Министерство науки и высшего образования РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное Государственное бюджетное образовательное   
учреждение высшего образования

«Поволжский государсТвенный технологический  
университет» (ФГБОУ ВО «ПГТУ»)

Кафедра Информатики

ОТЧЁТ

по практической работе № 3

**«Анализ протокола HTTP»**

Выполнил: студент группы ИСТ-38

Самойлова Е. В.

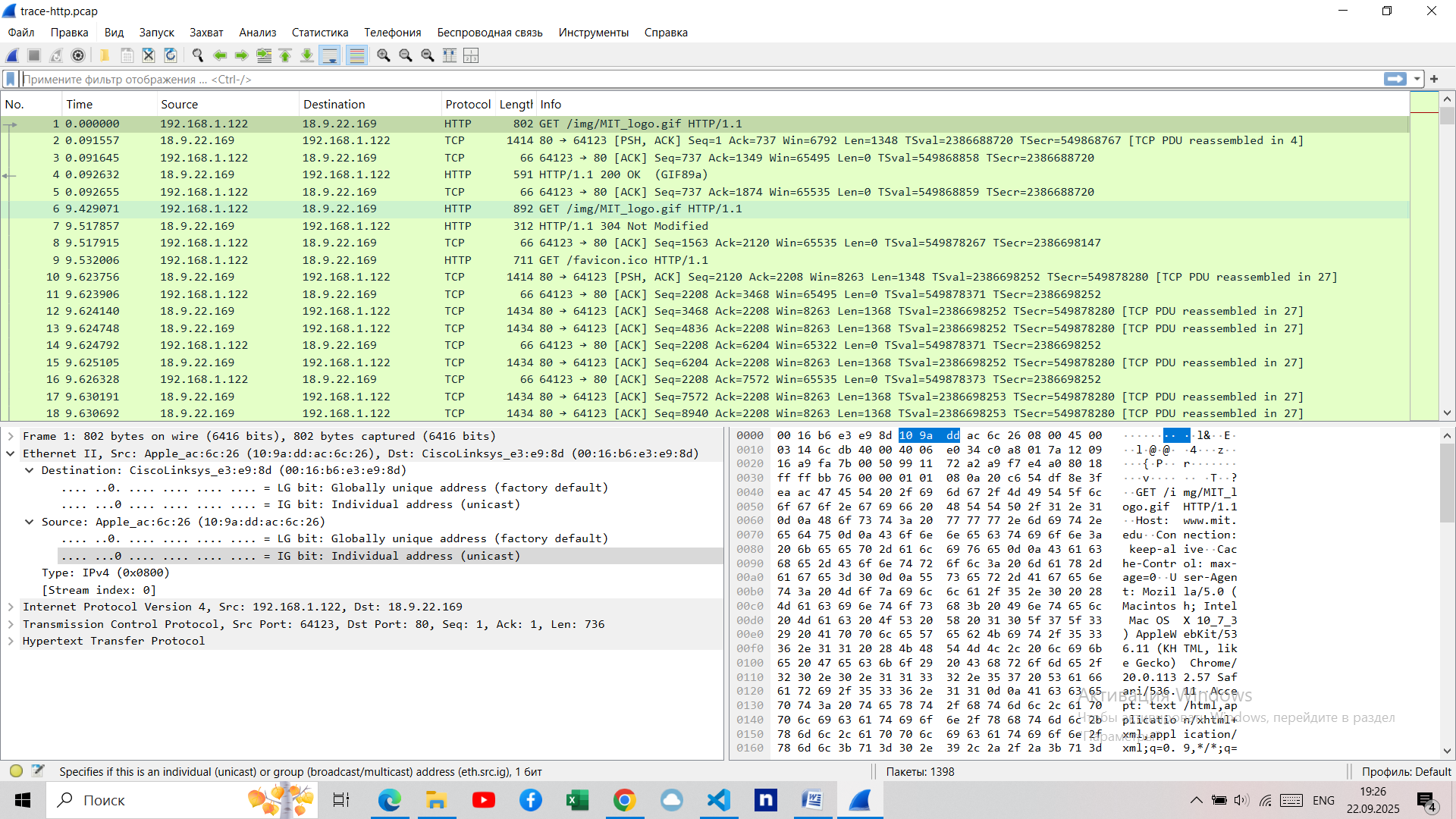
Проверил: преп. каф. Информатики

Уржумов Д. В.

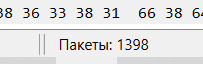
Йошкар-Ола

2025

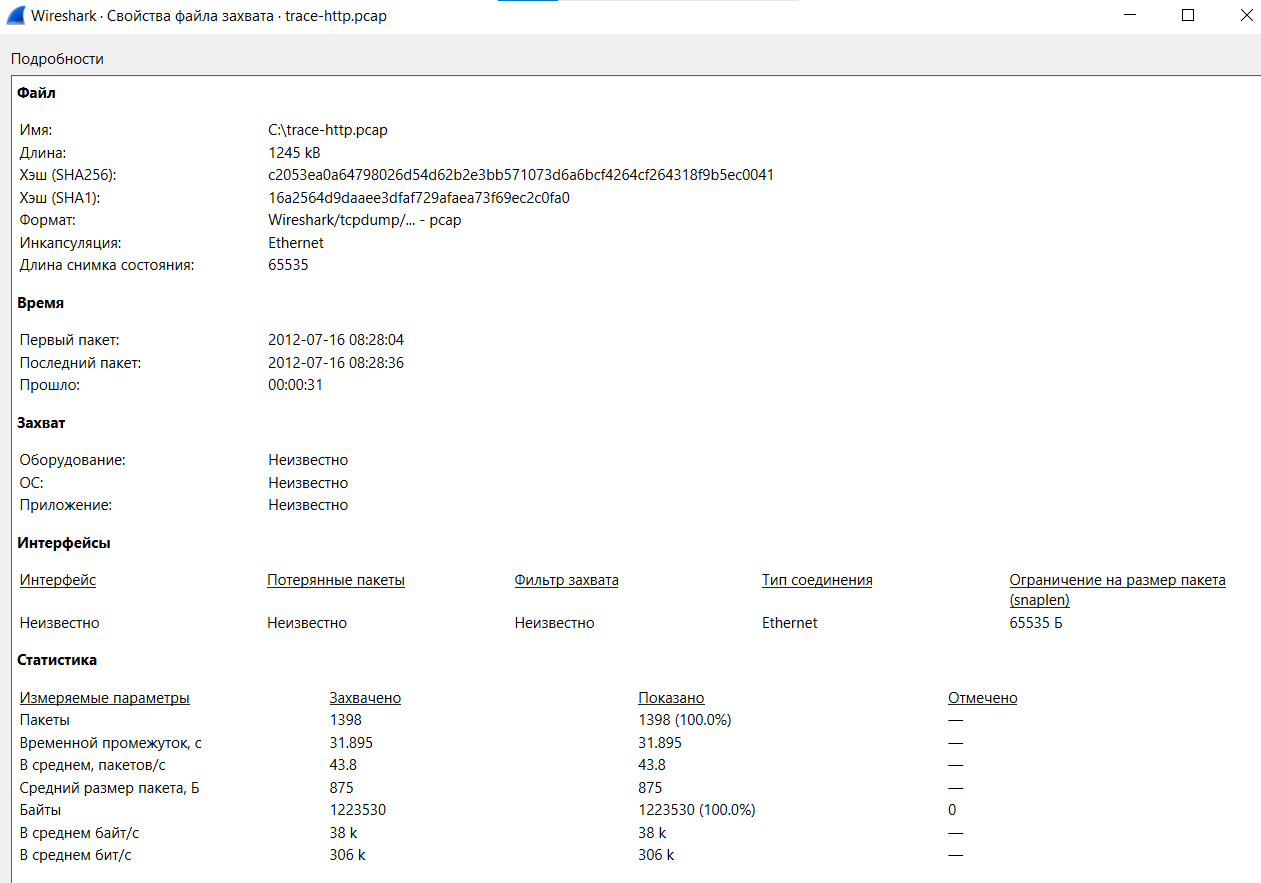
**Задание 1. Загрузка файла трассировки http**



* Количество пакетов, которые были перехвачены Wireshark и присутствуют в этом файле?

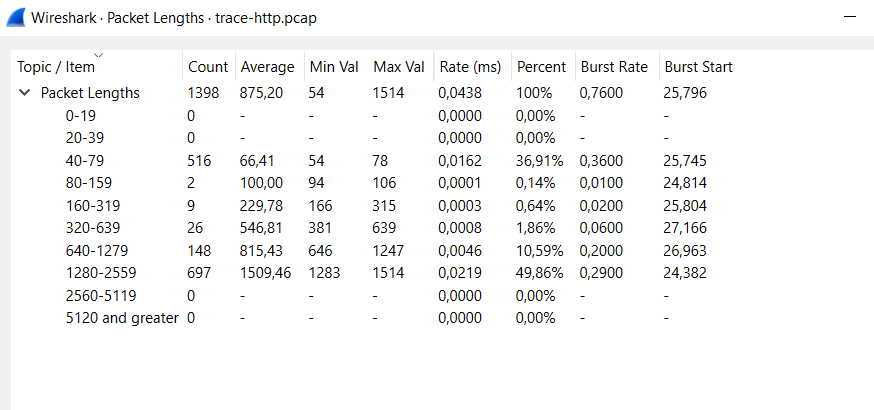


* Какой общий объем перехваченных пакетов (в байтах)?

