### Установить программу wireshark <https://www.wireshark.org/>

### Программа работает под управлением Windows, GNU/Linux, Mac OS X. Входит в состав Kali Linux. При установке вам предложат установить библиотеку Libpcap или Winpcap. Обязательно это сделайте, иначе программой не получится пользоваться.

### Далее выполните задания и составьте отчет (можно со скриншотами), описанием ваших действий и выводами.

### Важно. В качестве домашки сдается отчет в формате docx или pdf. Не присылайте дампы, это небезопасно и не будет зачтено.

#### 1. Найти нешифрованный http-сайт, где есть регистрация и логин. Отправить фейковые данные. Сможет ли злоумышленник перехватить пароль?

#### 2. Найти нешифрованный http-сайт с множеством картинок. Рекомендуется использовать Google Chrome. Сколько будет открыто tcp-соединений и почему?

#### 3. Повторите п.1 с TLS. Вопрос тот же.

#### 4 \*Какие интересные протоколы можно обнаружить, если подключиться Google Chrome к YouTube (сработать может не у всех)?

#### 5 \*Если хочется совсем сложное задание, попробуйте изучить трафик при подключении к ftp-cерверу (достаточно любого публичного нешифрованного ftp-сервера, например Yandex.Mirror).

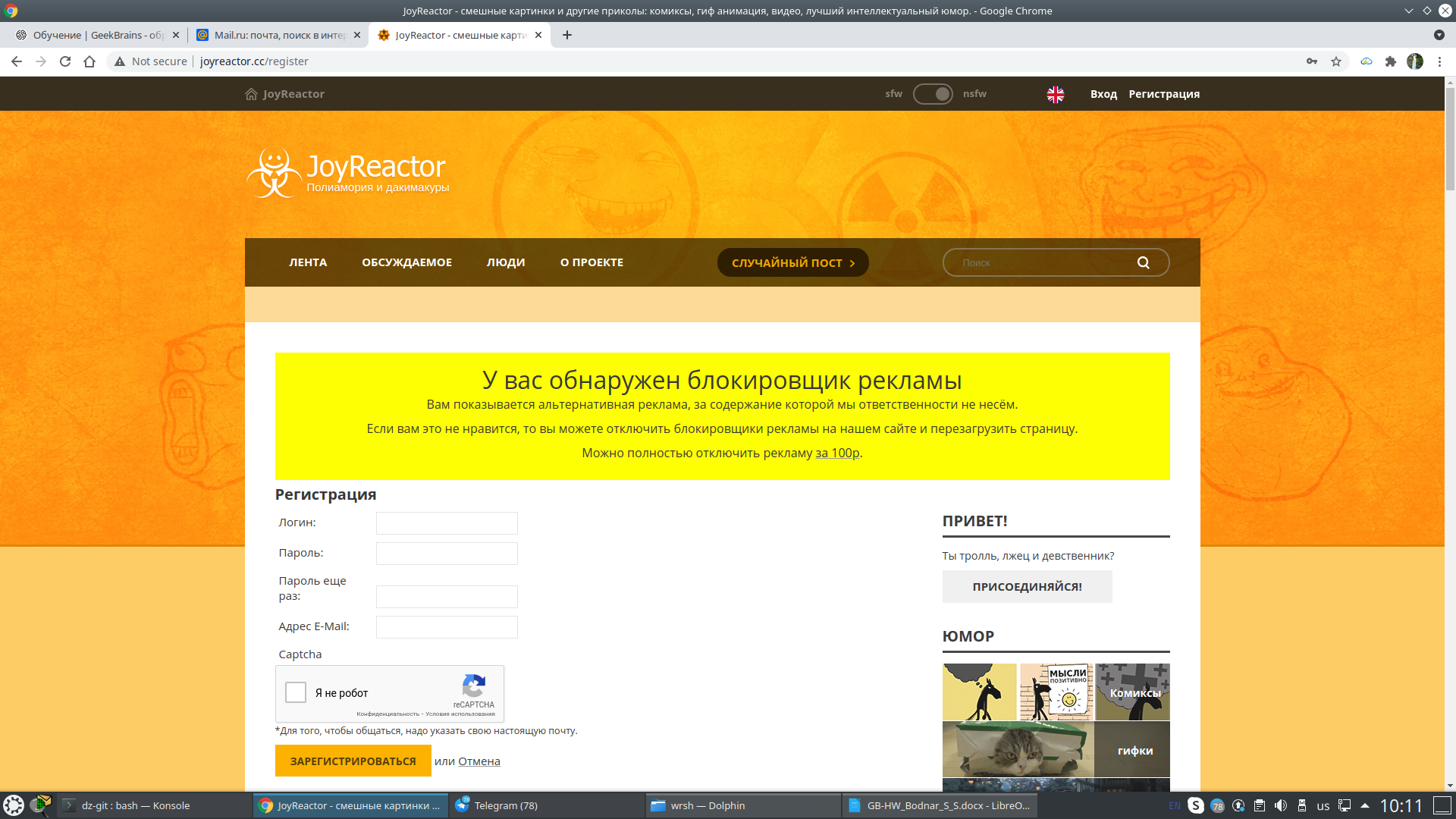
1. <http://joyreactor.cc/register>

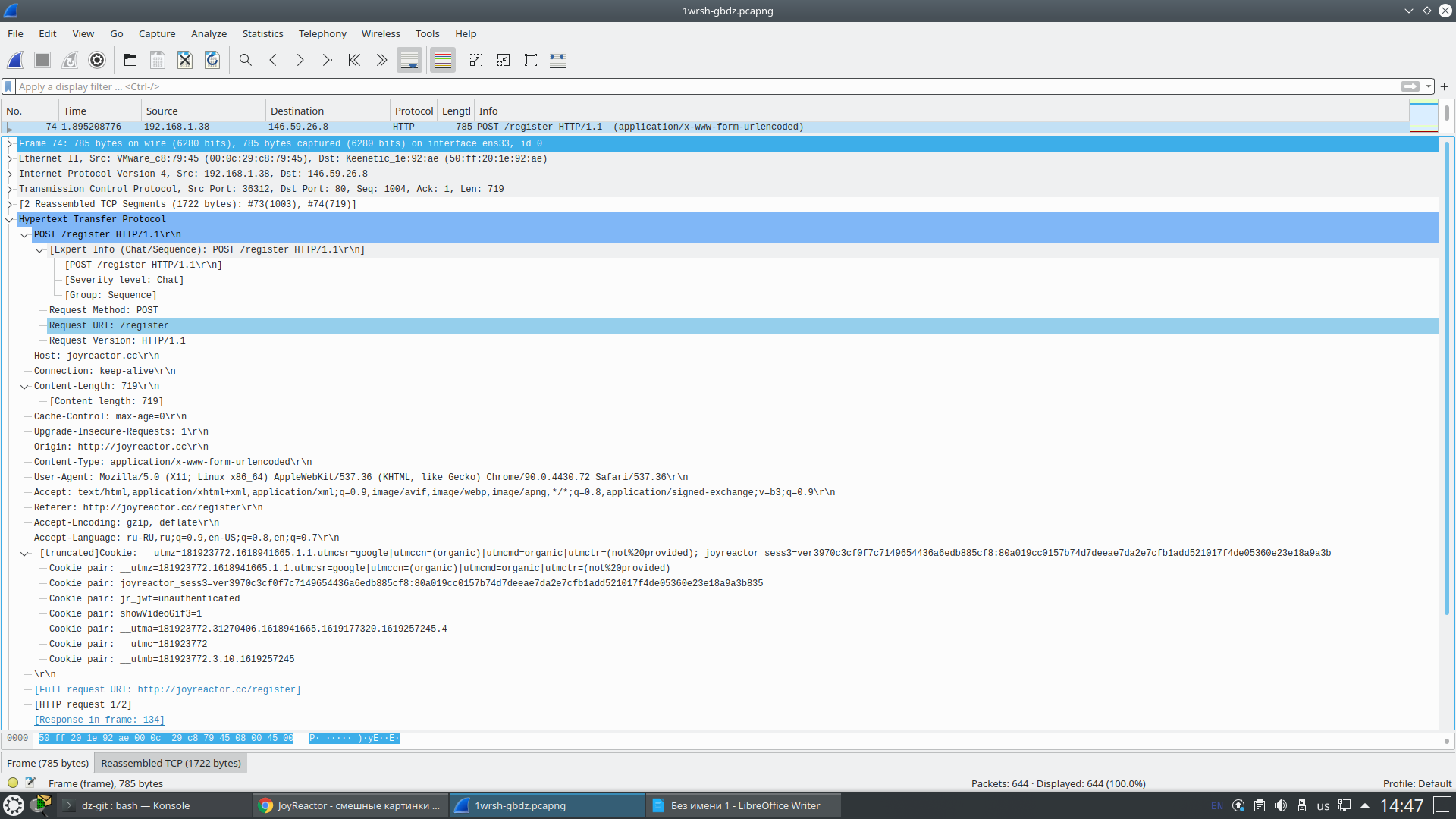
Порядок действий:

