Лабораторная работа №1. Анализ сетевого трафика с помощью программы “Wireshark”

Цель работы: ознакомление с работой программы Wireshark.

Ход работы

пункт меню Capture >> Options. В появившемся диалоговом окне установила следующие параметры захвата пакетов (Рисунок 1):

-захват 20 пакетов из сети Ethernet интерфейса Подключение по локальной сети;

- размер захваченных файлов 1 Мбайт;

- сохранять захваченные файлы на Рабочес столе в папке Ibragimova\lab1.

Через пункт меню Capture >> Interfaces запустить процесс захвата пакетов. Результат отображен на Рисунке 2.

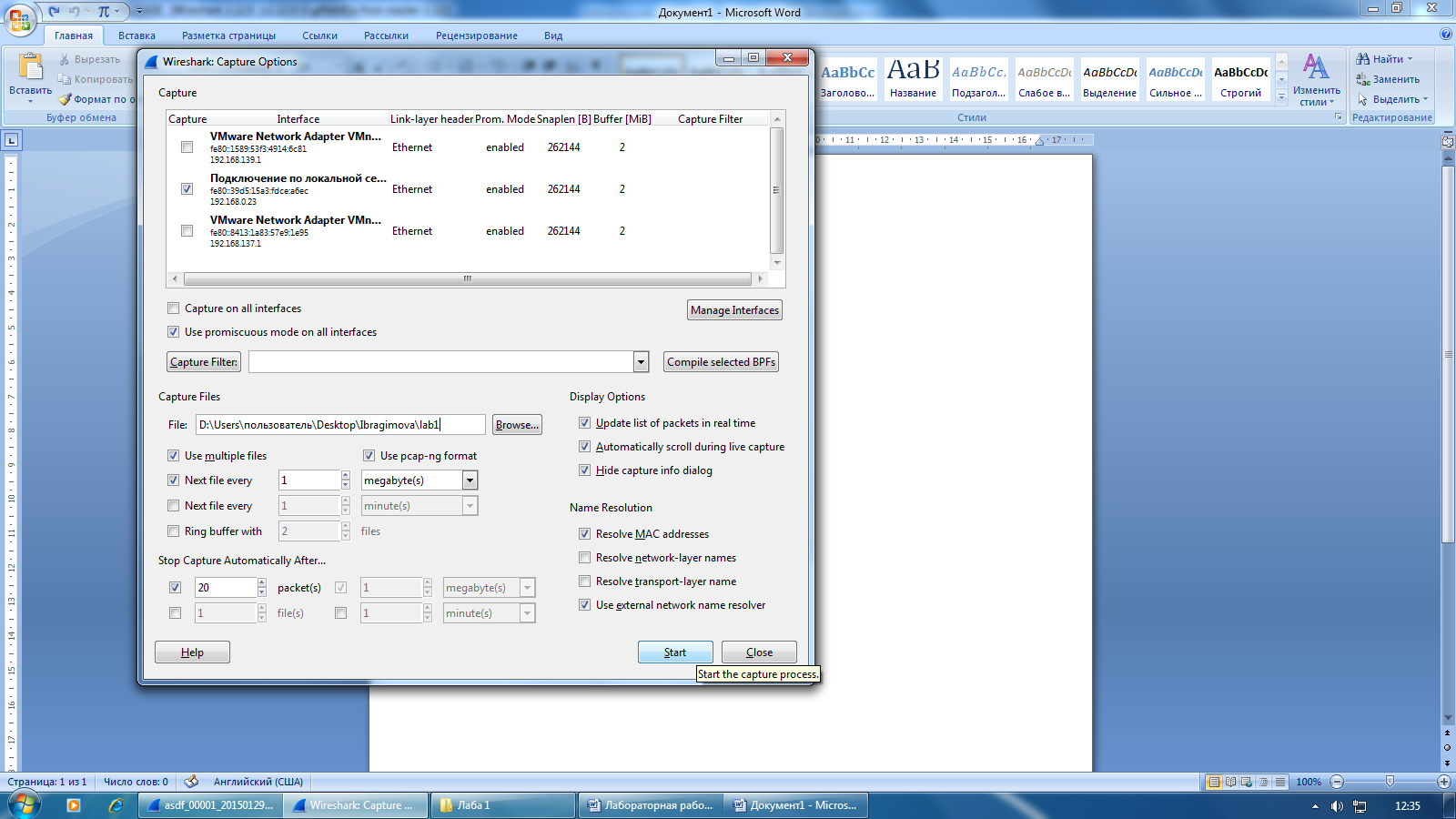


Рисунок 1 – Диалоговое окно Capture Options

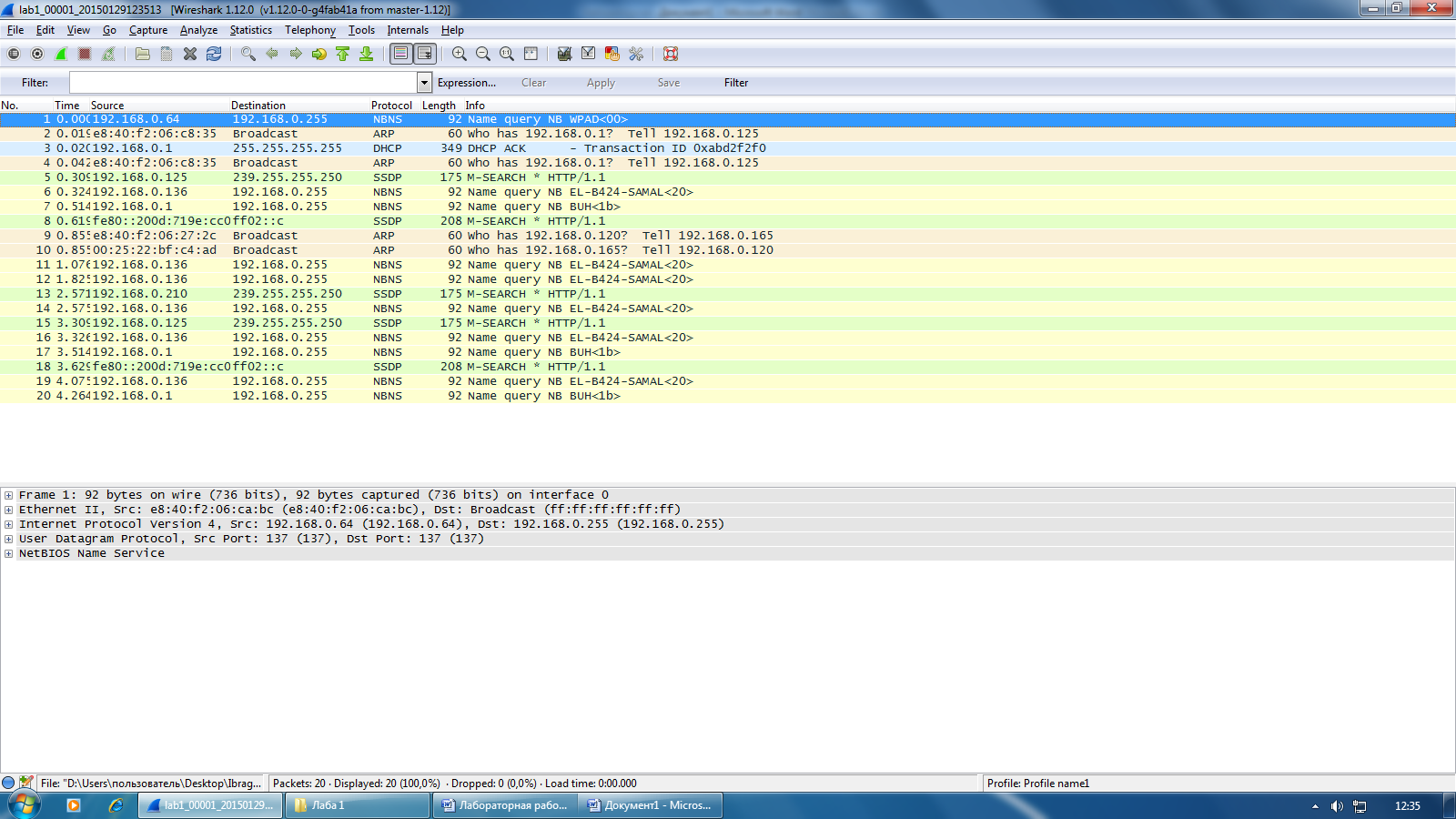


Рисунок 2 – Процесс захвата пакетов

Через пункт меню Edit >> Configuration Profile >> New создала новый профиль Leila (Рисунок 3).

