**МИНИСТЕРСТВО науки и высшего ОБРАЗОВАНИЯ РОссИЙСКОЙ ФЕДЕРАЦИИ**

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ**

**УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**

**«МОСКОВСКИЙ АВИАЦИОННЫЙ ИНСТИТУТ**

(национальный исследовательский университет)»

Институт №3 «Системы управления, информатика и электроэнергетика»

Кафедра № 304 «Вычислительные машины, системы и сети»

Web-технологии

Отчет по лабораторной работе № 3

Модель OSI. Стек протоколов. Wireshark

Выполнил студент группы М3О-409Б-20:

Мокшин И. Н.

Проверил, Титов Ю.П.

Москва 2023 г.

1. Рассмотреть классическую многоуровневую модель OSI. Определить какие протоколы возможны на каждом уровне (привести по 2 примера на каждом уровне).

**Физический**

Протоколы: USB, RJ

Описание: Работа со средой передачи, сигналами и двоичными данными

**Канальный**

Протоколы: IEEE 802.3, DSL

Описание: Физическая адресация

**Сетевой**

Протоколы: Internet Protocol (IP), Internet Control Message Protocol (ICMP)

Описание: Определение маршрута и логическая адресация

**Транспортный**

Протоколы: TCP, UDP

Описание: Прямая связь между конечными пунктами и надёжность

**Сеансовый**

Протоколы: L2TP, gRPC

Описание: Управление сеансом связи

**Представления**

Протоколы: SSL, gzip

Описание: Представление и шифрование данных

**Прикладной**

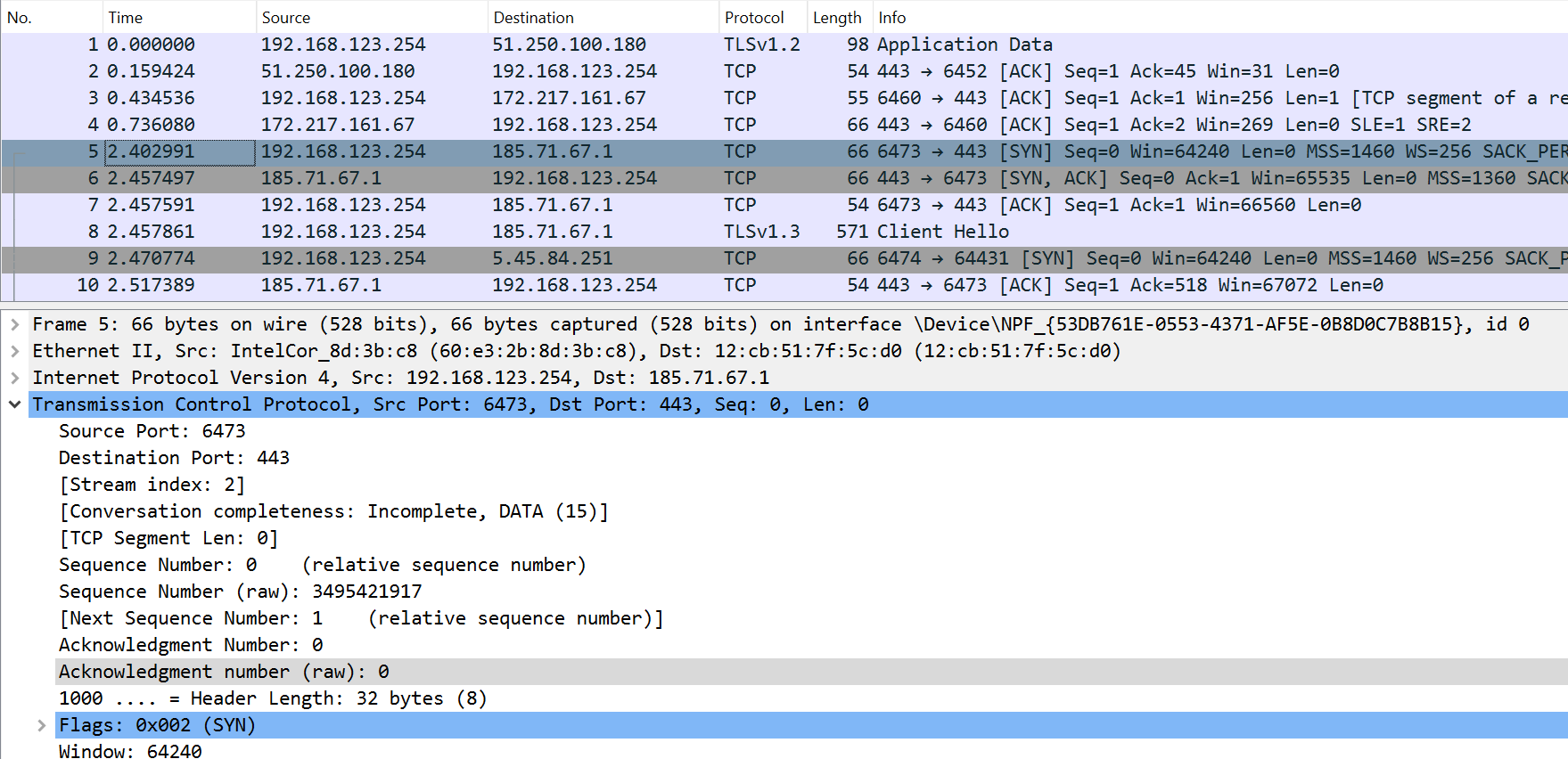
Протоколы: HTTP, FTP

Описание: Доступ к сетевым службам

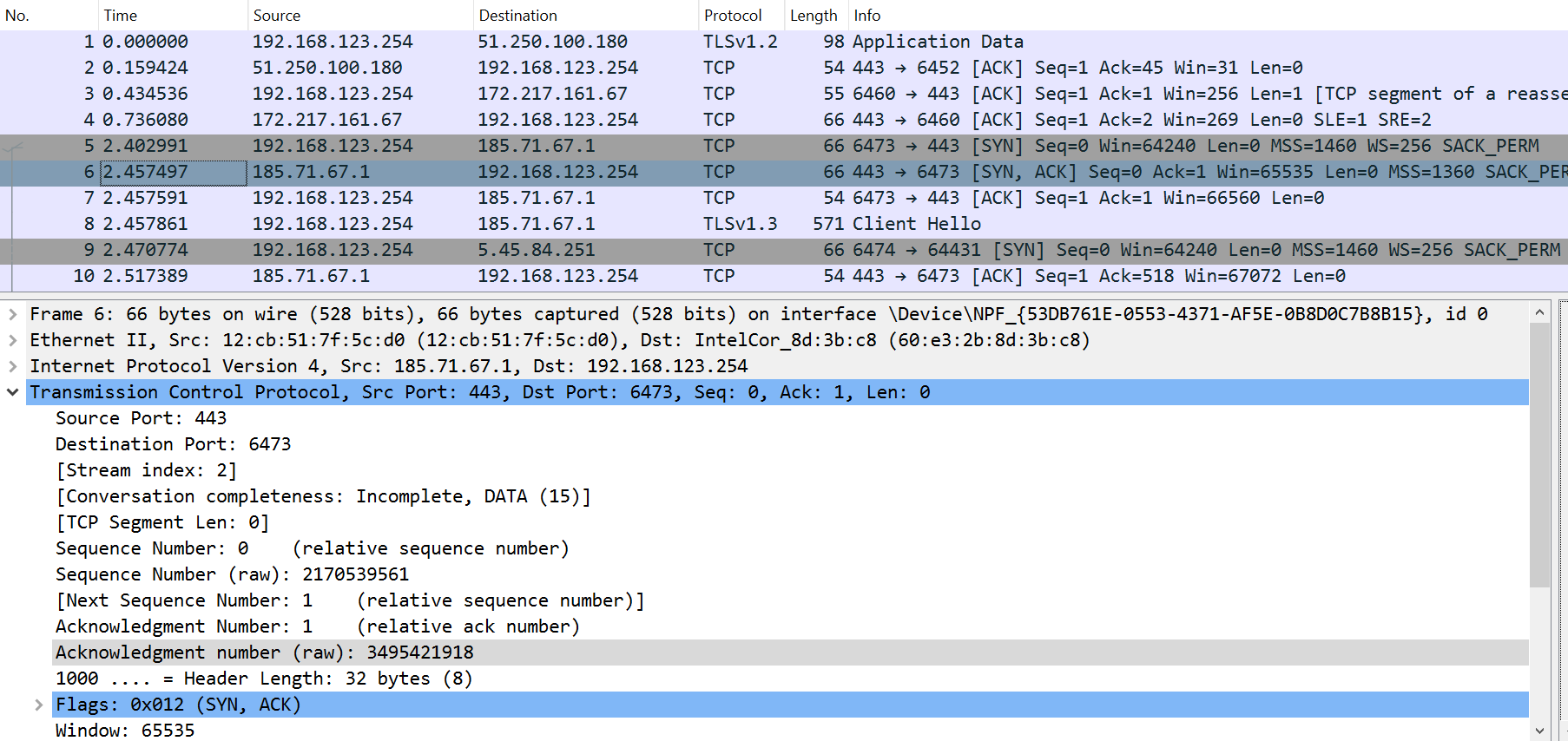
1. Запустить программу Wireshark. Настроить программу на активное интернет соединение. Рассмотреть различные TCP и UDP соединения. Привести в трехстороннее рукопожатие и трехстороннее завершение сеанса TCP (привести не только сам процесс, но и раскрыть флаги для каждого пакета).

**TCP Рукопожатие**

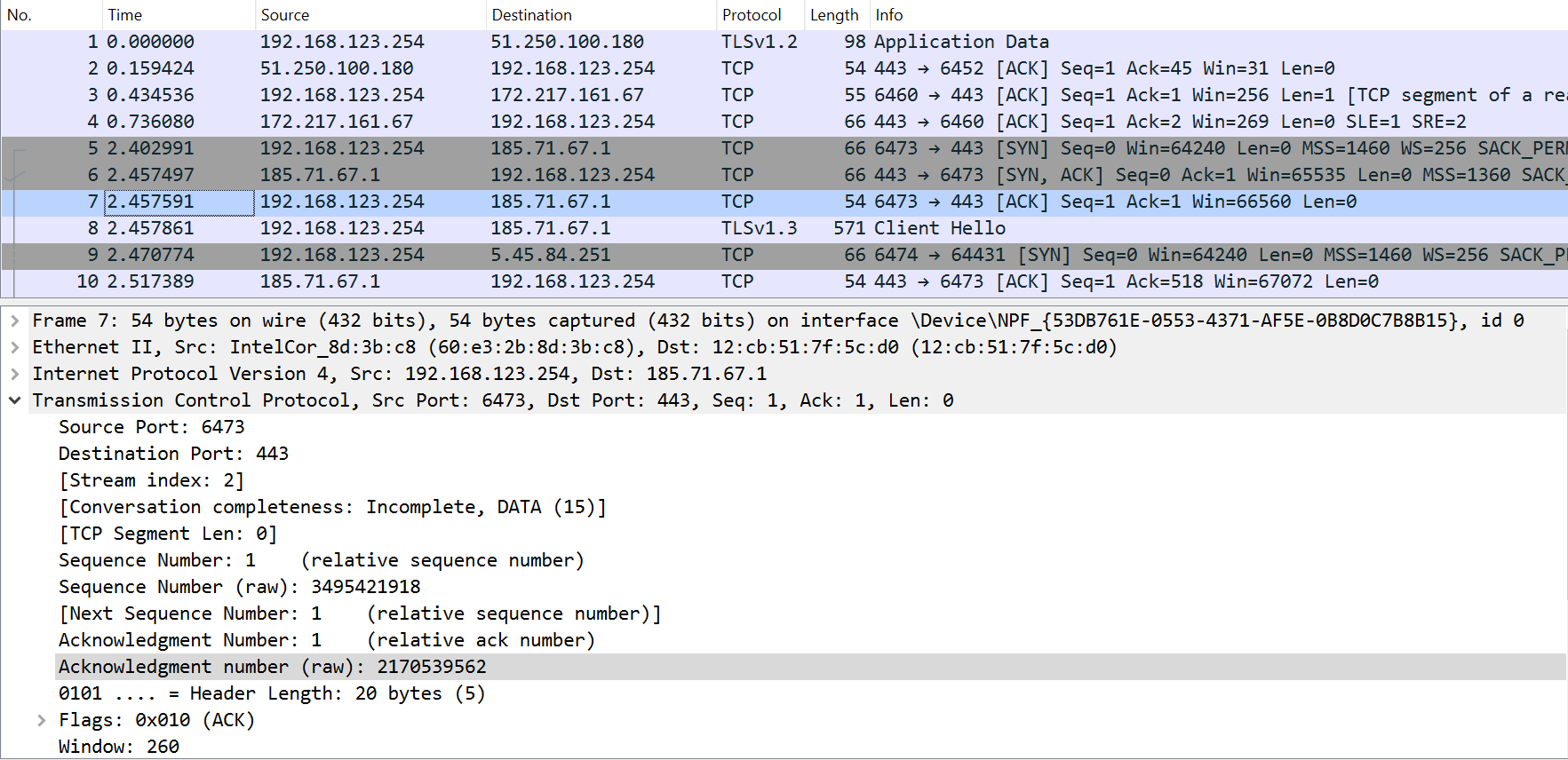
1) Клиент отправляет сегмент с установленным флагом SYN. При этом сегменту присваивается произвольный порядковый номер (sequence number) в интервале от 1 до 232 (т.н. initial sequence number), относительно которого будет вестись дальнейший отсчет последовательности сегментов в соединении.



