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:

Ерниязов Т.Е.
Группа: 43501/3

Проверил:

Новопашенный А.Г.

Санкт-Петербург
2018 г.

# Содержание

# 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

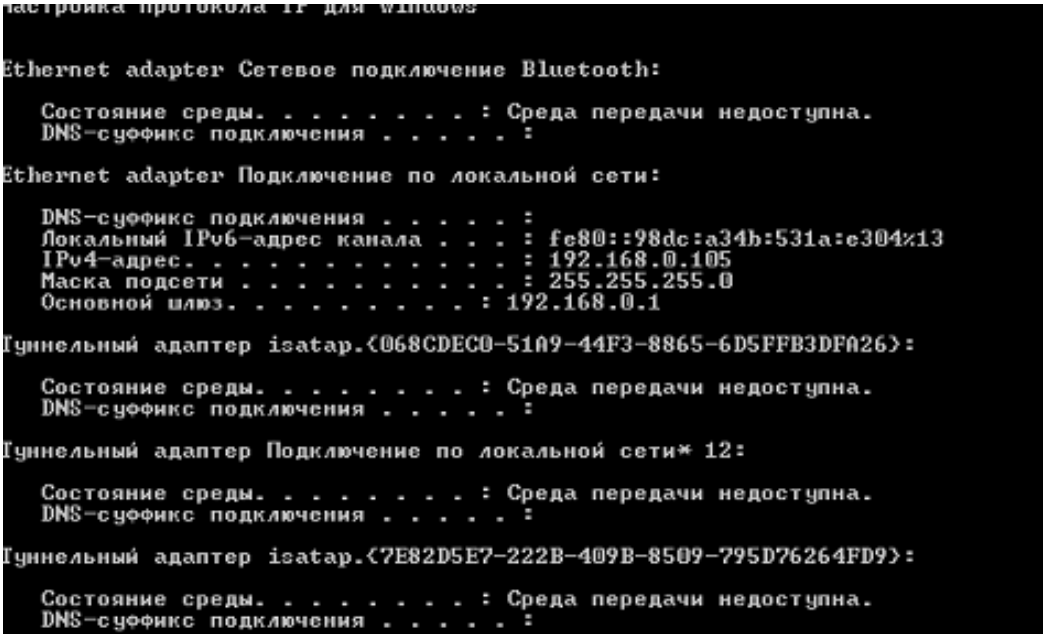


Рис. 1.1: Сетевые параметры компьютера

## 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 производиться. При нормальном или активном FTP, управляющее соединение инициируется со стороны клиента, а подключение для передачи данных инициируется со стороны сервера. В пассивном режиме, как управляющее соединение так и соединение для передачи данных инициируется 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):


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.