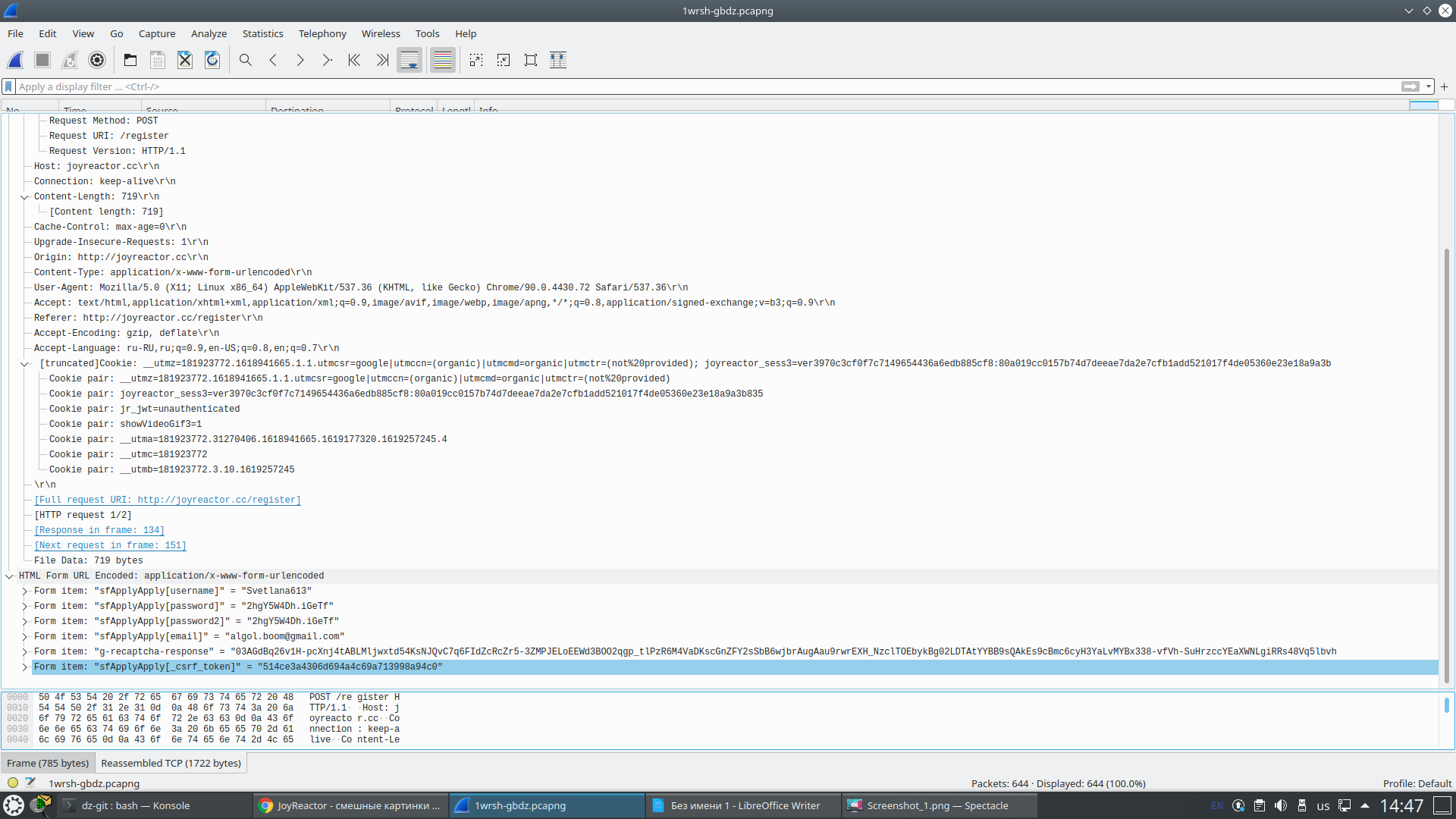
* Открыла в браузере сайт http://joyreactor.cc/
* Ввела данные для регистрации
* Перешла в Wireshark и запустила захват трафика
* На сайте отправила данные на регистрацию нового аккаунта
* Остановила захват трафика
* С помощью фильтра http отсортировала соединения
* Открыла записаные данные относящиеся к регистрации
* Удостоверилась, что при регистрации на незащищенном сайте, злоумышленник может перехватить данные создаваемого аккаунта.