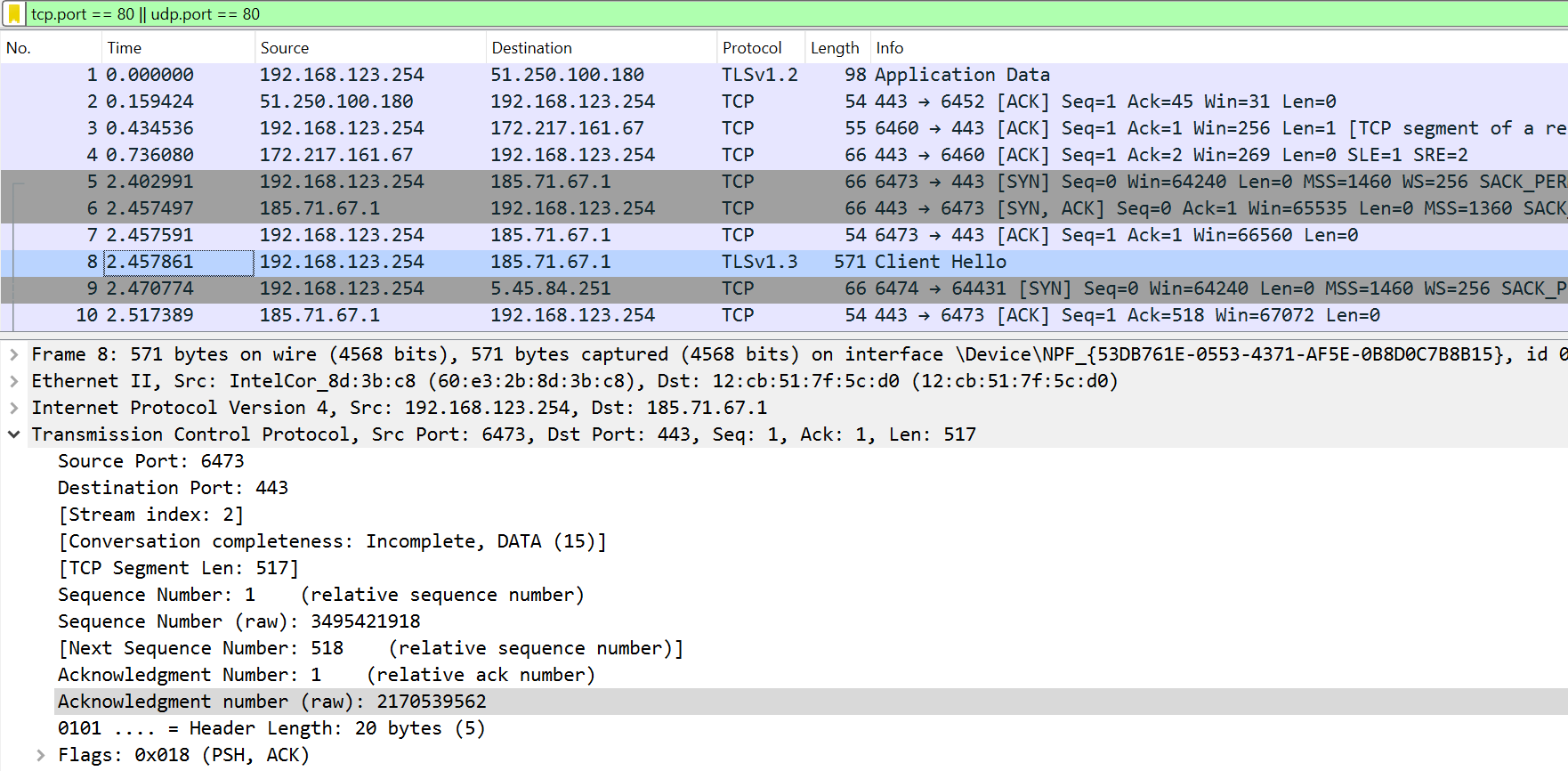
2) Сервер получает запрос и отправляет ответный сегмент с одновременно установленными флагами SYN+ACK, при этом записывает в поле «номер подтверждения» (acknowledgement number), полученный порядковый номер, увеличенный на 1 (что подтверждает получение первого сегмента), а также устанавливает свой порядковый номер, который, как и в SYN-сегменте, выбирается произвольно.



3) После получения клиентом сегмента с флагами SYN+ACK соединение считается установленным, клиент, в свою очередь, отправляет в ответ сегмент с флагом ACK, обновленными номерами последовательности, и не содержащий полезной нагрузки.

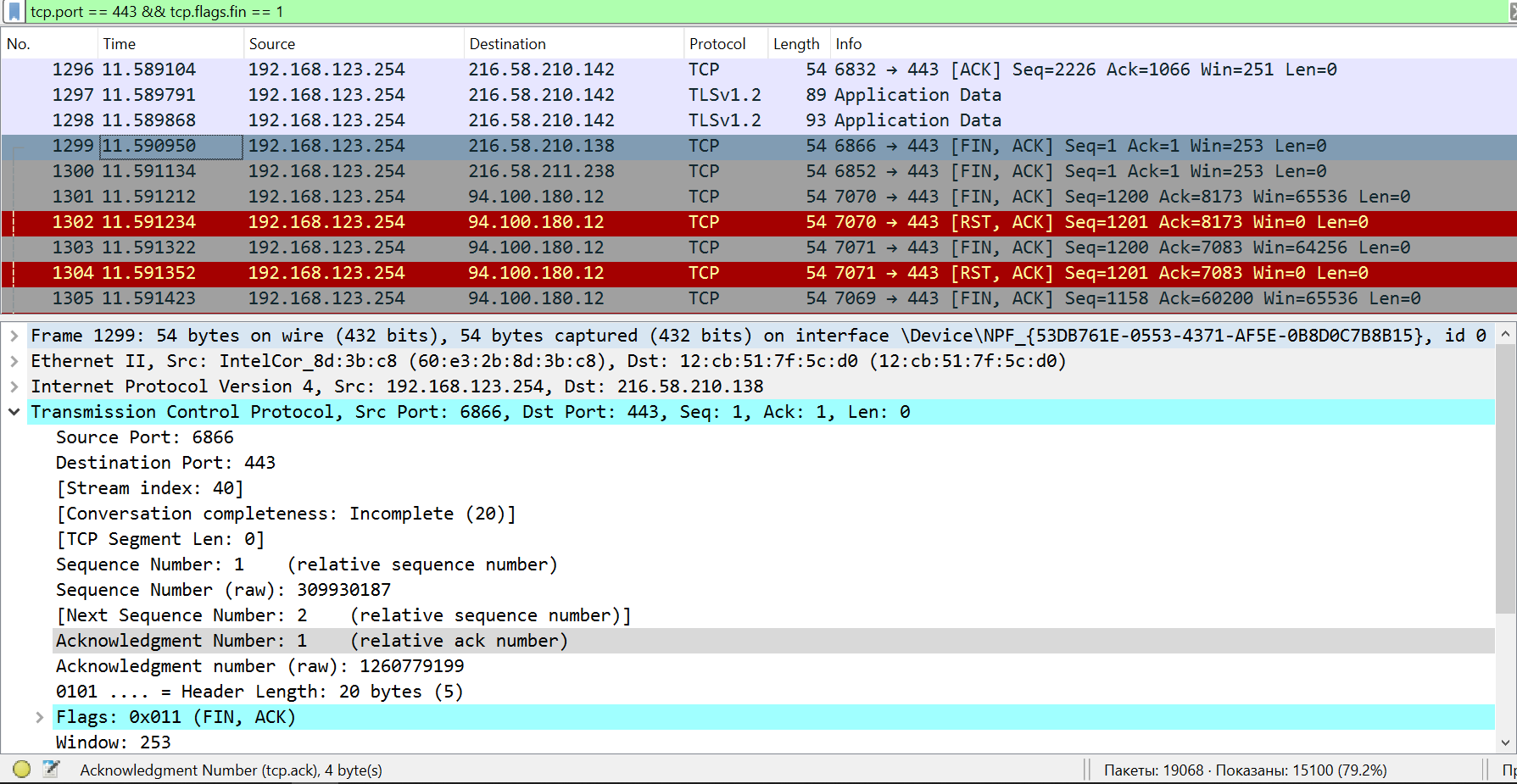


4) Начинается передача данных.