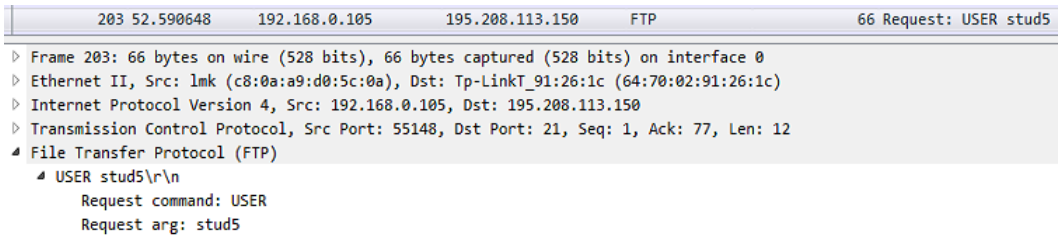


Рис. 1.3: Отправка имени пользователя командой USER

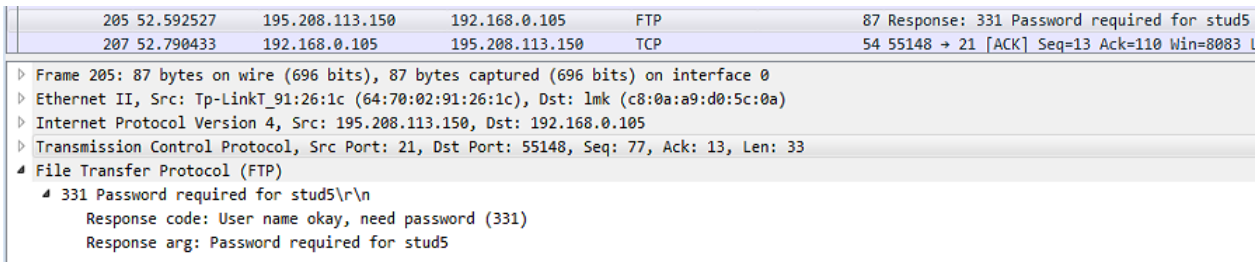


Рис. 1.4: Реакция сервера на правильное имя пользователя

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.

```
327 87.230397    192.168.0.105    195.208.113.150    FTP    68 Request: PASS NFhtkrf
Frame 327: 68 bytes on wire (544 bits), 68 bytes captured (544 bits) on interface 0
Ethernet II, Src: lmk (c8:0a:a9:d0:5c:0a), Dst: Tp-LinkT_91:26:1c (64:70:02:91:26:1c)
Internet Protocol Version 4, Src: 192.168.0.105, Dst: 195.208.113.150
Transmission Control Protocol, Src Port: 55148, Dst Port: 21, Seq: 13, Ack: 110, Len: 14
File Transfer Protocol (FTP)
  PASS NFhtkrf\r\n
      Request command: PASS
      Request arg: NFhtkrf
```

Рис. 1.5: Отправка пароля командой PASS

```
328 87.242533    195.208.113.150    192.168.0.105    FTP    80 Response: 230 User stud5 logged in
Frame 328: 80 bytes on wire (640 bits), 80 bytes captured (640 bits) on interface 0
Ethernet II, Src: Tp-LinkT_91:26:1c (64:70:02:91:26:1c), Dst: lmk (c8:0a:a9:d0:5c:0a)
Internet Protocol Version 4, Src: 195.208.113.150, Dst: 192.168.0.105
Transmission Control Protocol, Src Port: 21, Dst Port: 55148, Seq: 110, Ack: 27, Len: 26
File Transfer Protocol (FTP)
  230 User stud5 logged in\r\n
      Response code: User logged in, proceed (230)
      Response arg: User stud5 logged in
```