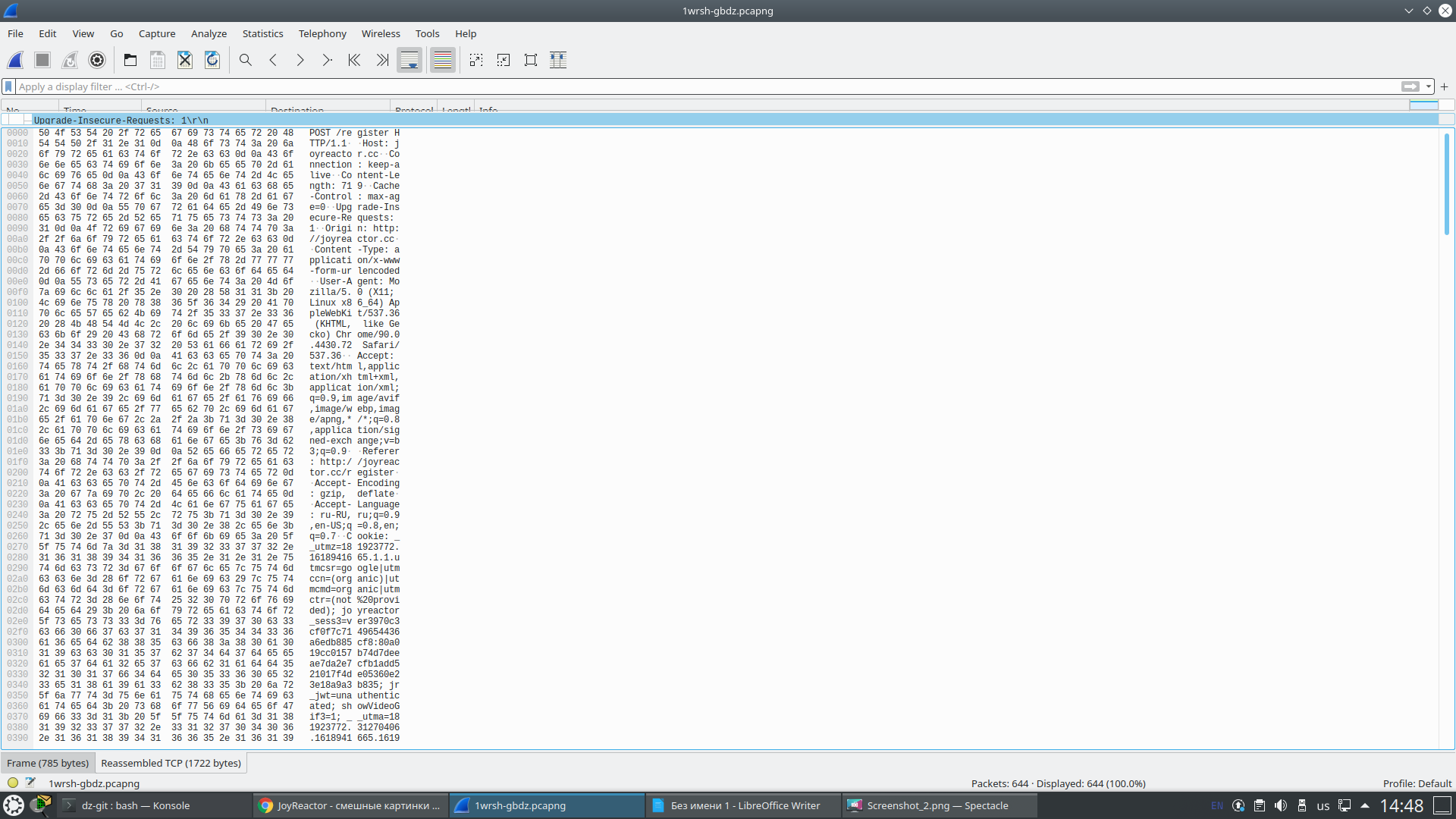
74 1.895208776 192.168.1.38 146.59.26.8 HTTP 785 POST /register HTTP/1.1 (application/x-www-form-urlencoded)

