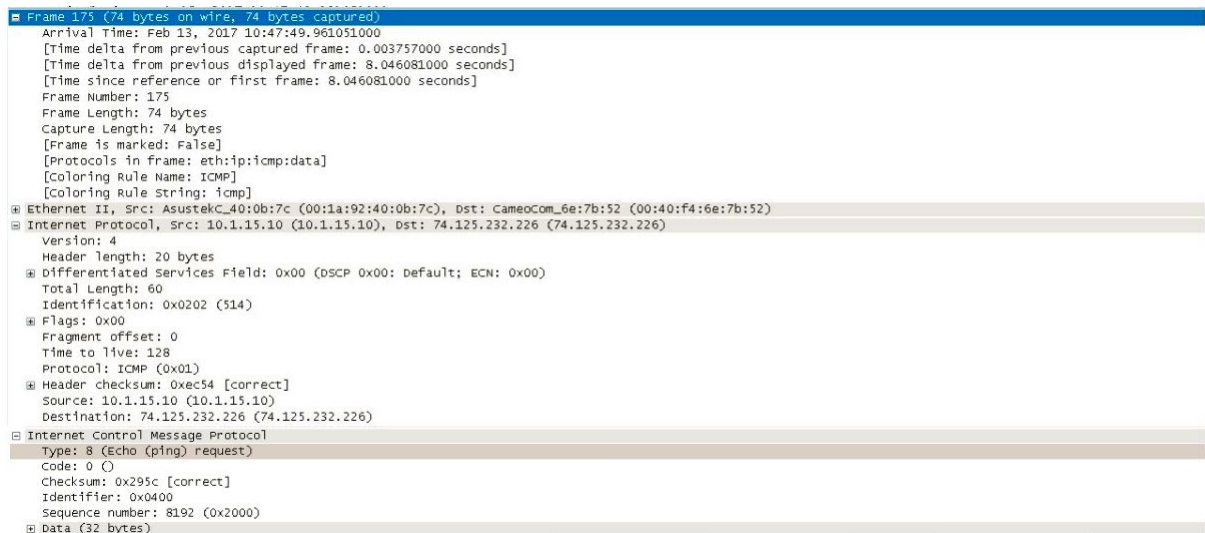


Рис. 2: ICMP эхо запрос

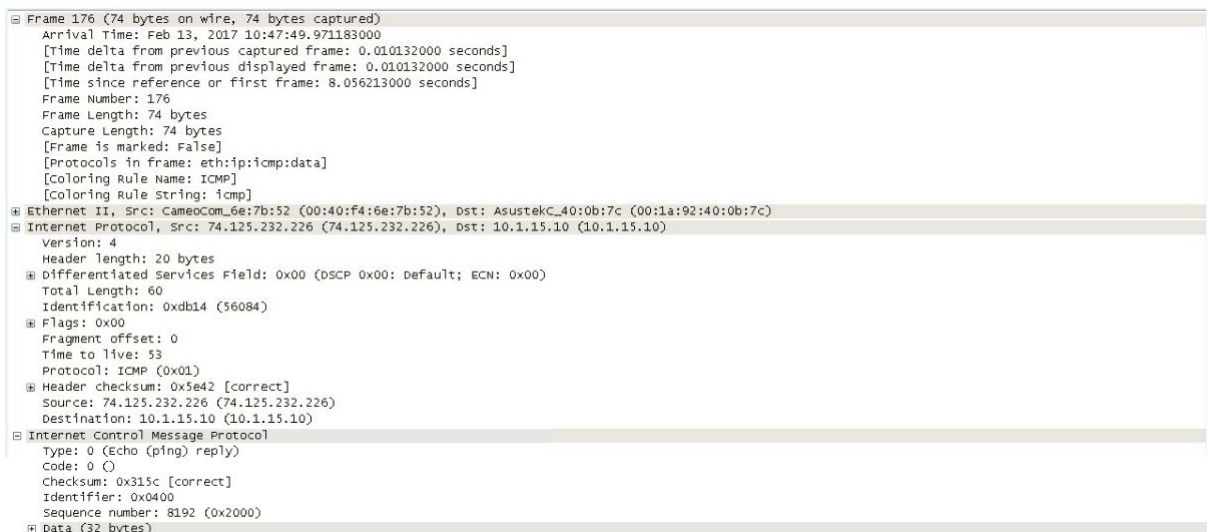


Рис. 3: ICMP эхо ответ

Как видно, в поле Destination указан IP-адрес google.com, поле Source показывает IP-адрес текущего компьютера. Тип сообщения равный 8 означает эхо-запрос, а тип 0 означает эхо-ответ.

4.2 Работа утилиты tracer

В основе работы данной утилиты лежит протокол icmp. Команда TRACERT определяет путь до точки назначения с помощью послыки в точку назначения эхо-сообщений протокола Control Message Protocol (ICMP) с постоянным увеличением значений срока жизни (Time to Live, TTL). Попробуем пронаблюдать трассировку маршрута пакетов до узла spbstu.ru при помощи протокола ICMP и утилиты tracer.

Первый пакет трассировки маршрута отправляется с TTL равным 1. Это значит, что на первом же маршрутизаторе пакет будет уничтожен и нам придет сообщение об ошибке.

```
C:\Users\Georgiy>tracert spbstu.ru

Трассировка маршрута к spbstu.ru [195.209.230.198]
с максимальным числом прыжков 30:

  1      13 ms      5 ms      6 ms  192.168.1.1
  2      6 ms      6 ms     14 ms  192.168.25.2
  3      3 ms      4 ms      7 ms  1p-1.47.255.92.net.unnet.ru [92.255.47.1]
  4      3 ms     30 ms      5 ms  92.255.2.233
  5      7 ms      6 ms      4 ms  1p-51.97.104.89.net.unnet.ru [89.104.97.51]
  6      3 ms      3 ms      3 ms  h16-1-gv.spb.runnet.ru [194.190.255.29]
  7     78 ms     123 ms      5 ms  stu.spb.runnet.ru [194.85.36.238]
  8      4 ms      4 ms      6 ms  195.209.230.190
  9      4 ms      3 ms      4 ms  end.spbstu.ru [195.209.230.198]
```