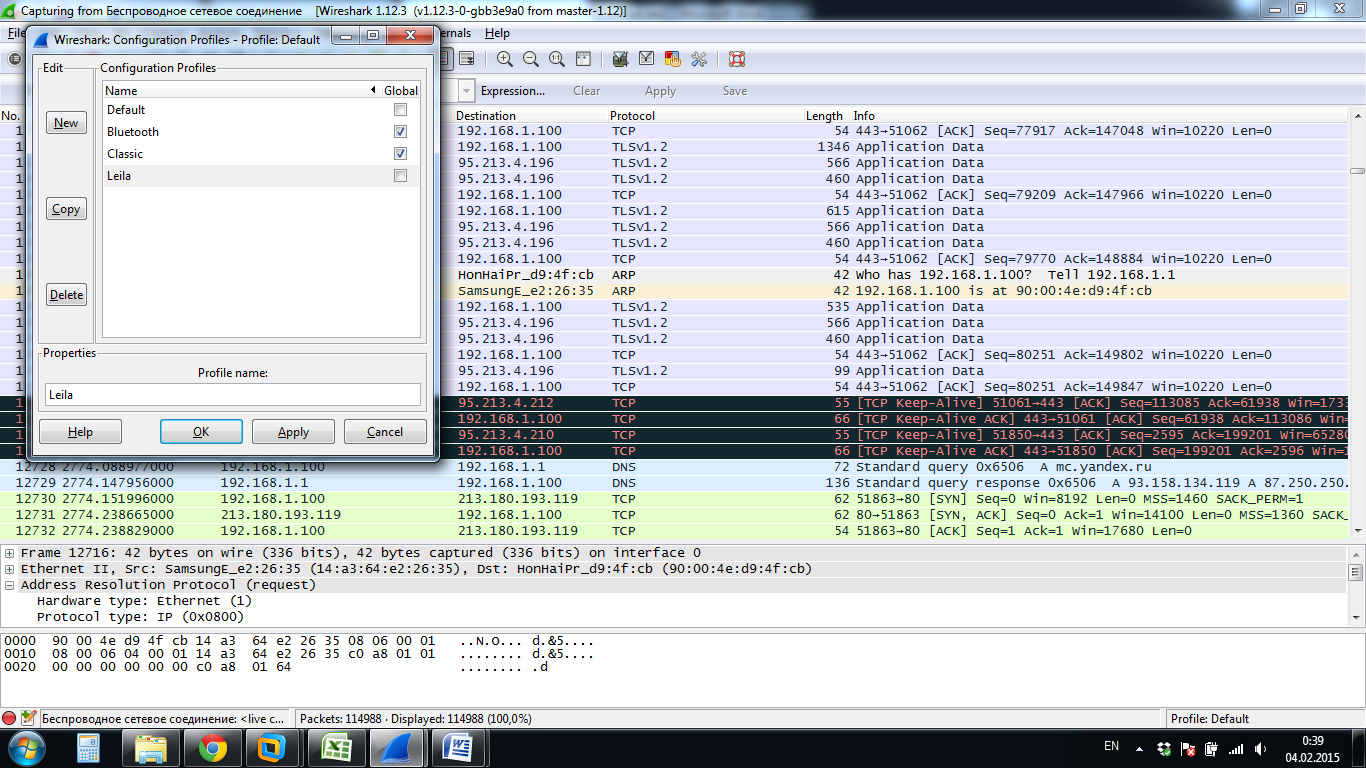


Рисунок 3 – Создание нового профиля

В программе Wireshark фильтрацию можно осуществить несколькими способами. Рассмотрим фильтрацию с помощью строки фильтра, используя логические функции:

- перехват трафика от определенного IP-адреса (Рисунок 4);

- перехват трафика кроме определенного IP-адреса (Рисунок 5);

- перехват трафика с длиной пакета больше N, где N задается (Рисунок 6);

- перехват трафика с длиной пакета меньше N, где N задается (Рисунок 7);

- перехват трафика от определенного IP-адреса, использующего определенный протокол (Рисунок 8);

- перехват трафика от нескольких IP-адресов (Рисунок 9);

- перехват трафика по определенному протоколу (Рисунок 10).

