Минобрнауки России

Федеральное государственное бюджетное образовательное

учреждение высшего образования

**НИЖЕГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

**ИМ. Р.Е. АЛЕКСЕЕВА**

ИНСТИТУТ РАДИОЭЛЕКТРОНИКИ И ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

|  |  |  |
| --- | --- | --- |
|  |  |  |

Курс “Сети и телекоммуникации”

**Отчет по лабораторной работе №3**

Выполнил: Потапов В. В.

Проверил: Гай В.Е.

Нижний Новгород 2020

**Задание на работу:**

Для экспериментов использовать схему из первой лабораторной работы. Все ip-адреса (или маски) необходимо поменять так, чтобы адрес сети у всех компьютеров был один. Все действия должны быть выполнены в симуляторе сетей CORE.

**Часть 1. Формирование запроса и получение ответа**

1. Начать захват пакетов при помощи WireShark.

2. Сформировать кадр ARP-запроса с помощью утилиты PackETH и отправить его в сеть (компьютеры выбрать самостоятельно).

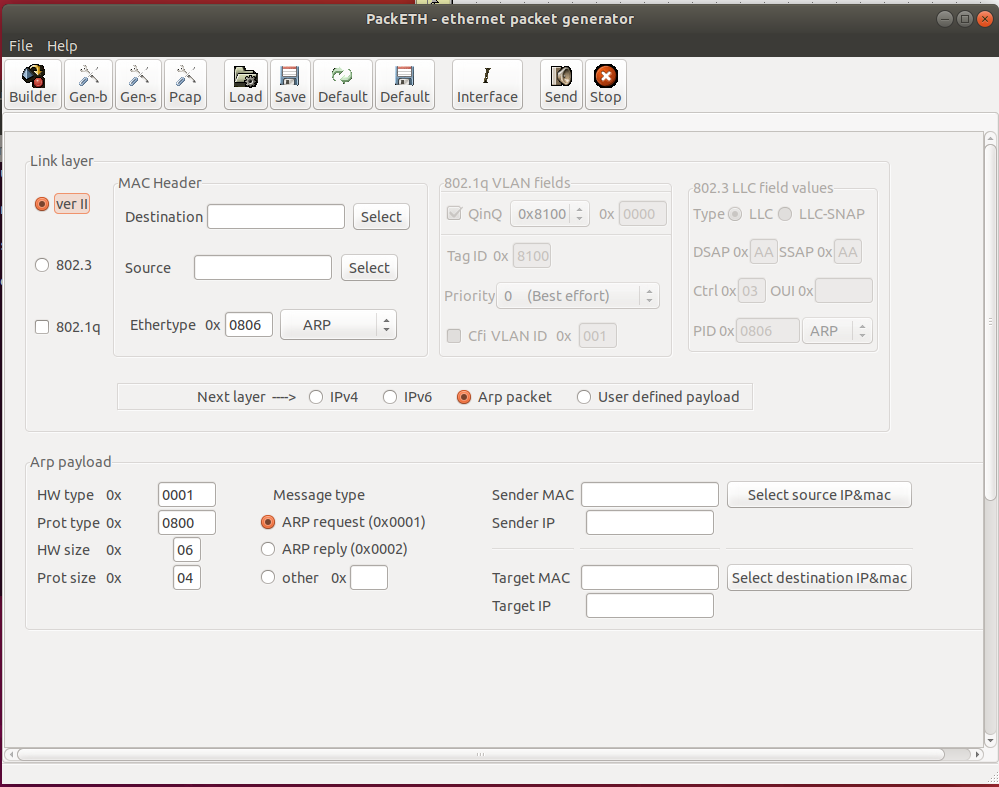
3. Убедиться, что был получен кадр ARP-ответа, соответствующий посланному запросу. Захваченные пакеты сохранить для отчета. Вывести arp таблицу (команда «arp»).

4. Прекратить захват пакетов.

**Часть 2**. **ARP-спуфинг**

1. Выделить на схеме и обозначить три компьютера: A, B, Сервер.

2. Подготовить кадр ARP-ответа, направляемый Сервером хосту А с помощью программы PackETH. Кадр должен быть составлен так, чтобы MAC-адресу Сервера соответствовал IP-адрес хоста В. Вывести arp таблицу на хосте А. Отправить сформированный пакет от Сервера хосту А.