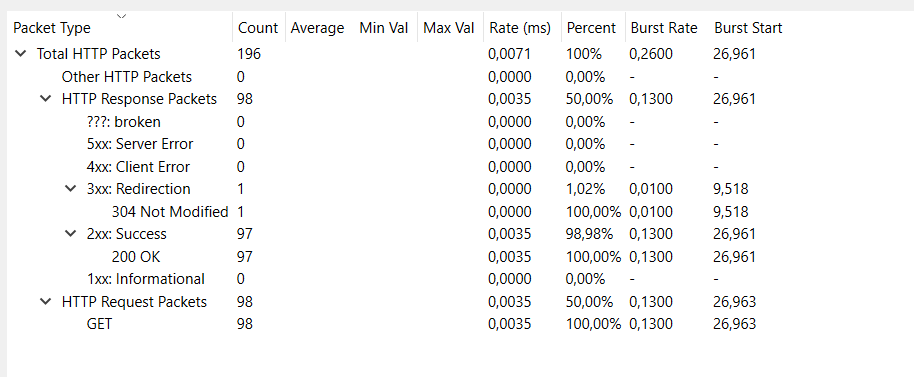


Через Статистику –> Свойства файла захвата видим, что перехваченных пакетов в байтах – 122350.

* Количество пакетов HTTP Request? (98)



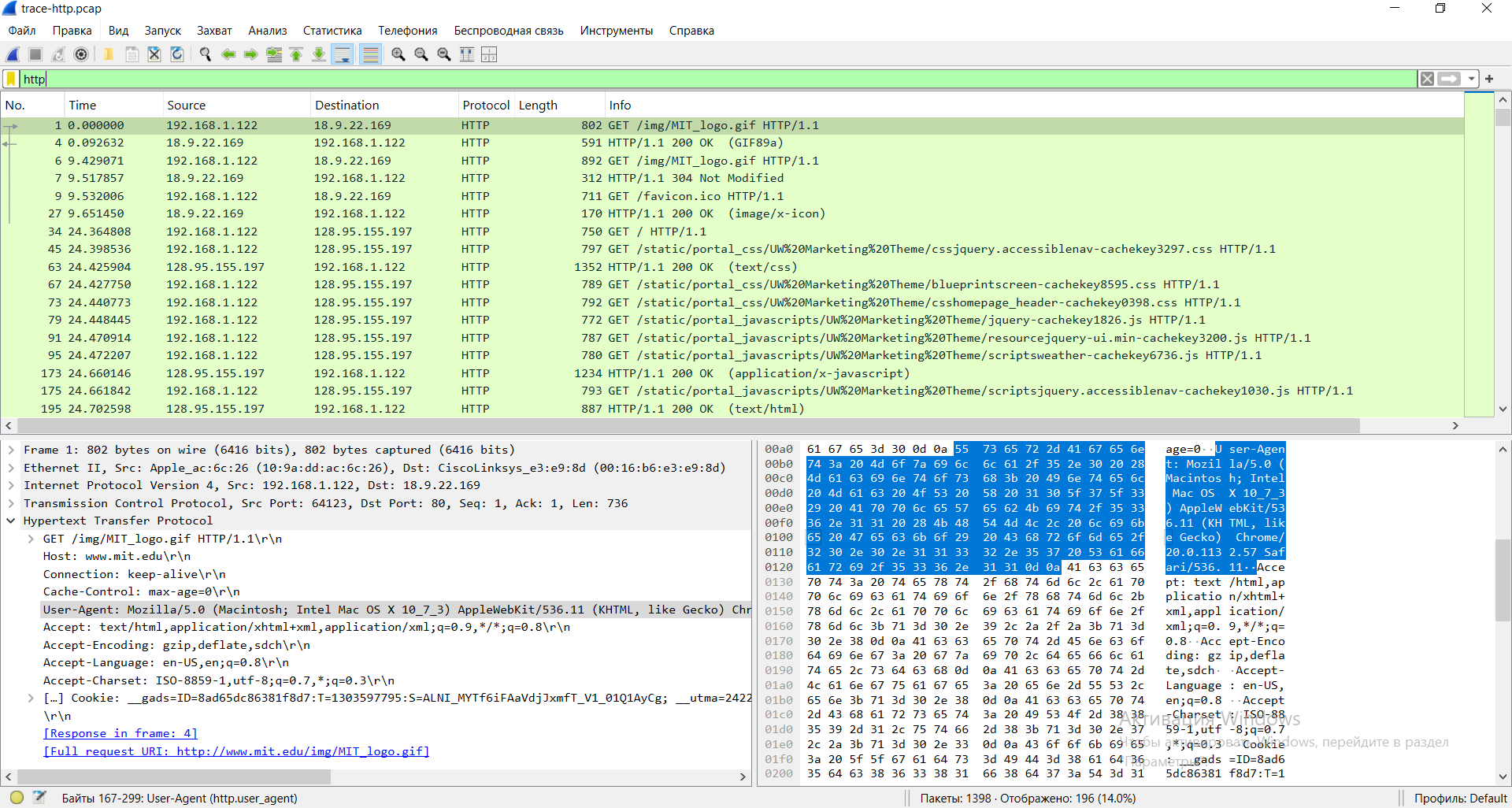
* Количество пакетов HTTP Response? (98)



**Задание 2. Изучение трассировки запроса и ответа**

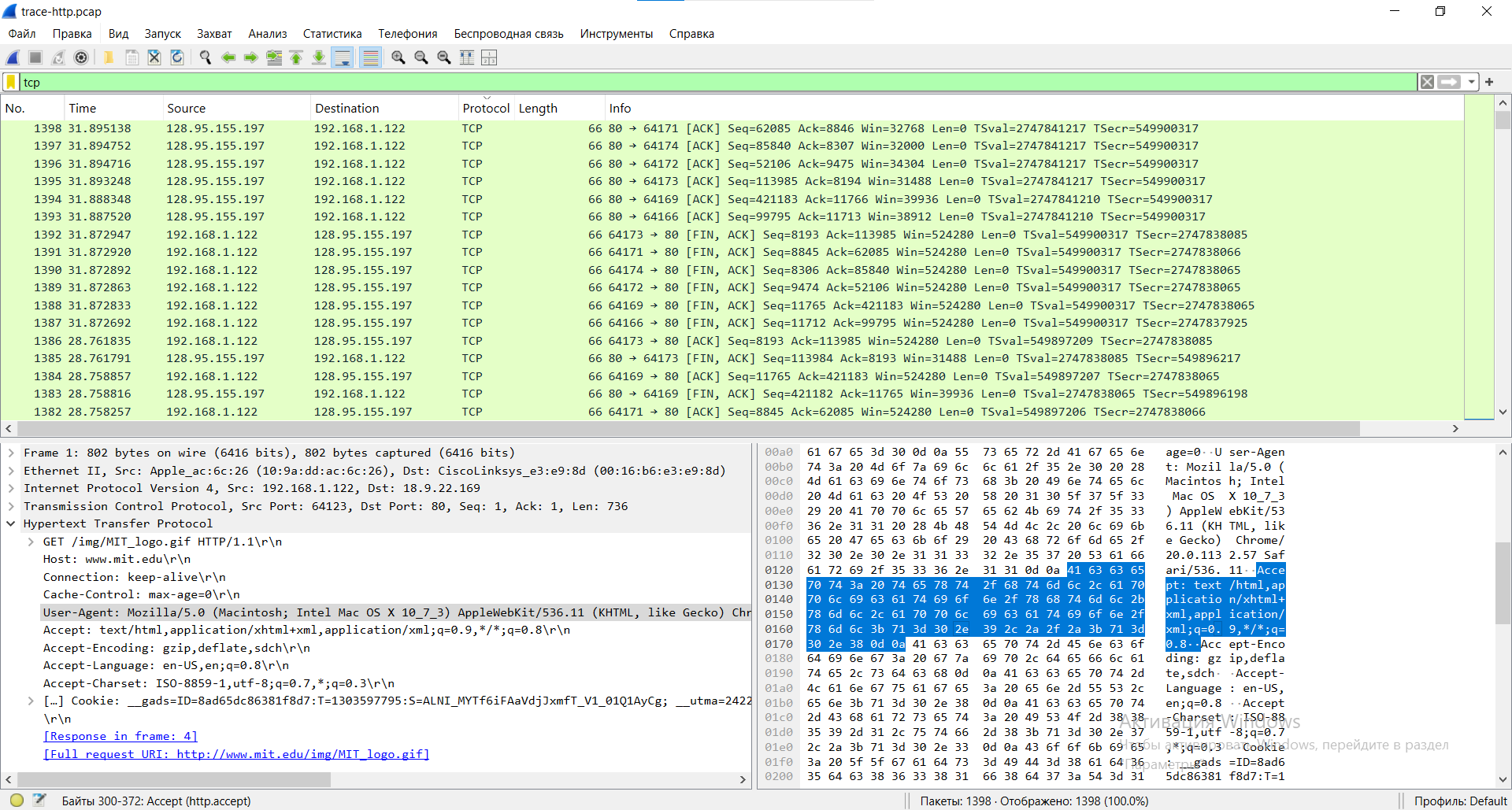
**Применение фильтра**

Чтобы сосредоточиться на трафике HTTP, введем и применим выражение фильтра «http». С помощью выражения фильтра «http» мы скроем промежуточные TCP-пакеты и увидим только HTTP-запросы и ответы.