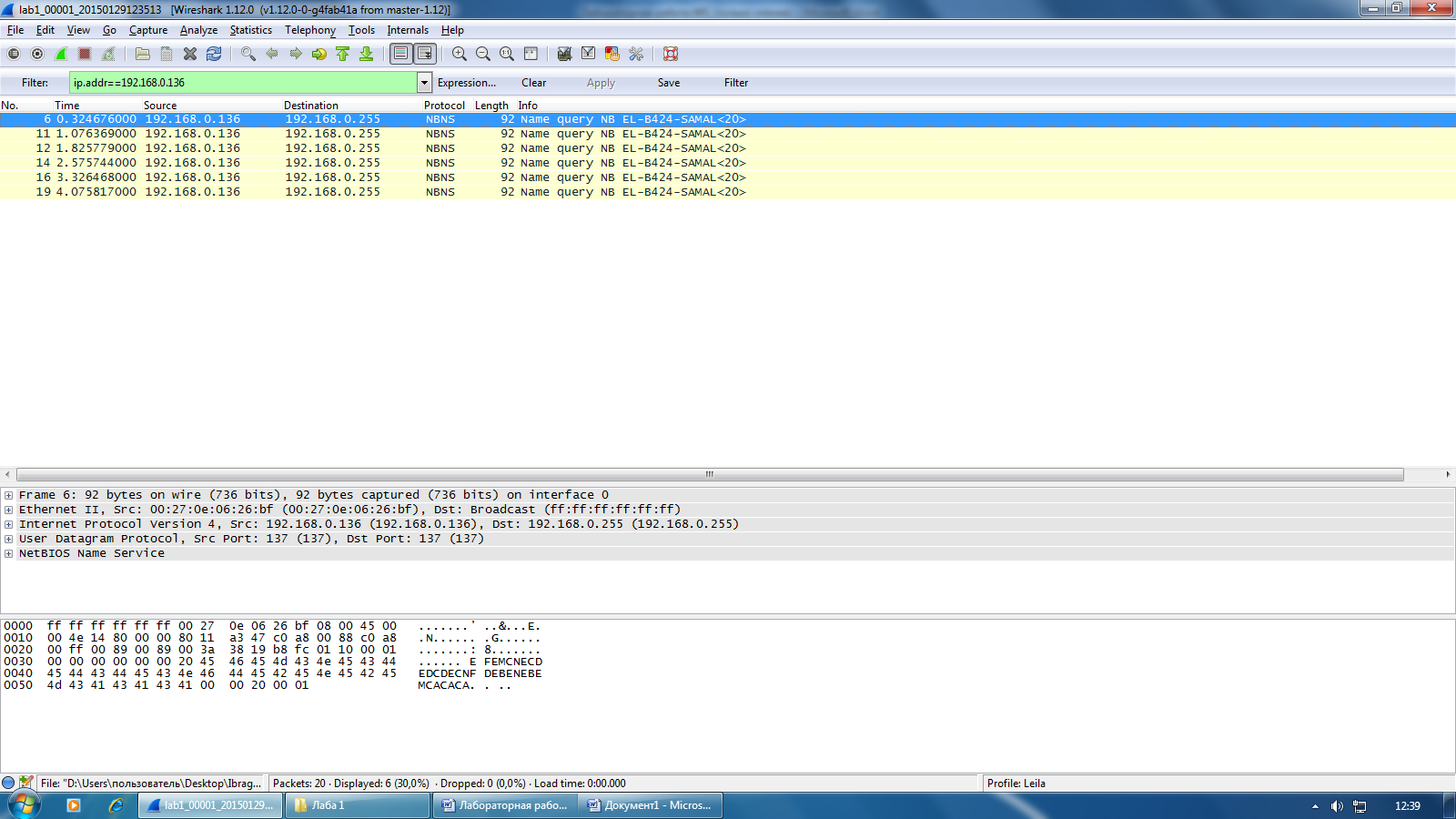


Рисунок 4 – Фильтрация по ip-адресу = 192.168.0.136

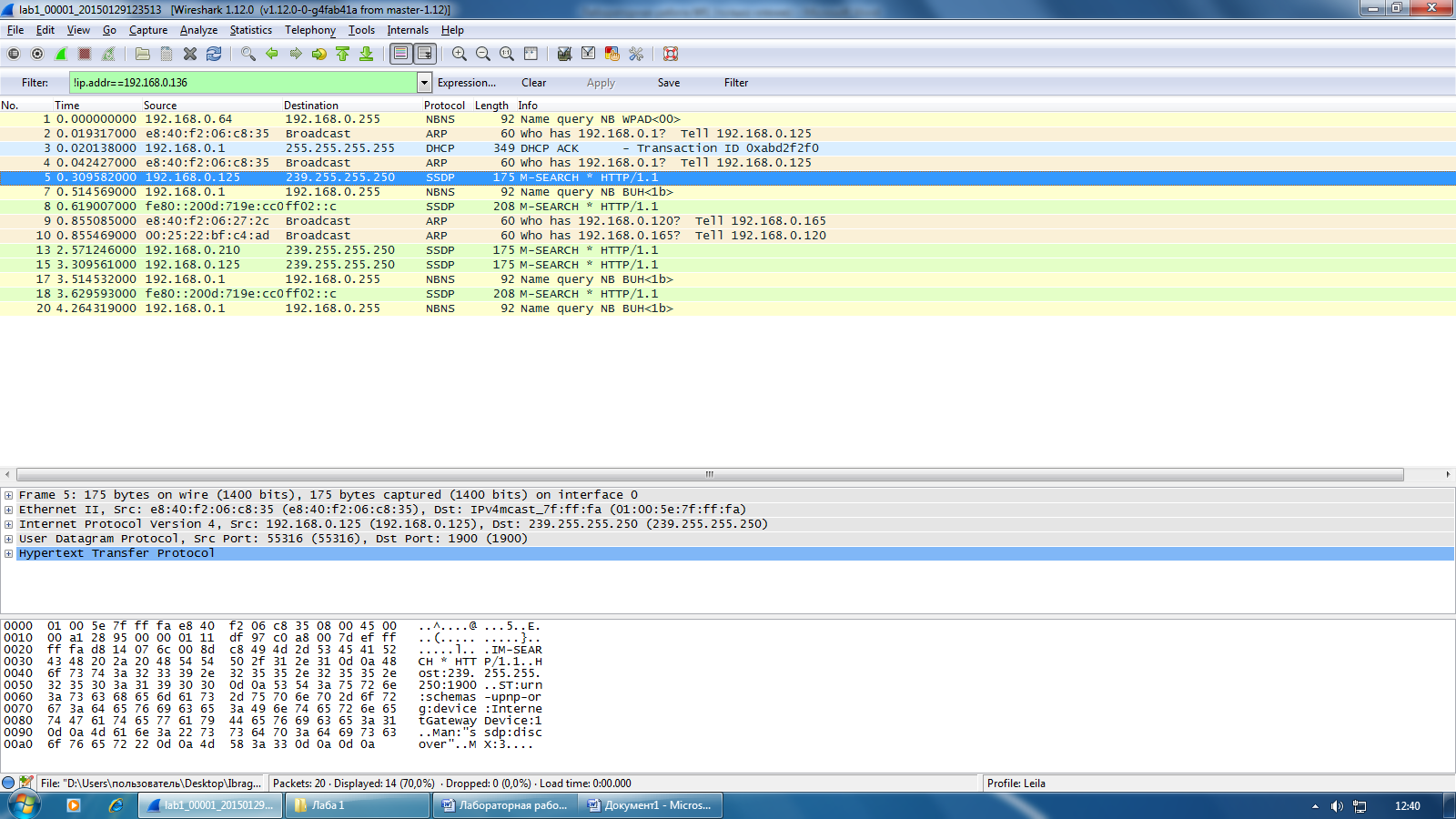


Рисунок 5 – Фильтрация по ip-адресу != 192.168.0.136

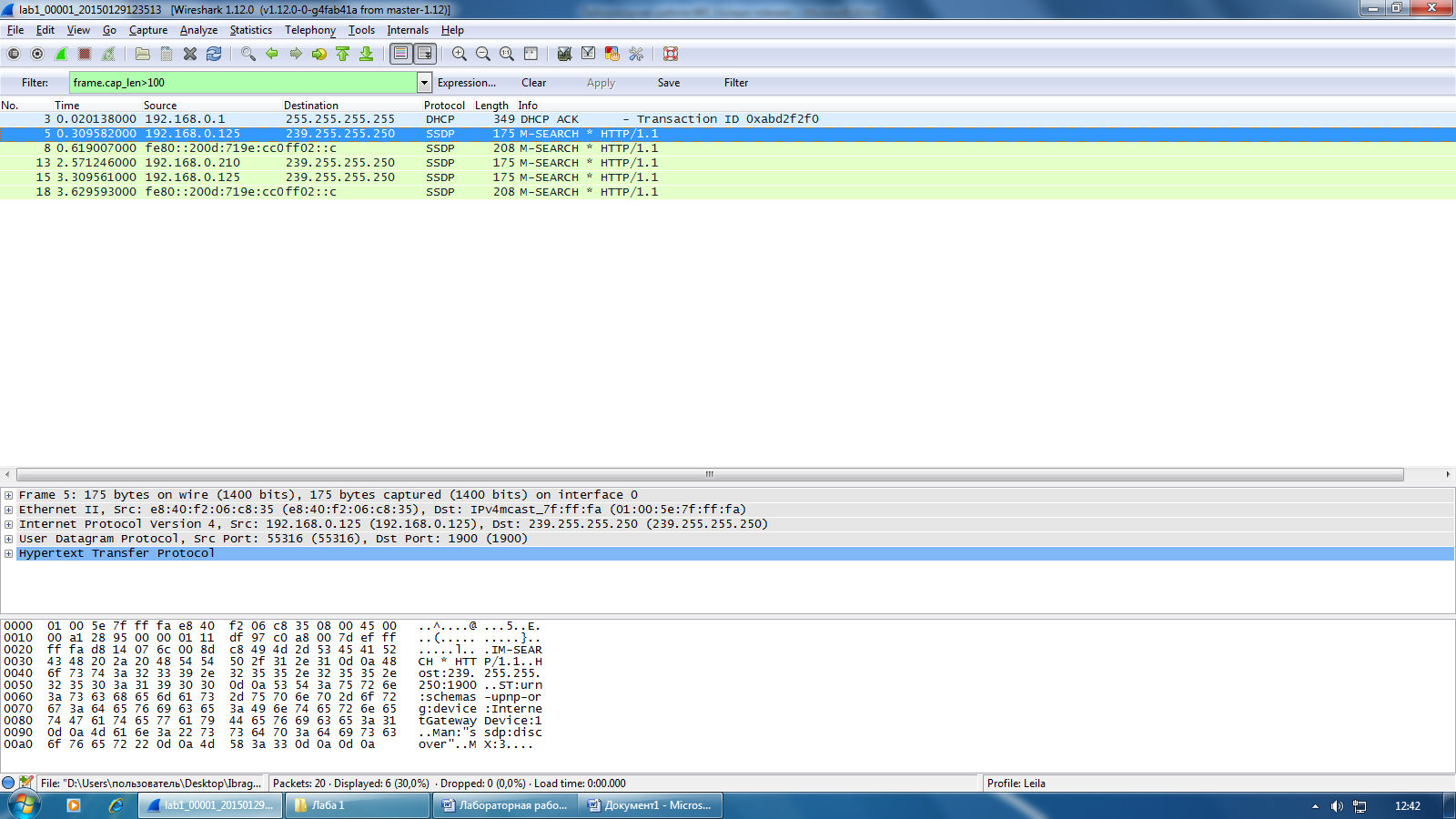