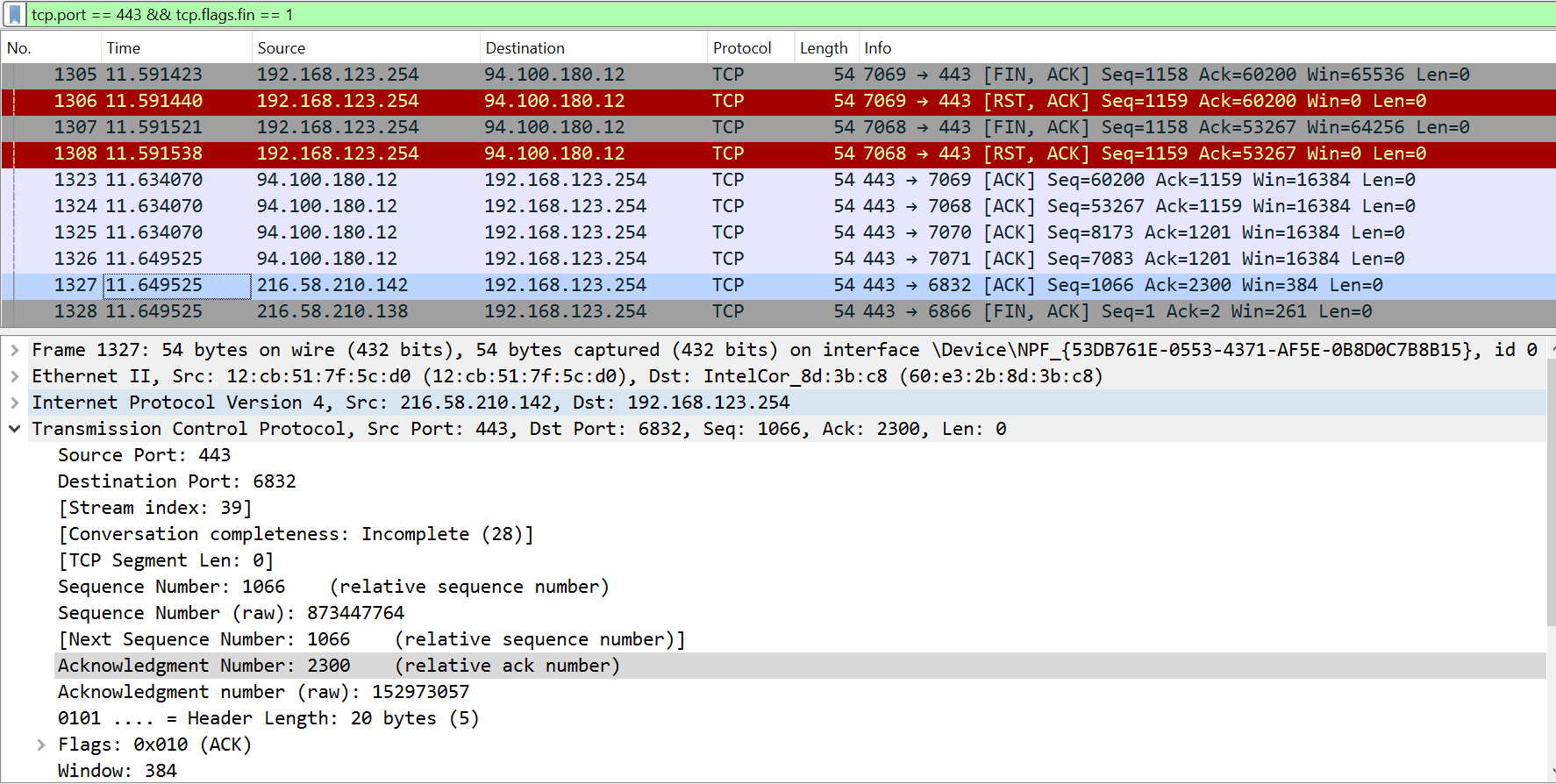


**TCP завершение сеанса**

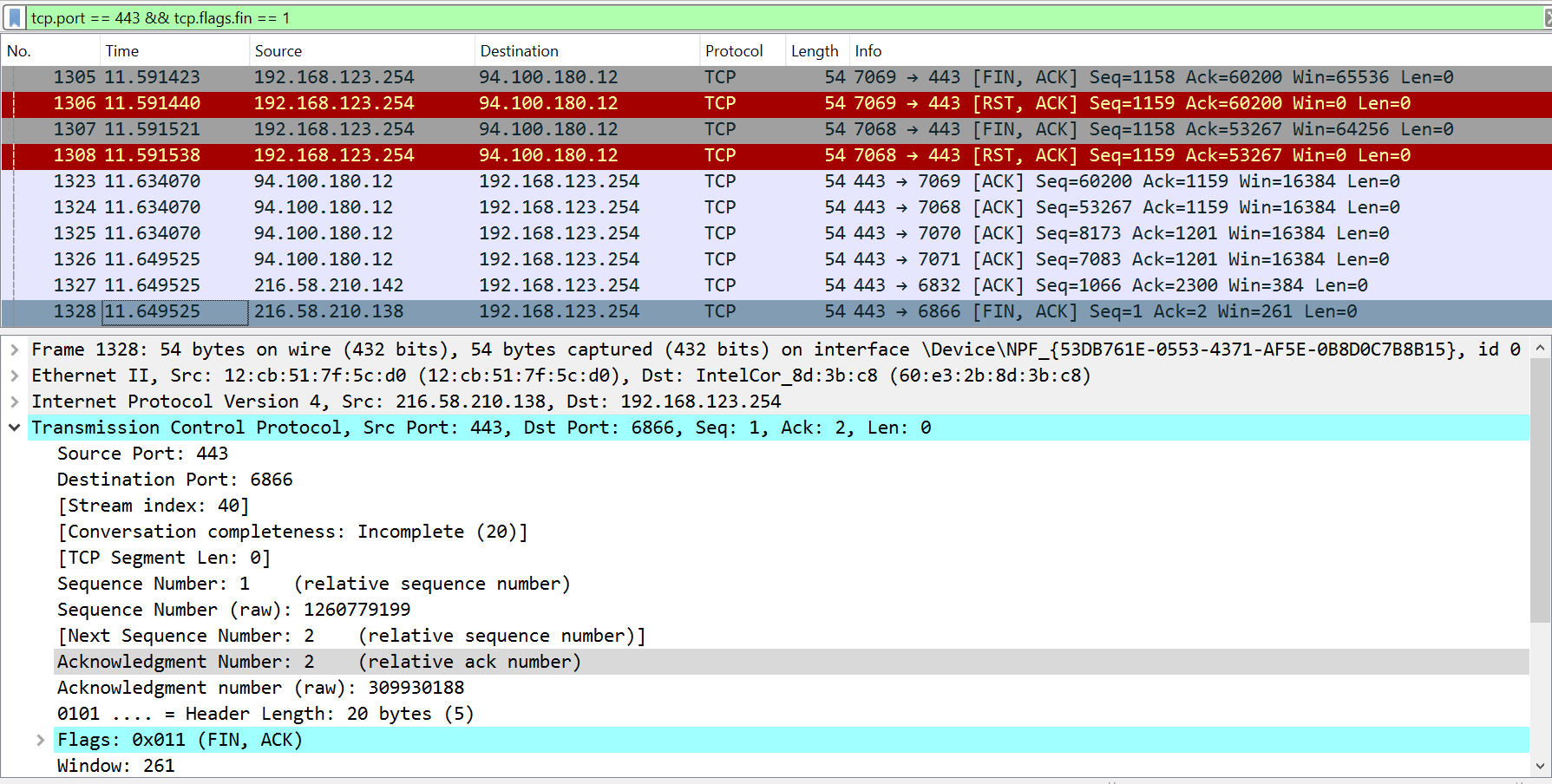
1. Один из задействованных процессов приложения сигнализирует своему уровню TCP, что сеанс связи больше не нужен. Со стороны этого устройства отправляется сообщение с установленным флагом FIN (этот пакет не обязательно должен быть пустым, он также может содержать полезную нагрузку), чтобы сообщить другому устройству о своем желании завершить открытое соединение.



1. Затем получение этого сообщения подтверждается (сообщение от отвечающего устройства с установленным флагом ACK, говорящем о получении сообщения FIN).

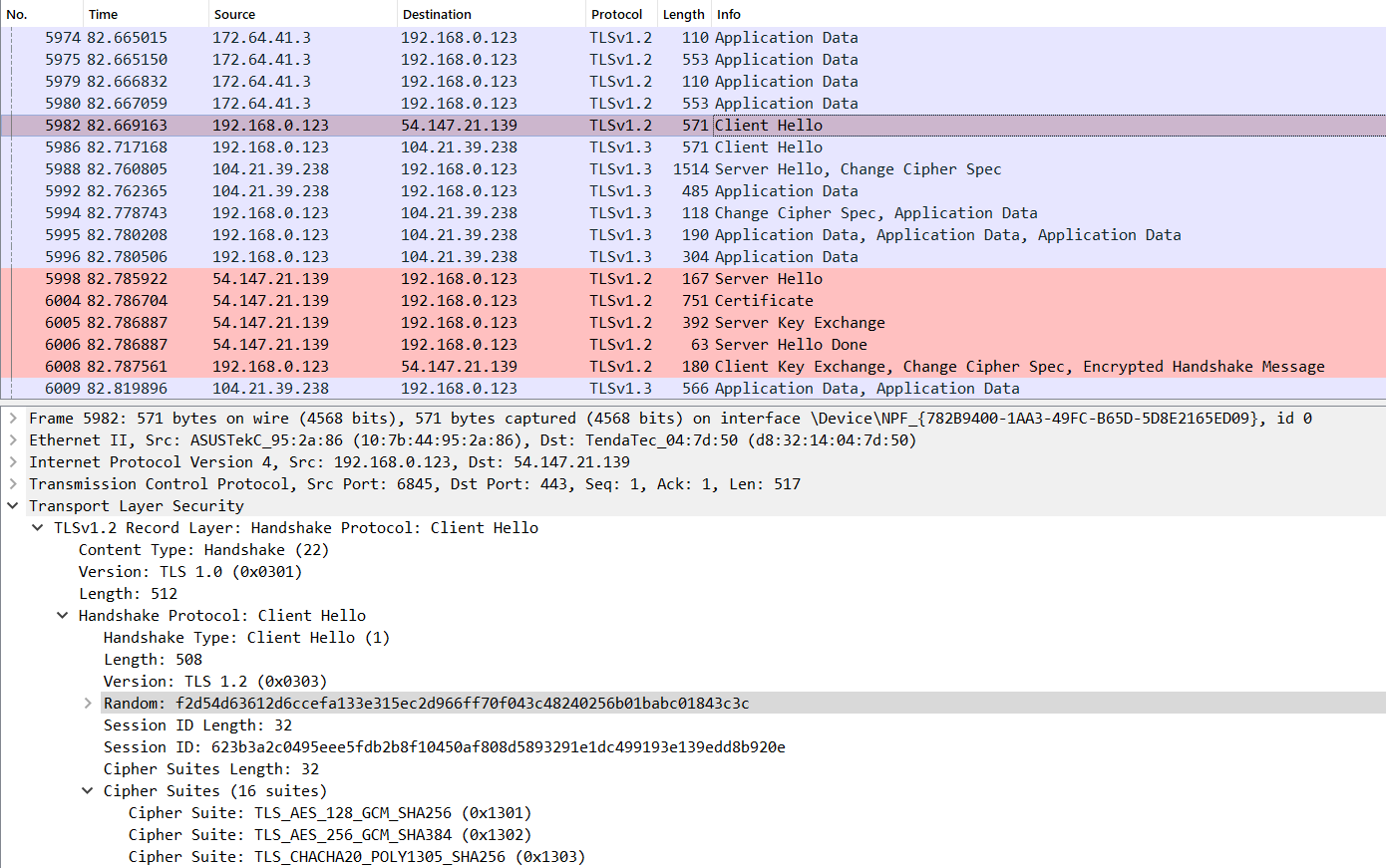


1. Когда отвечающее устройство готово, оно также отправляет сообщение с установленным флагом FIN, и, после получения в ответ подтверждающего получение сообщения с установленным флагом ACK или ожидания определенного периода времени, предусмотренного для получения ACK, сеанс полностью закрывается.

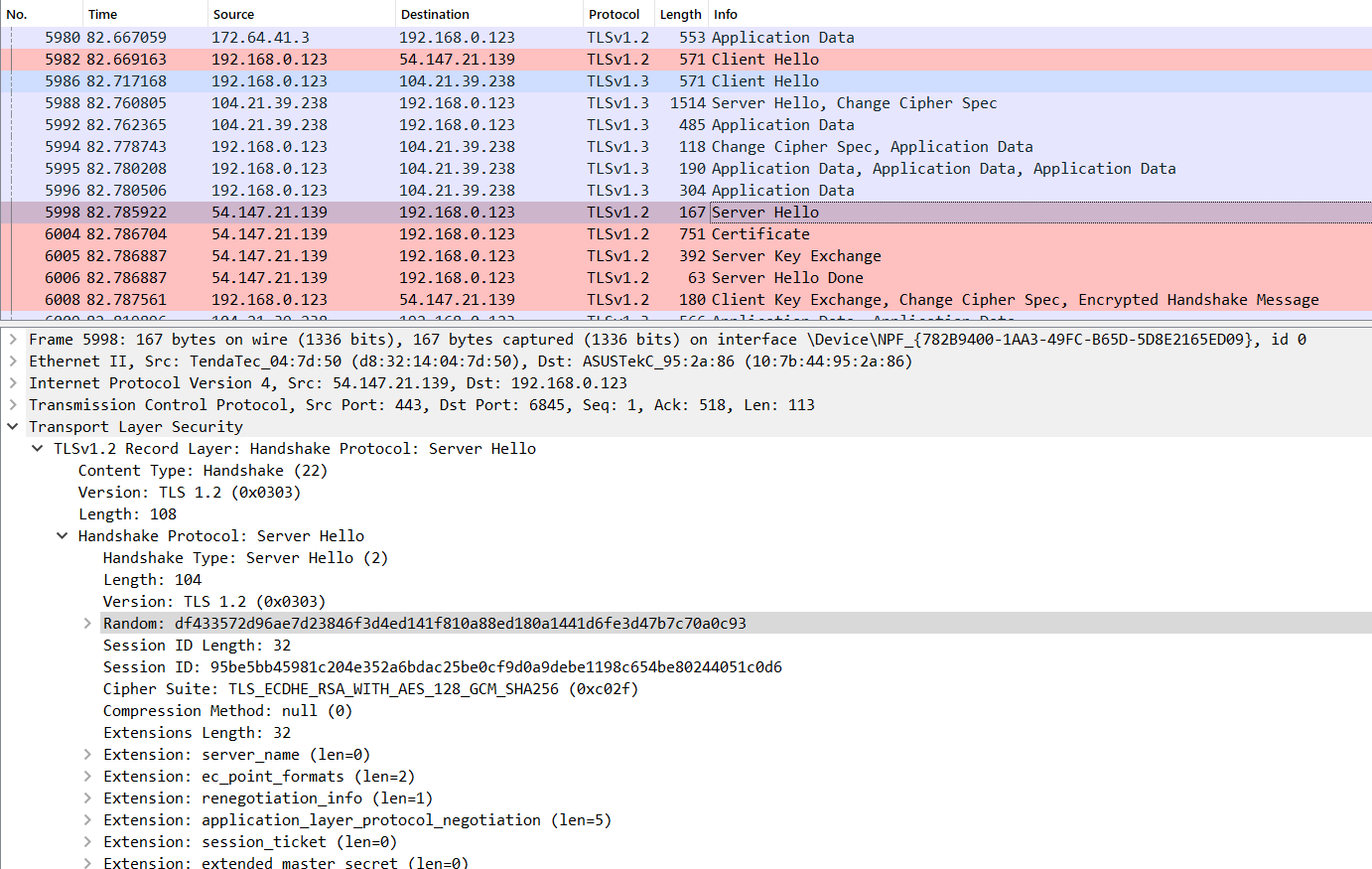


1. Рассмотреть процесс установления TLS соединения. В отчете привести заголовки стека протоколов основных сообщений при создании TSL соединения.