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

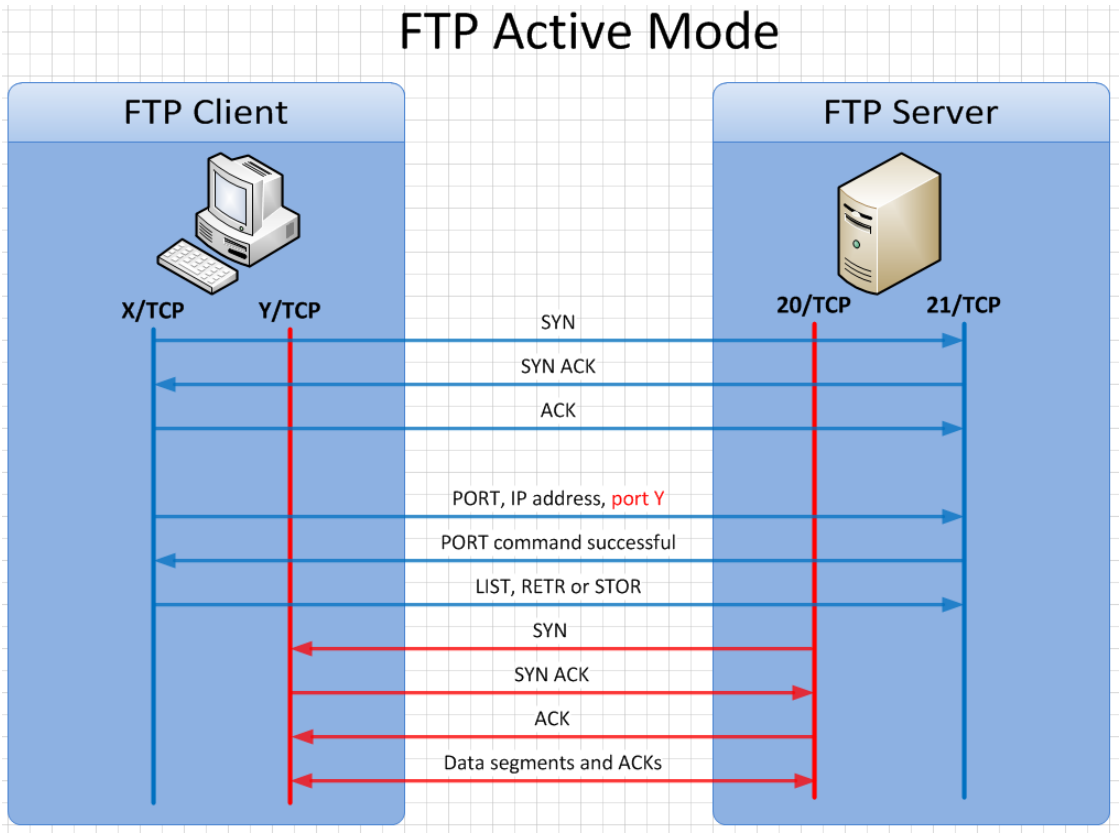


Рис. 1.7: Клиент-серверное взаимодействие в активном режиме FTP

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, PETR, STORE и др.

Запросим список файлов в текущей директории командой ls:

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

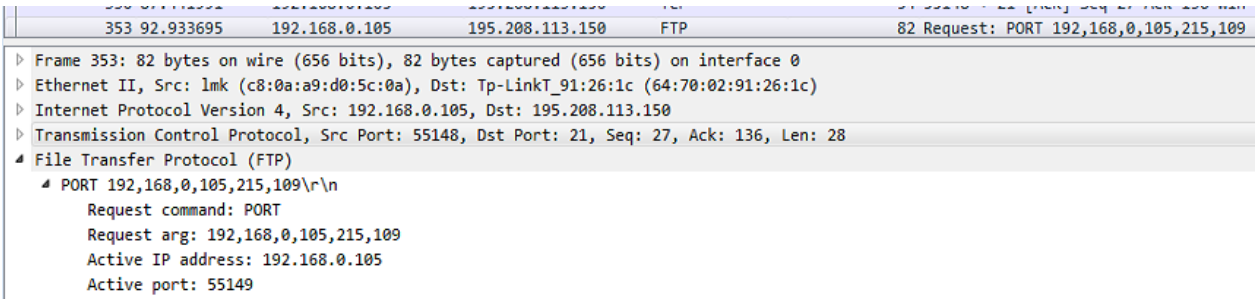


Рис. 1.8: Переход в активный режим

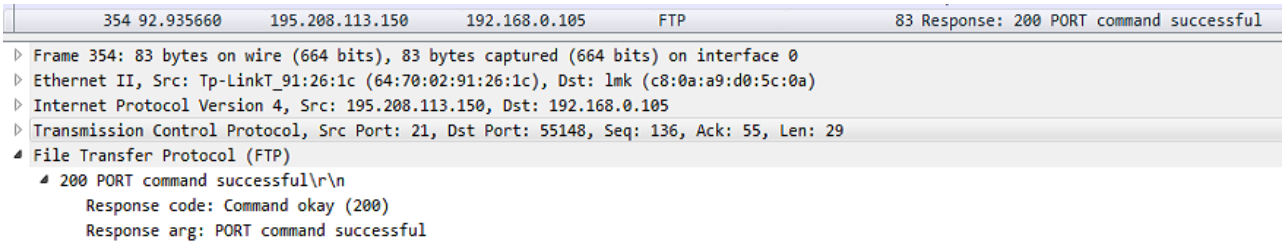


Рис. 1.9: Реакция сервера на успешный переход в активный режим

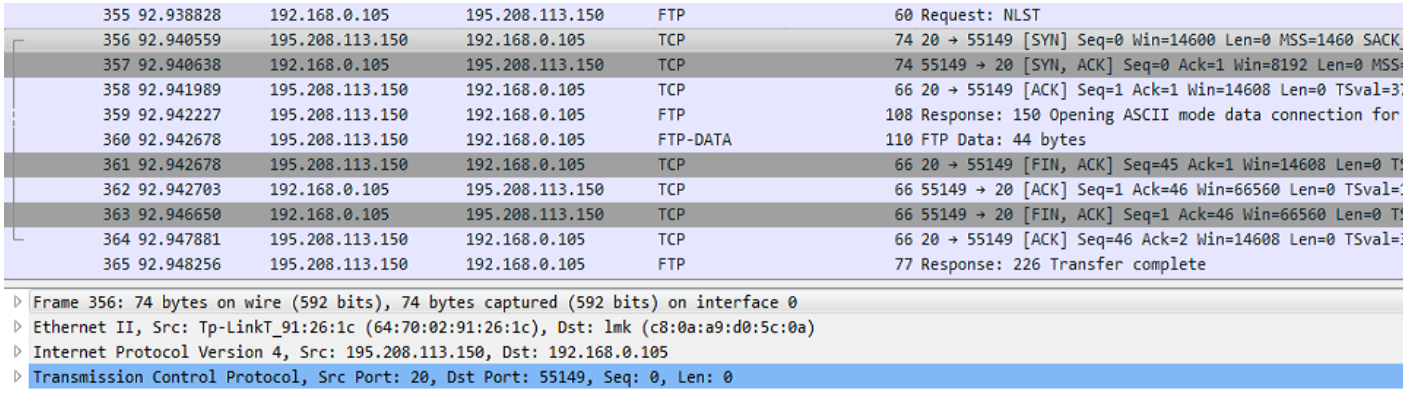


Рис. 1.10: Запрос списка файлов

### 1.4.5 Пассивный режим FTP

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

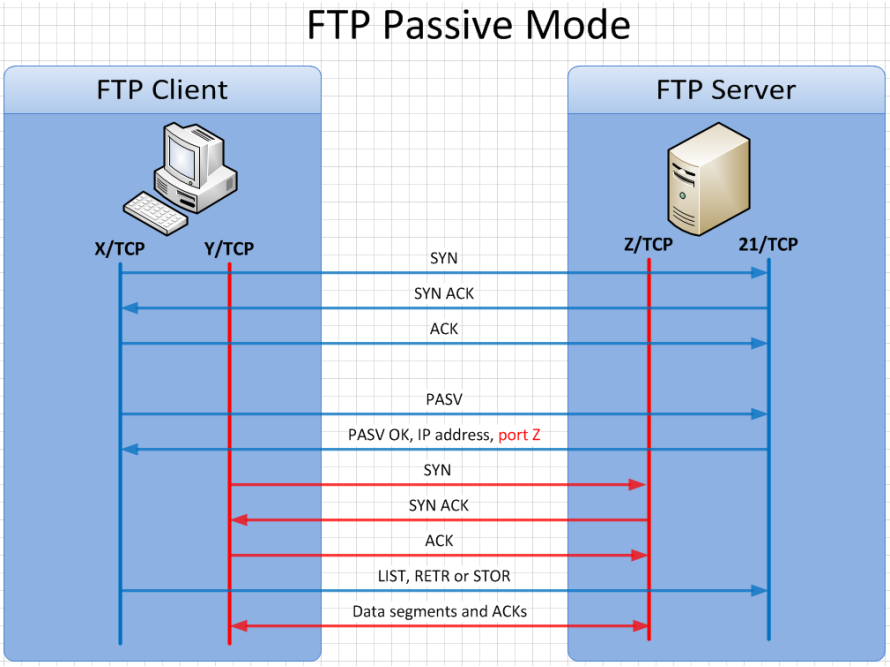


Рис. 1.11: Клиент-серверное взаимодействие в пассивном режиме FTP

| | | | | |
|---|---|---|---|---|
| 6140 725.592980 | 192.168.0.105 | 209.132.180.131 | FTP | 55 Request: U |
| 6141 725.768143 | 209.132.180.131 | 192.168.0.105 | TCP | 60 21 → 55262 [ACK] Seq=24 Ack=2 Win=14720 Len=0 |
| 6142 725.768157 | 192.168.0.105 | 209.132.180.131 | FTP | 55 Request: S |
| 6143 725.941826 | 209.132.180.131 | 192.168.0.105 | TCP | 60 21 → 55262 [ACK] Seq=24 Ack=3 Win=14720 Len=0 |
| 6144 725.941845 | 192.168.0.105 | 209.132.180.131 | FTP | 55 Request: E |
| 6145 726.115604 | 209.132.180.131 | 192.168.0.105 | TCP | 60 21 → 55262 [ACK] Seq=24 Ack=4 Win=14720 Len=0 |
| 6146 726.115631 | 192.168.0.105 | 209.132.180.131 | FTP | 55 Request: R |
| 6148 726.289418 | 209.132.180.131 | 192.168.0.105 | TCP | 60 21 → 55262 [ACK] Seq=24 Ack=5 Win=14720 Len=0 |
| 6149 726.569222 | 192.168.0.105 | 209.132.180.131 | FTP | 55 Request: |
| 6150 726.743874 | 209.132.180.131 | 192.168.0.105 | TCP | 60 21 → 55262 [ACK] Seq=24 Ack=6 Win=14720 Len=0 |
| 6165 729.729138 | 192.168.0.105 | 209.132.180.131 | FTP | 55 Request: a |
| 6166 729.903211 | 209.132.180.131 | 192.168.0.105 | TCP | 60 21 → 55262 [ACK] Seq=24 Ack=7 Win=14720 Len=0 |
| 6167 729.960673 | 192.168.0.105 | 209.132.180.131 | FTP | 55 Request: n |