Рисунок 6 – Фильтрация по длине пакета больше 100

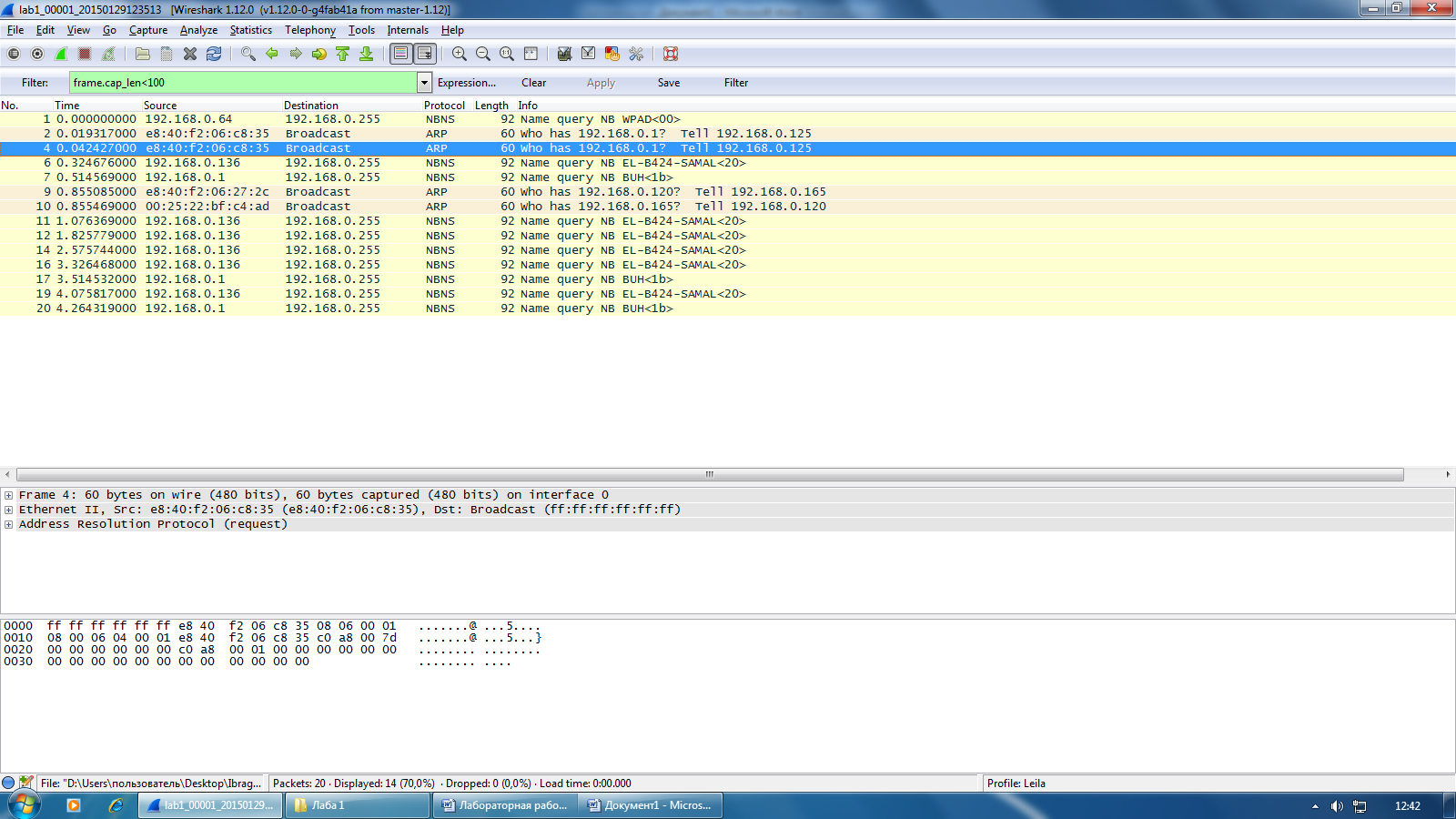


Рисунок 7 – Фильтрация по длине пакета меньше 100

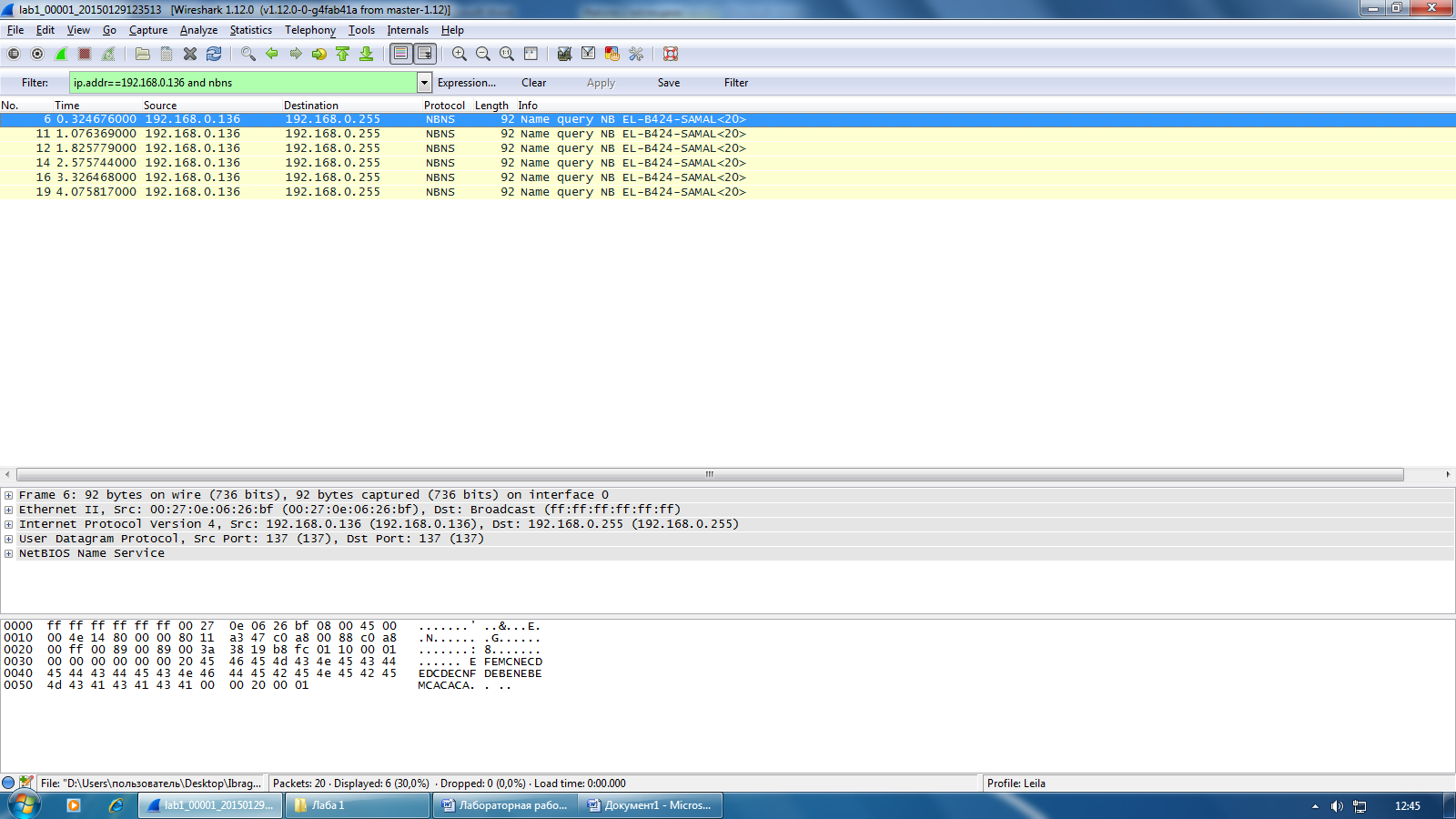


Рисунок 8 – Фильтрация по ip-адресу и протоколу

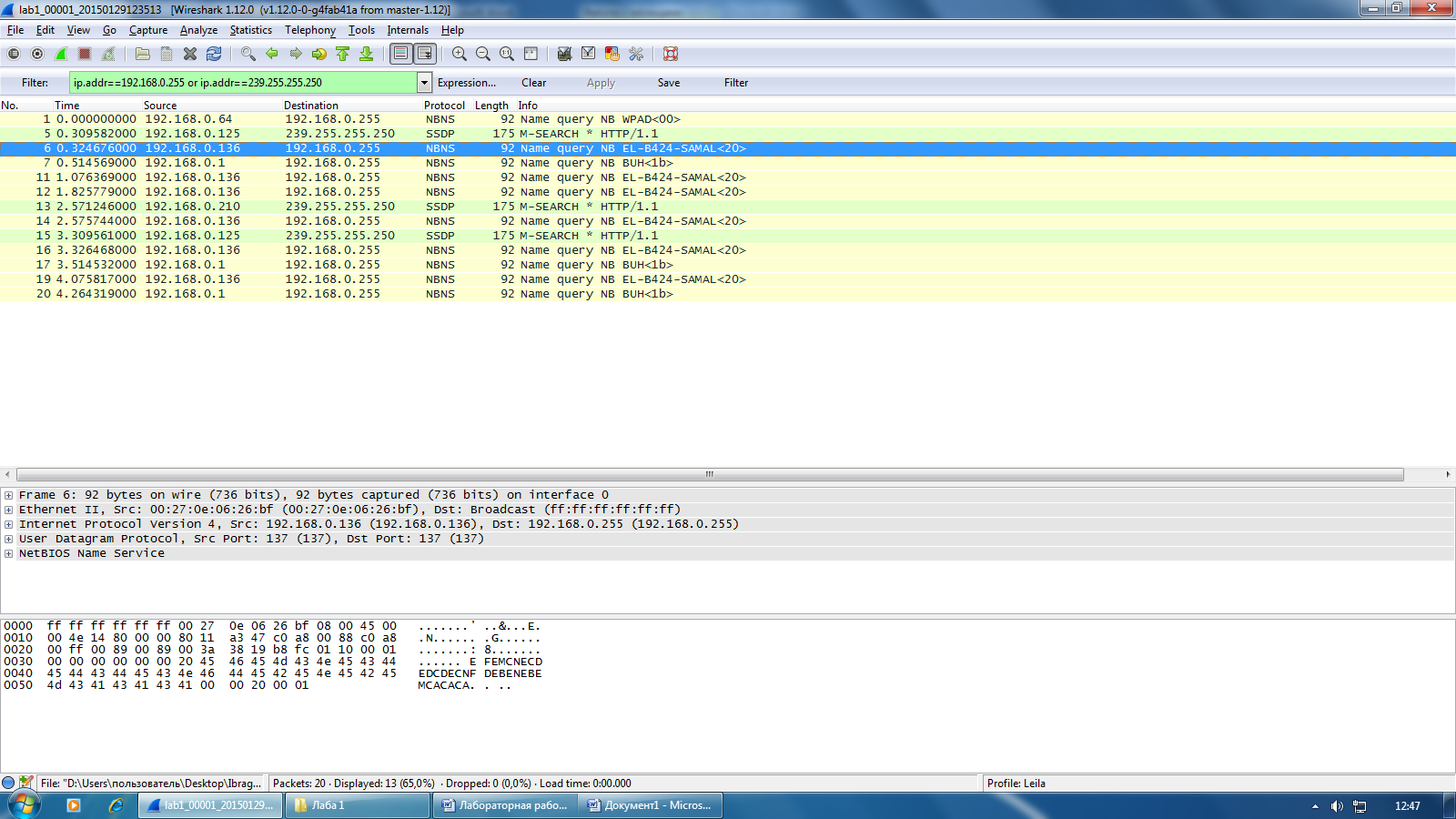