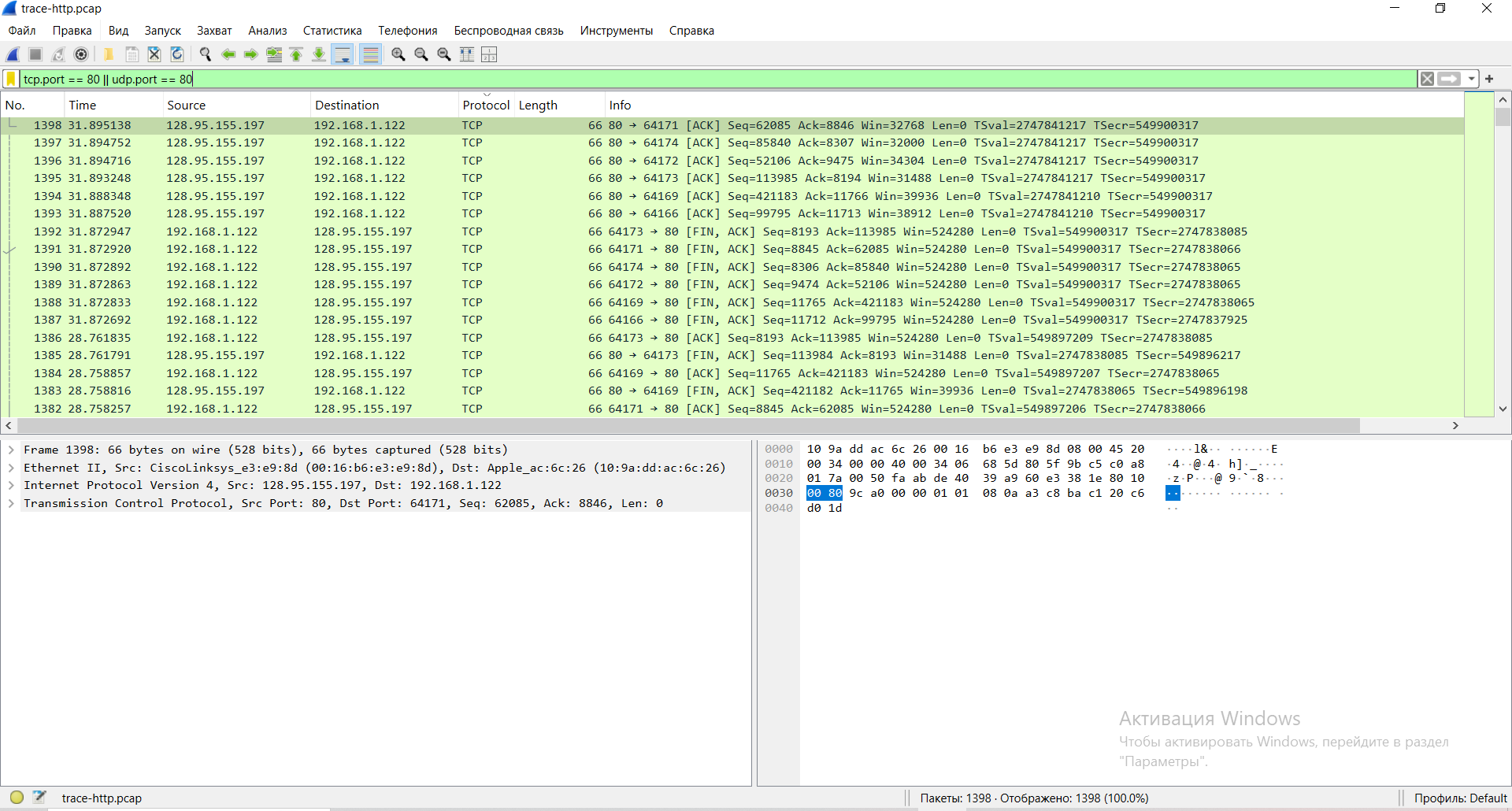


Также выполните фильтрацию по протоколам:

* TCP

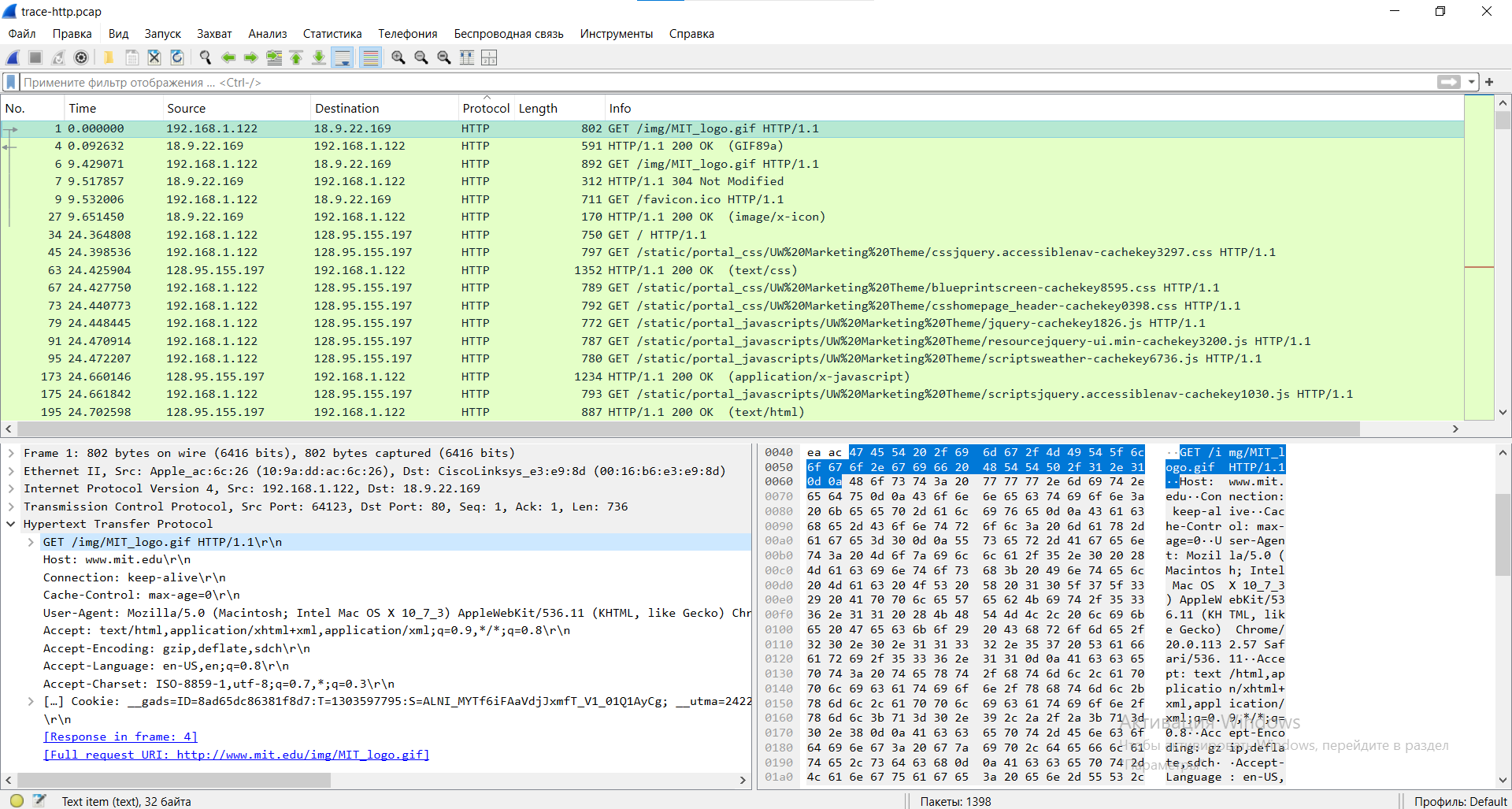


* TCP или UDP с использованием порта 80



**Просмотр GET запроса и изучение HTTP-заголовка**

Выберите первый GET в трассировке и разверните его HTTP-блок:

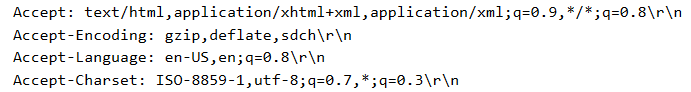


Посмотрите имеются ли в файле трассировки какие-либо из общих заголовков:

* Host. Обязательный заголовок, он идентифицирует имя (и порт) сервера. 
* User-Agent. Вид браузера и его возможности.



* Accept, Accept-Encoding, Accept-Charset, Accept-Language. Описания форматов, которые будут приняты в ответе, например, text/html, включая его кодировку, например, gzip, и язык.



* Cookie. Имя и значение cookie-файлов, которые браузер хранит для веб-сайта.