Рис. 4: Результат трассировки маршрута в консоли

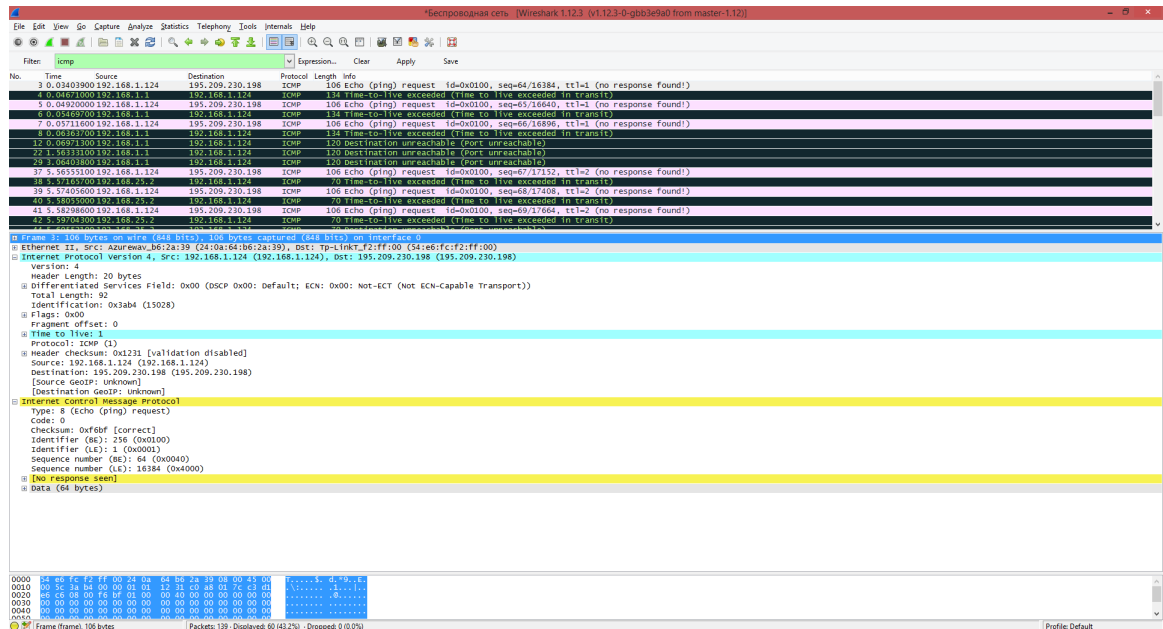


Рис. 5: Первый пакет трассировки маршрута

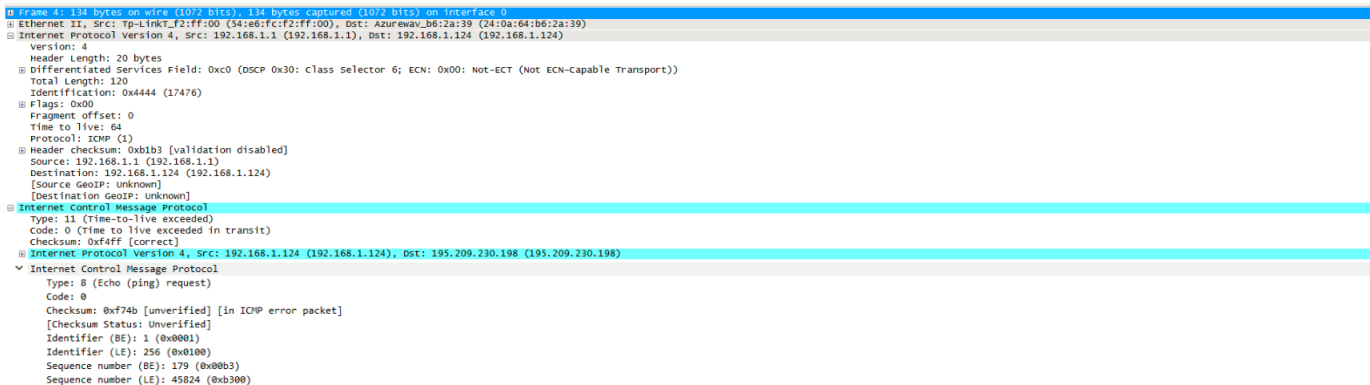


Рис. 6: Ответ на первый пакет трассировки

В сообщении об ошибке указан тип ICMP-пакета – 11.0, что означает, что время жизни пакета истекло. Сообщение пришло от маршрутизатора сети, который имеет адрес 192.168.1.1. Аналогично продолжается трассировка маршрута дальше с постепенным инкрементом параметра TTL.

```

> Frame 7: 106 bytes on wire (848 bits), 106 bytes captured (848 bits) on interface 0
> Ethernet II, Src: HonHaiPr_49:6d:bd (14:2d:27:49:6d:bd), Dst: D-Link_3e:a6:d6 (f0:7d:68:3e:a6:d6)
▼ Internet Protocol Version 4, Src: 192.168.0.104, Dst: 91.151.191.13
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
    > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 92
    Identification: 0x5e05 (24069)
    > Flags: 0x00
    Fragment offset: 0
    > Time to live: 2
    Protocol: ICMP (1)
    Header checksum: 0x7ee7 [validation disabled]
    [Header checksum status: Unverified]
    Source: 192.168.0.104
    Destination: 91.151.191.13
    [Source GeoIP: Unknown]
    [Destination GeoIP: Unknown]
▼ Internet Control Message Protocol
    Type: 8 (Echo (ping) request)

```

Рис. 7: Второй пакет трассировки

```

> Frame 8: 70 bytes on wire (560 bits), 70 bytes captured (560 bits) on interface 0
> Ethernet II, Src: D-Link_3e:a6:d6 (f0:7d:68:3e:a6:d6), Dst: HonHaiPr_49:6d:bd (14:2d:27:49:6d:bd)
▼ Internet Protocol Version 4, Src: 80.70.224.142, Dst: 192.168.0.104
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
    > Differentiated Services Field: 0xc0 (DSCP: CS6, ECN: Not-ECT)
    Total Length: 56
    Identification: 0x9746 (38726)
    > Flags: 0x00
    Fragment offset: 0
    Time to live: 254
    Protocol: ICMP (1)
    Header checksum: 0x32d9 [validation disabled]
    [Header checksum status: Unverified]
    Source: 80.70.224.142
    Destination: 192.168.0.104
    [Source GeoIP: Unknown]
    [Destination GeoIP: Unknown]
▼ Internet Control Message Protocol
    Type: 11 (Time-to-live exceeded)

```

Рис. 8: Ответ на второй пакет трассировки

Таким образом составляется примерный маршрут прохождения IP-пакета до узла с адресом spbstu.ru.

4.3 Протокол ICMP

4.3.1 Фрагментированный ping

Попробуем отослать фрагментированный ping-запрос. Данный вид запроса использует ICMP-протокол. Для фрагментации пакета необходимо указать его размер, превышающий MTU (maximum transmission unit) - максимальный размер полезного блока данных одного пакета, который может быть передан протоколом без фрагментации. Для протокола Ethernet обычно это чуть больше 1500 байт. Для фрагментации пакета на 3 части укажем размер – 4000 байт.

25	3.00459300	192.168.1.3	8.8.8.8	IPv4	1514	Fragmented IP protocol (proto=ICMP 1, off=0, ID=4e44) [Reassembled in #27]
26	3.00461900	192.168.1.3	8.8.8.8	IPv4	1514	Fragmented IP protocol (proto=ICMP 1, off=1480, ID=4e44) [Reassembled in #27]
27	3.00462600	192.168.1.3	8.8.8.8	ICMP	682	Echo (ping) request id=0x0100, seq=28/7168, ttl=128 (reply in 32)

Рис. 9: Сегментирование пакетов

Как видно, первый пакет был отправлен по протоколу ICMP, следующие фрагментированные пакеты передавались по протоколу IPv4 уже без заголовка ICMP.

