

САНКТ-ПЕТЕРБУРГСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ ПЕТРА ВЕЛИКОГО

КАФЕДРА КОМПЬЮТЕРНЫХ СИСТЕМ И ПРОГРАММНЫХ ТЕХНОЛОГИЙ

Отчёт по лабораторной работе №2

Курс: «Защита информации»

Тема: «Исследование сетевого трафика протокола FTP»

Выполнил студент:

Волкова М.Д.

Группа: 43501/3

Проверил:

Новопашенный А.Г.

Санкт-Петербург
2018 г.

Содержание

1	Лабораторная работа №2	2
1.1	Цель работы	2
1.2	Программа работы	2
1.3	Конфигурация сети	2
1.4	Ход работы	2
1.4.1	Протокол FTP	2
1.4.2	Установление управляющего соединения	3
1.4.3	Аутентификация	3
1.4.4	Активный режим FTP	4
1.4.5	Пассивный режим FTP	5
1.5	Вывод	10

Лабораторная работа №2

1.1 Цель работы

Получение навыков по исследованию сетевого трафика.

1.2 Программа работы

При помощи программы Wireshark продемонстрировать сетевой трафик для:

- Протокола FTP
 - В пассивном режиме
 - В активном режиме
 - Установление соединения и авторизация

1.3 Конфигурация сети

```
Настройка протокола IP для Windows

Адаптер Ethernet Ethernet:

    DNS-суффикс подключения . . . . . : 
    Локальный IPv6-адрес канала . . . : fe80::7c09:33ce:9061:fd1f%5
    IPv4-адрес. . . . . : 192.168.0.110
    Маска подсети . . . . . : 255.255.255.0
    Основной шлюз. . . . . : 192.168.0.1

Туннельный адаптер Teredo Tunneling Pseudo-Interface:

    DNS-суффикс подключения . . . . . : 
    IPv6-адрес. . . . . : 2001:0:9d38:6ab8:3cdb:14bd:430d:9231
    Локальный IPv6-адрес канала . . . : fe80::3cdb:14bd:430d:9231%24
    Основной шлюз. . . . . : ::
```

Рис. 1.1: Сетевые параметры компьютера

1.4 Ход работы

1.4.1 Протокол FTP

В отличие от большинства других протоколов, при работе с протоколом FTP создается два типа соединений, первое служит для аутентификации и передачи команд, второе — для передачи данных. По второму соединению определяется режим работы, он может быть активным, а может быть пассивным. Отличаются

они между собой стороной выступающей инициатором подключения для передачи данных и портами, на которых эта передача собственно производится. При нормальном или активном FTP, управляющее соединение иницируется со стороны клиента, а подключение для передачи данных иницируется со стороны сервера. В пассивном режиме, как управляющее соединение так и соединение для передачи данных иницируется клиентом.

Для демонстрации работы протокола FTP в активном режиме был выбран сервер ftp://ftp.funet.fi, а для пассивного - sourceware.org.

1.4.2 Установление управляющего соединения

Первый этап - установление TCP соединения с 21 портом сервера (управляющее соединение):

3062	117.813393	192.168.0.110	193.166.3.2	TCP	66 49807 → 21 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=1 SACK_PERM=1
3063	117.832306	193.166.3.2	192.168.0.110	TCP	66 21 → 49807 [SYN, ACK] Seq=0 Ack=1 Win=49640 Len=0 MSS=1460 WS=1 SACK_PERM=1
3064	117.832372	192.168.0.110	193.166.3.2	TCP	54 49807 → 21 [ACK] Seq=1 Ack=1 Win=8192 Len=0

Рис. 1.2: Установление TCP соединения с 21 портом сервера

Аналогичное установление соединения происходит с потоком данных после аутентификации и перехода в активный/пассивный режим.

1.4.3 Аутентификация

Следующий этап - аутентификация клиента с помощью имени пользователя и пароля. Имя пользователя отправляется на сервер командой USER, а пароль - командой PASS.