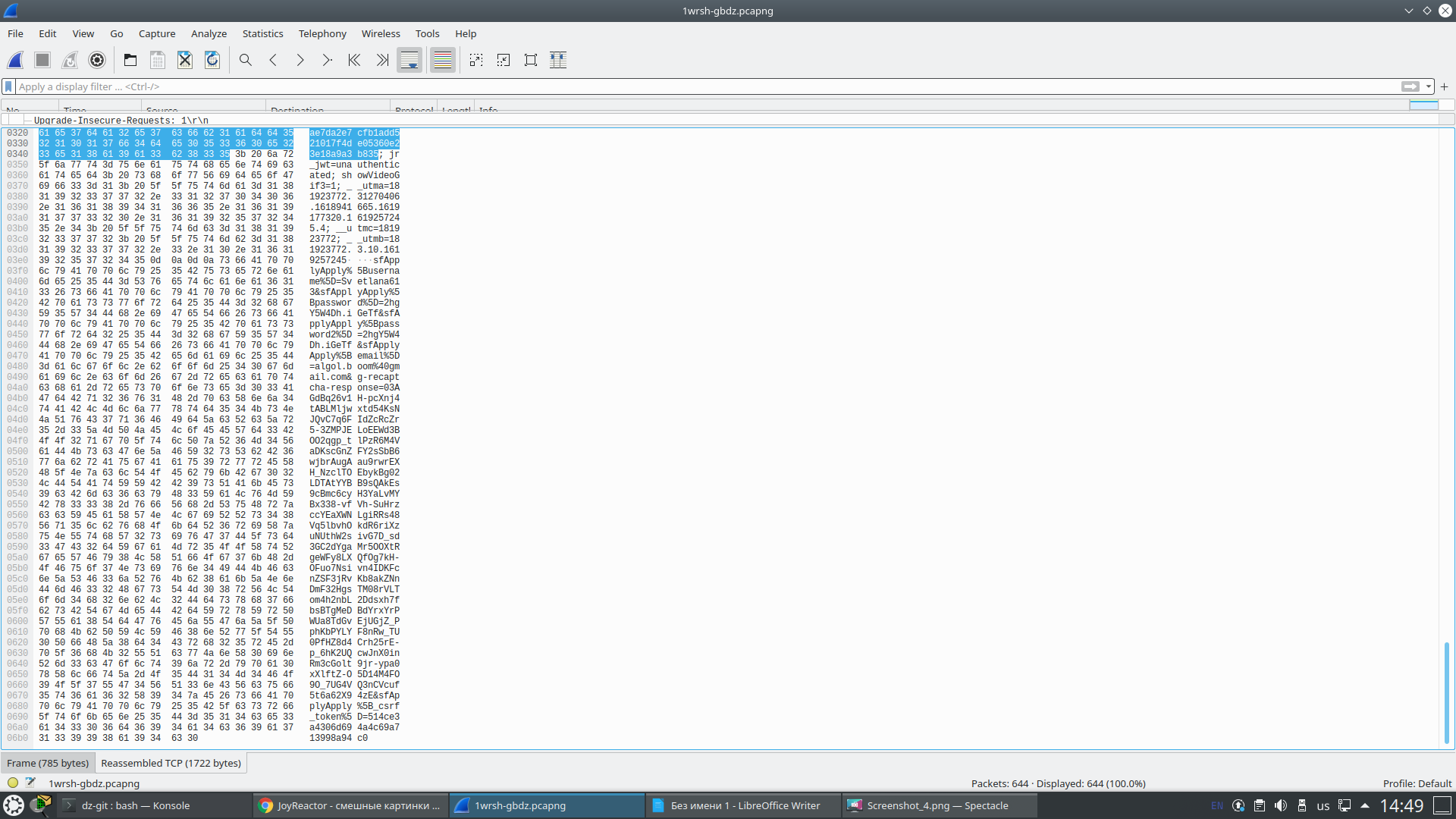
При регистрации на незащищенном http злоумышленник может перехватить данные, при последующих входах подобное невозможно.