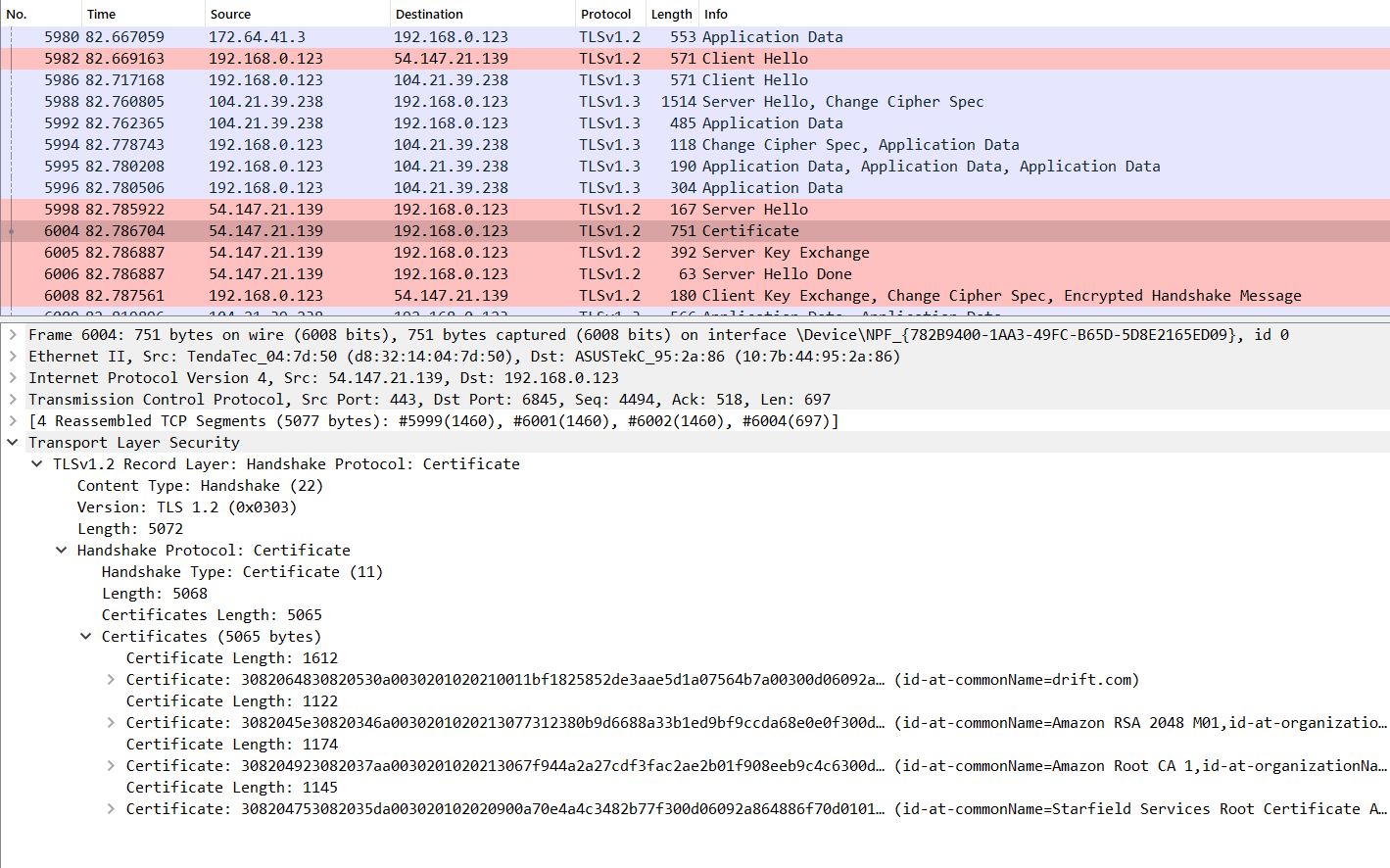
Клиент посылает сообщение ClientHello



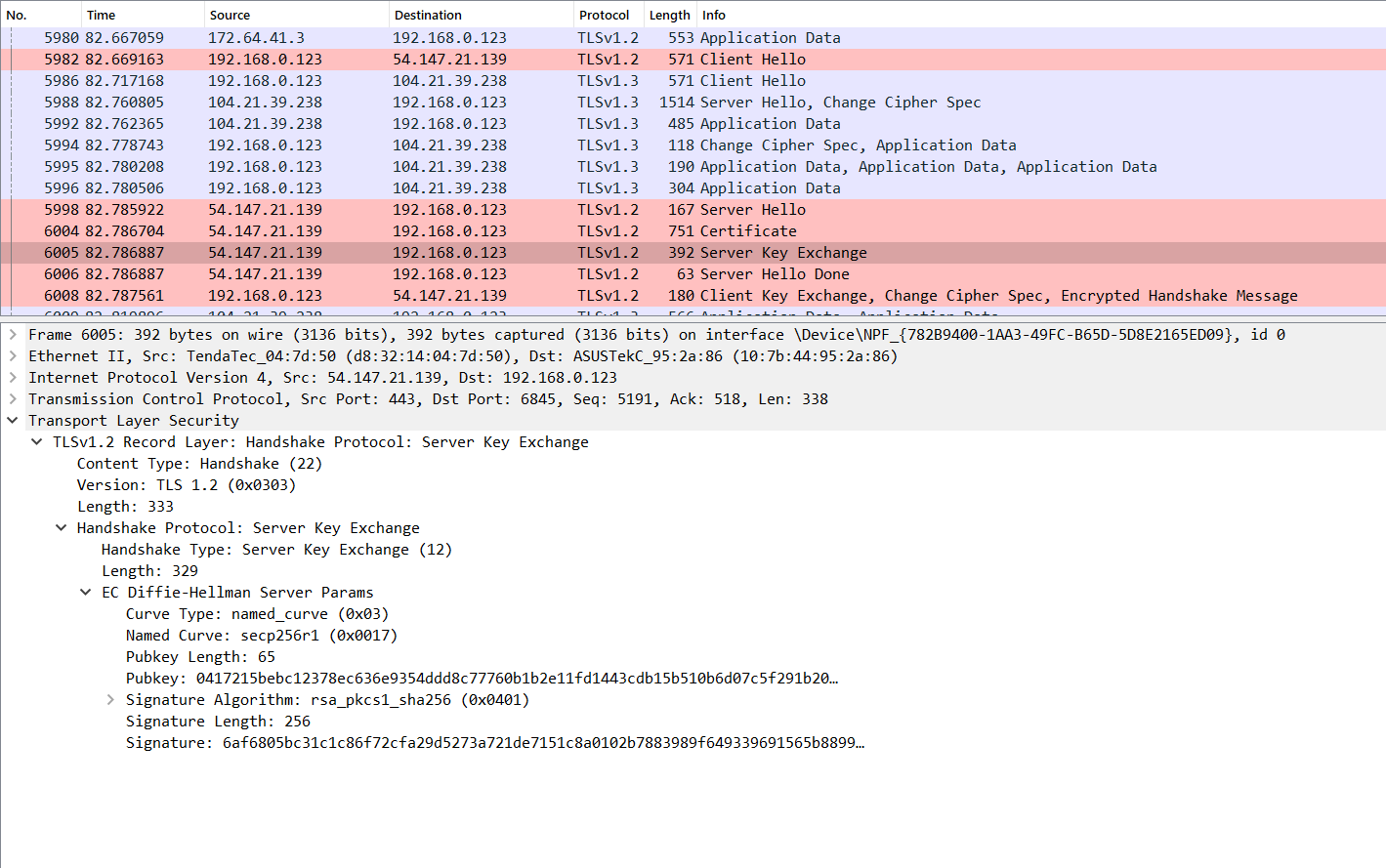
Сервер посылает сообщение Server hello



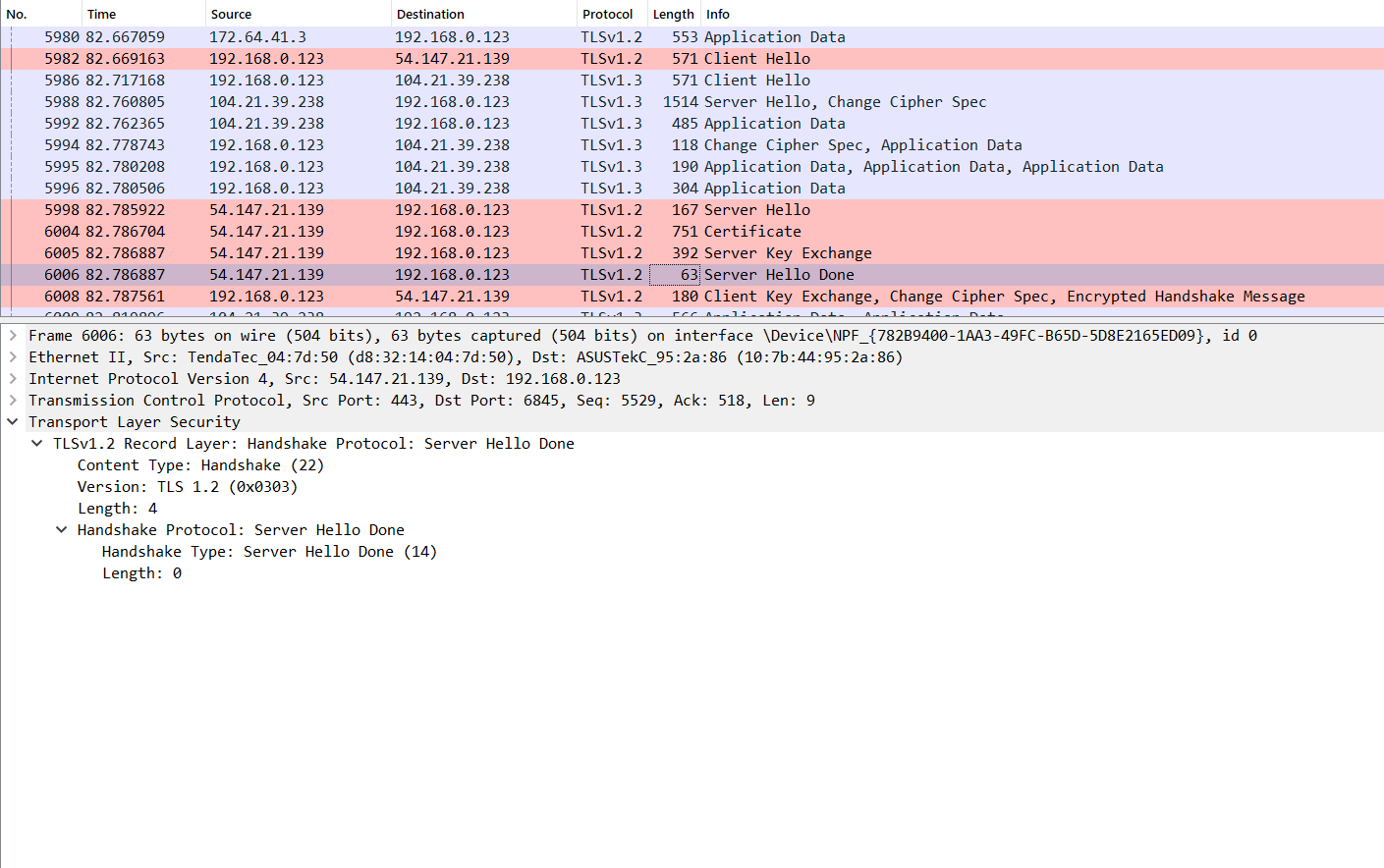
Сервер посылает сертификаты



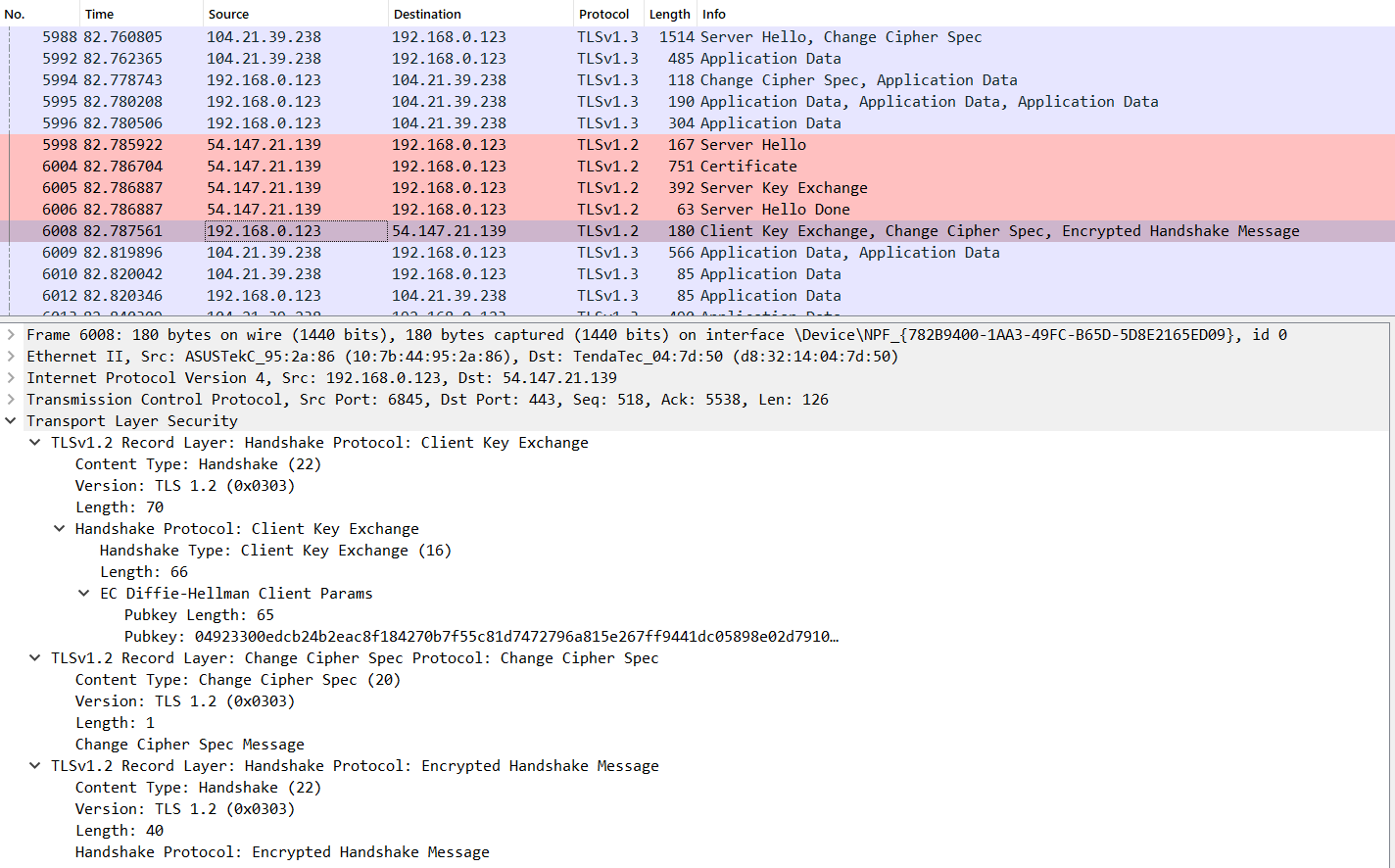
Сервер посылает открытый ключ



Сервер посылает сообщение Server Hello Done



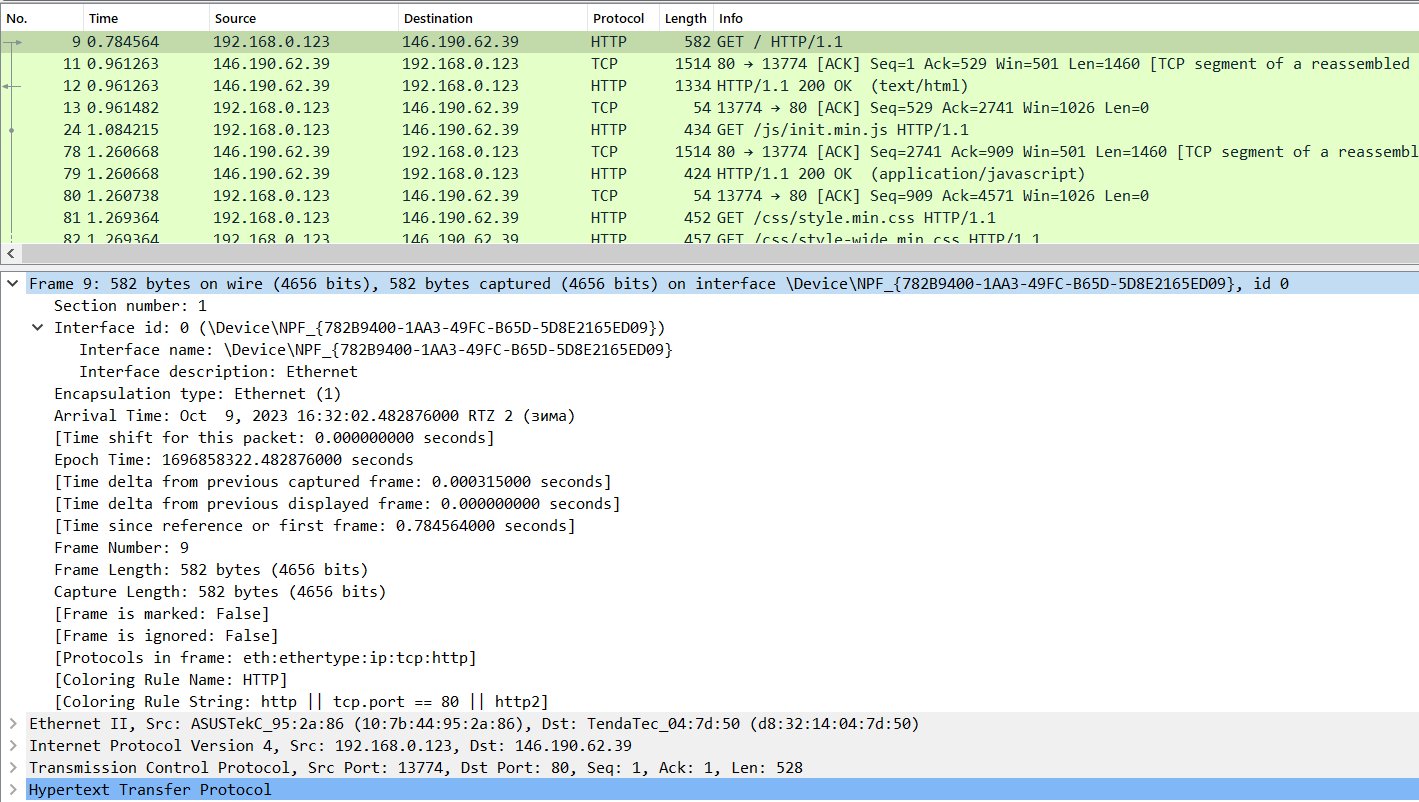
Клиент посылает свой открытый ключ



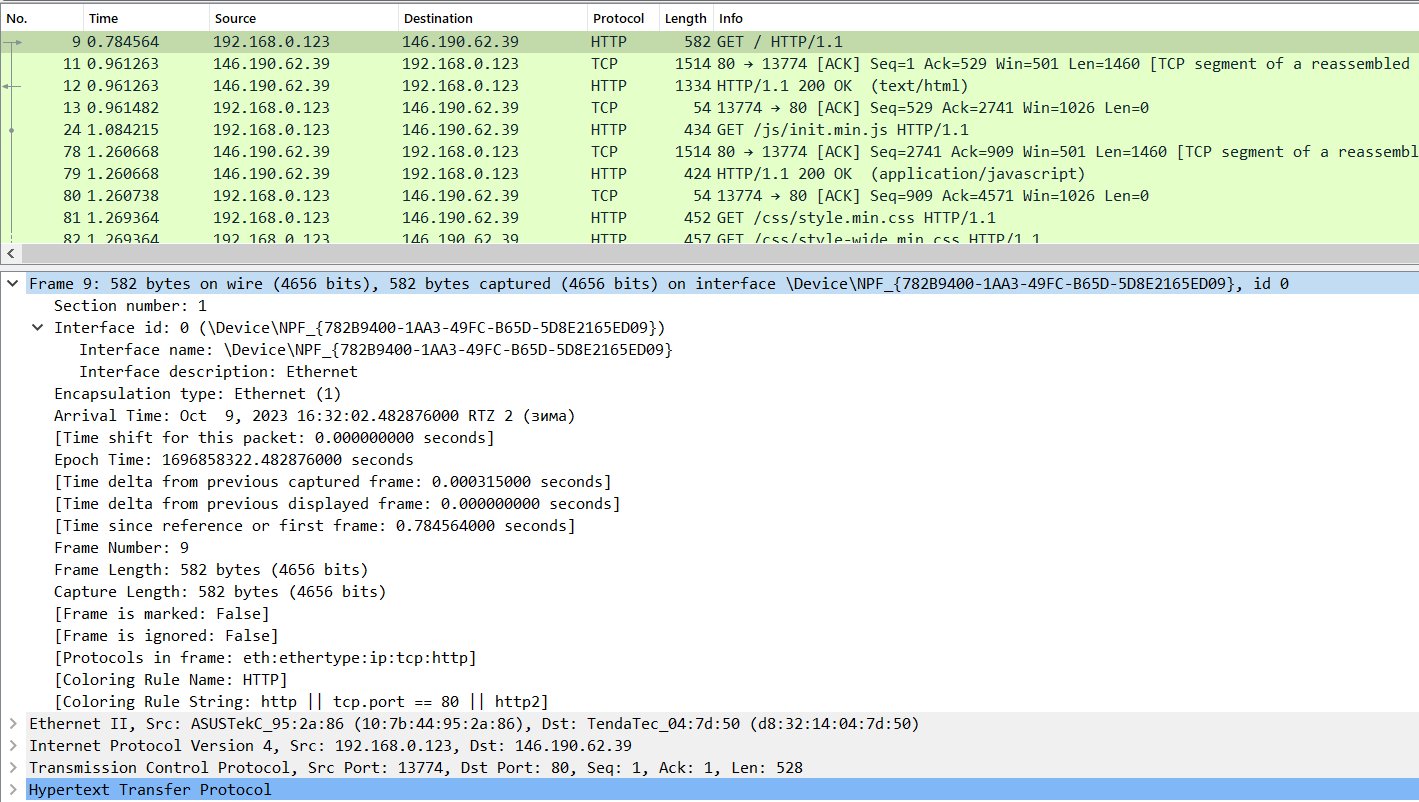
1. С помощью браузера отправить запрос для получения HTML страницы. Найти соответствующий запрос и ответ в программе Wireshark. В отчете полностью отобразить все содержимое стека протоколов на каждом уровне для получения HTML (не приводить полностью код HTML, а только до конца страницы и блок HTTP chunked response)
2. Проанализировать содержимое данных, отправленных в запросе и полученных в ответе. Расписать в отчете основные моменты соединения для каждого из протоколов.