Хост, обеспечивающий FTP-сервис, может предоставить анонимный доступ к FTP. Пользователи обычно входят в систему как «anonymous» в качестве имени пользователя. Хотя обычно пользователей просят прислать адрес их электронной почты вместо пароля, никакой проверки фактически не производится. Многие FTP-хосты, предоставляющие обновления программного обеспечения, поддерживают анонимный доступ.

3106	134.879146	192.168.0.110	193.166.3.2	FTP	70 Request: USER anonymous
Frame 3106: 70 bytes on wire (560 bits), 70 bytes captured (560 bits) on interface 0 Ethernet II, Src: Micro-St_10:cc:bf (00:24:21:10:cc:bf), Dst: Tp-LinkT_91:26:1c (64:70:02:91:26:1c) Internet Protocol Version 4, Src: 192.168.0.110, Dst: 193.166.3.2 Transmission Control Protocol, Src Port: 49807, Dst Port: 21, Seq: 15, Ack: 318, Len: 16 File Transfer Protocol (FTP) > USER anonymous\r\n					

Рис. 1.3: Отправка имени пользователя командой USER

3108	134.899061	193.166.3.2	192.168.0.110	FTP	1465 Response: 331-Welcome to the FUNET anonymous ftp archive
Frame 3108: 1465 bytes on wire (11720 bits), 1465 bytes captured (11720 bits) on interface 0 Ethernet II, Src: Tp-LinkT_91:26:1c (64:70:02:91:26:1c), Dst: Micro-St_10:cc:bf (00:24:21:10:cc:bf) Internet Protocol Version 4, Src: 193.166.3.2, Dst: 192.168.0.110 Transmission Control Protocol, Src Port: 21, Dst Port: 49807, Seq: 318, Ack: 31, Len: 1411 File Transfer Protocol (FTP) 331-Welcome to the FUNET anonymous ftp archive\r\n Response code: User name okay, need password (331) Response arg: Welcome to the FUNET anonymous ftp archive					

Рис. 1.4: Реакция сервера на правильное имя пользователя

Реакция сервера на имя пользователя - это FTP пакет с кодом 331 (имя пользователя корректно). Если имя пользователя не существует, то специальный FTP пакет с кодом ошибки не посылается (только на следующем этапе проверки пары имени пользователя и пароля). Это сделано для того, чтобы клиент не мог определить какие пользователи существуют на сервере.

```

3112 137.968402 192.168.0.110 193.166.3.2 FTP 64 Request: PASS lol
Frame 3112: 64 bytes on wire (512 bits), 64 bytes captured (512 bits) on interface 0
Ethernet II, Src: Micro-St_10:cc:bf (00:24:21:10:cc:bf), Dst: Tp-LinkT_91:26:1c (64:70:02:91:26:1c)
Internet Protocol Version 4, Src: 192.168.0.110, Dst: 193.166.3.2
Transmission Control Protocol, Src Port: 49807, Dst Port: 21, Seq: 31, Ack: 1729, Len: 10
File Transfer Protocol (FTP)
  PASS lol\r\n
    Request command: PASS
    Request arg: lol

```

Рис. 1.5: Отправка пароля командой PASS

```

3113 137.987347 193.166.3.2 192.168.0.110 FTP 82 Response: 230 Any password will work
Frame 3113: 82 bytes on wire (656 bits), 82 bytes captured (656 bits) on interface 0
Ethernet II, Src: Tp-LinkT_91:26:1c (64:70:02:91:26:1c), Dst: Micro-St_10:cc:bf (00:24:21:10:cc:bf)
Internet Protocol Version 4, Src: 193.166.3.2, Dst: 192.168.0.110
Transmission Control Protocol, Src Port: 21, Dst Port: 49807, Seq: 1729, Ack: 41, Len: 28
File Transfer Protocol (FTP)
  230 Any password will work\r\n
    Response code: User logged in, proceed (230)
    Response arg: Any password will work

```

Рис. 1.6: Реакция сервера на правильный пароль

Если пара имя пользователь и пароль правильная, то сервер возвращает FTP пакет с кодом 230 (пользователь идентифицирован), если нет, то 530 (вход не выполнен).