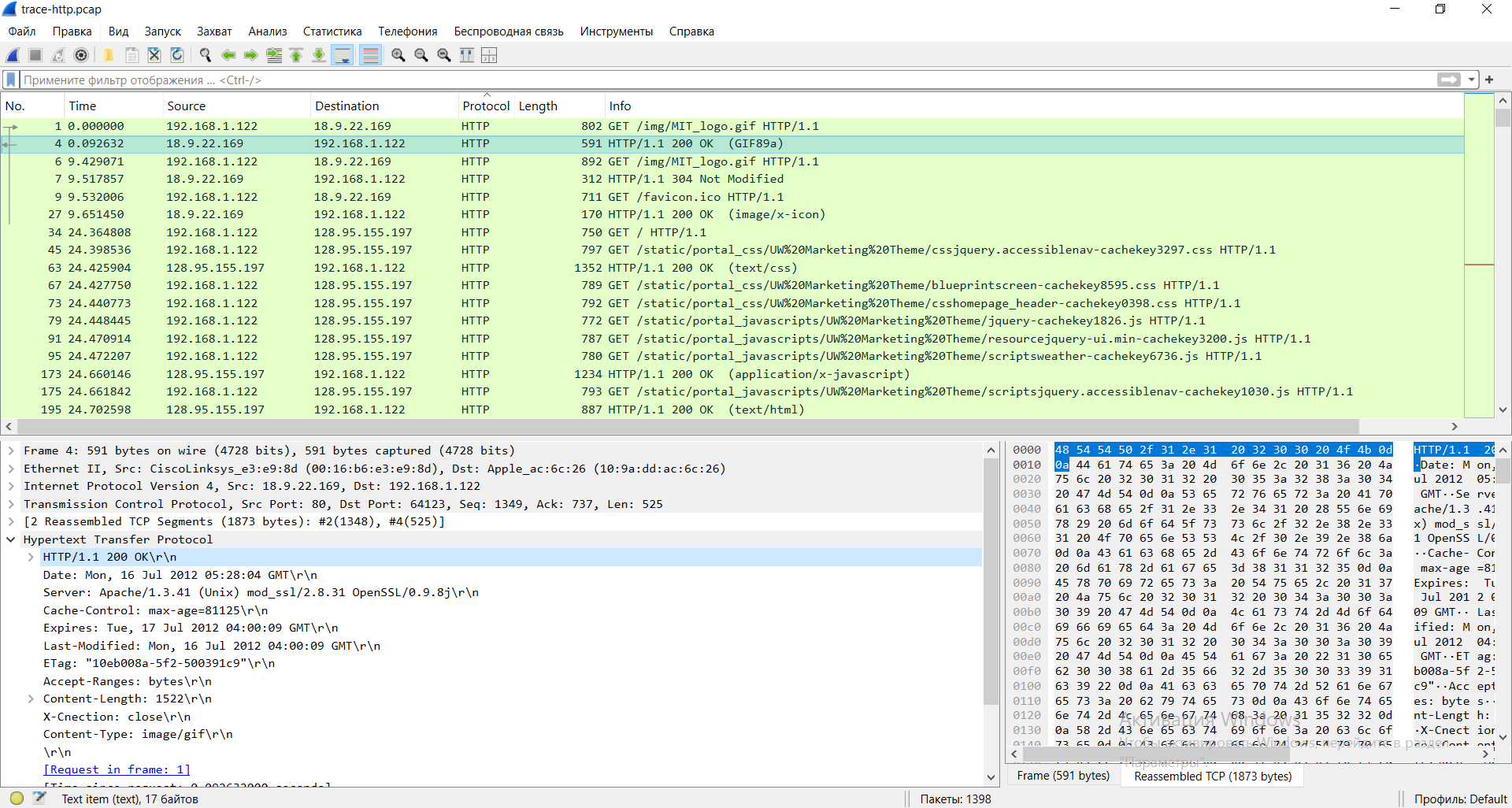


* Cache-Control. Информация о том, как ответ может быть кэширован.



**Просмотр GET ответа**

Выберите ответ, который соответствует первому GET в трассировке, и разверните его HTTP-блок:



Информация для этого пакета будет указывать «200 OK» в случае нормальной, успешной передачи.

Изучите общие заголовки, такие как:

* Server. Тип сервера и его возможности.
* Date, Last-Modified. Время ответа и время последнего изменения контента.
* Cache-Control, Expires, Etag. Информация о том, как можно кэшировать ответ.

В качестве примера рассмотрим первый GET-запрос в трассировке (HTTP/1.1) и соответствующий ему ответ с кодом «200 OK».

**Полезная информация в заголовках ответа:**

* Server: Apache/1.3.41 (Unix) mod\_ssl/2.8.31 OpenSSL/0.9.8j

Этот заголовок указывает на тип и версию программного обеспечения, работающего на веб-сервере. В данном случае это веб-сервер Apache версии 1.3.41, установленный на операционной системе Unix. Также указаны установленные модули: mod\_ssl версии 2.8.31 для поддержки HTTPS и библиотека OpenSSL версии 0.9.8j.  Эта информация может быть полезна администраторам для диагностики проблем, связанных с конкретными версиями ПО, но часто скрывается в целях безопасности.

* Date: Mon, 16 Jul 2012 05:28:04 GMT

Указывает дату и время, когда ответ был сгенерирован сервером по Гринвичскому времени (GMT). Это важно для корректного кэширования и синхронизации времени между клиентом и сервером.

* Last-Modified: Mon, 16 Jul 2012 04:00:09 GMT

Содержит дату и время последнего изменения запрашиваемого ресурса (например, HTML-страницы). Браузер использует эту метку в последующих запросах в заголовке If-Modified-Since, чтобы проверить, нужно ли загружать ресурс заново.

* Cache-Control: max-age=81125

Директива, управляющая кэшированием. Значение max-age=81125 указывает браузеру, что он может хранить полученный ресурс в своем кэше в течение 81125 секунд (примерно 22 часа и 32 минуты). В течение этого времени браузер будет использовать локальную копию, не отправляя запрос на сервер.

* Expires: Tue, 17 Jul 2012 04:00:09 GMT

Устаревший, но поддерживаемый заголовок, который указывает конкретную календарную дату и время истечения срока действия кэшированной копии. В данном случае срок действия истекает 17 июля 2012 года в 04:00:09. Если присутствует Cache-Control: max-age, браузер отдает ему приоритет.

* Etag: "10eb008a-5f2-500391c9"

Это уникальный идентификатор (хэш) текущей версии ресурса. Если ресурс изменяется, его Etag также меняется. При повторном запросе браузер может отправить заголовок If-None-Match со значением предыдущего Etag. Сервер сравнит их и, если они совпадают, отправит статус 304 Not Modified, экономя трафик.

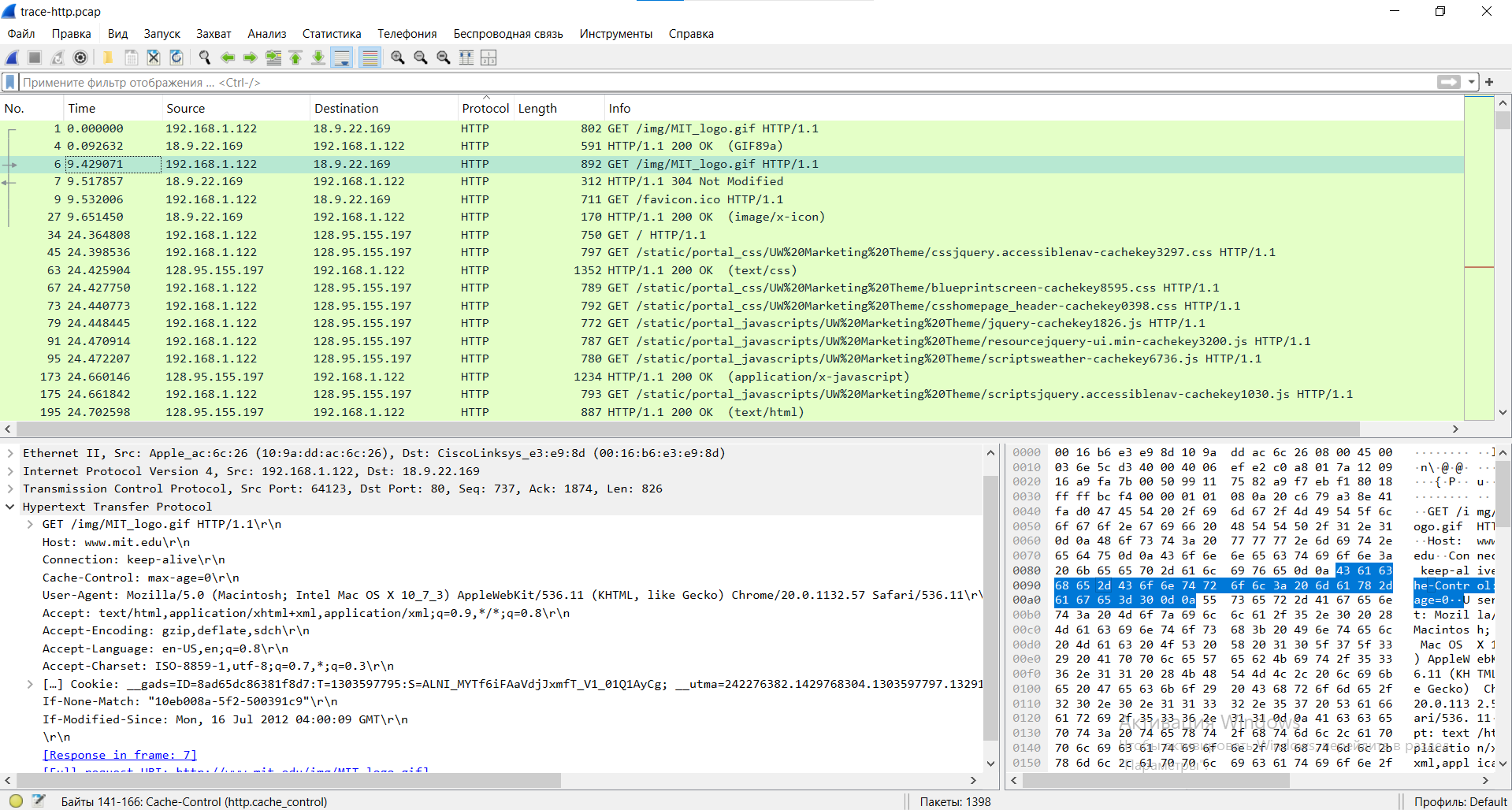
* Accept-Ranges: bytes

Сообщает клиенту, что сервер поддерживает докачку файлов (запросы с заголовком Range). Это позволяет возобновить прерванную загрузку больших файлов (например, видео) с места обрыва, а не начинать её заново.