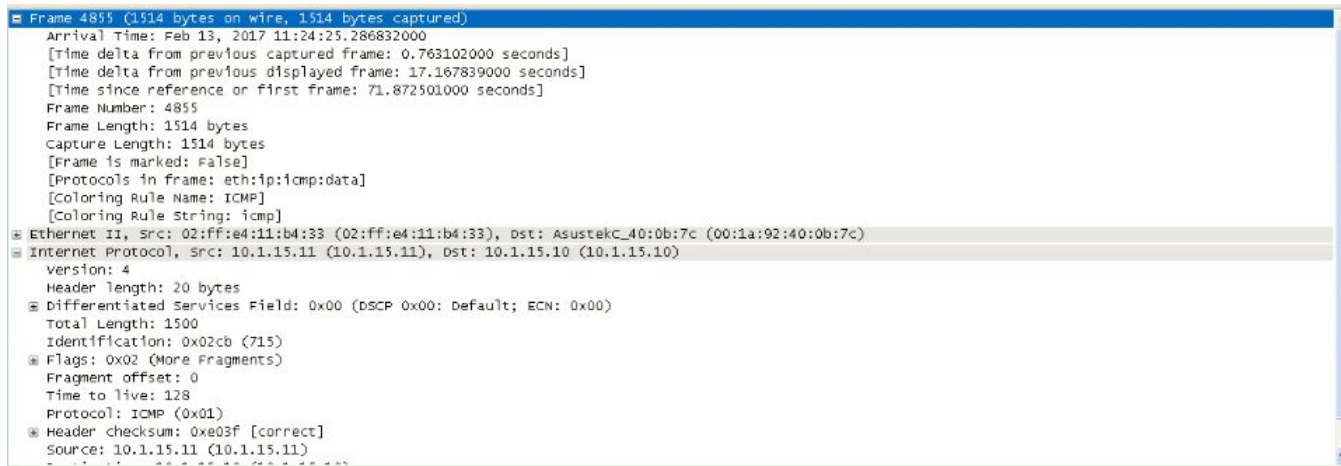


Рис. 10: Первый фрагмент пакета ping-запроса

О фрагментированности пакета свидетельствуют флаги пакета IP (0x01 – имеются еще фрагменты). О том, что это первый пакет из фрагментированных, свидетельствует нулевое смещение фрагмента. При этом во всех трех IP пакетах содержится ICMP-пакет с одним и тем же идентификатором.

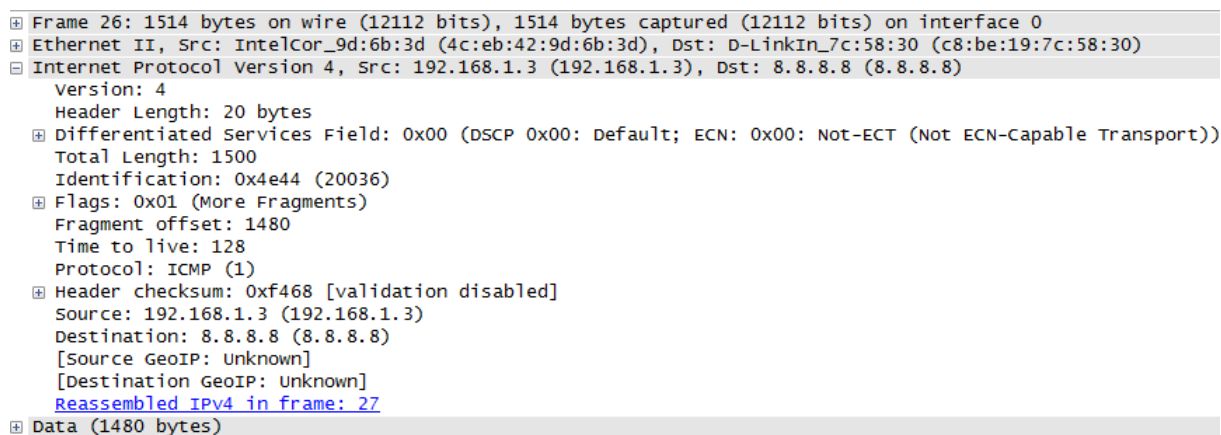


Рис. 11: Второй фрагмент запроса

Видим, что смещение в данном случае уже ненулевое. Последний пакет выглядит следующим образом:

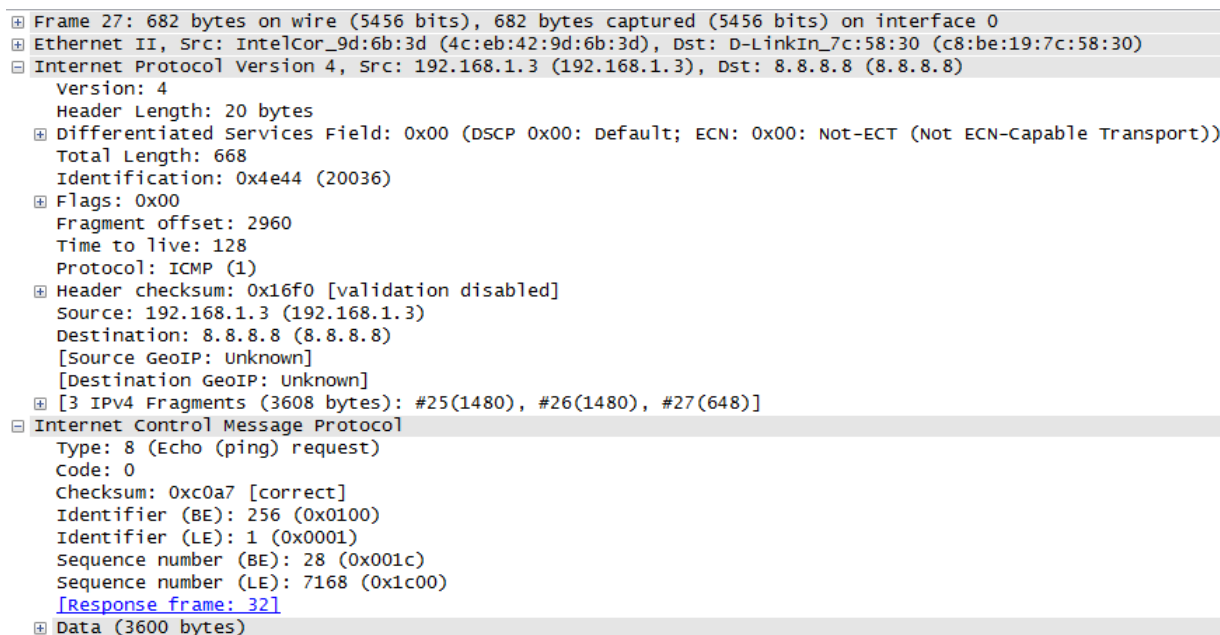


Рис. 12: Последний фрагментированный пакет

Здесь флаг, присутствующий в предыдущих пакетах, не установлен, что свидетельствует о том, что фрагмент последний.