Стоит отметить, что ни имя пользователя, ни пароль не зашифровываются.

1.4.4 Активный режим FTP

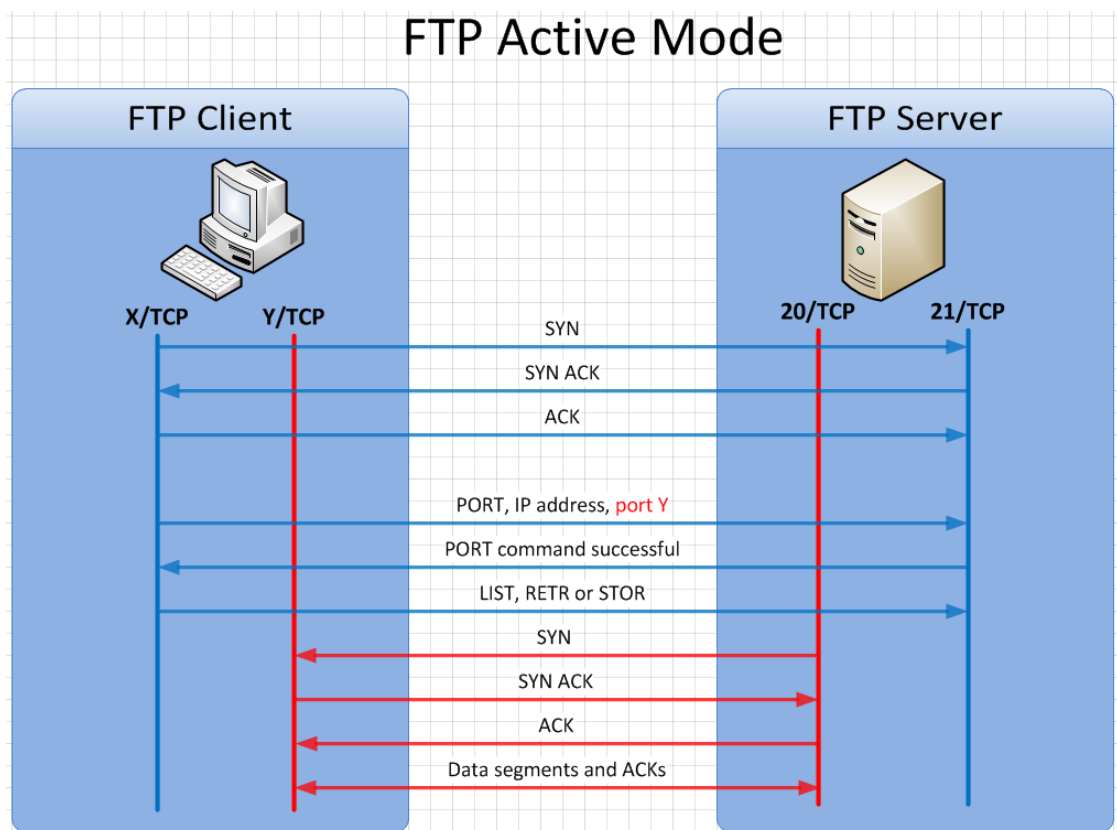


Рис. 1.7: Клиент-серверное взаимодействие в активном режиме FTP

Для перехода в активный режим клиент явно указывает собственный IP адрес (первые 4 байта аргумента) и порт подключения (последние 2 байта аргумента) командой PORT:

В случае успешного перехода в активный режим сервер возвращает FTP пакет с кодом 200 (команда корректна). После этого становятся доступными команды LIST, RETR, STORE и др.

Запросим список файлов в текущей директории командой ls:

Заметим, что установление соединения канала данных инициируется сервером с 20 порта.