**Задание 3. Кэширование контента**

Второй блок записей в трассировке — это повторная выборка по первому URL. Этот блок дает нам возможность рассмотреть кэширование в действии, поскольку весьма вероятно, что изображение или документ не изменились и, следовательно, не нуждаются в повторной загрузке. Механизмы кэширования HTTP должны предоставить такую возможность.

Щелкните на третью строку в файле трассировки, т.е. на 892 GET /img/MIT\_logo.gif HTTP 1.1 и разверните HTTP-блок в нижней панели:

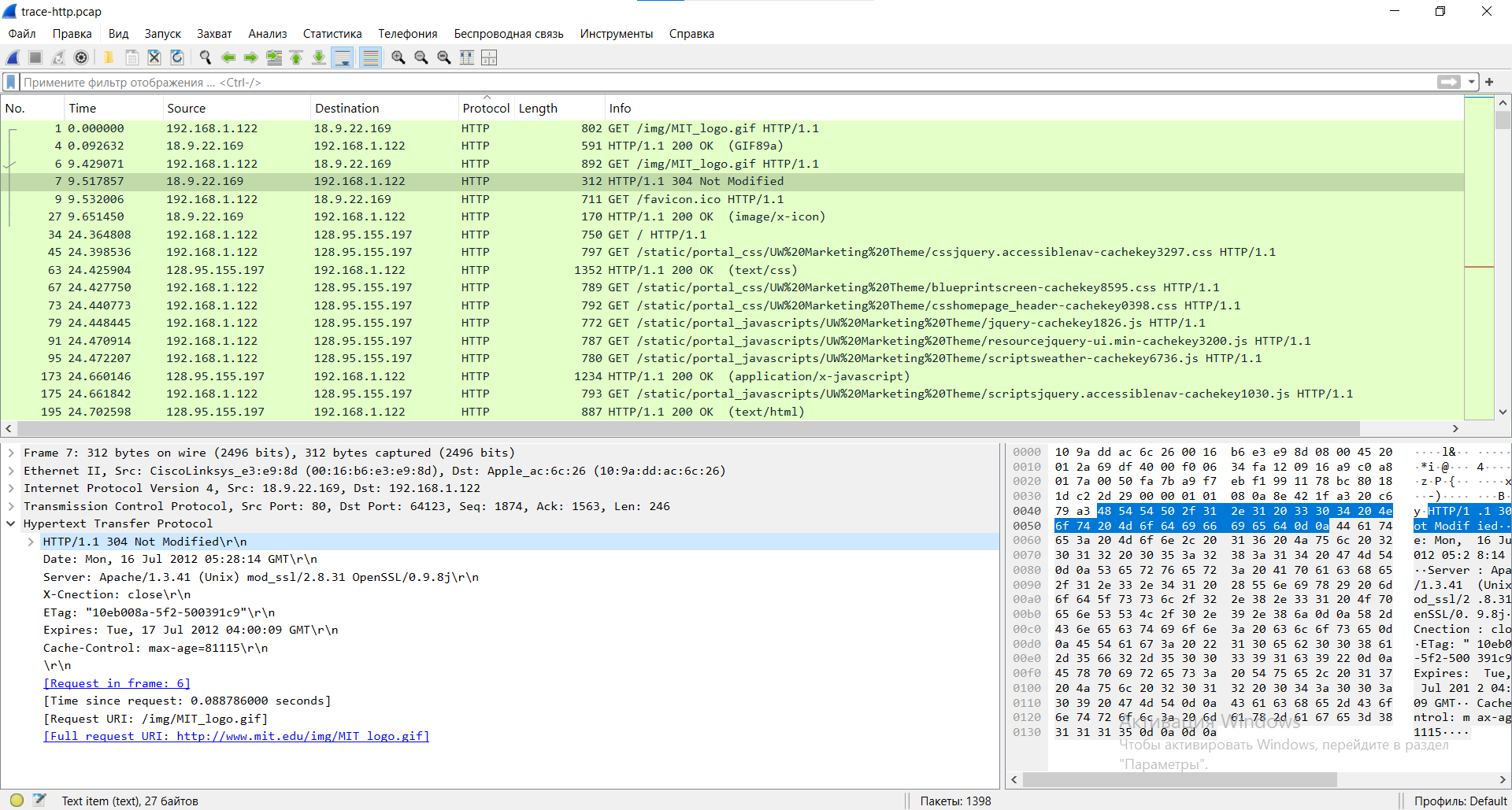


Найдите заголовок, который позволит серверу решить, нужно ли отправлять свежий контент. Серверу нужно будет отправлять свежий контент только в том случае, если контент изменился с момента последней загрузки браузером. Чтобы сделать это, браузер включает временную метку, взятую из предыдущей загрузки для контента, который он кэшировал. Этот заголовок отсутствовал в первом GET, поскольку кэш браузера был очищен, поэтому у браузера не было предыдущей загрузки контента, который он мог бы использовать. В большинстве других отношений этот запрос будет таким же, как и первый запрос.

Заголовок называется «If-Modified-Since», т.е. он просит сервер отправить контент, если он был изменен с определенного времени.



Теперь выберем ответ на повторную выборку и развернем его HTTP-блок.

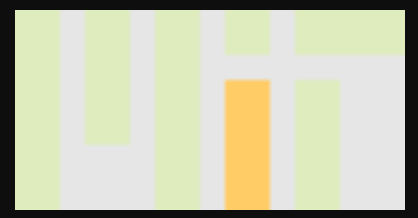


Мы видим, что кэширование сработало, как и ожидалось, этот ответ не будет содержать контент. Вместо этого код статуса ответа — «304 Not Modified». Это сообщает браузеру, что контент не изменился по сравнению с предыдущей копией, и кэшированный контент затем может быть отображен.

**Описание заголовков для строки трассировки 7:**

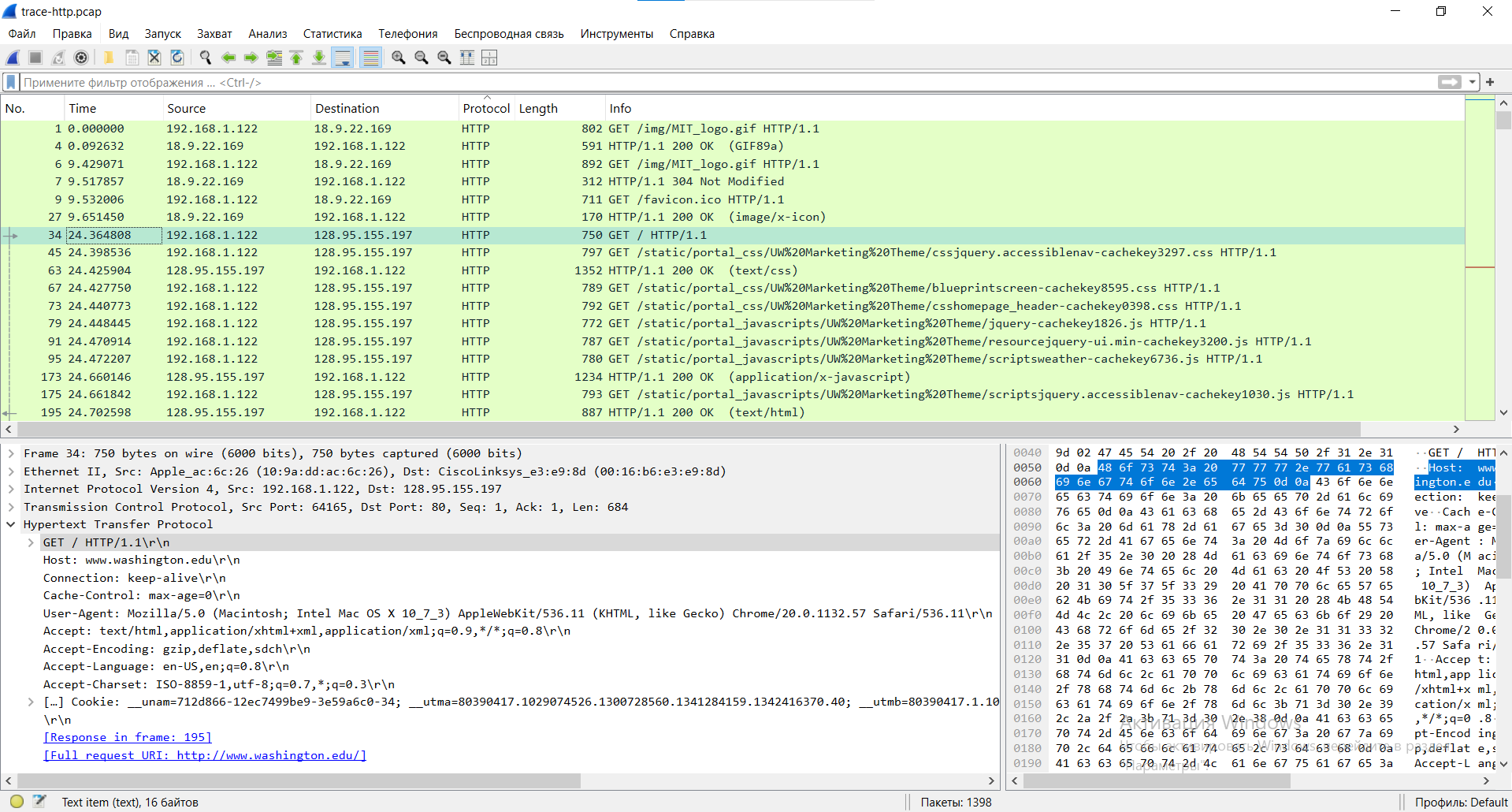
* Date: Mon, 16 Jul 2012 05:28:14 GMT — Время, когда сервер сгенерировал данный ответ.
* Server: Apache/1.3.41 (Unix)... — Информация о сервере, аналогичная рассмотренной в Задании 2.
* ETag: "10eb008a-5f2-500391c9" — Сервер подтверждает уникальный идентификатор версии ресурса. Браузер отправил этот же ETag в заголовке If-None-Match (на основе предыдущего ответа 200 OK). Совпадение значений ETag является основанием для отправки статуса 304.
* Expires: Tue, 17 Jul 2012 04:00:09 GMT и Cache-Control: max-age=81115 — Сервер обновляет информацию о кэшировании, указывая новый срок актуальности кэшированной копии (в данном случае ~22.5 часа с момента ответа). Браузер обновит таймер для этого ресурса.