4.3.2 Несуществующий хост

Попробуем пронаблюдать ошибку типа 3.1 (целевой узел недостижим). Для этого отправим ping-запрос на адрес, которого не существует. В пакете можно наблюдать типичный ping-запрос (ICMP-пакет типа 8.0).

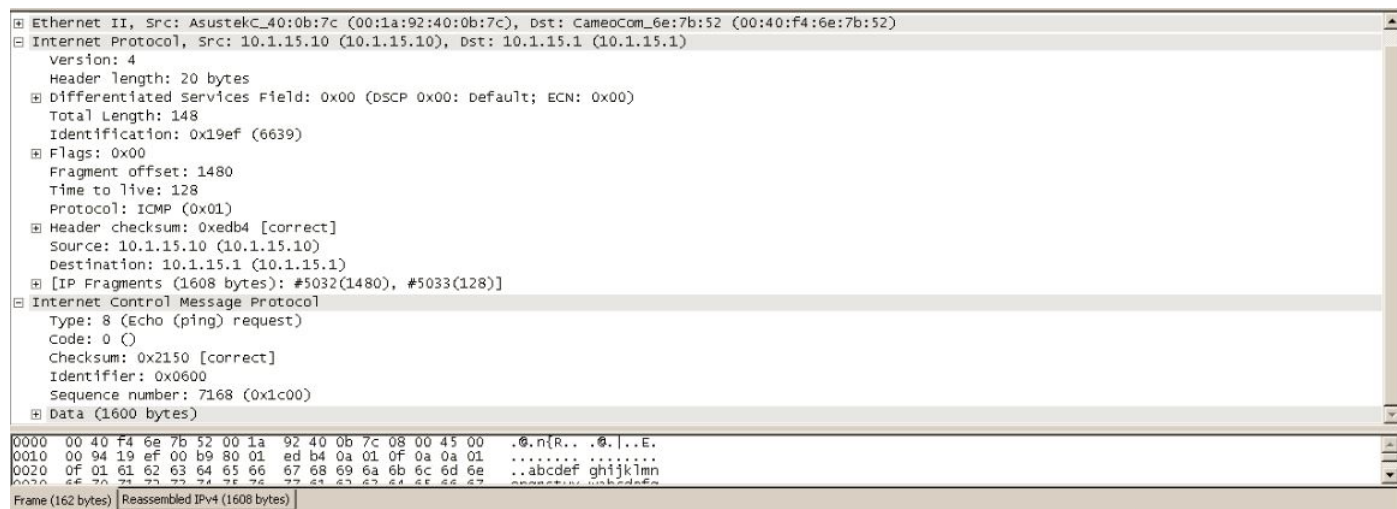


Рис. 13: Ping-запрос

А вот ответом на указанный выше запрос будет ICMP-пакет типа 3.1, свидетельствующий об ошибке «целевой узел недостижим». При этом, в ответе, в качестве данных пакета отправляется заголовок того пакета, на который пришел ответ.

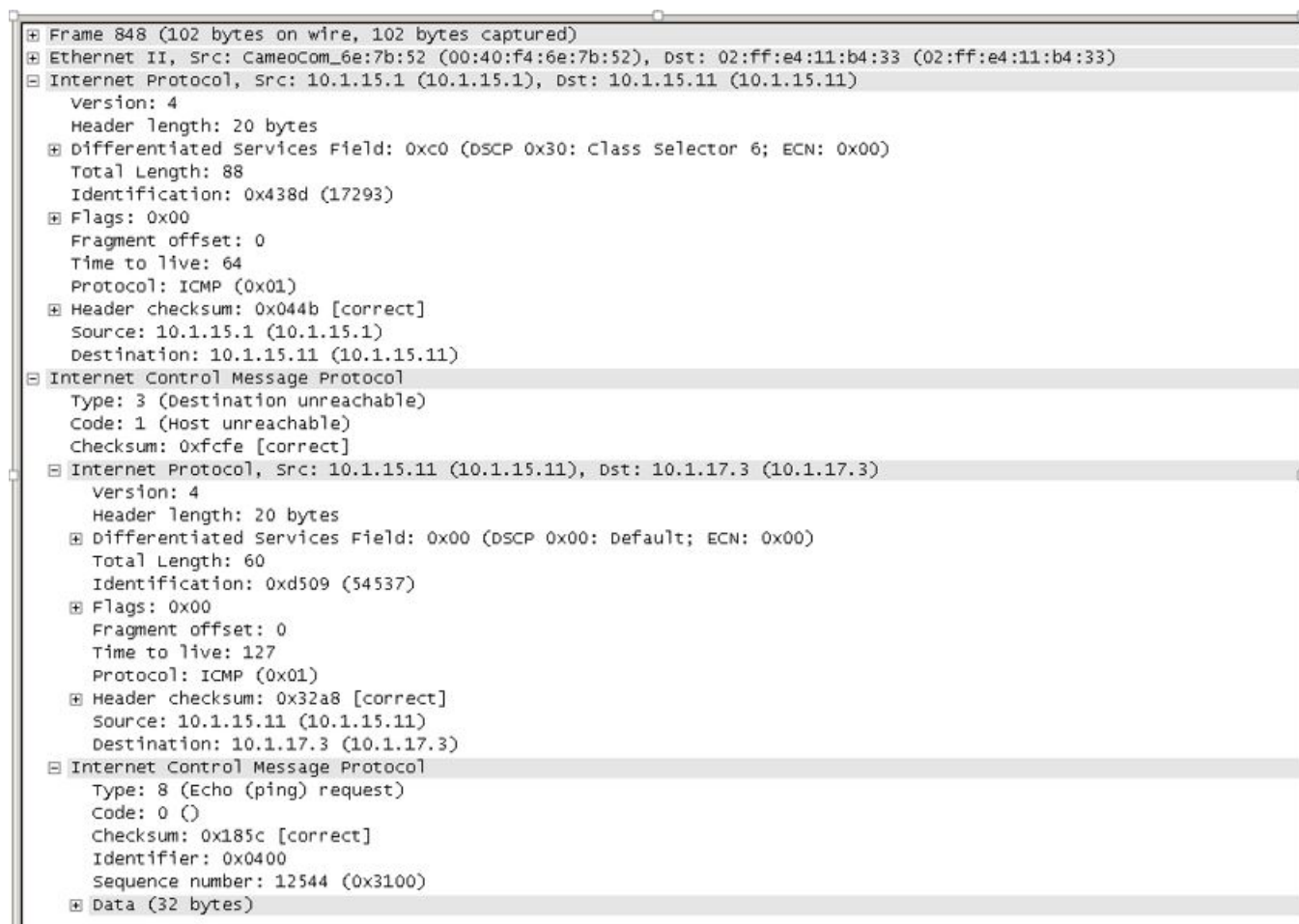


Рис. 14: ICMP-ответ

4.4 ARP протокол

В пакете указывается его тип (поле Opcode) – запрос, а так же целевой IP-адрес для которого запрашивается MAC-адрес. MAC-адрес цели при этом обнулен.