Соединение производилось с: <http://httpforever.com/>

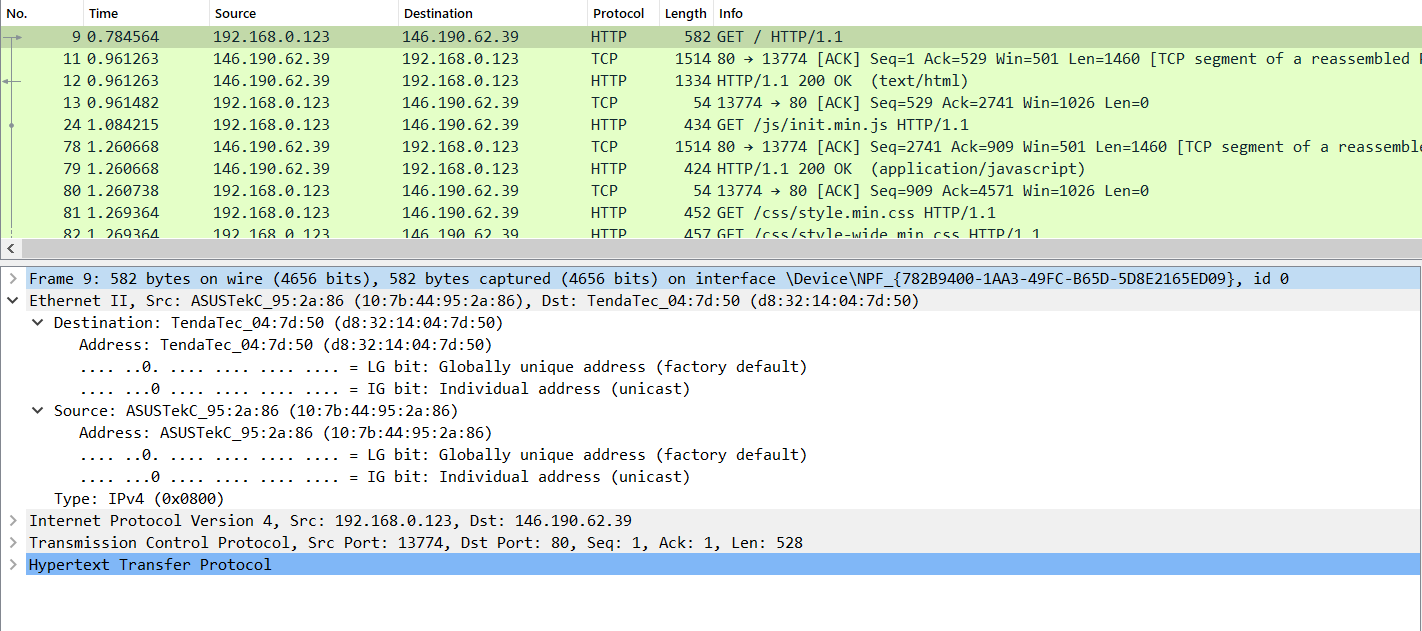
**Запрос**



**Физический уровень**

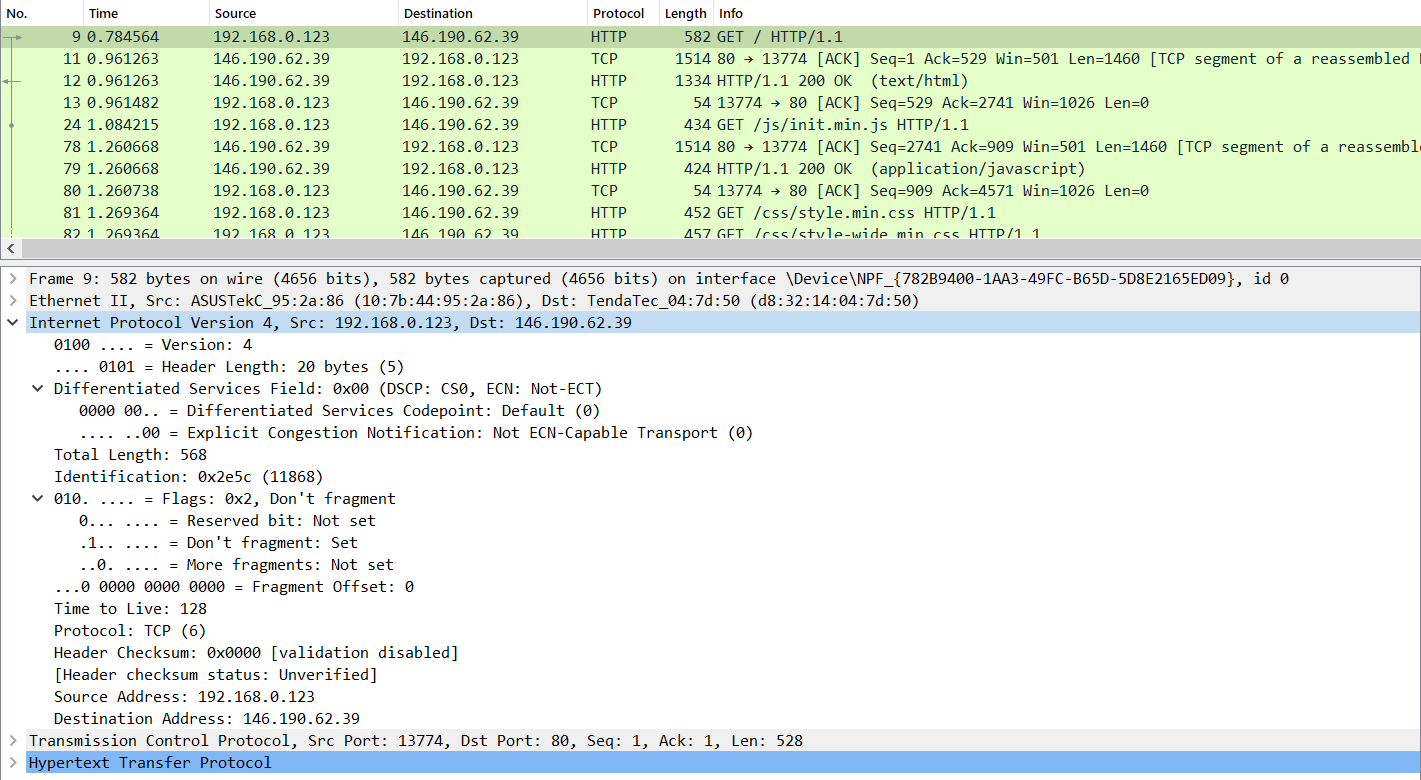


**Канальный уровень**



Данные передаются по интерфейсу Ethernet. На скриншоте видны MAC-адреса отправителя и получателя.

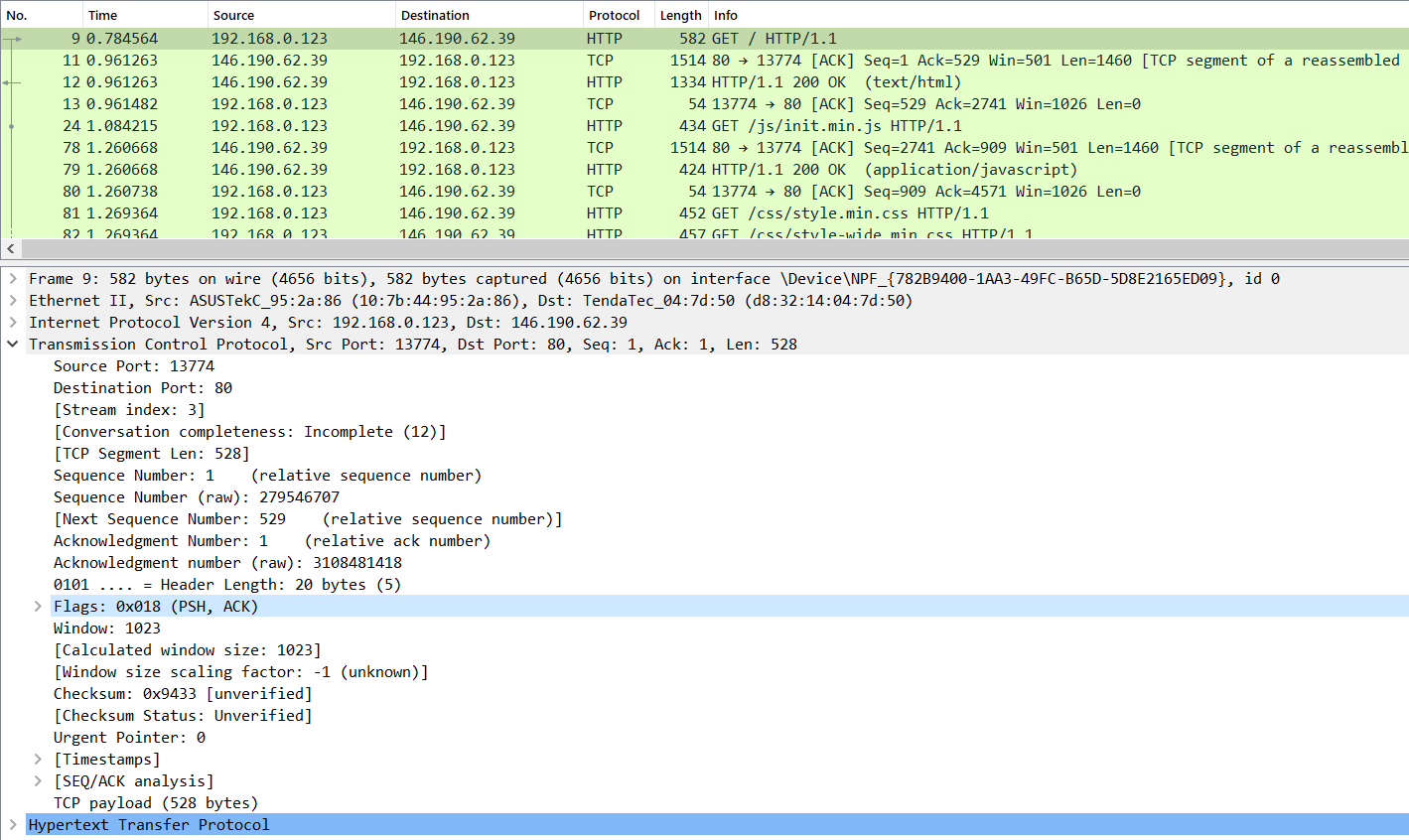
**Сетевой уровень**



Данные передаются по протоколу IPv4 без фрагментации.

Отправитель 192.168.0.123, Получатель: 146.190.62.39

**Транспортный уровень**



Протокол TCP. Порт отправителя 13774 (случайно выдан системой) Порт получателя 80 (определяется протоколом HTTP).