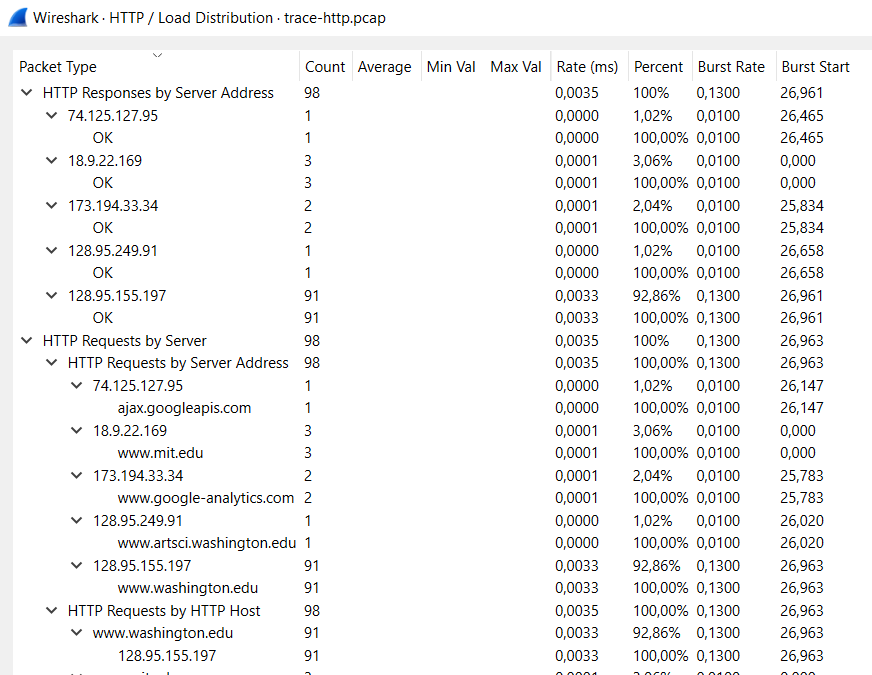
**Задание 4. Сложные страницы**

Теперь мы рассмотрим четвертую выборку в трассировке. Эта выборка была для более сложной веб-страницы, которая, скорее всего, будет иметь встроенные ресурсы. Поэтому браузер загрузит исходный HTML плюс все встроенные ресурсы, необходимые для отображения страницы, плюс другие ресурсы, которые запрашиваются во время выполнения сценариев страницы. В этой выборке мы увидим, что одна страница может включать много GET. Щелкните по номеру 34, который является 750 GET / HTTP/1.1:



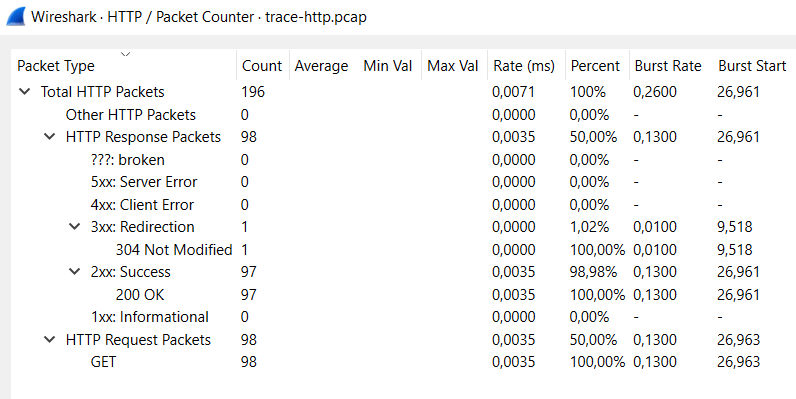
Обобщите GET для этой страницы, вызвав панель HTTP Load Distribution. Эту панель можно найти в разделах Статистика и HTTP.

На этой панели показана информация, сколько запросов было сделано на какие серверы. При серфинге есть вероятность, что fetch запросит контент с других серверов, о которых, мы могли и не подозревать. Эти другие серверы могут включать третьи стороны, такие как сети распространения контента, рекламные и аналитические сети. Наша панель показана ниже — при извлечении страницы было сделано 98 запросов на 5 разных серверов.



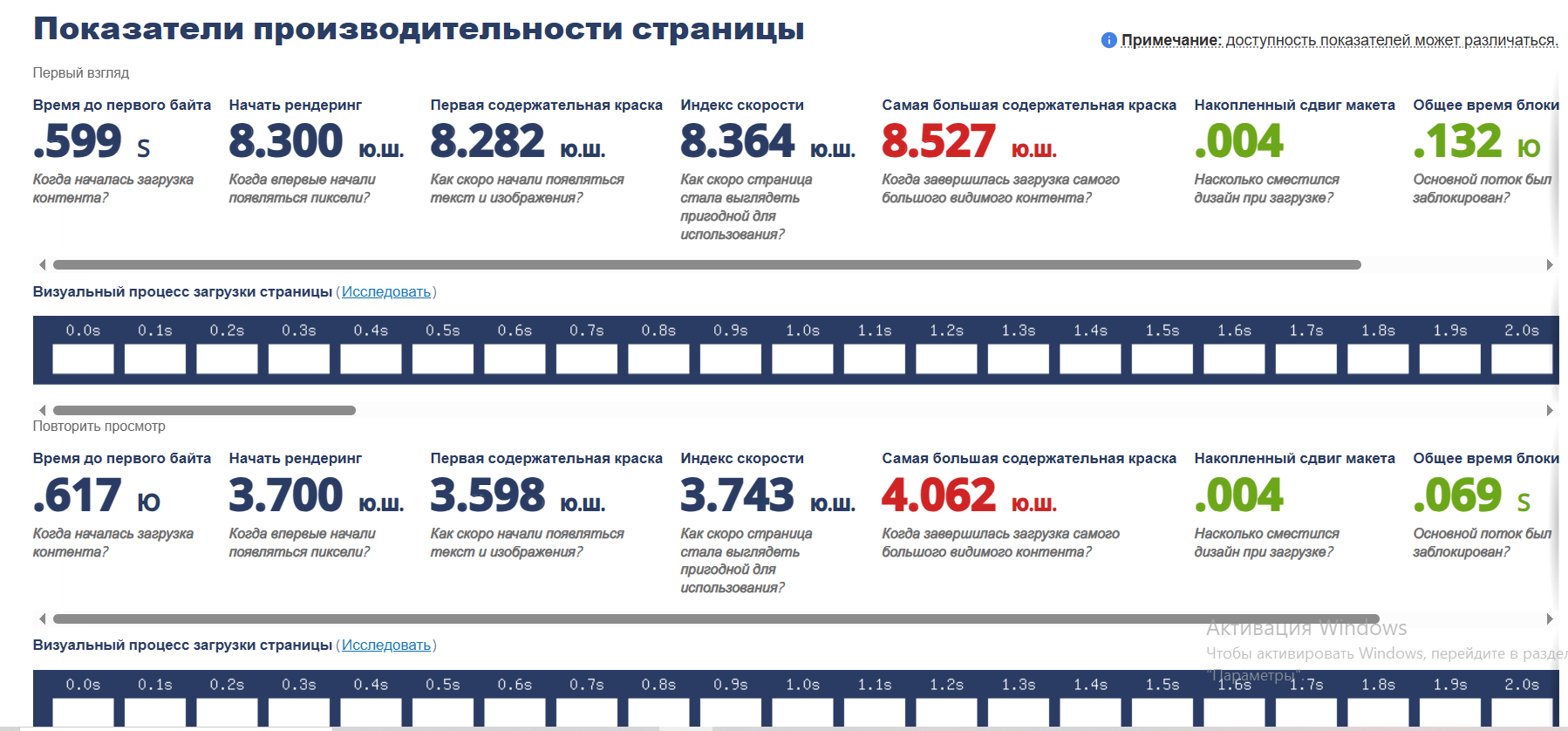
Для получения другого вида сводки GET-запросов откройте панель счетчика HTTP-пакетов.