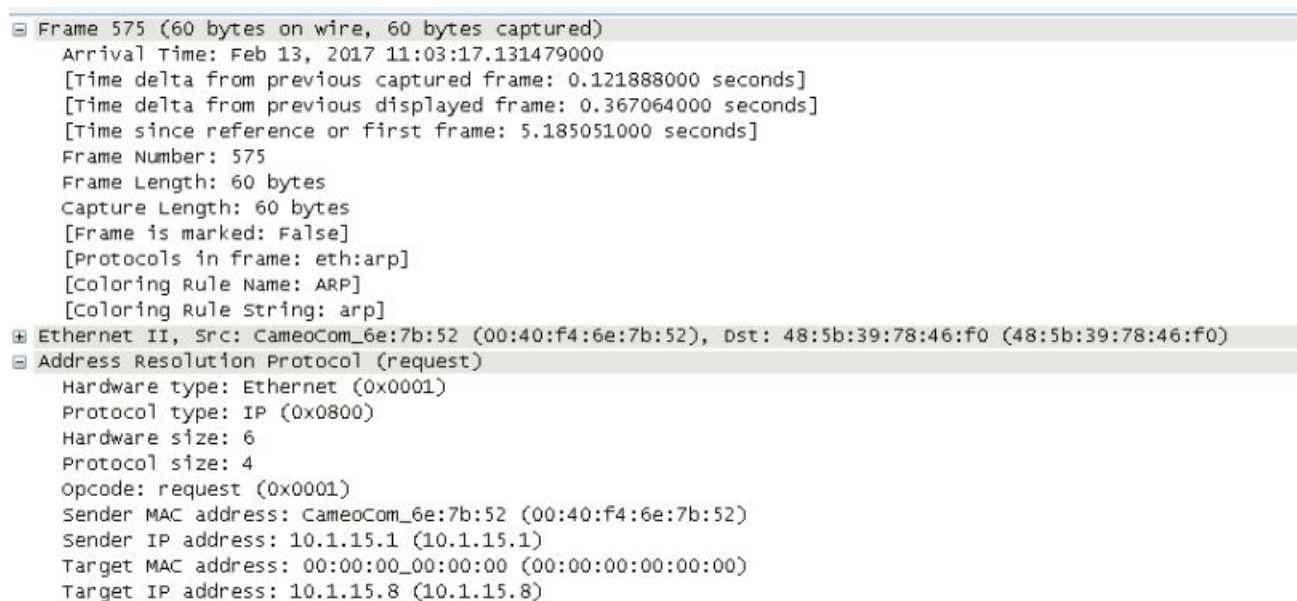


Рис. 15: ARP запрос

ARP-ответ отсылается уже на тот адрес, с которого исходил ARP-запрос. В пакете указывается его

тип (поле Opcode) – ответ, а так же заполненный MAC-адрес цели.

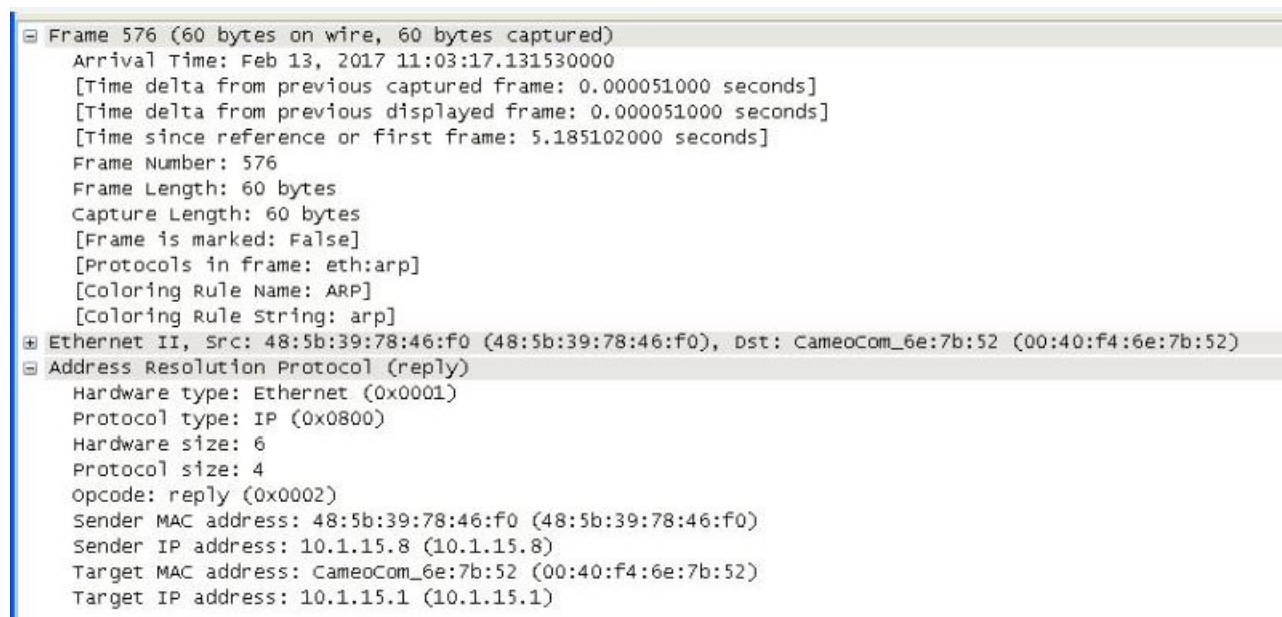


Рис. 16: ARP ответ

4.5 TCP-протокол

4.5.1 Установление соединения

Эта операция происходит следующим образом: Клиент, посылает серверу сегмент с номером последовательности и флагом SYN.

В заголовке TCP-пакета можно увидеть следующие поля:

1. Sequence Number - порядковый номер: 32 бита Порядковый номер первого октета данных в сегменте при отсутствии флага SYN. Если в сегменте присутствует бит SYN, поле номера содержит значение начального порядкового номера (ISN), а первый октет данных имеет номер ISN+1.
2. Acknowledgment Number - номер подтверждения: 32 бита Если бит ACK установлен, это поле содержит значение следующего порядкового номера, который отправитель сегмента ожидает получить. После организации соединения это значение передается всегда.