Рисунок 9 – Фильтрация по нескольким ip-адресам

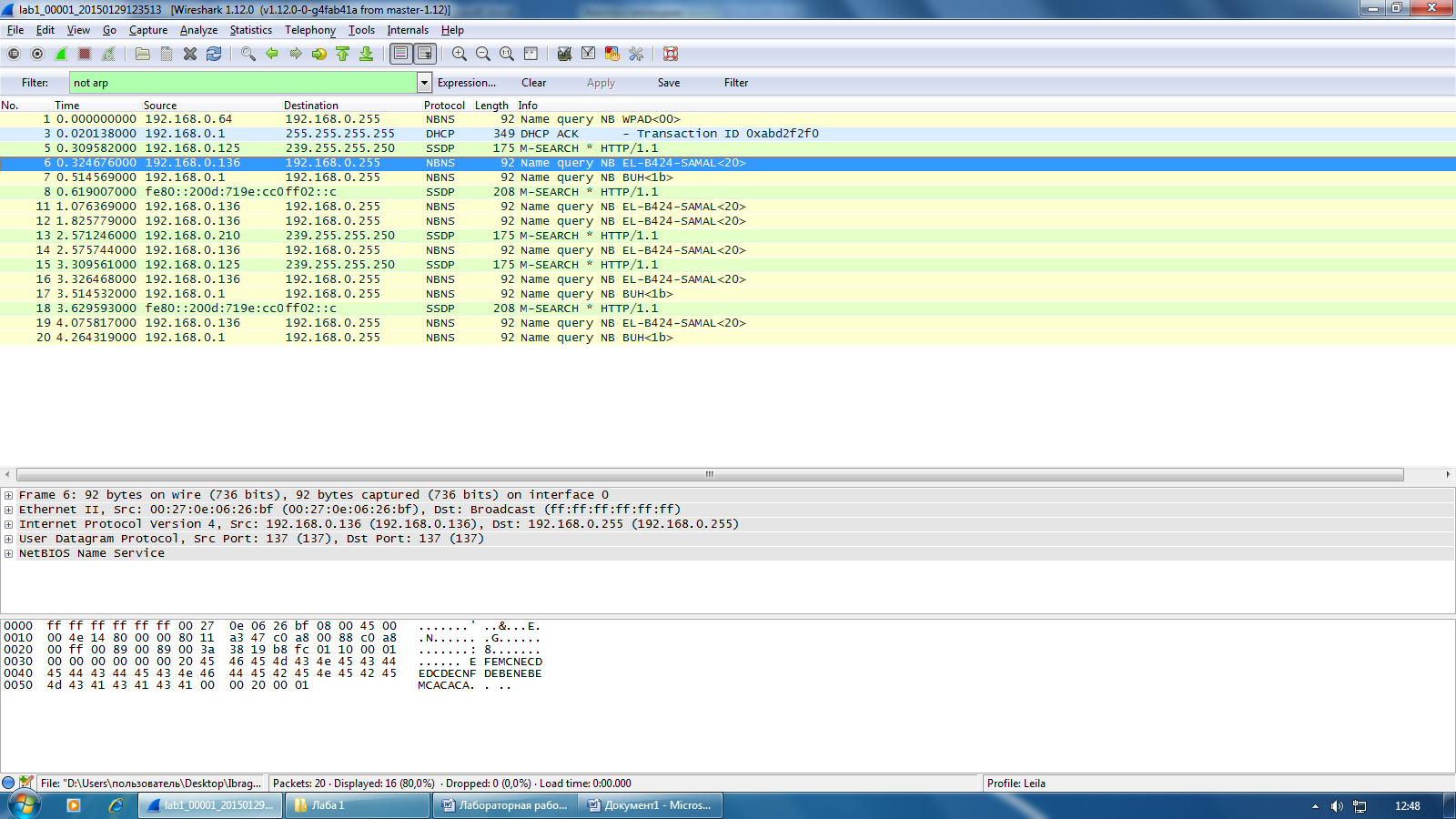


Рисунок 10 – Фильтрация по протоколу ARP

Теперь рассмотрим фильтрацию через поле Capture Filter диалогового окна Capture Options (Рисунок 11).