3131	142.823477	192.168.0.110	193.166.3.2	FTP	82 Request: PORT 192,168,0,110,194,144
> Frame 3131: 82 bytes on wire (656 bits), 82 bytes captured (656 bits) on interface 0 > Ethernet II, Src: Micro-St_10:cc:bf (00:24:21:10:cc:bf), Dst: Tp-LinkT_91:26:1c (64:70:02:91:26:1c) > Internet Protocol Version 4, Src: 192.168.0.110, Dst: 193.166.3.2 > Transmission Control Protocol, Src Port: 49807, Dst Port: 21, Seq: 41, Ack: 1757, Len: 28 > File Transfer Protocol (FTP)					
> PORT 192,168,0,110,194,144\r\n Request command: PORT Request arg: 192,168,0,110,194,144 Active IP address: 192.168.0.110 Active port: 49808					

Рис. 1.8: Переход в активный режим

3132	142.842646	193.166.3.2	192.168.0.110	FTP	83 Response: 200 PORT command successful
Frame 3132: 83 bytes on wire (664 bits), 83 bytes captured (664 bits) on interface 0 Ethernet II, Src: Tp-LinkT_91:26:1c (64:70:02:91:26:1c), Dst: Micro-St_10:cc:bf (00:24:21:10:cc:bf) Internet Protocol Version 4, Src: 193.166.3.2, Dst: 192.168.0.110 Transmission Control Protocol, Src Port: 21, Dst Port: 49807, Seq: 1757, Ack: 69, Len: 29 File Transfer Protocol (FTP)					
> 200 PORT command successful\r\n Response code: Command okay (200) Response arg: PORT command successful					

Рис. 1.9: Реакция сервера на успешный переход в активный режим

3133	142.847472	192.168.0.110	193.166.3.2	FTP	60 Request: NLST
3134	142.866410	193.166.3.2	192.168.0.110	TCP	66 20 → 49808 [SYN] Seq=0 Win=49640 Len=0 MSS=1460 WS=1 SACK_
3135	142.951995	193.166.3.2	192.168.0.110	TCP	60 21 → 49807 [ACK] Seq=1786 Ack=75 Win=49640 Len=0
3137	144.001936	193.166.3.2	192.168.0.110	TCP	66 [TCP Retransmission] 20 → 49808 [SYN] Seq=0 Win=49640 Len=0
3141	146.272006	193.166.3.2	192.168.0.110	TCP	66 [TCP Retransmission] 20 → 49808 [SYN] Seq=0 Win=49640 Len=0
3147	150.791853	193.166.3.2	192.168.0.110	TCP	66 [TCP Retransmission] 20 → 49808 [SYN] Seq=0 Win=49640 Len=0
3148	150.792009	192.168.0.110	193.166.3.2	TCP	66 49808 → 20 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460
3149	150.810734	193.166.3.2	192.168.0.110	TCP	60 20 → 49808 [ACK] Seq=1 Ack=1 Win=49640 Len=0
3150	150.810899	193.166.3.2	192.168.0.110	FTP	84 Response: 150 Connecting to port 49808
3151	150.811162	193.166.3.2	192.168.0.110	FTP-DA...	121 FTP Data: 67 bytes
3152	150.811184	193.166.3.2	192.168.0.110	TCP	60 20 → 49808 [FIN, ACK] Seq=68 Ack=1 Win=49640 Len=0
3153	150.811205	192.168.0.110	193.166.3.2	TCP	54 49808 → 20 [ACK] Seq=1 Ack=69 Win=65536 Len=0
3154	150.811218	193.166.3.2	192.168.0.110	FTP	75 Response: 226 9 matches total
3155	150.811240	192.168.0.110	193.166.3.2	TCP	54 49807 → 21 [ACK] Seq=75 Ack=1837 Win=8084 Len=0
3156	150.823485	192.168.0.110	193.166.3.2	TCP	54 49808 → 20 [FIN, ACK] Seq=1 Ack=69 Win=65536 Len=0
3157	150.842213	193.166.3.2	192.168.0.110	TCP	60 20 → 49808 [ACK] Seq=69 Ack=2 Win=49640 Len=0