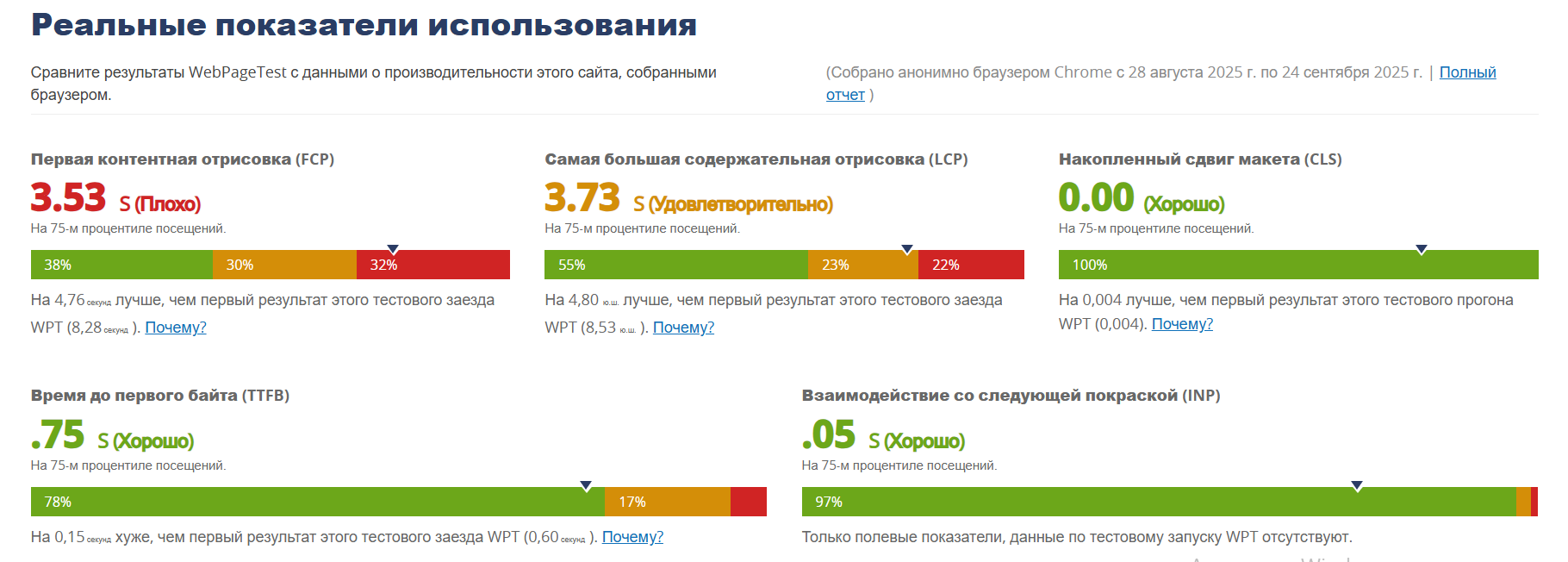
На панели отображаются типы запросов и ответов. Она состоит из запросов GET, которые соответствуют ответам 200 OK и 304 Not Modified. Однако есть множество других кодов ответов, которые наблюдаются в файле трассировки, например, когда ресурс уже кэширован.

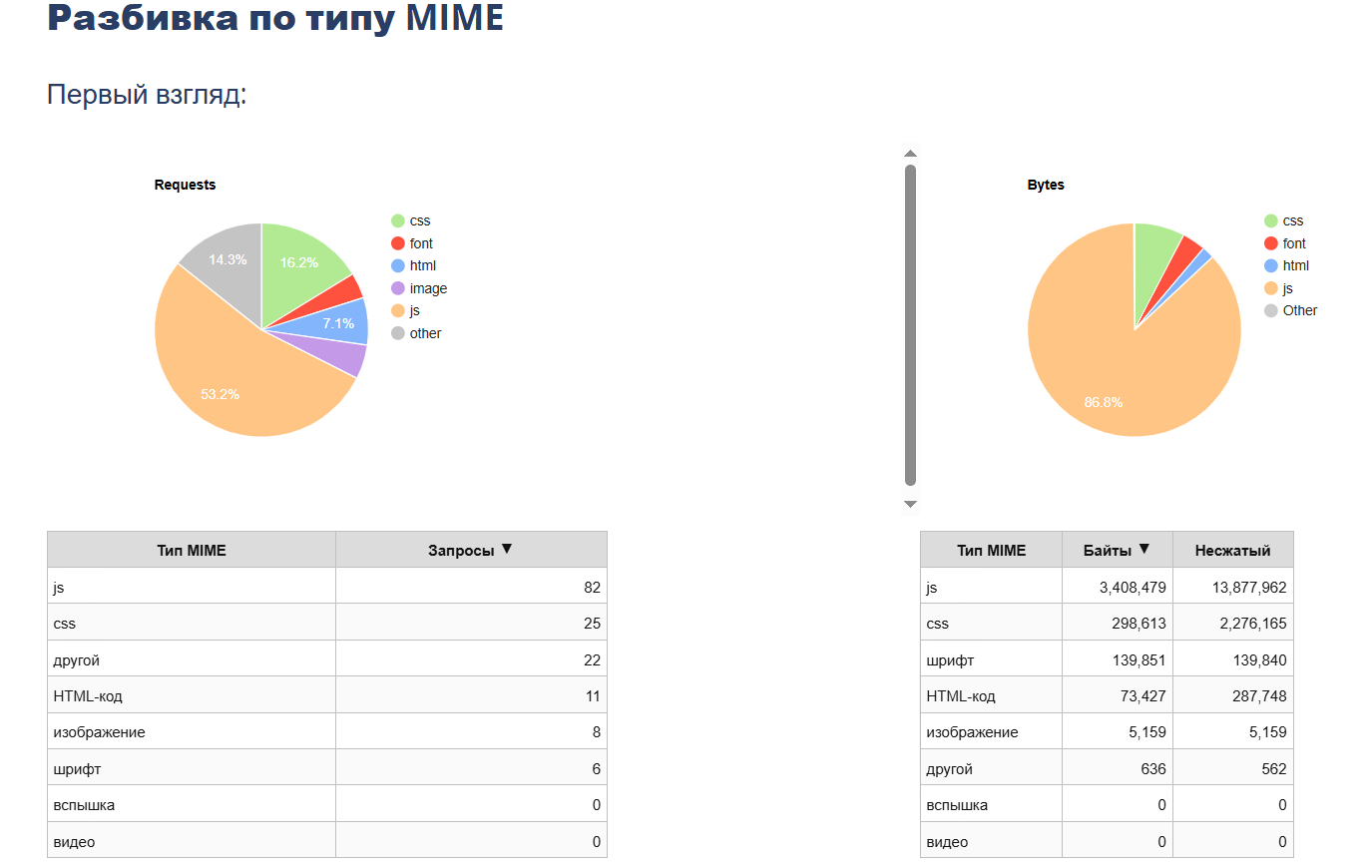


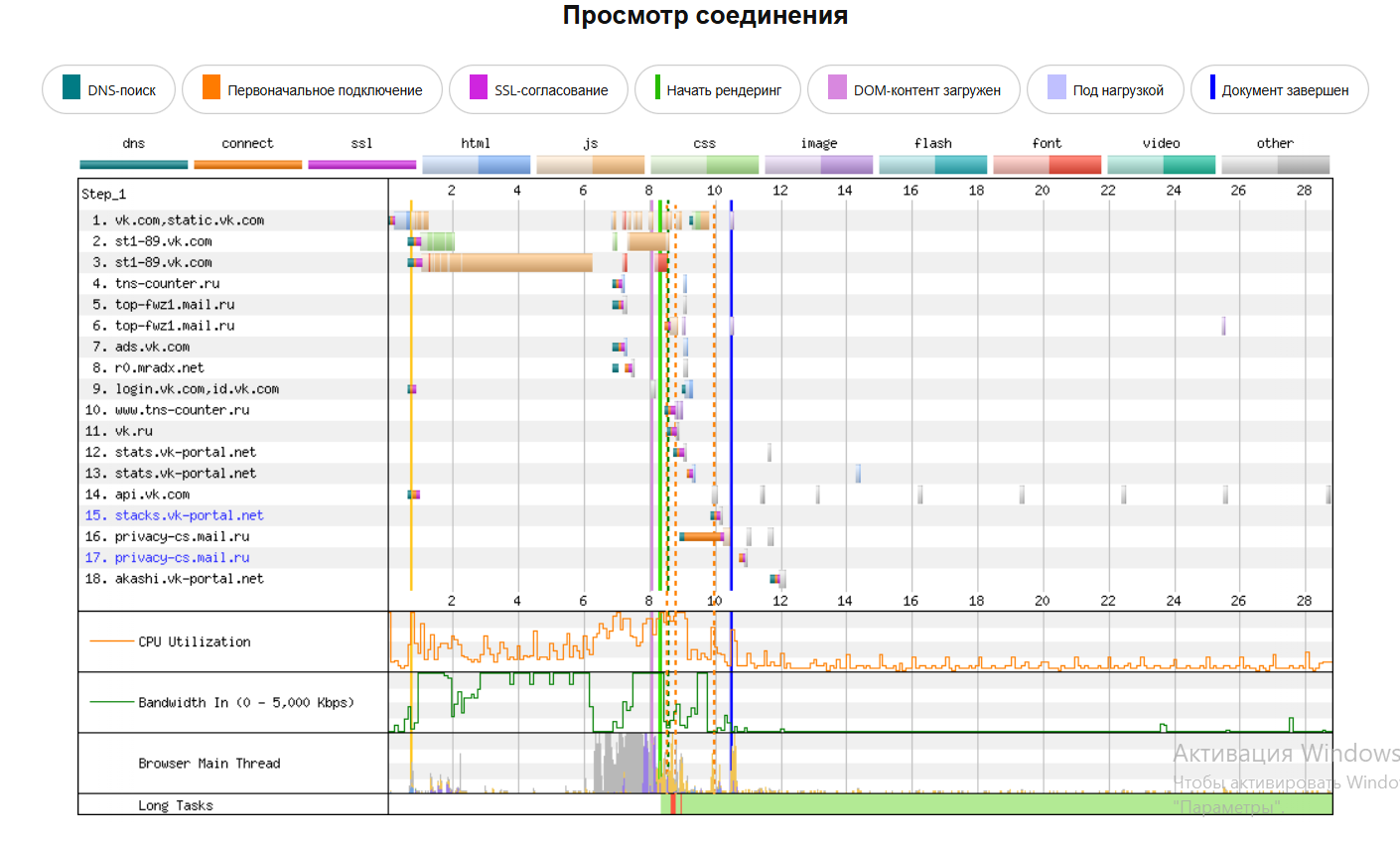
Попробуйте выполнить тестирование Интернет-ресурса, с которым мы работали при выполнении первой практической работы (сетевые утилиты) с помощью сервиса PageSpeed или <https://www.webpagetest.org/>.