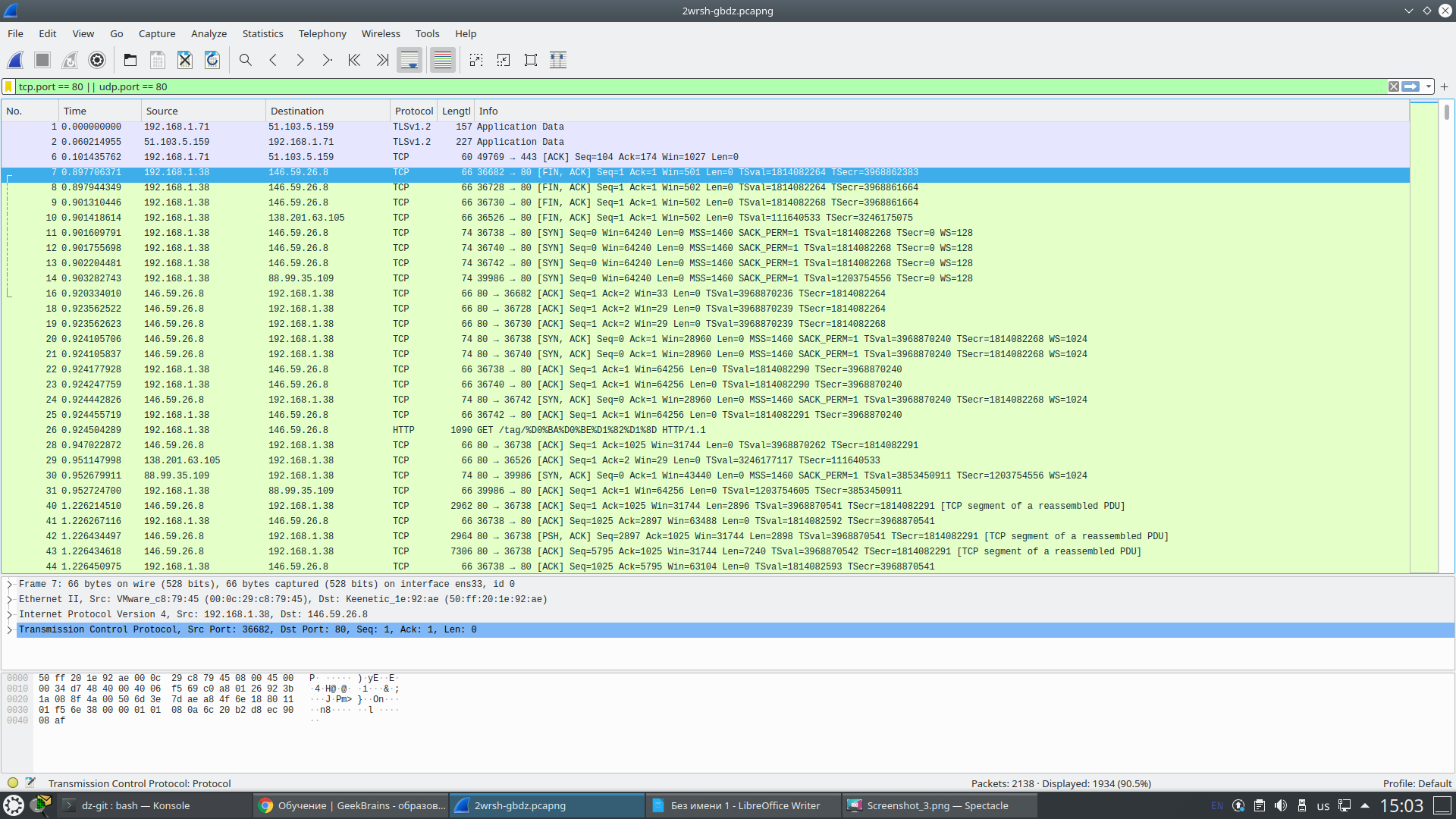


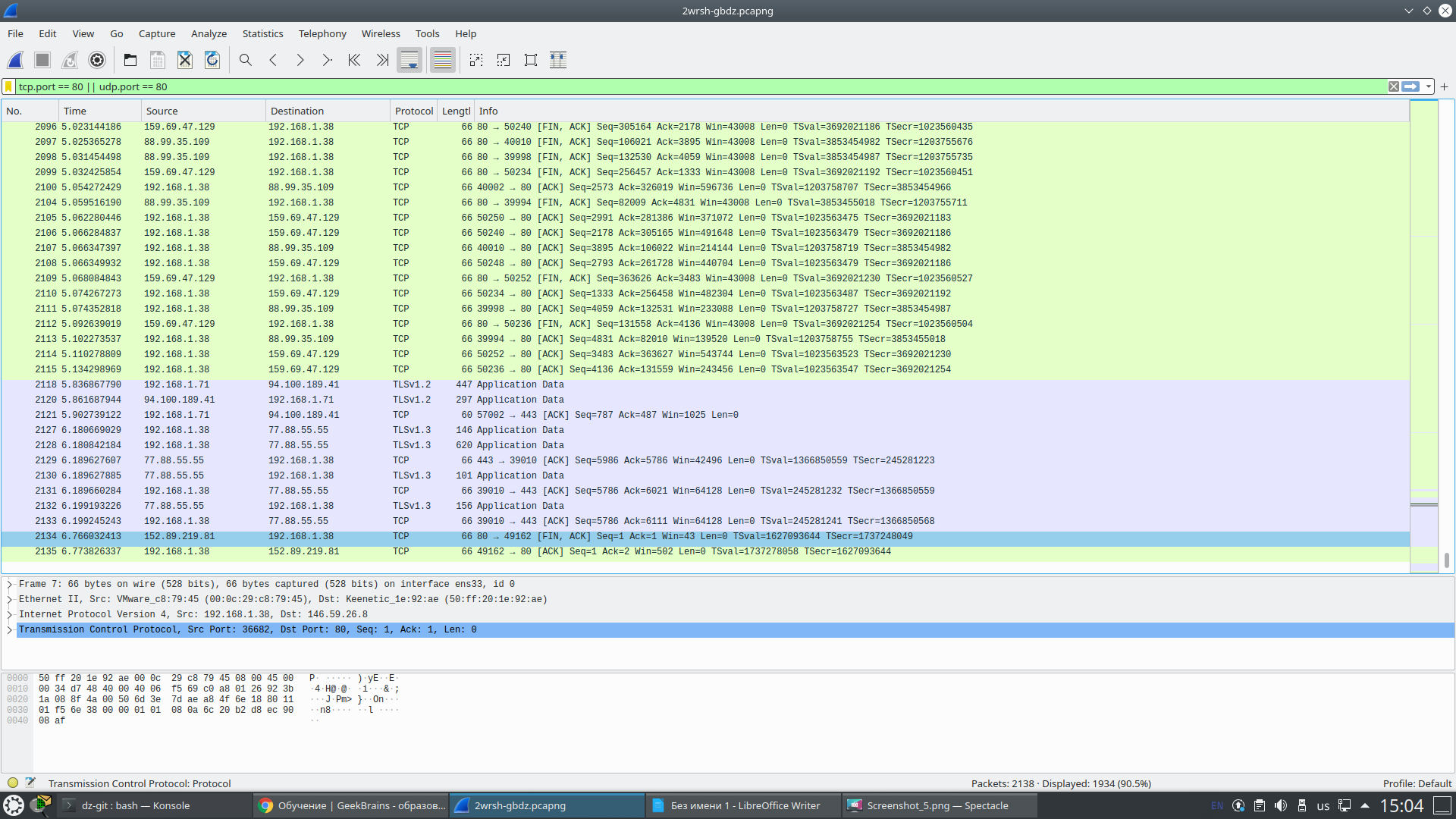
С помощью фильтра отсортировала соединения