Рис. 1.10: Запрос списка файлов

1.4.5 Пассивный режим FTP

Для подключения к sourceware.org воспользуемся утилитой telnet.
Данные в telnet передаются посимвольно:

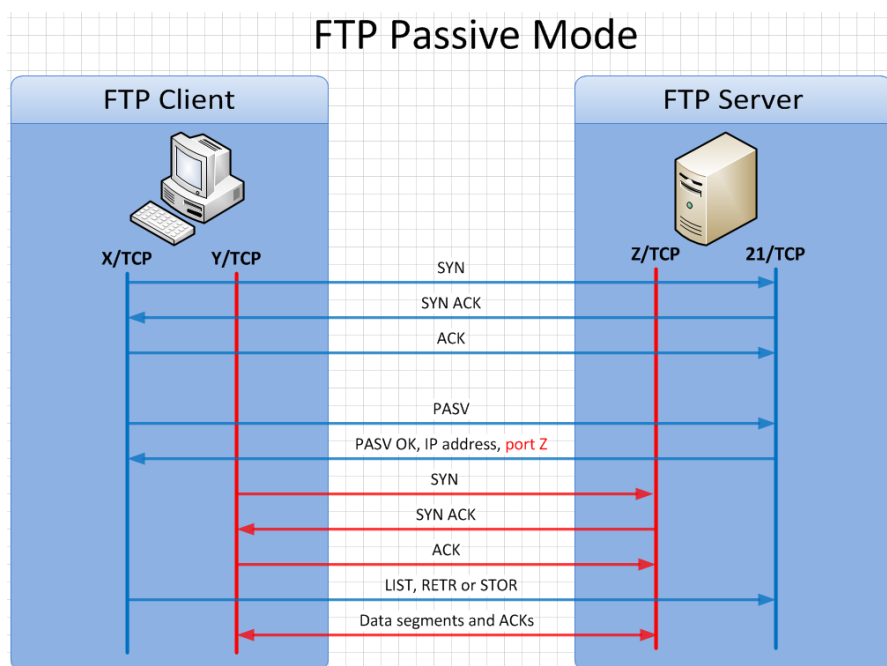


Рис. 1.11: Клиент-серверное взаимодействие в пассивном режиме FTP

6864	638.896522	192.168.0.110	209.132.180.131	TCP	54 49819 → 21 [ACK] Seq=1 ACK=24 Win=66560 Len=0
6870	642.391696	192.168.0.110	209.132.180.131	FTP	55 Request: U
6871	642.568592	209.132.180.131	192.168.0.110	TCP	60 21 → 49819 [ACK] Seq=24 Ack=2 Win=14720 Len=0
6872	642.568659	192.168.0.110	209.132.180.131	FTP	55 Request: S
6873	642.745406	209.132.180.131	192.168.0.110	TCP	60 21 → 49819 [ACK] Seq=24 Ack=3 Win=14720 Len=0
6874	642.823252	192.168.0.110	209.132.180.131	FTP	55 Request: E
6875	643.000013	209.132.180.131	192.168.0.110	TCP	60 21 → 49819 [ACK] Seq=24 Ack=4 Win=14720 Len=0
6876	643.023334	192.168.0.110	209.132.180.131	FTP	55 Request: R
6877	643.200093	209.132.180.131	192.168.0.110	TCP	60 21 → 49819 [ACK] Seq=24 Ack=5 Win=14720 Len=0
6880	643.807510	192.168.0.110	209.132.180.131	FTP	55 Request:
6881	643.984291	209.132.180.131	192.168.0.110	TCP	60 21 → 49819 [ACK] Seq=24 Ack=6 Win=14720 Len=0
6897	645.599687	192.168.0.110	209.132.180.131	FTP	55 Request: a
6899	645.776525	209.132.180.131	192.168.0.110	TCP	60 21 → 49819 [ACK] Seq=24 Ack=7 Win=14720 Len=0
6900	645.783229	192.168.0.110	209.132.180.131	FTP	55 Request: n
6901	645.960117	209.132.180.131	192.168.0.110	TCP	60 21 → 49819 [ACK] Seq=24 Ack=8 Win=14720 Len=0
6902	646.495422	192.168.0.110	209.132.180.131	FTP	55 Request: o
6903	646.672624	209.132.180.131	192.168.0.110	TCP	60 21 → 49819 [ACK] Seq=24 Ack=9 Win=14720 Len=0
6904	646.879300	192.168.0.110	209.132.180.131	FTP	55 Request: n
6906	647.056092	209.132.180.131	192.168.0.110	TCP	60 21 → 49819 [ACK] Seq=24 Ack=10 Win=14720 Len=0
6908	647.319433	192.168.0.110	209.132.180.131	FTP	55 Request: y
6909	647.496273	209.132.180.131	192.168.0.110	TCP	60 21 → 49819 [ACK] Seq=24 Ack=11 Win=14720 Len=0