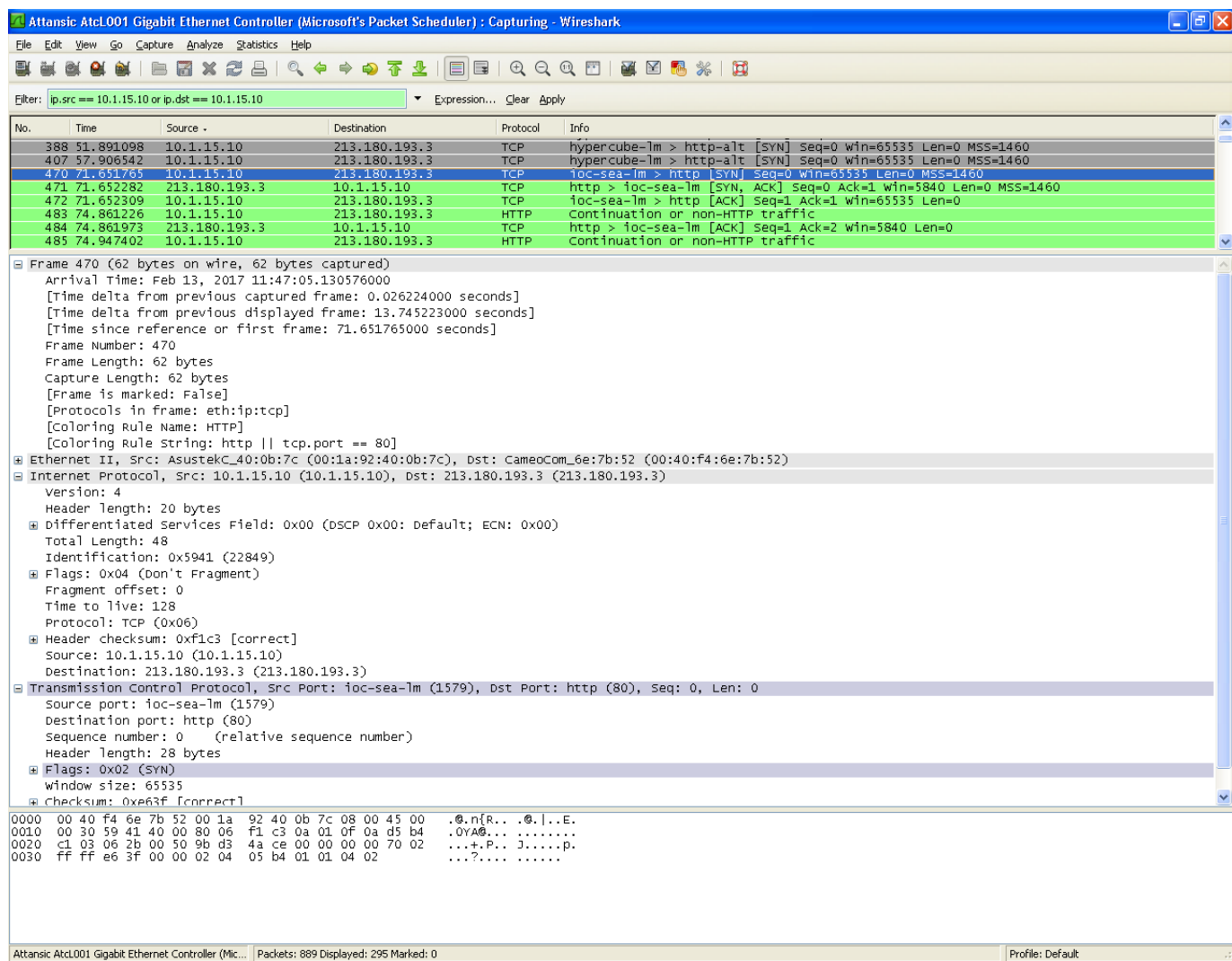


Рис. 17: TCP запрос на установление соединения SYN

Сервер получает сегмент, запоминает номер последовательности и посылает клиенту сегмент с номером последовательности и флагами SYN и ACK.

Frame 471 (62 bytes on wire, 62 bytes captured)	
Arrival Time: Feb 13, 2017 11:47:05.131093000	
[Time delta from previous captured frame: 0.000517000 seconds]	
[Time delta from previous displayed frame: 0.000517000 seconds]	
[Time since reference or first frame: 71.652282000 seconds]	
Frame Number: 471	
Frame Length: 62 bytes	
Capture Length: 62 bytes	
[Frame is marked: False]	
[Protocols in frame: eth:ip:tcp]	
[Coloring Rule Name: HTTP]	
[Coloring Rule String: http tcp.port == 80]	
Ethernet II, Src: CameoCom_6e:7b:52 (00:40:f4:6e:7b:52), Dst: AsustekC_40:0b:7c (00:1a:92:40:0b:7c)	
Internet Protocol, Src: 213.180.193.3 (213.180.193.3), Dst: 10.1.15.10 (10.1.15.10)	
Version: 4	
Header length: 20 bytes	
Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00)	
Total Length: 48	
Identification: 0x0000 (0)	
Flags: 0x04 (Don't Fragment)	
Fragment offset: 0	
Time to live: 64	
Protocol: TCP (0x06)	
Header checksum: 0x8b05 [correct]	
Source: 213.180.193.3 (213.180.193.3)	
Destination: 10.1.15.10 (10.1.15.10)	
Transmission Control Protocol, Src Port: http (80), Dst Port: ioc-sea-lm (1579), Seq: 0, Ack: 1, Len: 0	
Source port: http (80)	
Destination port: ioc-sea-lm (1579)	
Sequence number: 0 (relative sequence number)	
Acknowledgement number: 1 (relative ack number)	
Header length: 28 bytes	
Flags: 0x12 (SYN, ACK)	
window size: 5840	
0000	00 1a 92 40 0b 7c 00 40 f4 6e 7b 52 08 00 45 00 ...@. .@.n{R..E.
0010	00 30 00 00 40 00 40 06 8b 05 d5 b4 c1 03 0a 01 ...@.@.
0020	0f 0a 00 50 06 2b ee 86 77 bd 9b d3 4a cf 70 12 ...P.+.. w...j.p.
0030	16 d0 69 1a 00 00 02 04 05 b4 01 01 04 02 ...i.....

Рис. 18: Ответ сервера на установление TCP-соединения

Если клиент получает сегмент с флагом SYN, то он запоминает номер последовательности и посылает сегмент с флагом ACK.

Frame 2925: 54 bytes on wire (432 bits), 54 bytes captured (432 bits) on interface 0	
Ethernet II, Src: Azurewav_b6:2a:39 (24:0a:64:b6:2a:39), Dst: Tp-LinkT_f2:ff:00 (54:e6:fc:f2:ff:00)	
Internet Protocol Version 4, Src: 192.168.1.124 (192.168.1.124), Dst: 87.240.131.99 (87.240.131.99)	
Version: 4	
Header Length: 20 bytes	
Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00: Not-ECT (Not ECN-Capable Transport))	
Total Length: 40	
Identification: 0x29b0 (10672)	
Flags: 0x02 (Don't Fragment)	
Fragment offset: 0	
Time to live: 128	
Protocol: TCP (6)	
Header checksum: 0x33a8 [validation disabled]	
Source: 192.168.1.124 (192.168.1.124)	
Destination: 87.240.131.99 (87.240.131.99)	
[Source GeoIP: unknown]	
[Destination GeoIP: unknown]	
Transmission Control Protocol, Src Port: 22363 (22363), Dst Port: 80 (80), Seq: 1, Ack: 1, Len: 0	
Source Port: 22363 (22363)	
Destination Port: 80 (80)	
[Stream index: 29]	
[TCP segment Len: 0]	
Sequence number: 1 (relative sequence number)	
Acknowledgment number: 1 (relative ack number)	
Header Length: 20 bytes	
... 0000 0001 0000 = Flags: 0x010 (ACK)	
000. = Reserved: Not set	
...0 = Nonce: Not set	
.... 0... = Congestion window Reduced (CWR): Not set	
.... .0.. = ECN-Echo: Not set	
.... ..0. = Urgent: Not set	
.... ...1 = Acknowledgment: Set	
.... 0... = Push: Not set	
....0.. = Reset: Not set	
....0. = Syn: Not set	
....0 = Fin: Not set	
window size value: 64	
[Calculated window size: 16384]	
[window size scaling factor: 256]	
0000	54 e6 fc f2 ff 00 24 0a 64 b6 2a 39 08 00 45 00 T....\$. d.*9..E.
0010	00 28 29 b0 40 00 80 06 33 a8 c0 a8 01 7c 57 f0 .().@... 3.... w.
0020	83 63 57 5b 00 50 89 30 9d 10 9b b0 51 1d 50 10 .cw[.P.0Q.P.
0030	00 40 a7 62 00 00 ..@.b..