| 6168 730.134358 | 209.132.180.131 | 192.168.0.105 | TCP | 60 21 → 55262 [ACK] Seq=24 Ack=8 Win=14720 Len=0 |
| 6170 730.440765 | 192.168.0.105 | 209.132.180.131 | FTP | 55 Request: o |
| 6171 730.615392 | 209.132.180.131 | 192.168.0.105 | TCP | 60 21 → 55262 [ACK] Seq=24 Ack=9 Win=14720 Len=0 |
| 6172 730.720369 | 192.168.0.105 | 209.132.180.131 | FTP | 55 Request: n |
| 6173 730.894393 | 209.132.180.131 | 192.168.0.105 | TCP | 60 21 → 55262 [ACK] Seq=24 Ack=10 Win=14720 Len=0 |
| 6174 731.033012 | 192.168.0.105 | 209.132.180.131 | FTP | 55 Request: y |
| 6176 731.206721 | 209.132.180.131 | 192.168.0.105 | TCP | 60 21 → 55262 [ACK] Seq=24 Ack=11 Win=14720 Len=0 |
| 6177 731.639484 | 192.168.0.105 | 209.132.180.131 | FTP | 55 Request: m |
| 6178 731.813285 | 209.132.180.131 | 192.168.0.105 | TCP | 60 21 → 55262 [ACK] Seq=24 Ack=12 Win=14720 Len=0 |
| 6179 731.960467 | 192.168.0.105 | 209.132.180.131 | FTP | 55 Request: o |
| 6181 732.134313 | 209.132.180.131 | 192.168.0.105 | TCP | 60 21 → 55262 [ACK] Seq=24 Ack=13 Win=14720 Len=0 |
| 6182 732.215419 | 192.168.0.105 | 209.132.180.131 | FTP | 55 Request: u |
| 6183 732.389199 | 209.132.180.131 | 192.168.0.105 | TCP | 60 21 → 55262 [ACK] Seq=24 Ack=14 Win=14720 Len=0 |
| 6184 732.448625 | 192.168.0.105 | 209.132.180.131 | FTP | 55 Request: s |
| 6185 732.623225 | 209.132.180.131 | 192.168.0.105 | TCP | 60 21 → 55262 [ACK] Seq=24 Ack=15 Win=14720 Len=0 |
| 6186 733.007294 | 192.168.0.105 | 209.132.180.131 | FTP | 56 Request: |
| 6187 733.181659 | 209.132.180.131 | 192.168.0.105 | TCP | 60 21 → 55262 [ACK] Seq=24 Ack=17 Win=14720 Len=0 |
| 6188 733.181922 | 209.132.180.131 | 192.168.0.105 | FTP | 129 Response: 331 Anonymous login ok, send your complete ema |