Флаги: PSH, ACK (рукопожатие уже было произведено)

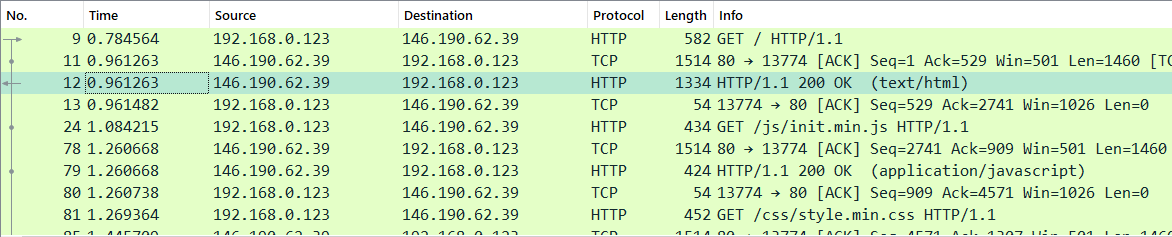
**Прикладной уровень**



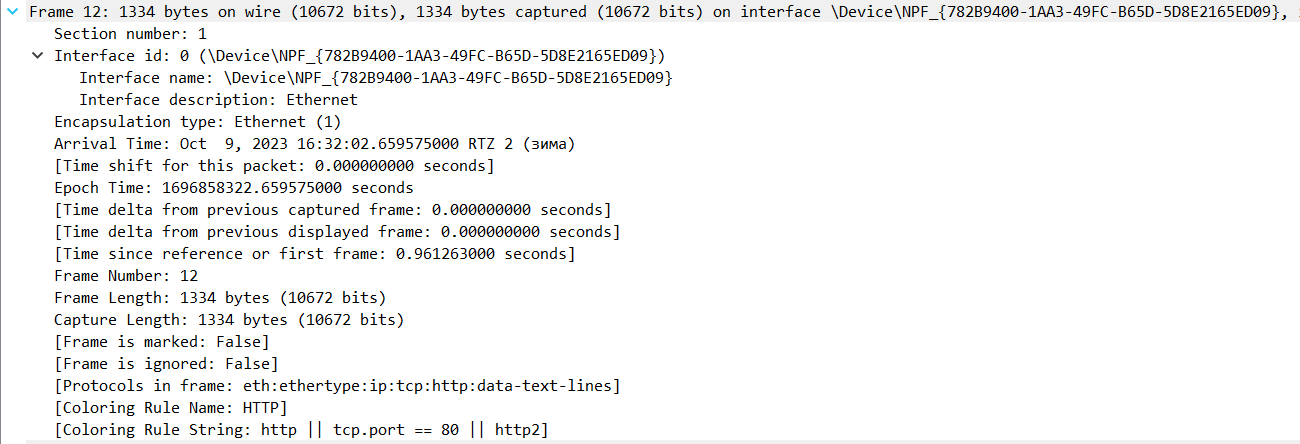
Протокол HTTP версии 1.1. Тип запроса GET. Хост: httpforever.com

Передаются различные заголовки, например, User-Agent (данные об устройстве, запрашивающем страницу), Accept-Encoding (поддерживаемые алгоритмы кодировки).

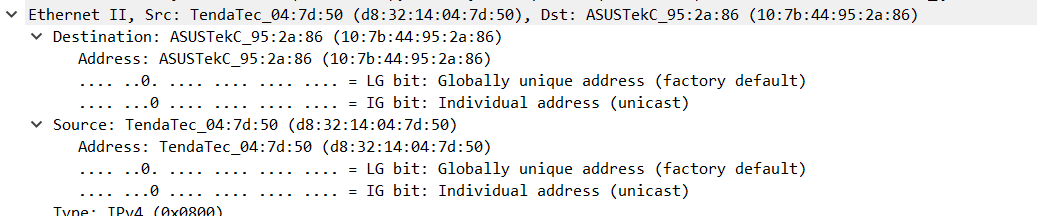
**Ответ**



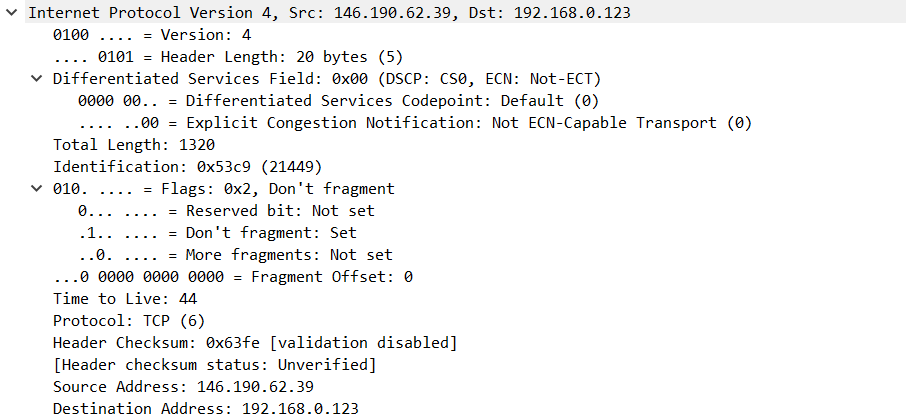
**Физический уровень**



**Канальный уровень**

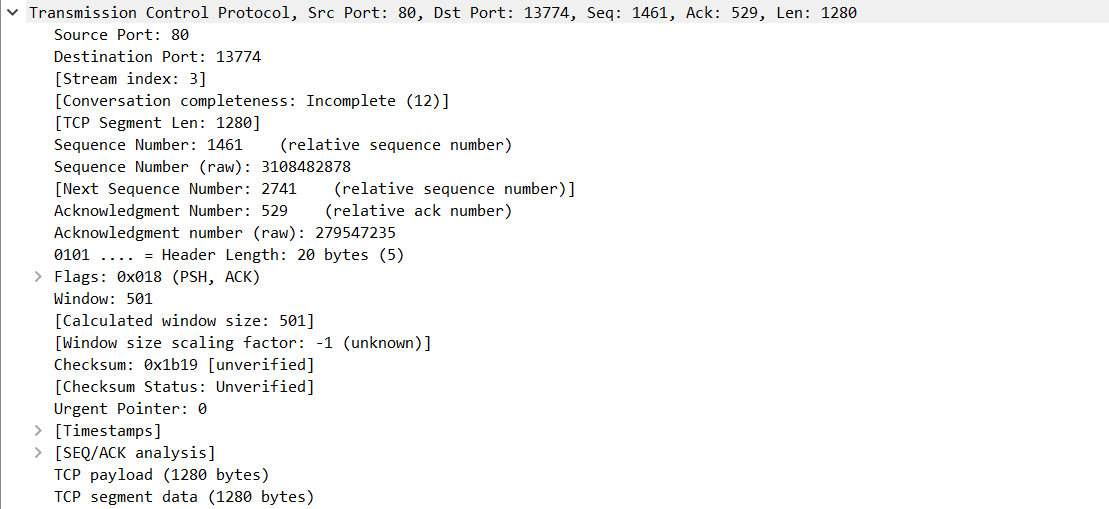


**Сетевой уровень**



Протокол IP. Получатель и отправитель теперь меняются местами

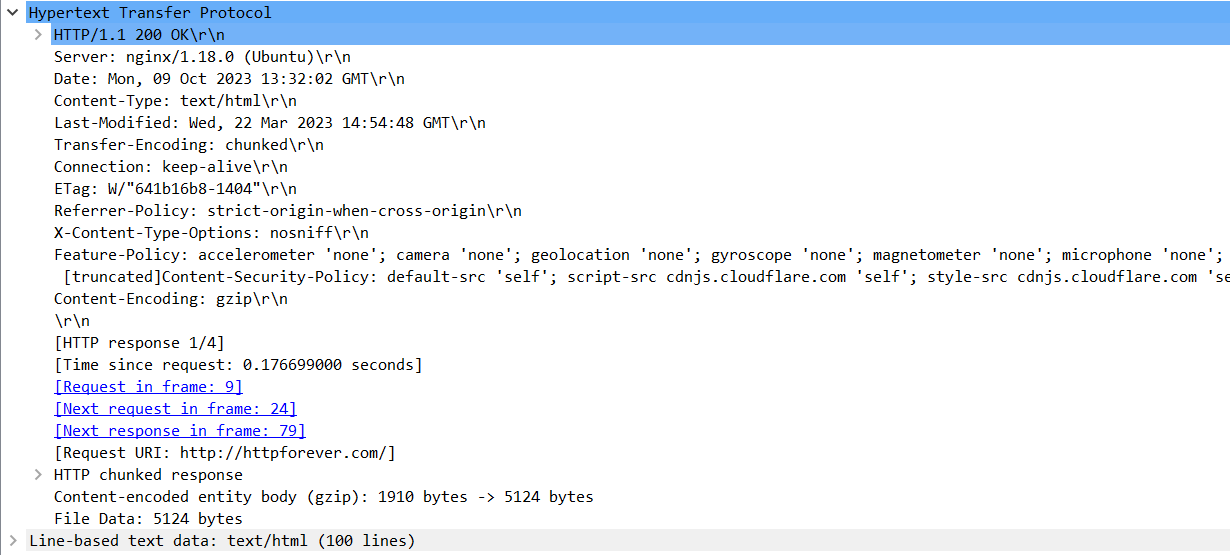
**Транспортный уровень**



Протокол TCP. Порты меняются местами, т.к. поменялись стороны.

Флаги: PSH, ACK (рукопожатие уже было произведено)

**Прикладной уровень**

T

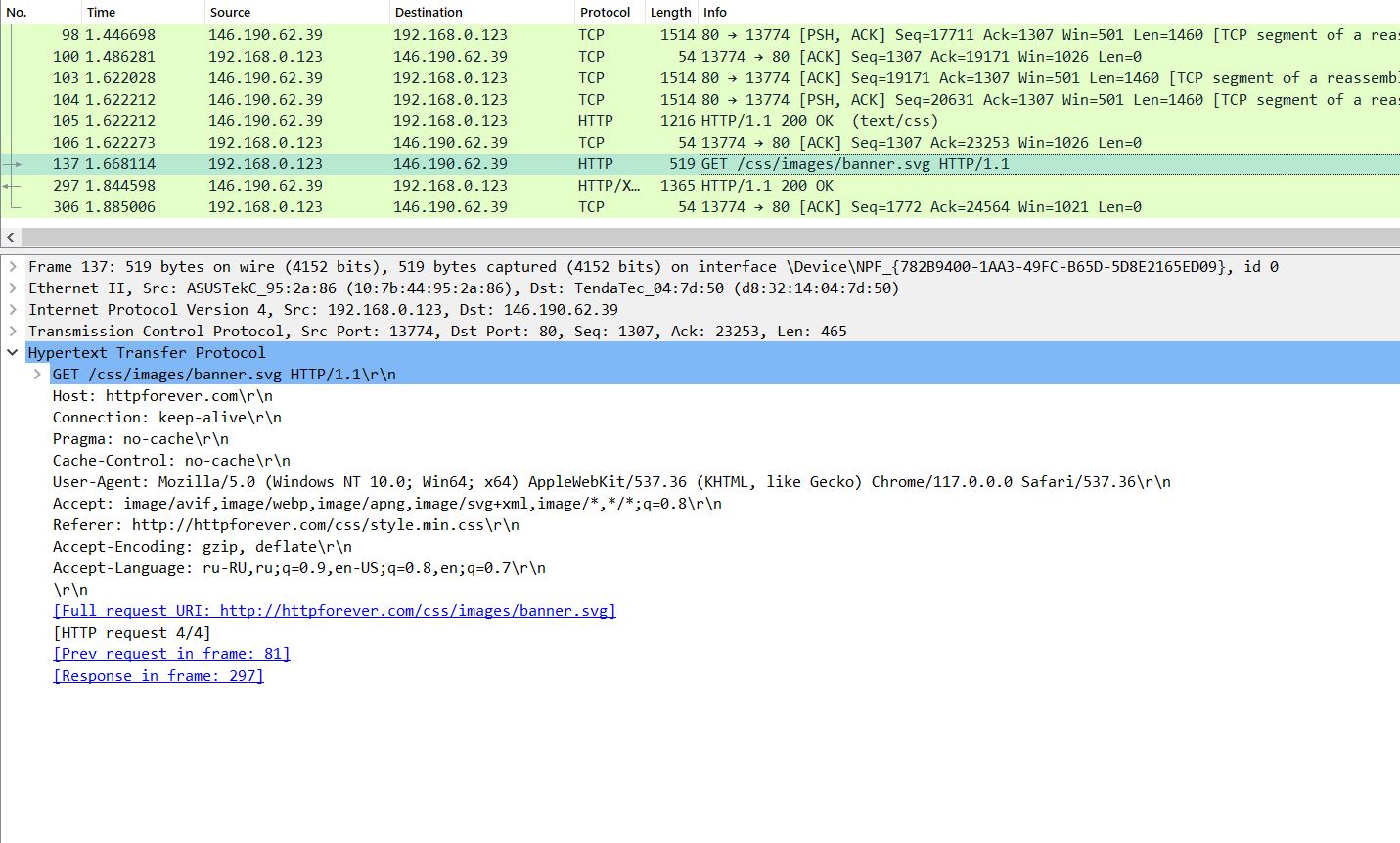
Протокол HTTP версии 1.1. Код ответа 200 (ОК).

Передаются различные заголовки, например, Server (информация о сервере), Content-Type (MIME-тип передаваемого файла. Известен как Media Type или Content Type).