Рис. 19: Подтверждение от клиента о получении ответа

4.5.2 Разрыв соединения

При разрыве соединения сервер отправляет клиенту пакет с установленным флагом RST.

No.	Time	Source	Destination	Protocol	Length	Info
39	6.078598000	191.238.224.150	192.168.1.124	TCP	54	80→22350 [RST, ACK] Seq=1 Ack=1 win=0 Len=0
6190	563.934595000	192.168.1.124	23.77.198.8	TCP	54	22424→443 [RST, ACK] Seq=913 Ack=5913 win=0 Len=0
11280	948.135387000	173.194.33.191	192.168.1.124	TCP	54	443→22462 [RST] Seq=2 win=0 Len=0
11597	987.991450000	87.245.198.35	192.168.1.124	TCP	54	80→22457 [RST] Seq=798 win=0 Len=0
13158	1280.495933000	192.168.1.1	192.168.1.124	TCP	54	1910→22482 [RST, ACK] Seq=1 Ack=1 win=0 Len=0
13162	1281.000311000	192.168.1.1	192.168.1.124	TCP	54	1910→22482 [RST, ACK] Seq=1 Ack=1 win=0 Len=0
13179	1281.548938000	192.168.1.1	192.168.1.124	TCP	54	1910→22482 [RST, ACK] Seq=1 Ack=1 win=0 Len=0
13196	1281.587947000	192.168.1.1	192.168.1.124	TCP	54	1910→22484 [RST, ACK] Seq=1 Ack=1 win=0 Len=0
13263	1288.102745000	192.168.1.1	192.168.1.124	TCP	54	1910→22484 [RST, ACK] Seq=1 Ack=1 win=0 Len=0

⊞	Frame 39: 54 bytes on wire (432 bits), 54 bytes captured (432 bits) on interface 0
⊞	Ethernet II, Src: Tp-LinkT_f2:ff:00 (54:e6:fc:f2:ff:00), Dst: Azurewav_b6:2a:39 (24:0a:64:b6:2a:39)
⊞	Internet Protocol Version 4, Src: 191.238.224.150 (191.238.224.150), Dst: 192.168.1.124 (192.168.1.124)
	Version: 4
	Header Length: 20 bytes
⊞	Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00: Not-ECT (Not ECN-Capable Transport))
	Total Length: 40
	Identification: 0x7135 (28981)
⊞	Flags: 0x02 (Don't Fragment)
	Fragment offset: 0
	Time to live: 108
	Protocol: TCP (6)
⊞	Header checksum: 0x3af1 [validation disabled]
	Source: 191.238.224.150 (191.238.224.150)
	Destination: 192.168.1.124 (192.168.1.124)
	[Source GeoIP: Unknown]
	[Destination GeoIP: Unknown]
⊞	Transmission Control Protocol, Src Port: 80 (80), Dst Port: 22350 (22350), Seq: 1, Ack: 1, Len: 0
	Source Port: 80 (80)
	Destination Port: 22350 (22350)
	[Stream index: 2]
	[TCP Segment Len: 0]
	Sequence number: 1 (relative sequence number)
	Acknowledgment number: 1 (relative ack number)
	Header Length: 20 bytes
⊞	... 0000 0001 0100 = Flags: 0x014 (RST, ACK)
	000. = Reserved: Not set
	...0 = Nonce: Not set
	...0... = Congestion window Reduced (CWR): Not set
0... = ECN-Echo: Not set
0... = Urgent: Not set
1... = Acknowledgment: Set
0... = Push: Not set
⊞1... = Reset: Set
⊞	[Expert Info (Warn/Sequence): Connection reset (RST)]
0... = Syn: Not set
0... = Fin: Not set
	window size value: 0
	[Calculated window size: 0]
	[window size scaling factor: -1 (unknown)]

0000	24 0a 64 b6 2a 39 54 e6 fc f2 ff 00 08 00 45 00	\$.d.*9T.E.
0010	00 28 71 35 40 00 6c 06 3a f1 bf ee e0 96 c0 a8	.(q5@.1.
0020	01 7c 00 50 57 4e e8 b3 80 8e 4f 0d 57 5b 50 14	..PWN.. ..O.W[P.
0030	00 00 e5 dd 00 00

Рис. 20: Пример пакета с флагом RST

4.5.3 Завершение соединения

При завершении соединения происходит обмен пакетами с флагами FIN и ACK.

```

+ Frame 13238: 54 bytes on wire (432 bits), 54 bytes captured (432 bits) on interface 0
+ Ethernet II, Src: D-LinkIn_7c:58:30 (c8:be:19:7c:58:30), Dst: IntelCor_9d:6b:3d (4c:eb:42:9d:6b:3d)
+ Internet Protocol Version 4, Src: 185.26.97.188 (185.26.97.188), Dst: 192.168.1.3 (192.168.1.3)
+ Transmission Control Protocol, Src Port: 80 (80), Dst Port: 51254 (51254), Seq: 1199, Ack: 1695, Len: 0
  Source Port: 80 (80)
  Destination Port: 51254 (51254)
  [Stream index: 258]
  [TCP Segment Len: 0]
  Sequence number: 1199 (relative sequence number)
  Acknowledgment number: 1695 (relative ack number)
  Header Length: 20 bytes
  ... 0000 0001 0001 = Flags: 0x011 (FIN, ACK)
    000. .... = Reserved: Not set
    ...0 .... = Nonce: Not set
    .... 0... = Congestion window Reduced (CWR): Not set
    .... .0.. = ECN-Echo: Not set
    .... ..0. = Urgent: Not set
    .... ...1 = Acknowledgment: Set
    .... .... 0... = Push: Not set
    .... ..... 0.. = Reset: Not set
    .... .... ..0. = Syn: Not set
  + .... .... ...1 = Fin: Set
  window size value: 1125
  [Calculated window size: 18000]
  [window size scaling factor: 16]
+ checksum: 0xf23b [validation disabled]
urgent pointer: 0