6911	647.807372	192.168.0.110	209.132.180.131	FTP	55 Request: m
6912	647.984084	209.132.180.131	192.168.0.110	TCP	60 21 → 49819 [ACK] Seq=24 Ack=12 Win=14720 Len=0
6914	648.343559	192.168.0.110	209.132.180.131	FTP	55 Request: o
6915	648.520372	209.132.180.131	192.168.0.110	TCP	60 21 → 49819 [ACK] Seq=24 Ack=13 Win=14720 Len=0
6916	648.591266	192.168.0.110	209.132.180.131	FTP	55 Request: u
6917	648.768007	209.132.180.131	192.168.0.110	TCP	60 21 → 49819 [ACK] Seq=24 Ack=14 Win=14720 Len=0
6918	648.775736	192.168.0.110	209.132.180.131	FTP	55 Request: s
6919	648.952457	209.132.180.131	192.168.0.110	TCP	60 21 → 49819 [ACK] Seq=24 Ack=15 Win=14720 Len=0
6920	649.255459	192.168.0.110	209.132.180.131	FTP	56 Request:

Рис. 1.12: Посимвольная передача логина

Для перехода в пассивный режим клиент отправляет на сервер FTP пакет с единственной командой PASV. В ответ сервер посылает FTP пакет с кодом 227 (переход в пассивный режим) и парой значений: IP адрес (первые 4 байта аргумента) и порт подключения (последние 2 байта аргумента):

12493	1133.071708	209.132.180.131	192.168.0.110	FTP	107 Response: 227 Entering Passive Mode (209,132,180,131,39,23)
> Frame 12493: 107 bytes on wire (856 bits), 107 bytes captured (856 bits) on interface 0					
> Ethernet II, Src: Tp-LinkT_91:26:1c (64:70:02:91:26:1c), Dst: Micro-St_10:cc:bf (00:24:21:10:cc:bf)					
> Internet Protocol Version 4, Src: 209.132.180.131, Dst: 192.168.0.110					
> Transmission Control Protocol, Src Port: 21, Dst Port: 49822, Seq: 1631, Ack: 123, Len: 53					
v File Transfer Protocol (FTP)					
v 227 Entering Passive Mode (209,132,180,131,39,234).\r\n					
Response code: Entering Passive Mode (227)					
Response arg: Entering Passive Mode (209,132,180,131,39,234).					
Passive IP address: 209.132.180.131					
Passive port: 10218					

Рис. 1.13: Переход в пассивный режим

Затем клиент инициирует установление соединения с произвольного порта на серверный порт, указанный в ответе на команду PASV ранее. Со второго терминала подключимся по полученному адресу:

12639	1171.172042	192.168.0.110	209.132.180.131	TCP	66 49910 → 10218 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
12644	1171.348473	209.132.180.131	192.168.0.110	TCP	66 10218 → 49910 [SYN, ACK] Seq=0 Ack=1 Win=14600 Len=0 MSS=1360 SACK_PERM=1 WS=128
12645	1171.348549	192.168.0.110	209.132.180.131	TCP	54 49910 → 10218 [ACK] Seq=1 Ack=1 Win=66560 Len=0
> Frame 12639: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface 0					
> Ethernet II, Src: Micro-St_10:cc:bf (00:24:21:10:cc:bf), Dst: Tp-LinkT_91:26:1c (64:70:02:91:26:1c)					
> Internet Protocol Version 4, Src: 192.168.0.110, Dst: 209.132.180.131					
> Transmission Control Protocol, Src Port: 49910, Dst Port: 10218, Seq: 0, Len: 0					

Рис. 1.14: Установление соединения по полученному адресу

Запросим данные по управляющему каналу. Сервер передаст их по каналу данных и инициирует закрытие соединения.

13135	1254.068403	209.132.180.131	192.168.0.110	FTP	118 Response: 150 Opening ASCII mode data connection for md5
13136	1254.072220	209.132.180.131	192.168.0.110	FTP-DA...	751 FTP Data: 697 bytes
13137	1254.072249	209.132.180.131	192.168.0.110	TCP	60 10218 → 49910 [FIN, ACK] Seq=698 Ack=1 Win=14720 Len=0
13138	1254.072295	192.168.0.110	209.132.180.131	TCP	54 49910 → 10218 [ACK] Seq=1 Ack=699 Win=65792 Len=0
13139	1254.078841	192.168.0.110	209.132.180.131	TCP	54 49910 → 10218 [FIN, ACK] Seq=1 Ack=699 Win=65792 Len=0
13140	1254.115589	192.168.0.110	209.132.180.131	TCP	54 49822 → 21 [ACK] Seq=210 Ack=1966 Win=66048 Len=0
13142	1254.251820	209.132.180.131	192.168.0.110	FTP	77 Response: 226 Transfer complete
13143	1254.255096	209.132.180.131	192.168.0.110	TCP	60 10218 → 49910 [ACK] Seq=699 Ack=2 Win=14720 Len=0
13145	1254.303010	192.168.0.110	209.132.180.131	TCP	54 49822 → 21 [ACK] Seq=210 Ack=1989 Win=66048 Len=0
13156	1259.285269	192.168.0.110	209.132.180.131	TCP	54 49822 → 21 [FIN, ACK] Seq=210 Ack=1989 Win=66048 Len=0
13157	1259.457983	209.132.180.131	192.168.0.110	TCP	60 21 → 49822 [FIN, ACK] Seq=1989 Ack=211 Win=14720 Len=0
13158	1259.458068	192.168.0.110	209.132.180.131	TCP	54 49822 → 21 [ACK] Seq=211 Ack=1990 Win=66048 Len=0