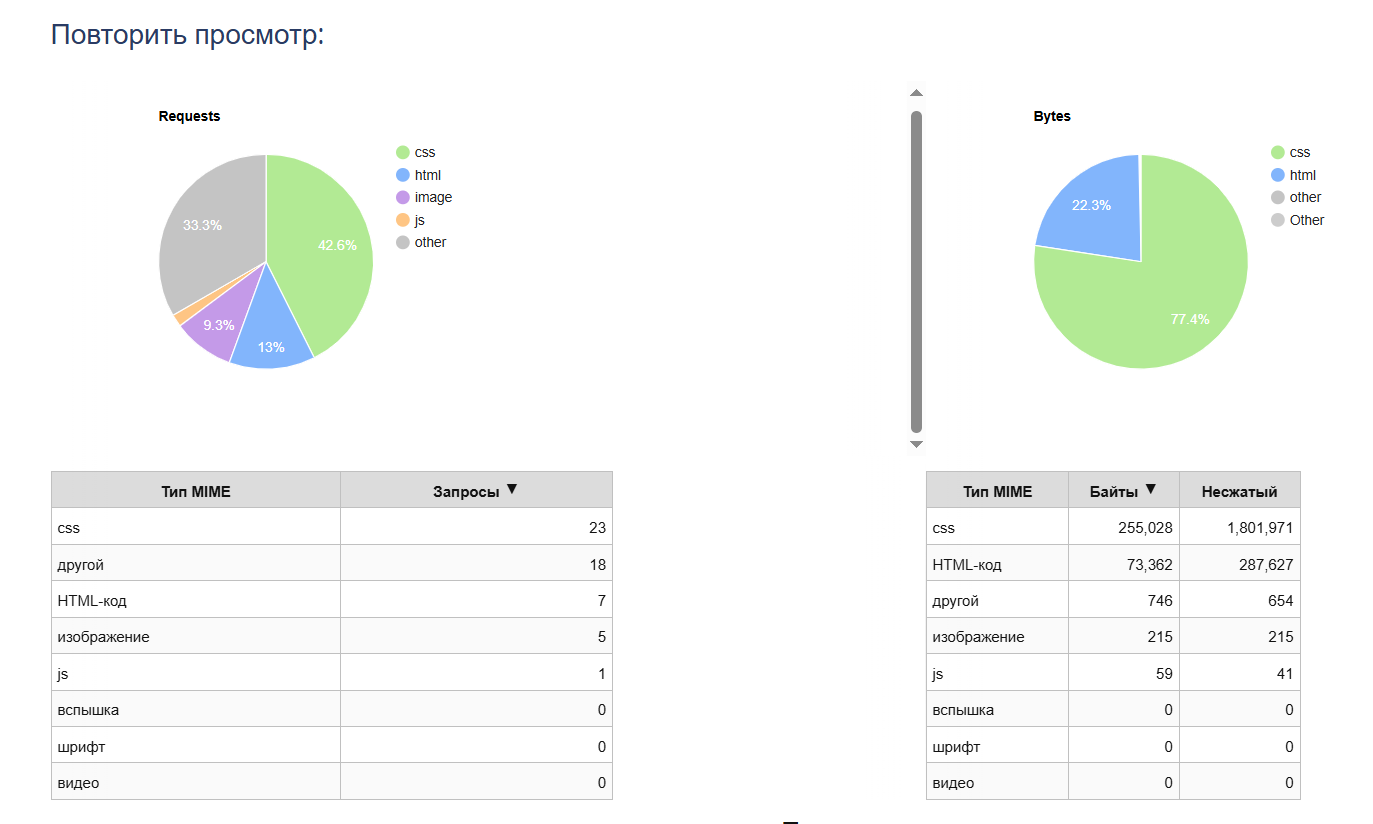
**Интернет-ресурс:** https://vk.com/ (Вконтакте)

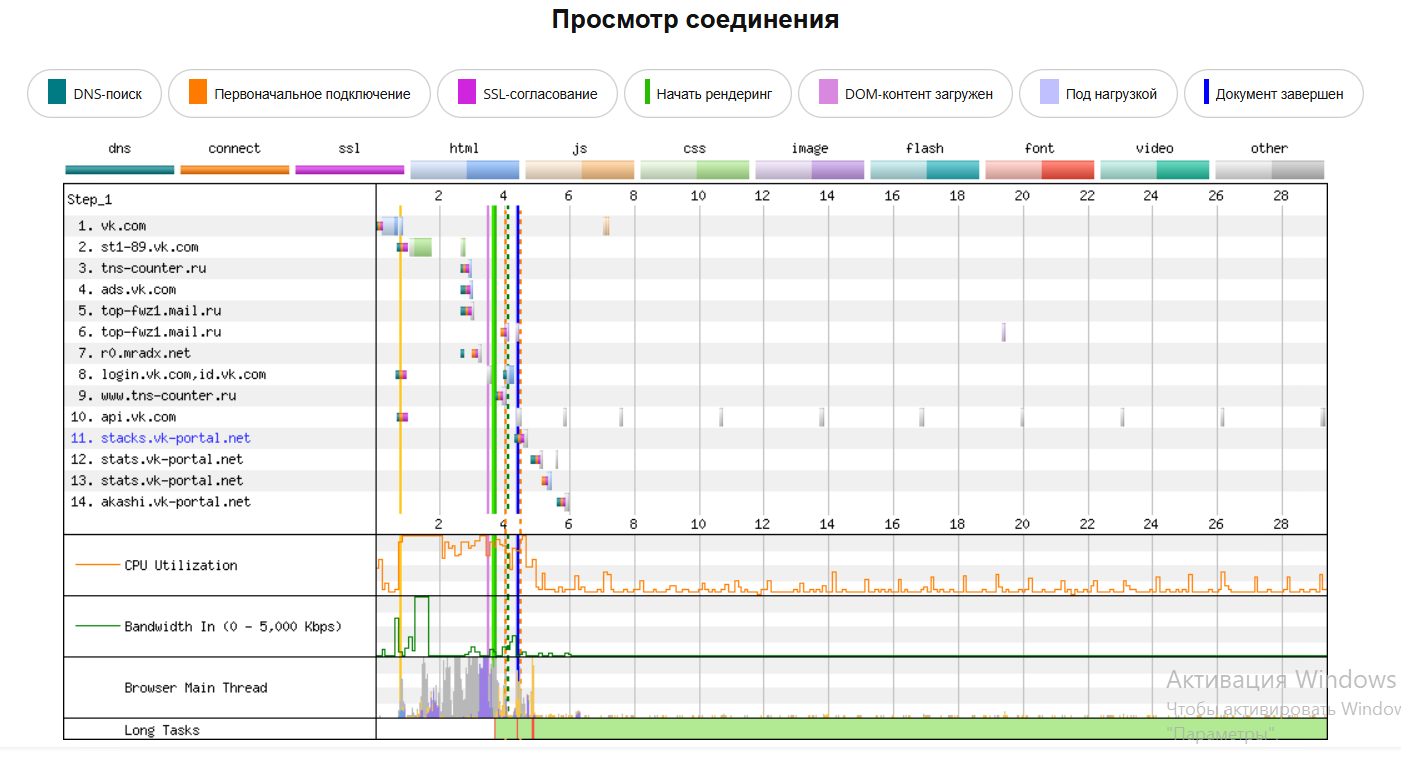
****

****

****

****

****

****

Панель производительности позволяет разработчикам и тестировщикам измерять производительность загрузки страницы веб-сайта. Это достигается путем запуска анализа производительности на веб-странице и генерации метрик по различным аспектам процесса загрузки страницы, таким как время, необходимое для отображения веб-страницы, загрузка различных сетевых ресурсов и интерактивность веб-страницы.

**Вывод:** Проведенный анализ протокола HTTP на примере трассировки Wireshark и тестирования веб-сайта [vk.com](https://vk.com/) показал важность механизмов кэширования и оптимизации загрузки ресурсов для повышения производительности. Полученные данные демонстрируют, что эффективное использование заголовков кэширования и минимизация количества запросов являются ключевыми факторами для обеспечения быстрой работы веб-приложений.