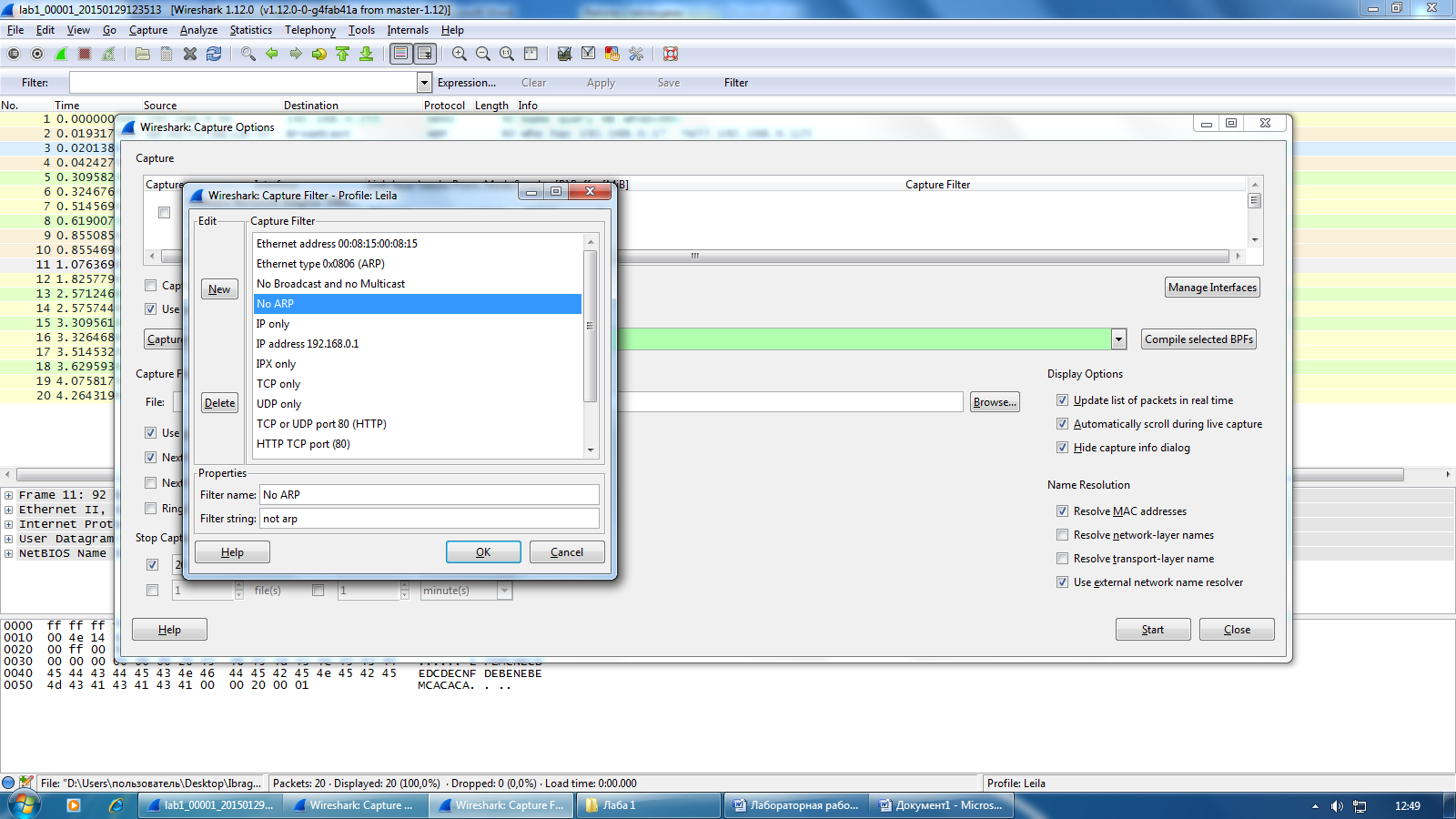


Рисунок 11 – Фильтрация по протоколу через Capture Options

Фильтры можно сохранять. Например, сохраним фильтр по протоколу и назовем его F.dhcp\_v6 (Рисунок 12).

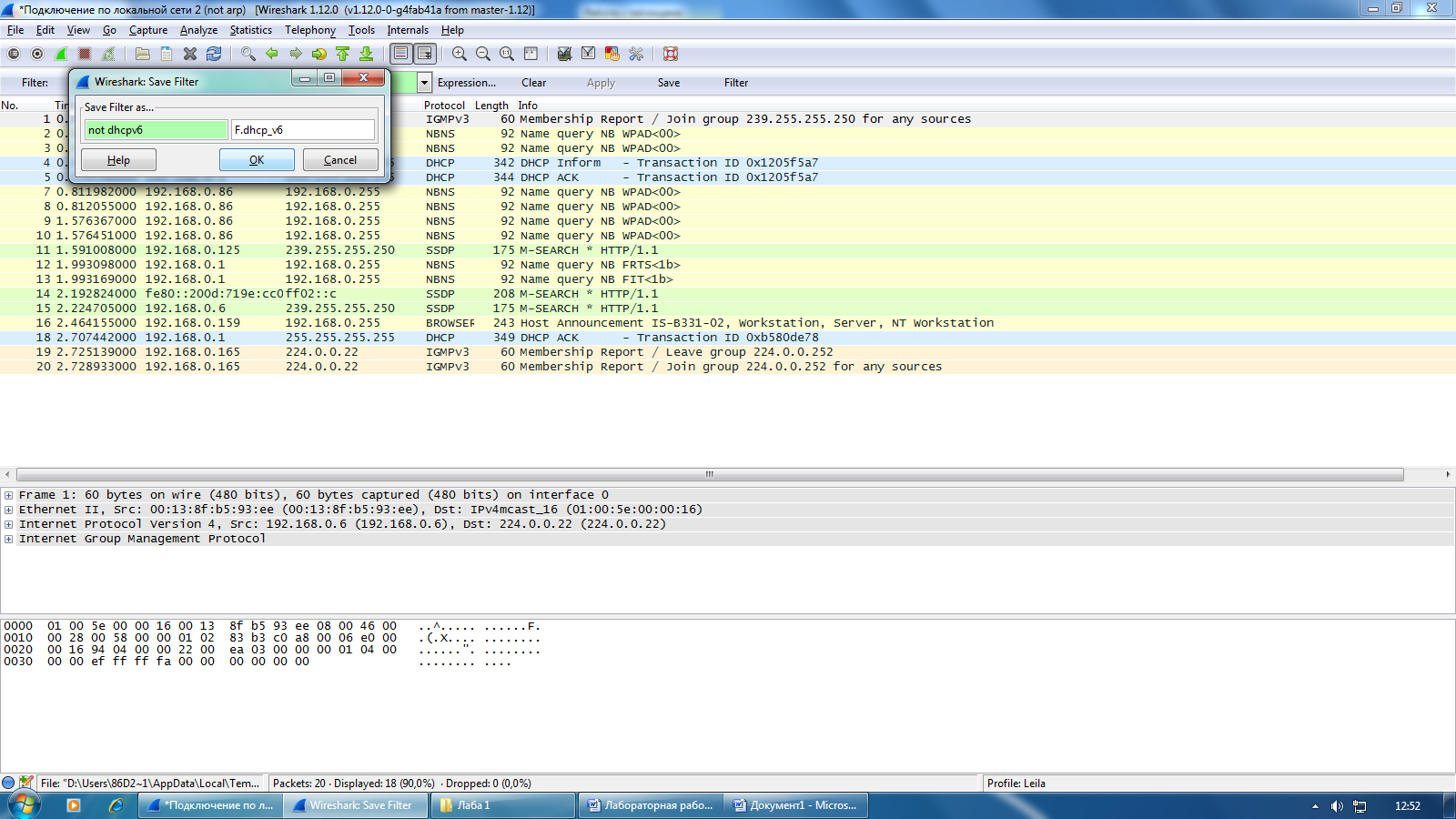
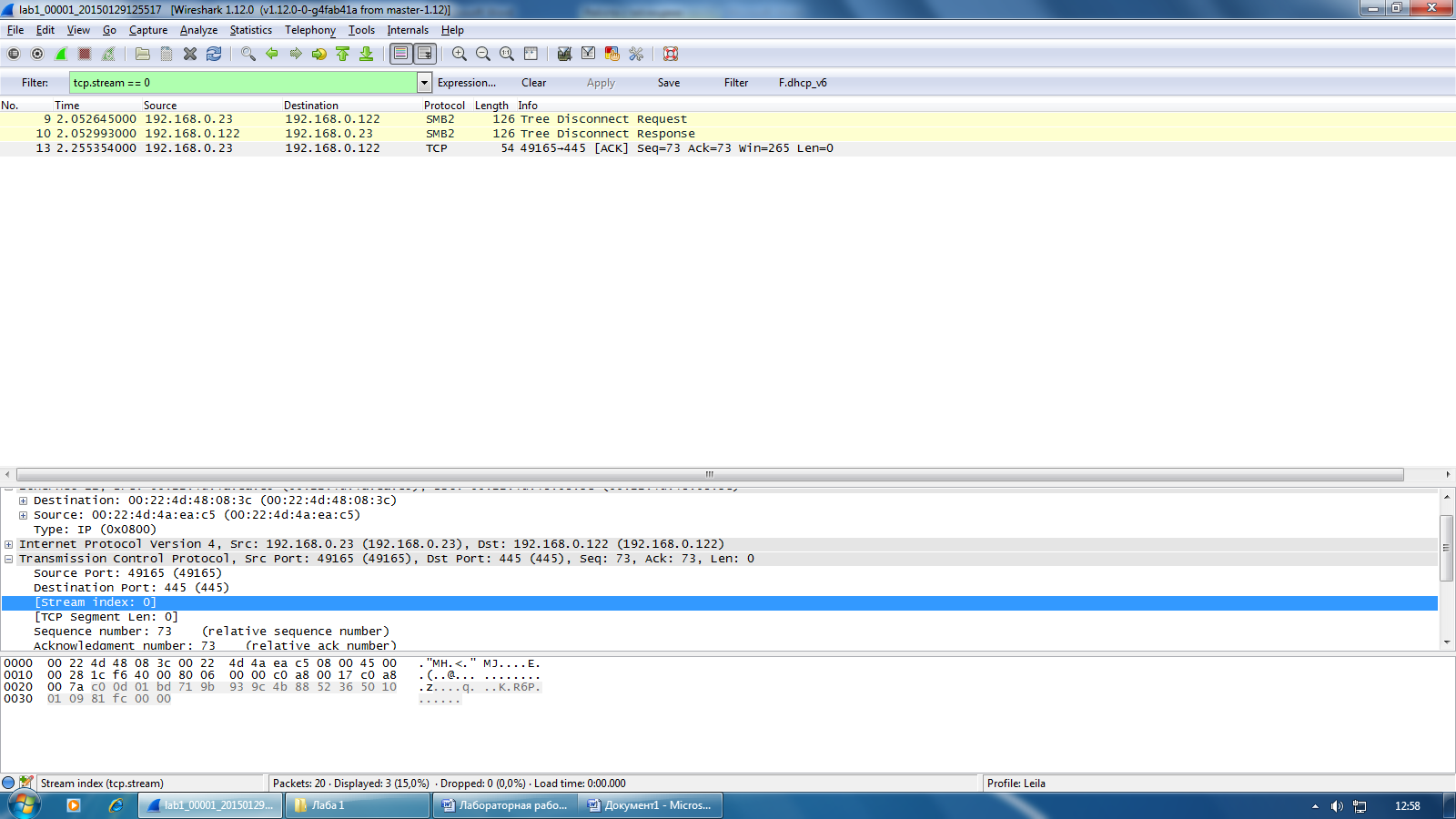
 