1. Рассмотреть загрузку других ресурсов сайта (CSS/ PNG/ JPG и т.д.) как запрос на данный ресурс, так и ответ.

**Загрузка SVG**

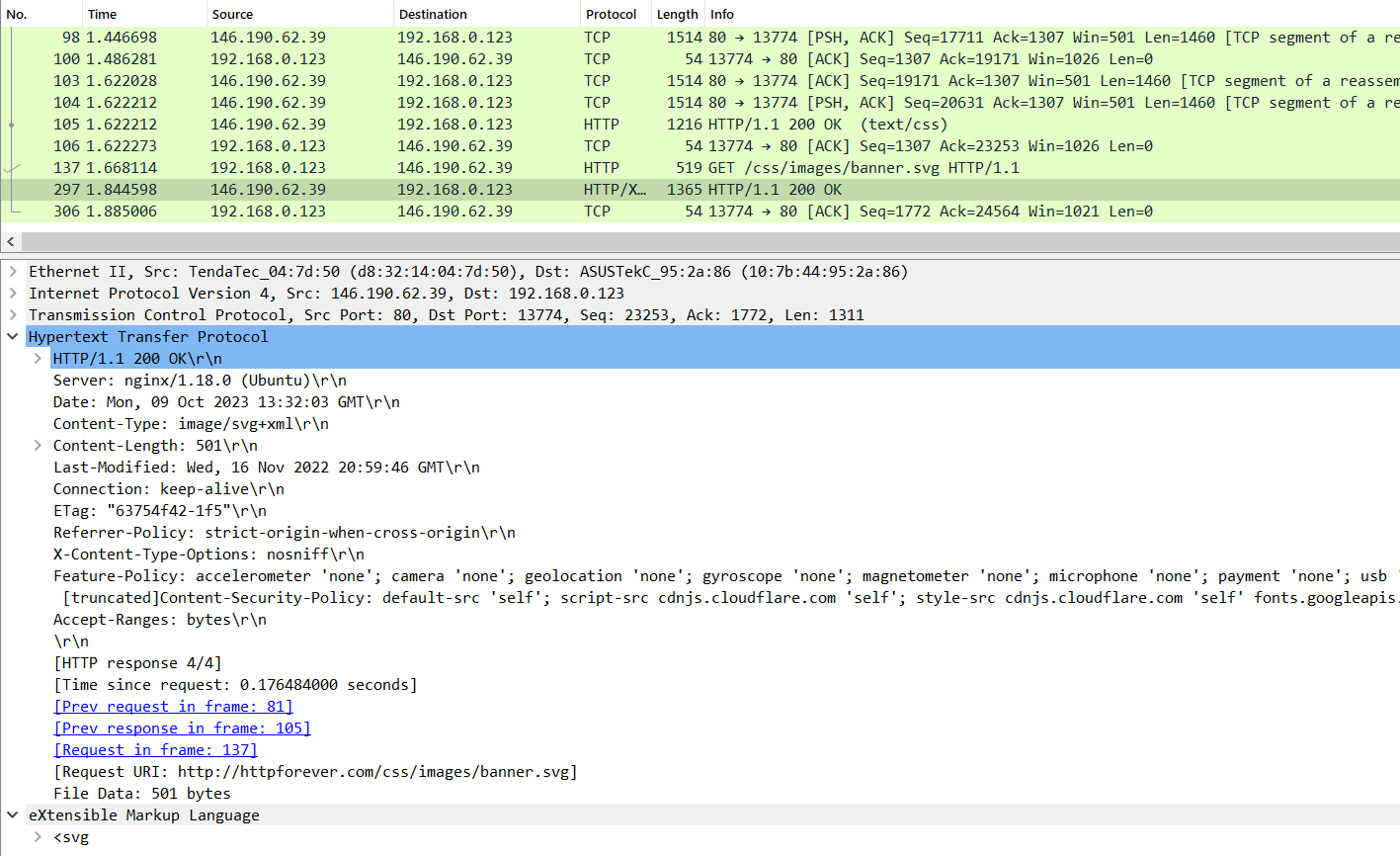
Запрос



Протокол HTTP версии 1.1. Тип запроса GET. Хост: httpforever.com.

Адрес запроса: /css/images/banner.svg

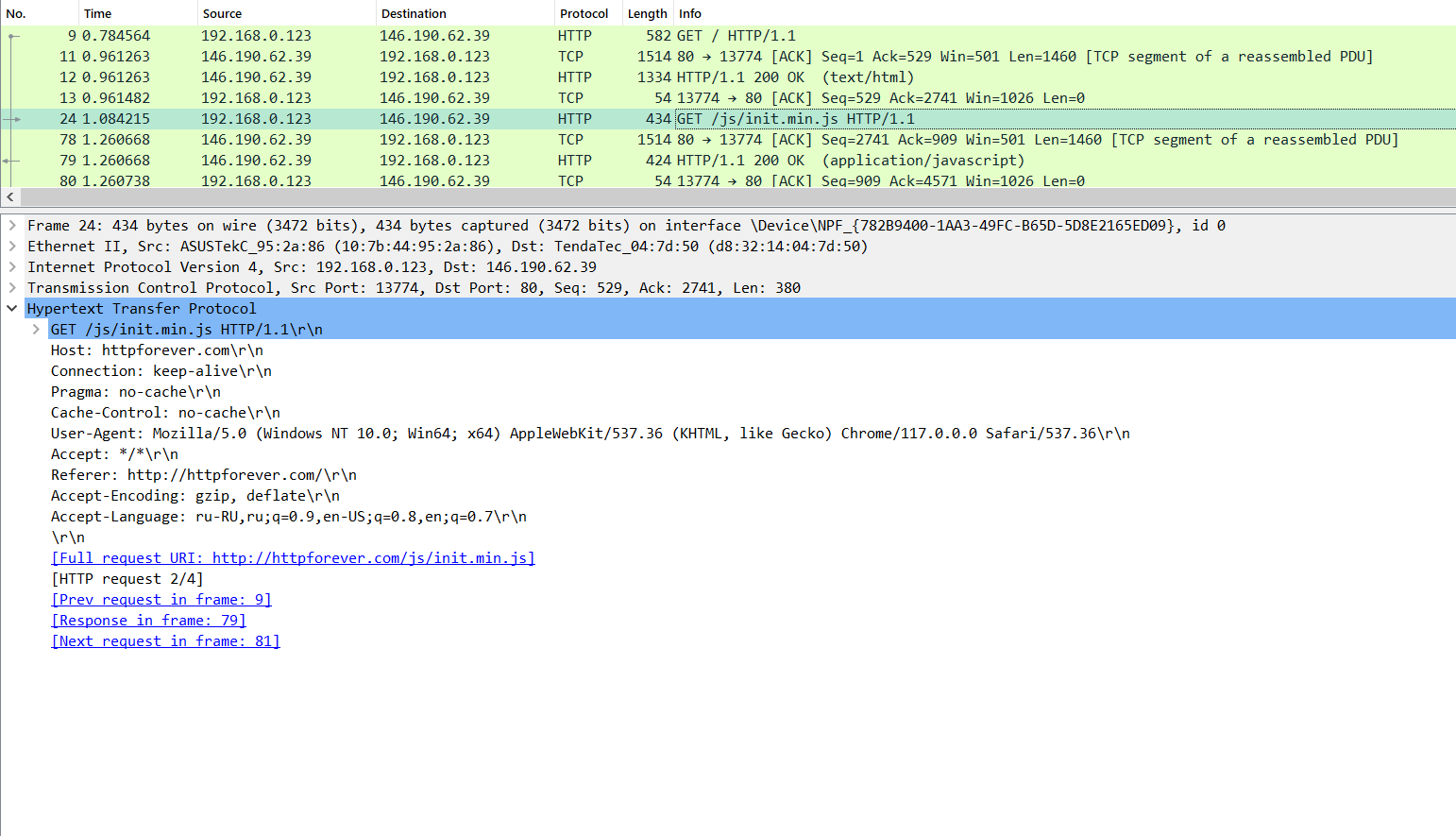
Ответ



Протокол HTTP версии 1.1. Код ответа 200 (ОК). Content-Type: image/svg+xml (векторное изображение)

**Загрузка JS**

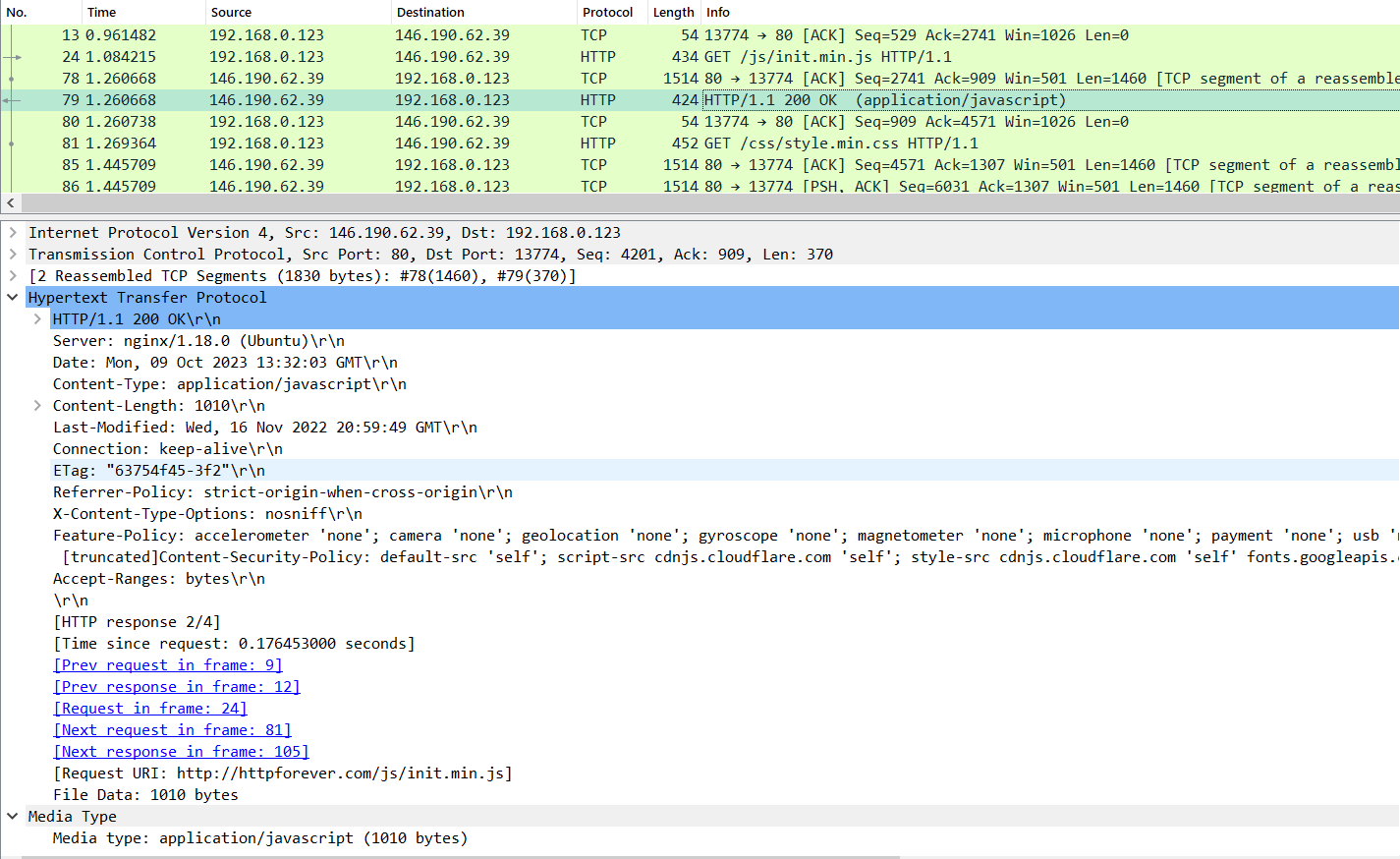
Запрос



Протокол HTTP версии 1.1. Тип запроса GET. Хост: httpforever.com.

Адрес запроса: /js/init.min.js

Ответ



Протокол HTTP версии 1.1. Код ответа 200 (ОК). Content-Type: application/javascript (код на js)