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):

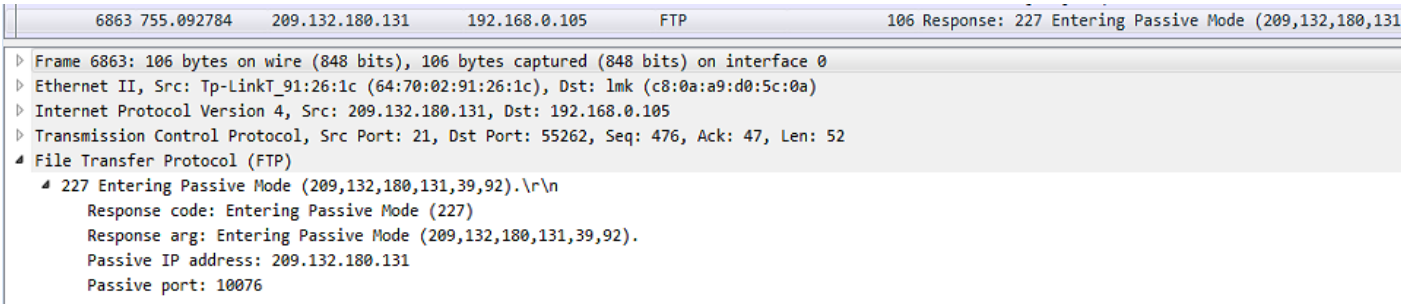


Рис. 1.13: Переход в пассивный режим

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:

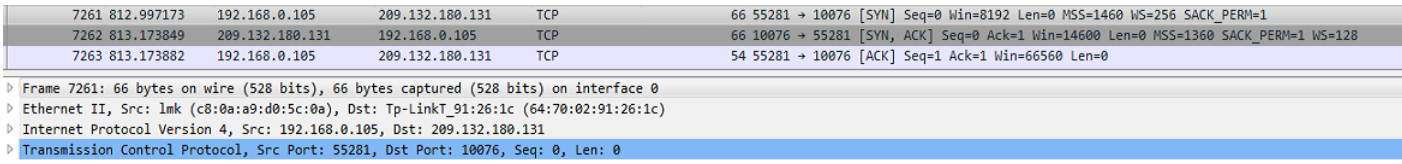


Рис. 1.14: Установление соединения по полученному адресу

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

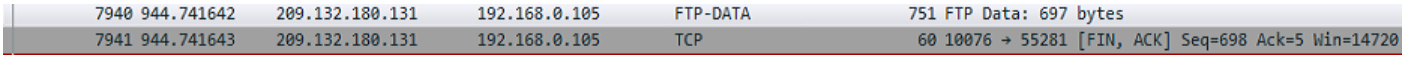


Рис. 1.15: Передача данных

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.