```

Рис. 21: Передача пакета с флагами FIN и ACK от сервера клиенту

Сервер посылает клиенту пакет с установленными флагами ACK, FIN. Сервер переходит из состояния ESTABLISHED в состояние FIN-WAIT-1.

```

+ Frame 13239: 54 bytes on wire (432 bits), 54 bytes captured (432 bits) on interface 0
+ Ethernet II, Src: IntelCor_9d:6b:3d (4c:eb:42:9d:6b:3d), Dst: D-LinkIn_7c:58:30 (c8:be:19:7c:58:30)
+ Internet Protocol Version 4, Src: 192.168.1.3 (192.168.1.3), Dst: 185.26.97.188 (185.26.97.188)
+ Transmission Control Protocol, Src Port: 51254 (51254), Dst Port: 80 (80), Seq: 1695, Ack: 1200, Len: 0
  Source Port: 51254 (51254)
  Destination Port: 80 (80)
  [Stream index: 258]
  [TCP Segment Len: 0]
  Sequence number: 1695 (relative sequence number)
  Acknowledgment number: 1200 (relative ack number)
  Header Length: 20 bytes
  ... 0000 0001 0000 = Flags: 0x010 (ACK)
    000. .... = Reserved: Not set
    ...0 .... = Nonce: Not set
    .... 0... = Congestion window Reduced (CWR): Not set
    .... .0.. = ECN-Echo: Not set
    .... ..0. = Urgent: Not set
    .... ...1 = Acknowledgment: Set
    .... .... 0... = Push: Not set
    .... ..... 0.. = Reset: Not set
    .... .... ..0. = Syn: Not set
    .... .... ...0 = Fin: Not set
  window size value: 7892
  [Calculated window size: 31568]
  [window size scaling factor: 4]
+ Checksum: 0xd7cc [validation disabled]
urgent pointer: 0
+ [SEQ/ACK analysis]

```

Рис. 22: Подтверждение получения пакета

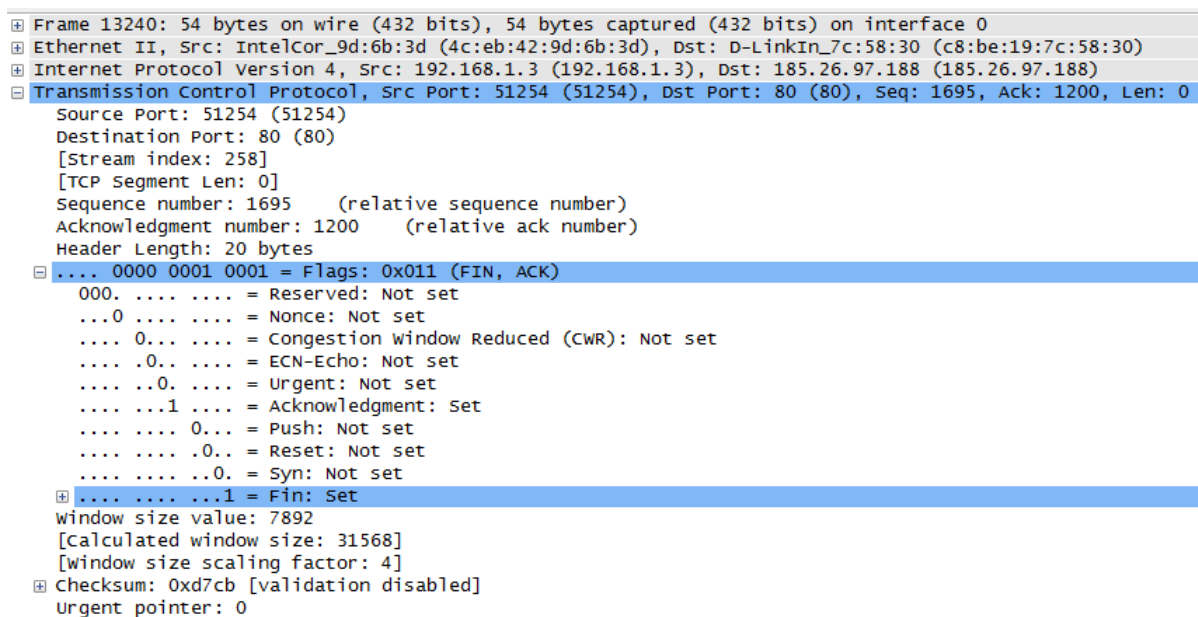


Рис. 23: Подтверждение завершения соединения от клиента серверу

Клиент посылает второй пакет с флагами FIN, ACK, после его отсылки клиент переходит в состояние LAST-ACK, а сервер в состояние TIME-WAIT.

4.5.4 Установка соединения с отсутствующим портом

При попытке подключения к отсутствующему порту, не приходит ACK и RST, поэтому клиент находится в подвешенном состоянии и ожидает ответа.

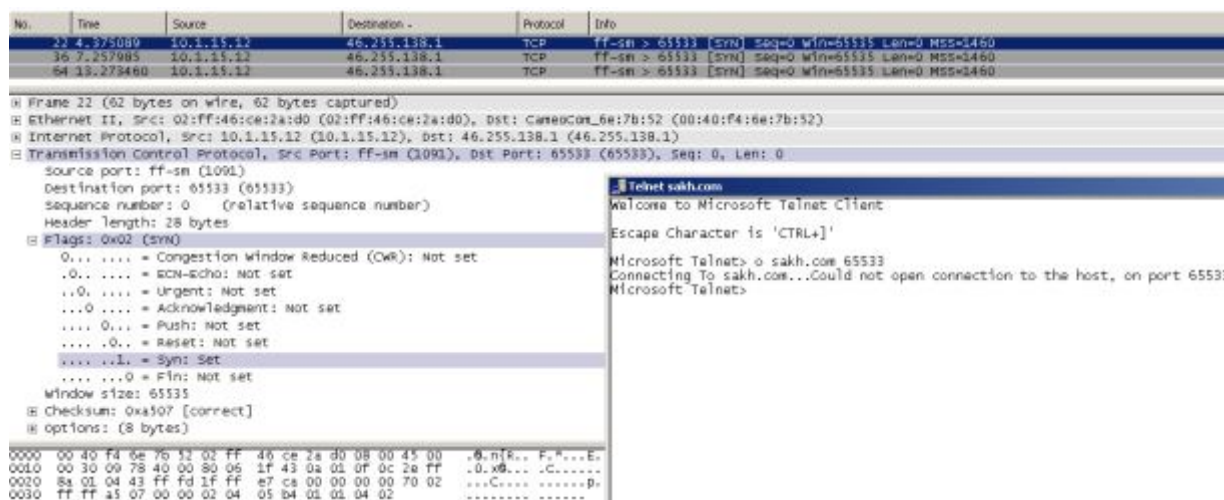


Рис. 24: Попытка tcp - соединения на sakh.com:65533

5 Выводы

В ходе работы были получены навыки работы в программе Wireshark и закреплены знания о сетевых протоколах ARP, ICMP, TCP. Были рассмотрены:

1. работу утилит ping и tracert;
2. работа ARP-протокола;
3. работа протокола ICMP, включая такие типовые случаи, как: отправка фрагментированного пакета, возникновение ошибки 3.1, трассировка маршрута;
4. установка, разрыв и завершение TCP соединения;