Рисунок 12 – Сохранение фильтра

Осуществим фильтрацию по протоколам HTTP (Рисунок 13), TCP (Рисунок 14), DNS (Рисунок 15).

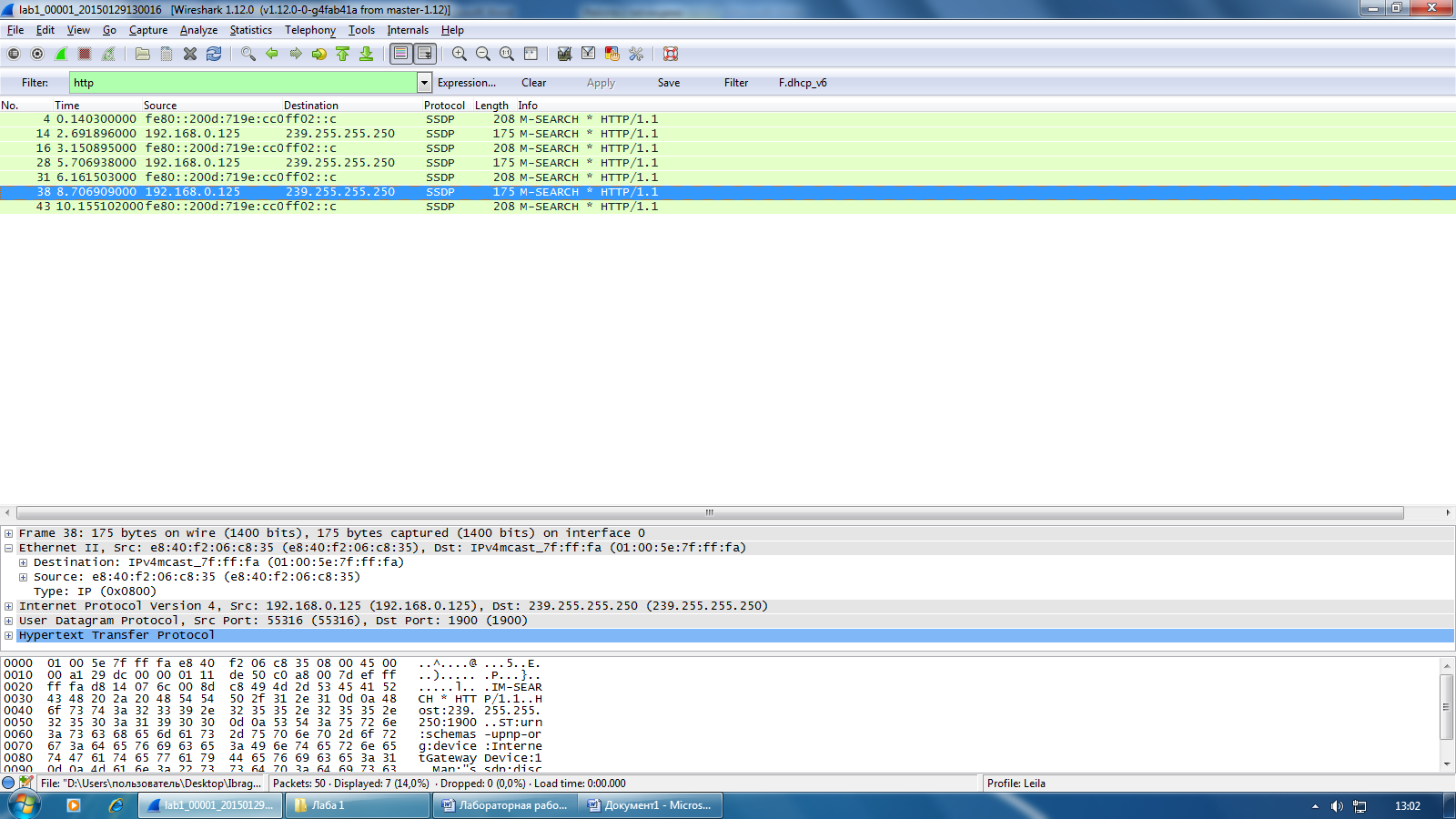


Рисунок 13 – Фильтрация по протоколу HTTP

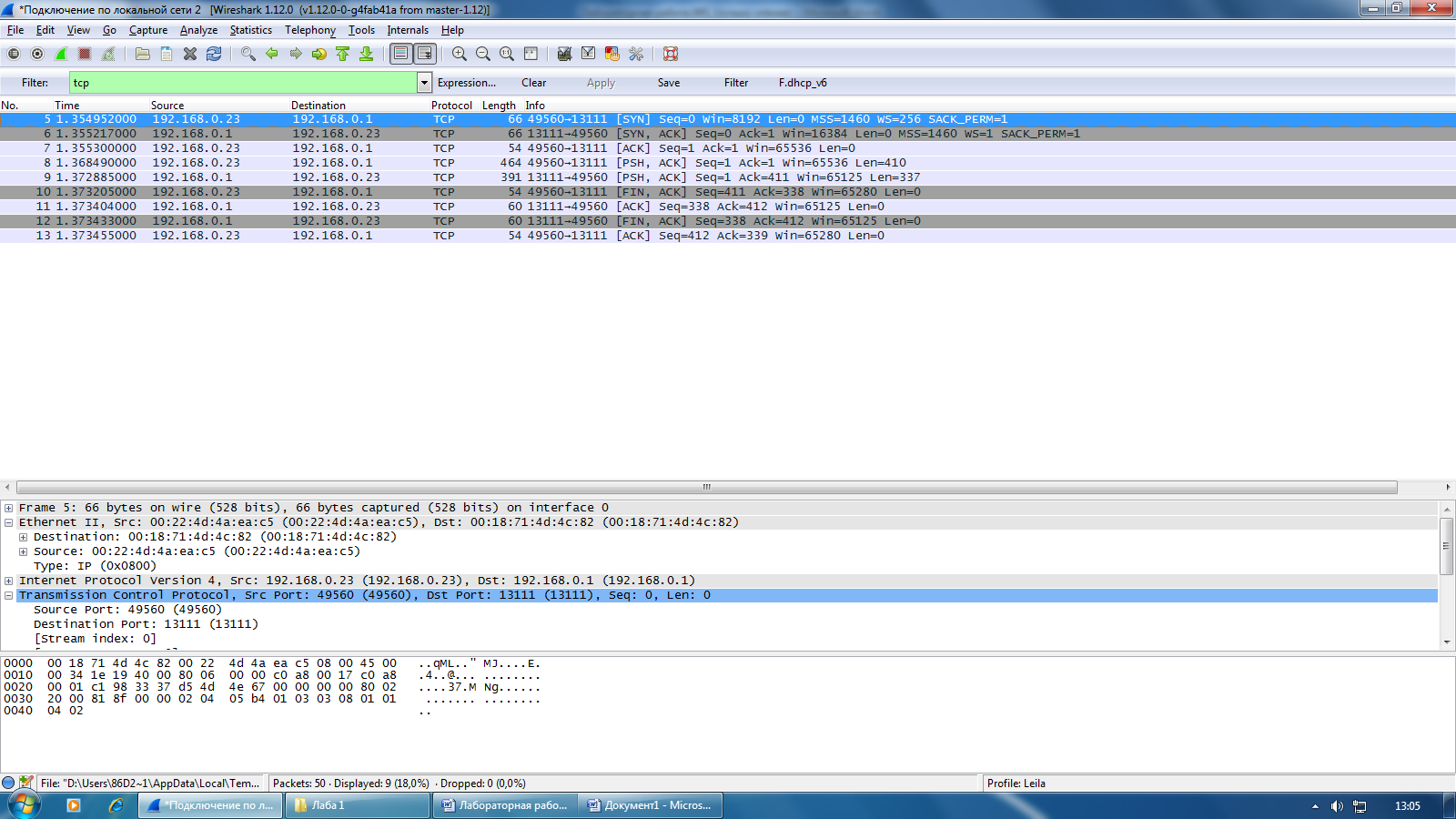


Рисунок 14 – Фильтрация по протоколу TCP

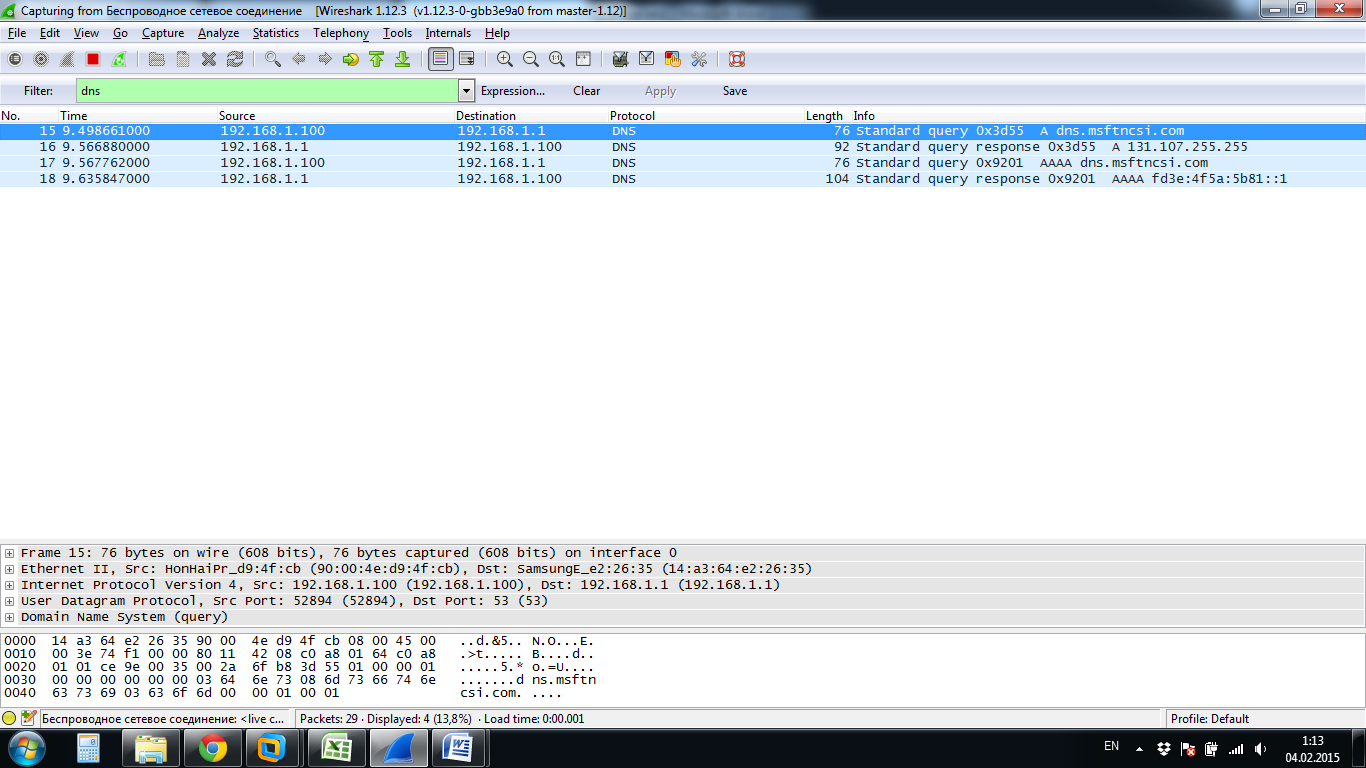


Рисунок 15 – Фильтрация по протоколу DNS

Проведем анализ третьего кадра сеанса TCP (Рисунок 16). Здесь порт 53274 и порт 443 являются портами источника и назначения соответственно. Выражение Seq=1903685773 является последовательным номером клиента. Поле подтверждения отсутствует. Длина заголовка = 0. Установлен бит флагов [SYN].

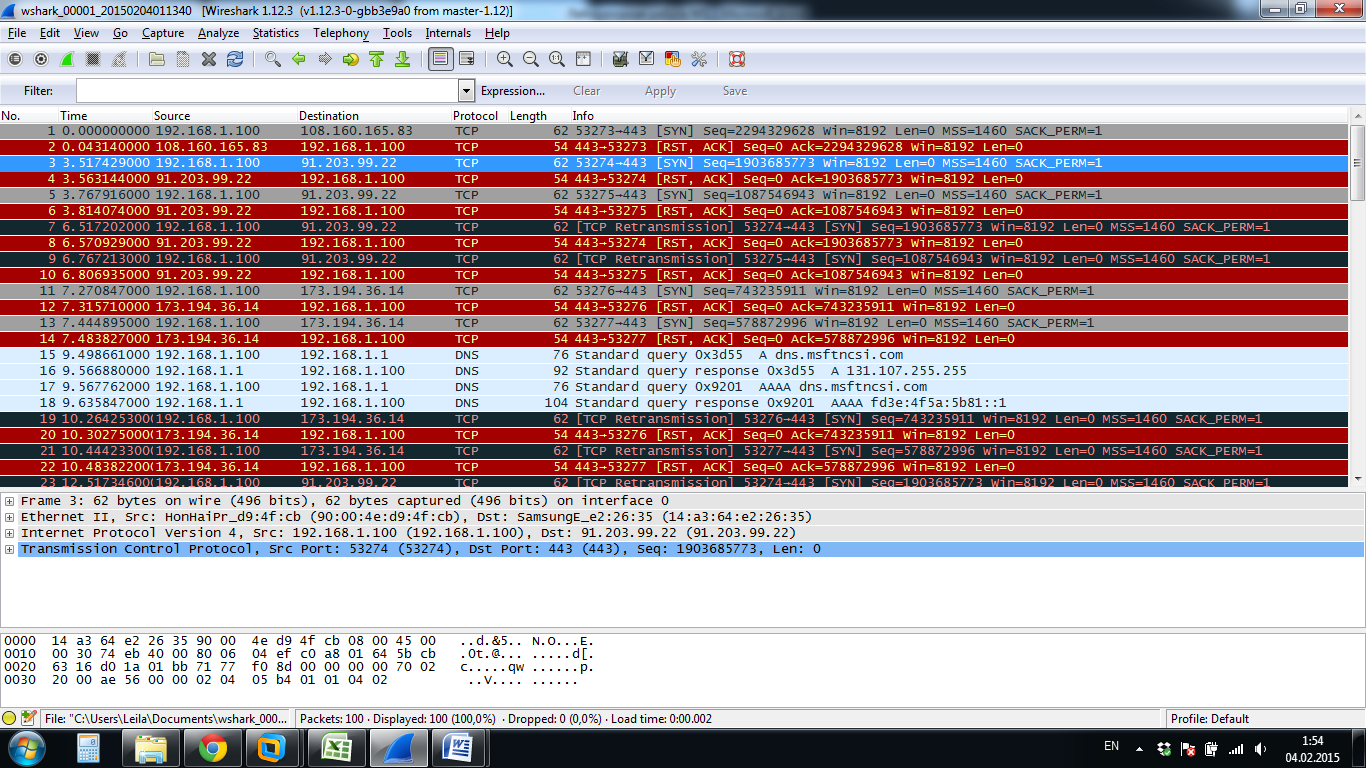


Рисунок 16 – сеансы TCP

Определим реальные последовательные номера в рамках сеансов TCP (Рисунок 17).