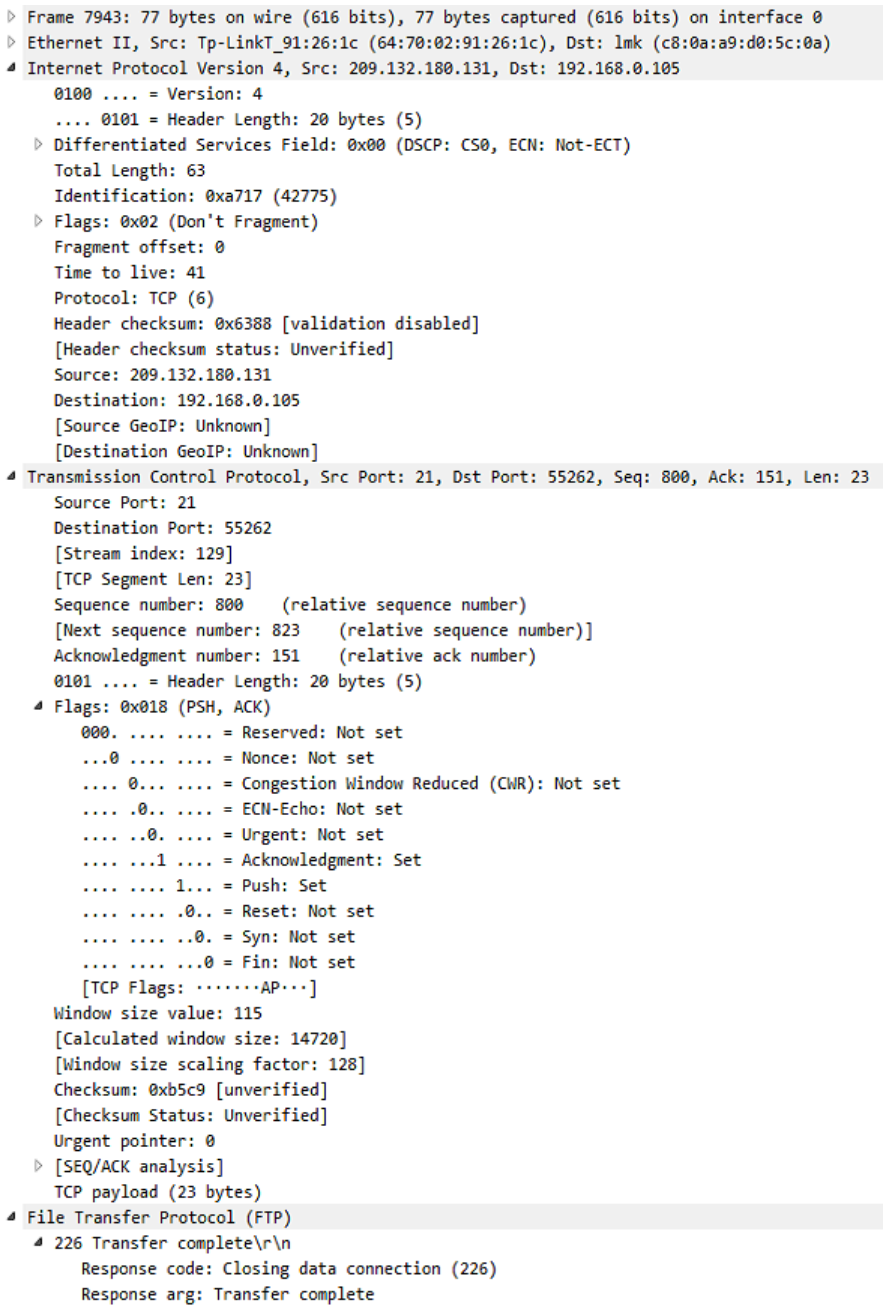


Рис. 1.16: Пакет с запрошенными данными

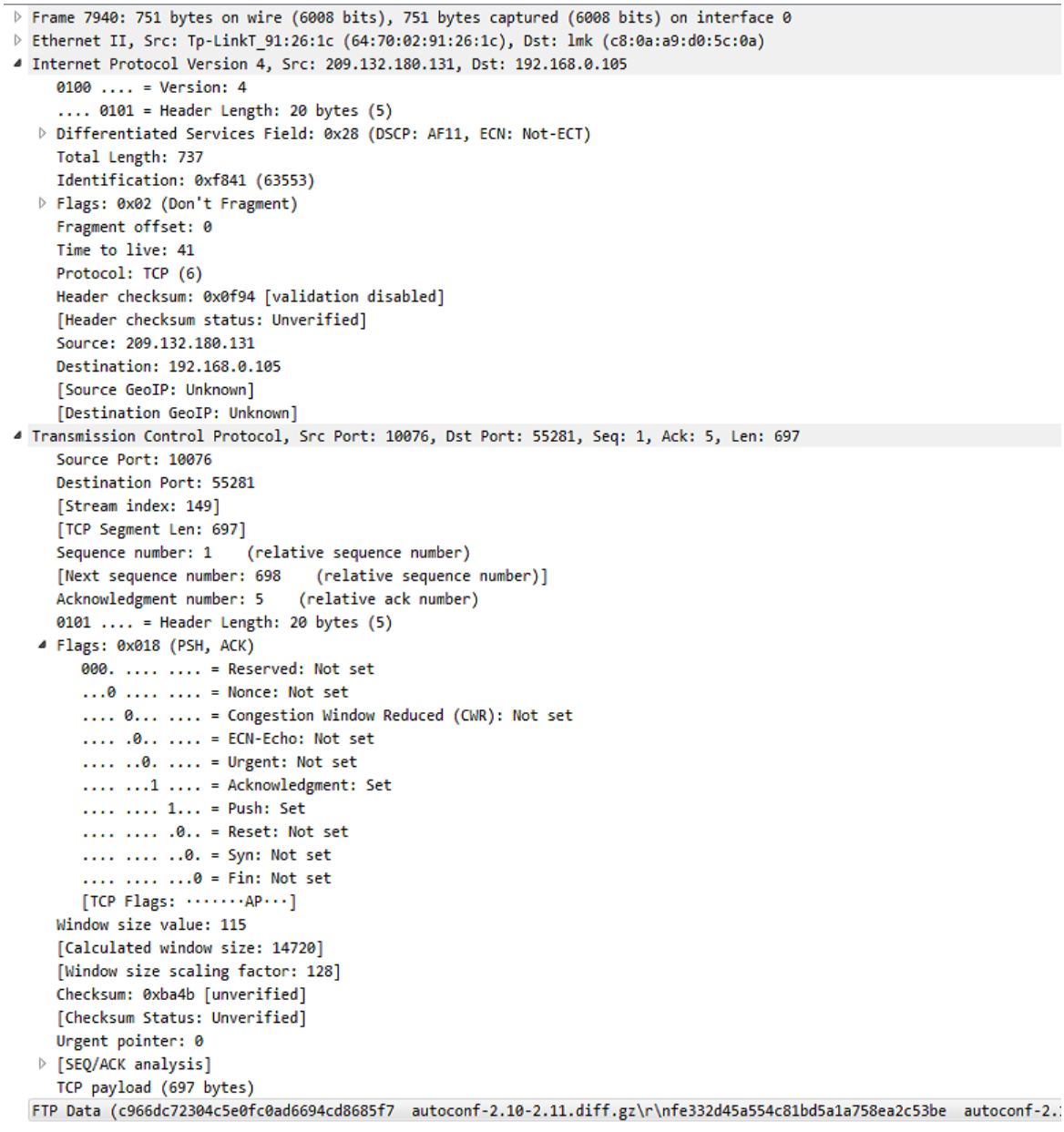


Рис. 1.17: Пакет с кодом закрытия канала

## 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 том, 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.