Рис. 1.15: Передача данных

Финальным этапом является завершение соединения канала данных: клиенту отправляется сообщение с кодом 226 (закрытие канала, обмен завершен успешно), а канал данных обменивается флагами FIN и соединение завершается.


```

> Ethernet II, Src: Tp-LinkT_91:26:1c (64:70:02:91:26:1c), Dst: Micro-St_10:cc:bf (00:24:21:10:cc:bf)
✓ Internet Protocol Version 4, Src: 209.132.180.131, Dst: 192.168.0.110
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
    > Differentiated Services Field: 0x28 (DSCP: AF11, ECN: Not-ECT)
        Total Length: 737
        Identification: 0xa2a2 (41634)
    > Flags: 0x02 (Don't Fragment)
        Fragment offset: 0
        Time to live: 41
        Protocol: TCP (6)
        Header checksum: 0x652e [validation disabled]
        [Header checksum status: Unverified]
        Source: 209.132.180.131
        Destination: 192.168.0.110
        [Source GeoIP: Unknown]
        [Destination GeoIP: Unknown]
✓ Transmission Control Protocol, Src Port: 10218, Dst Port: 49910, Seq: 1, Ack: 1, Len: 697
    Source Port: 10218
    Destination Port: 49910
    [Stream index: 128]
    [TCP Segment Len: 697]
    Sequence number: 1 (relative sequence number)
    [Next sequence number: 698 (relative sequence number)]
    Acknowledgment number: 1 (relative ack number)
    0101 .... = Header Length: 20 bytes (5)
    > Flags: 0x018 (PSH, ACK)
        Window size value: 115
        [Calculated window size: 14720]
        [Window size scaling factor: 128]
        Checksum: 0x8b4b [unverified]
        [Checksum Status: Unverified]
        Urgent pointer: 0
    > [SEQ/ACK analysis]
        TCP payload (697 bytes)
    FTP Data (c966dc72304c5e0fc0ad6694cd8685f7 autoconf-2.10-2.11.diff.gz\r\nfe332d45a554c81bd5a1a758ea2c53be

```

Рис. 1.16: Пакет с запрошенными данными

```

> Ethernet II, Src: Tp-LinkT_91:26:1c (64:70:02:91:26:1c), Dst: Micro-St_10:cc:bf (00:24:21:10:cc:bf)
v Internet Protocol Version 4, Src: 209.132.180.131, Dst: 192.168.0.110
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
    > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
        Total Length: 63
        Identification: 0xd7c3 (55235)
    > Flags: 0x02 (Don't Fragment)
        Fragment offset: 0
        Time to live: 41
        Protocol: TCP (6)
        Header checksum: 0x32d7 [validation disabled]
        [Header checksum status: Unverified]
        Source: 209.132.180.131
        Destination: 192.168.0.110
        [Source GeoIP: Unknown]
        [Destination GeoIP: Unknown]
v Transmission Control Protocol, Src Port: 21, Dst Port: 49822, Seq: 1966, Ack: 210, Len: 23
    Source Port: 21
    Destination Port: 49822
    [Stream index: 39]
    [TCP Segment Len: 23]
    Sequence number: 1966 (relative sequence number)
    [Next sequence number: 1989 (relative sequence number)]
    Acknowledgment number: 210 (relative ack number)
    0101 .... = Header Length: 20 bytes (5)
    > Flags: 0x018 (PSH, ACK)
        Window size value: 115
        [Calculated window size: 14720]
        [Window size scaling factor: 128]
        Checksum: 0xb0a1 [unverified]
        [Checksum Status: Unverified]
        Urgent pointer: 0
    > [SEQ/ACK analysis]
        TCP payload (23 bytes)
v File Transfer Protocol (FTP)
    v 226 Transfer complete\r\n
        Response code: Closing data connection (226)
        Response arg: Transfer complete

```

Рис. 1.17: Пакет с кодом закрытия канала

1.5 Вывод

В ходе работы был исследован сетевой трафик протокола FTP в активном и пассивном режиме.

Протокол FTP не безопасен, потому что не поддерживает шифрование данных. Это обусловлено тем, что во времена создания протокола проблема защиты данных не была так актуальна. Для решения проблемы безопасности были созданы защищенные вариации FTP, такие как:

- FTPS
- SFTP
- FTP через SSH

В большинстве случаев используется пассивный режим FTP соединения. Это обусловлено тем, что в пассивном режиме все соединения инициирует клиент и поэтому к нему нет никаких требований, он может находиться за NAT и брандмауэром, а также не иметь выделенного IP-адреса.

В активном режиме основная проблема возникает у клиента. Если брандмауэр настроен отбрасывать не инициированные изнутри входящие соединения, то сервер не сможет установить соединение для передачи данных. А так как порт для данных является динамическим, то возникают определенные сложности с настройкой брандмауэра. Наиболее правильным будет указать в клиенте диапазон используемых портов и создать для них разрешающее правило брандмауэра.