2.http://joyreactor.cc/tag/%D0%BA%D0%BE%D1%82%D1%8D

Порядок действий:

* Открыла в браузере сайт <http://joyreactor.cc/>
* Перешла в Wireshark и запустила захват трафика
* Открыла страницу с картинками котиков
* Остановила захват трафика
* Отфильтровала трафик при помощи фильтра tcp



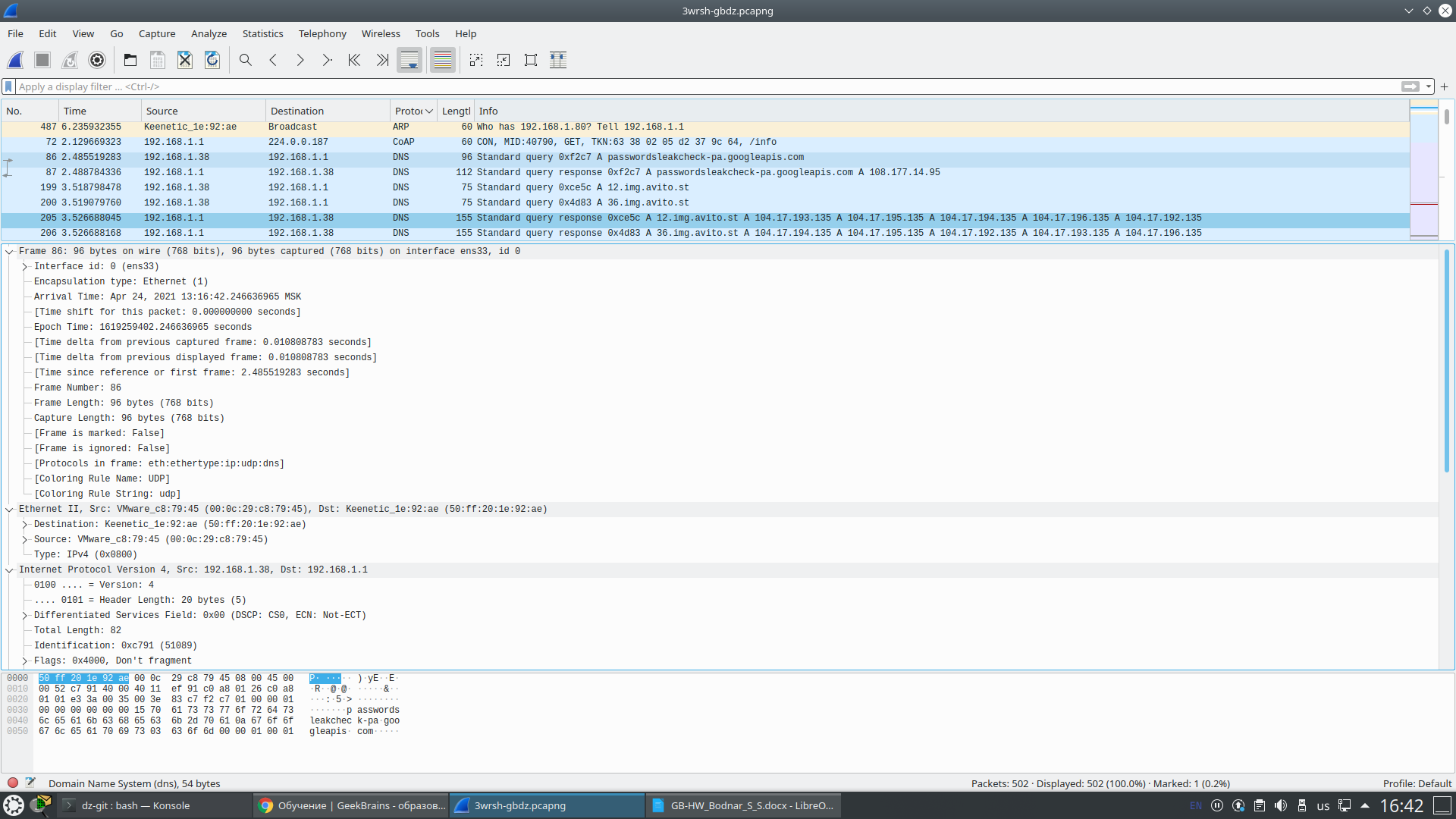


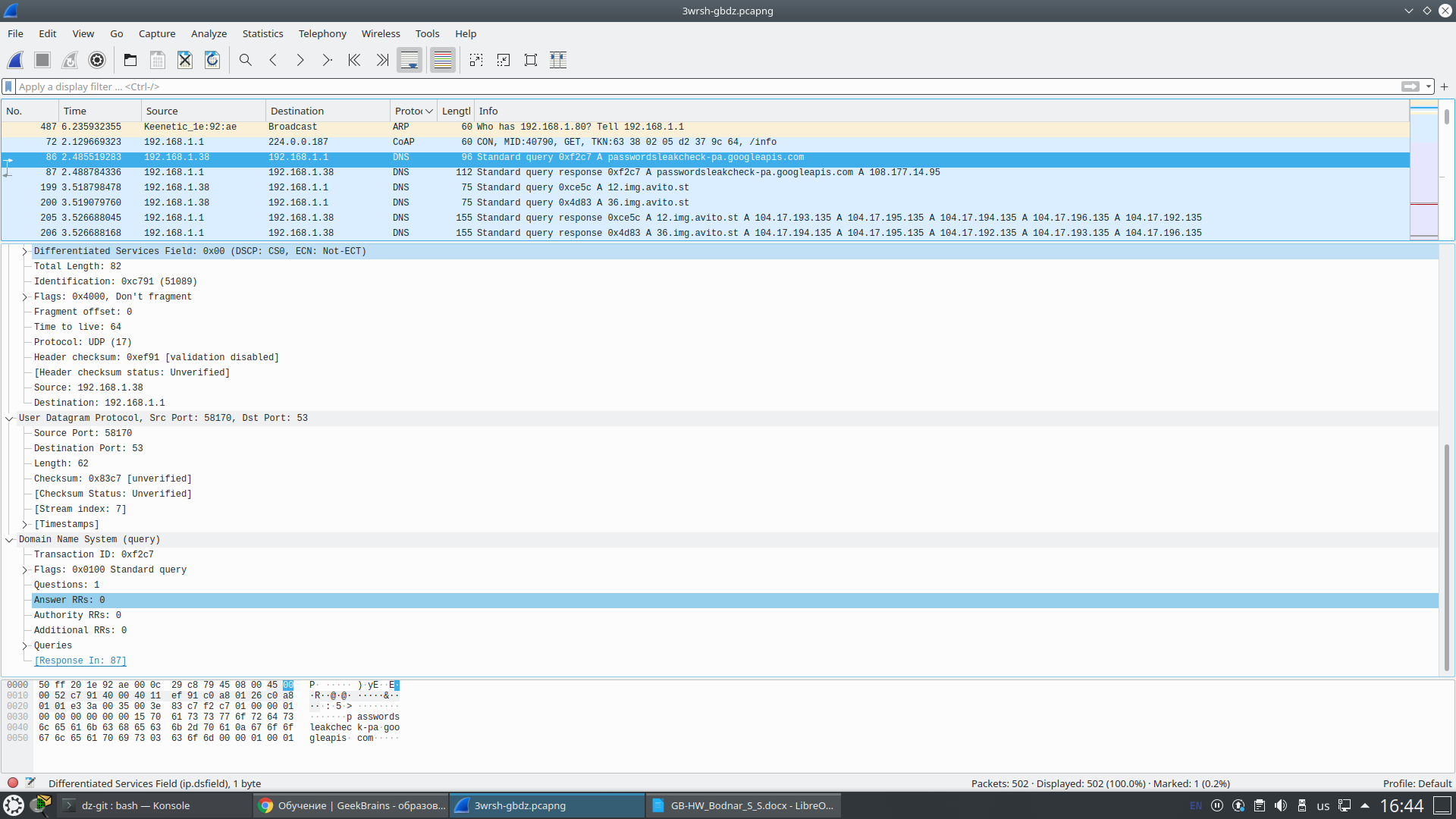
Открыто более 2000 tcp соединенй, для каждого изображения существует отдельный запрос, так как его загрузка происходит отдельно.

3. https://www.avito.ru/

Порядок действий:

* Открыла в браузере сайт https://www.avito.ru/
* Ввела данные для регистрации
* Перешла в Wireshark и запустила захват трафика
* На сайте отправила данные на регистрацию нового аккаунта
* Остановила захват трафика
* Провела поиск соединения отвечающего за регистрацию, как видно на скриншотах, данные никаким образом не выводятся.





4. У меня не отобразилось, но при выяснении вопроса нашла ответ.

Можно найти протокол QUIC работает над UDP.

QUIC (сокр. от англ. Quick UDP Internet Connections; произносится quick) —

экспериментальный интернет-протокол, разработанный Google[2][3] в конце 2012 года

[4]

.

QUIC позволяет мультиплексировать несколько потоков данных между двумя компьютерами,

работая поверх протокола UDP, и содержит возможности шифрования,

эквивалентные TLS и SSL. Имеет более низкую задержку соединения и передачи, чем TCP.

Хорошо переносит потерю части пакетов путём выравнивания границ криптографических

блоков по границам пакетов. В протокол заложена возможность прямой (упреждающей)

коррекции ошибок на уровне пакетов, но на практике она отключена.