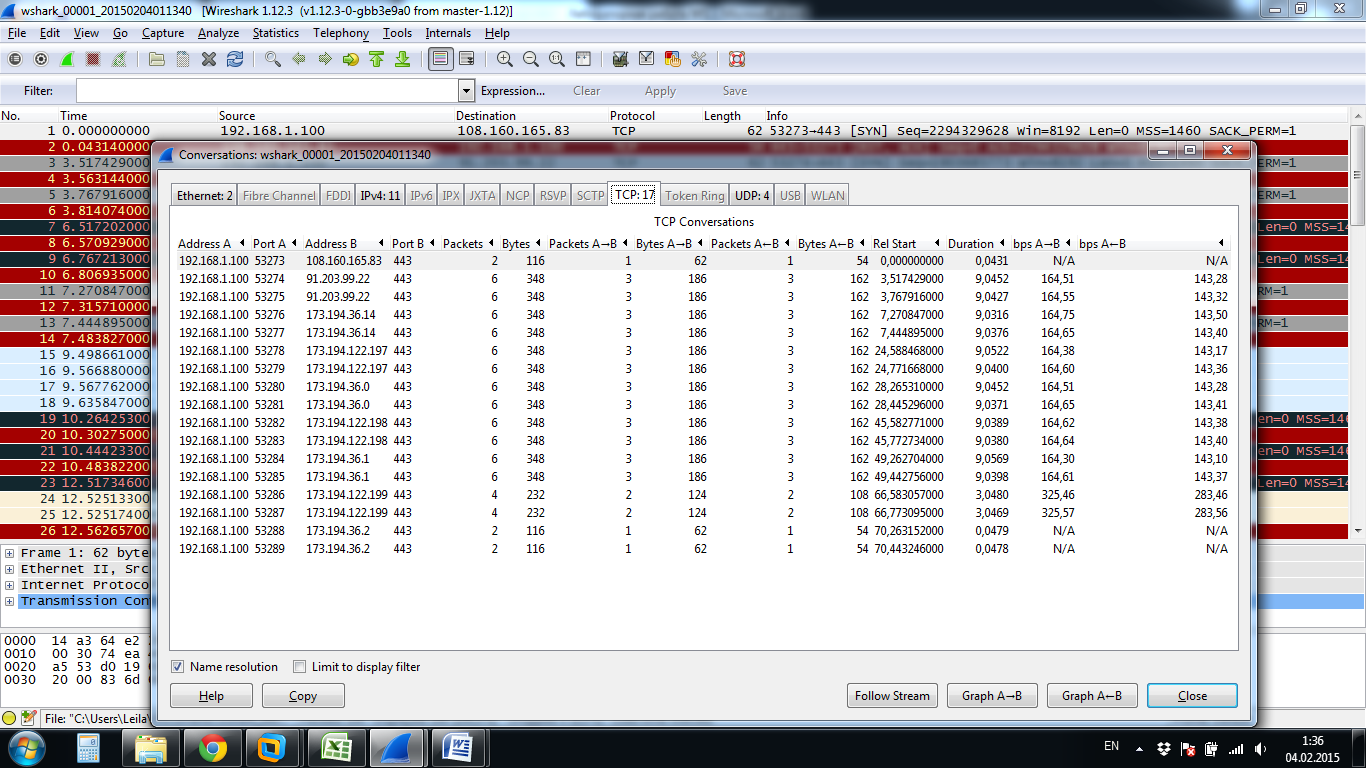


Рисунок 17 – Статистика по сеансам TCP

Выберем первый сеанс и с помощью контекстного меню Apply as Filter >> Selected >> A<—>B отобразим в буфере кадры, принадлежащие этому сеансу (Рисунок 18).

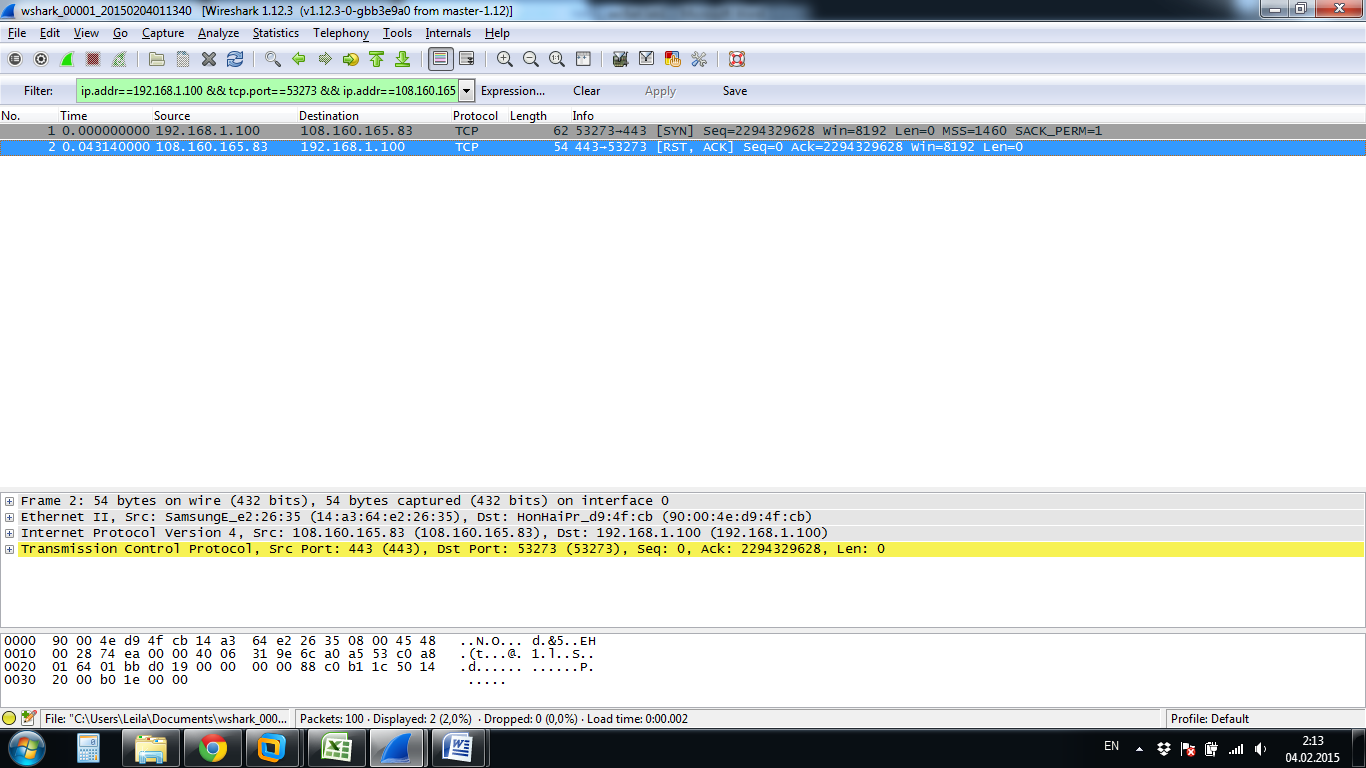


Рисунок 18 – Кадры, принадлежащие первому сеансу

С помощью функции IO Graph пункта меню Statistics можно вывести полученную информацию в графическом режиме (Рисунок 19).

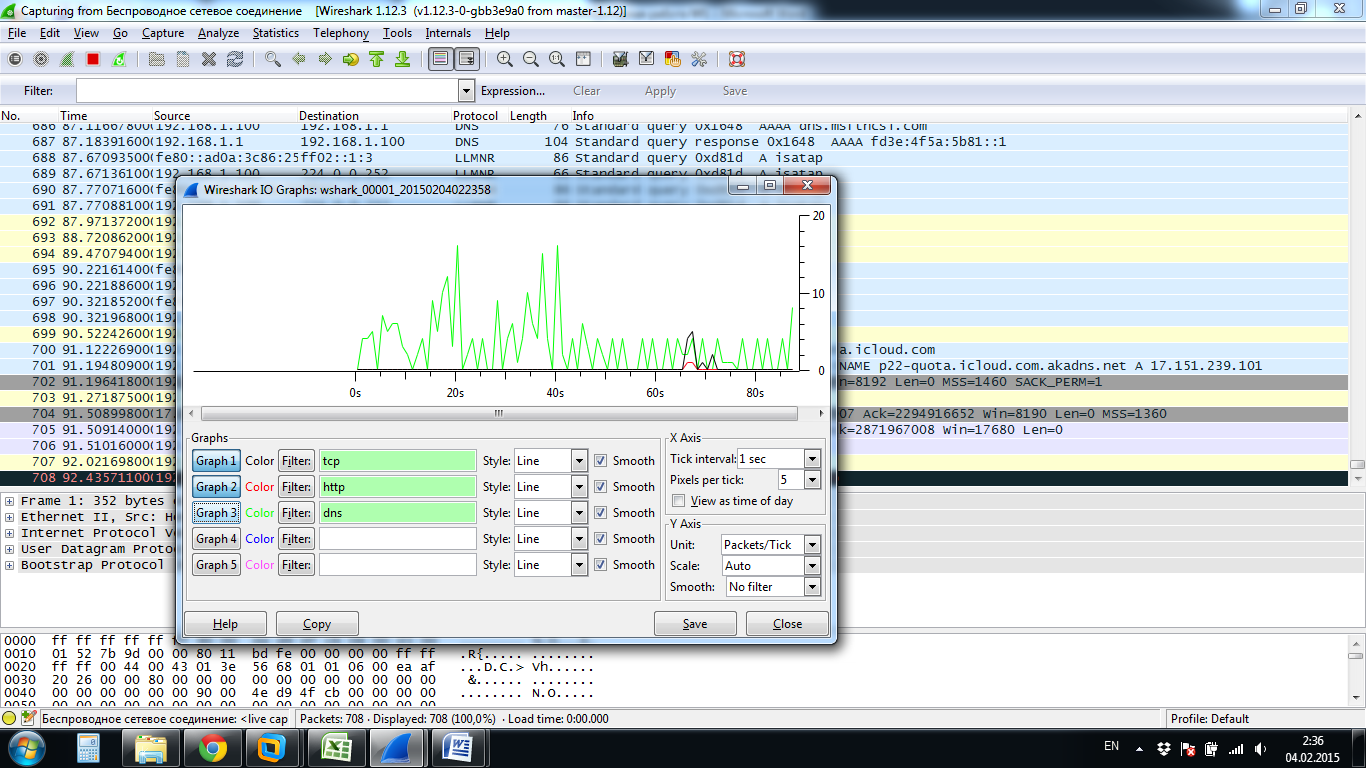


Рисунок 19 – IO Graph

Заключение