Для запуска packEth в консоли выполните команду «xhost +», в консоли узла «DISPLAY=:0 packeth» или «DISPLAY=:0 packeth --sync».

netcat (англ. net сеть + cat) — утилита Unix, позволяющая устанавливать соединения TCP и UDP, принимать оттуда данные и передавать их.

Организация чата между узлами с помощью netcat:

1. На первом узле (192.168.1.100):

$ nc -lp 9000

2. На втором узле:

$ nc 192.168.1.100 9000

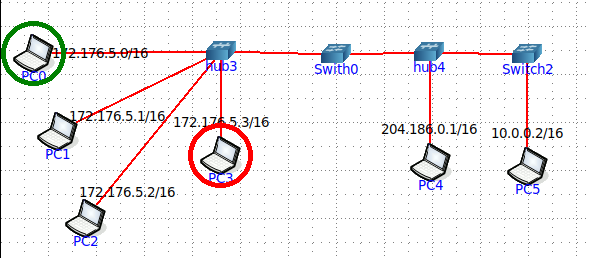
3. Начать захват пакетов при помощи WireShark на Сервере.

4. Попытаться установить соединение между хостом А и хостом В с помощью программы netcat (А отправляет сообщения В). Убедиться, что запросы от хоста A, направленные хосту В поступают на Сервер.

5. Прекратить захват пакетов.

6. Сохранить для отчета отправленный кадр ARP-ответа и несколько перехваченных пакетов, переданных на Сервер, arp таблицу хоста А.

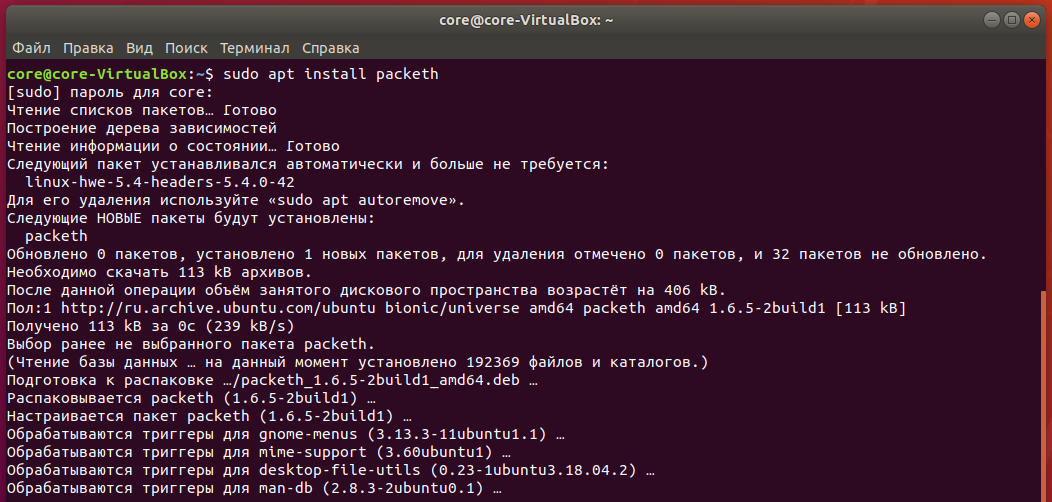
**Структура сети:**



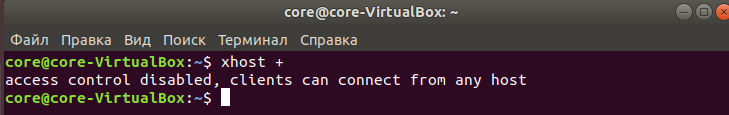
**Доказательства работы:**

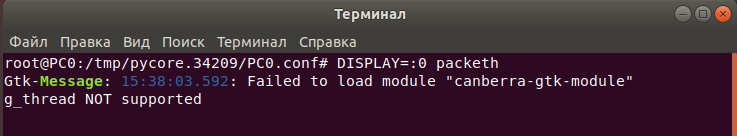
**Часть 1. Формирование запроса и получение ответа**

Устанавливаем необходимый пакет программ:

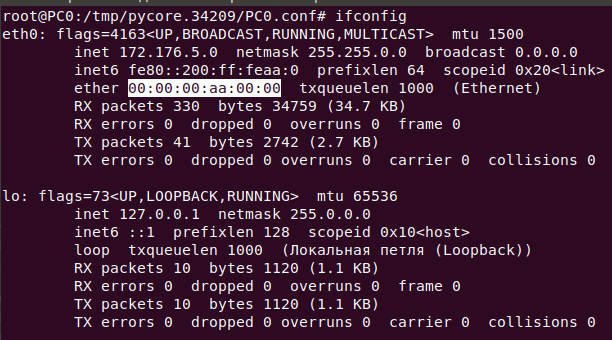


Для запуска **packEth** в консоли выполняем команду «**xhost +**», в консоли узла «**DISPLAY=:0 packeth**»

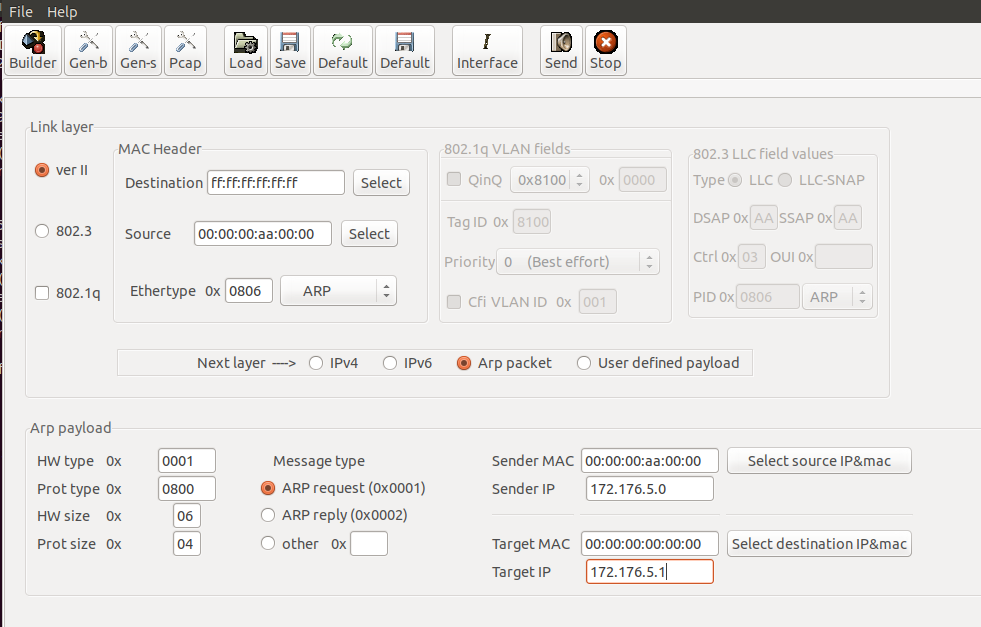




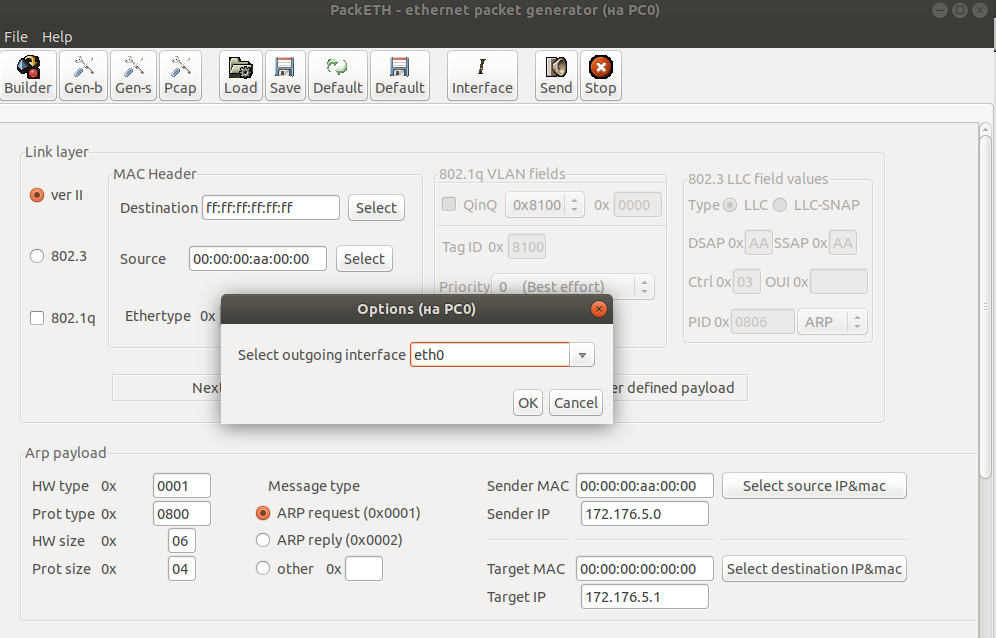
Узнаем MAC-адрес отправителя:



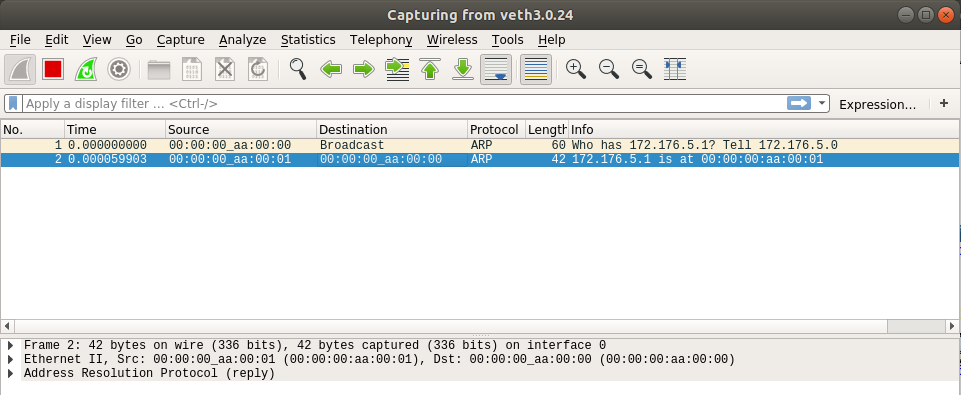
Настраиваем **PackEth**:

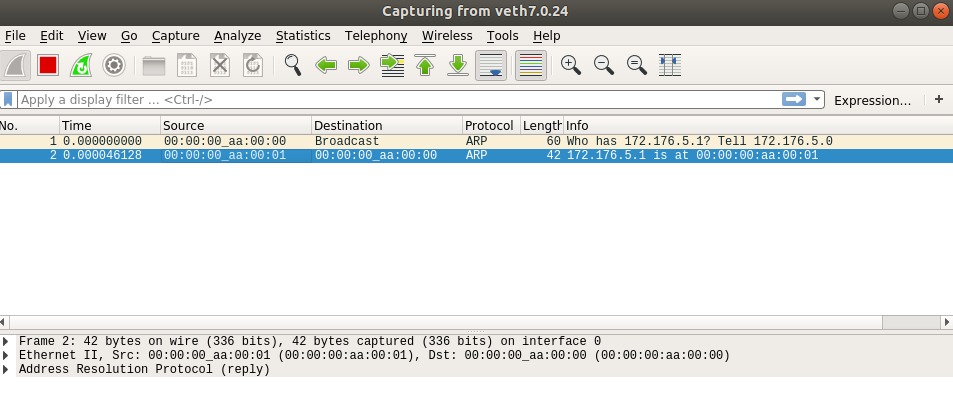


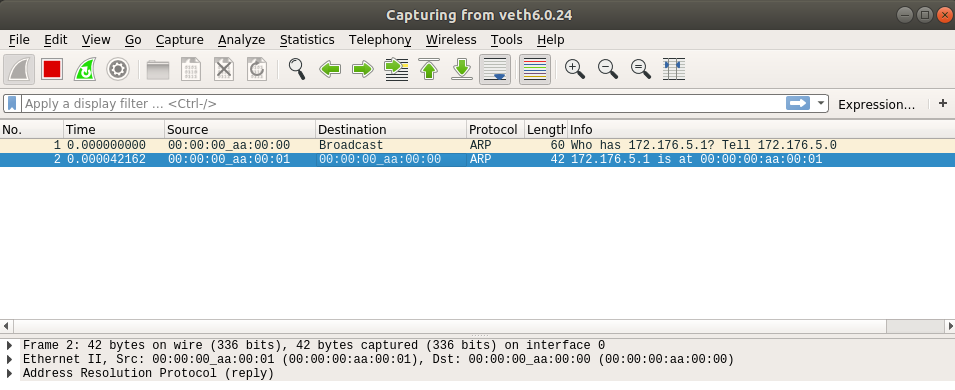
Выбираем интерфейс **eth0**:

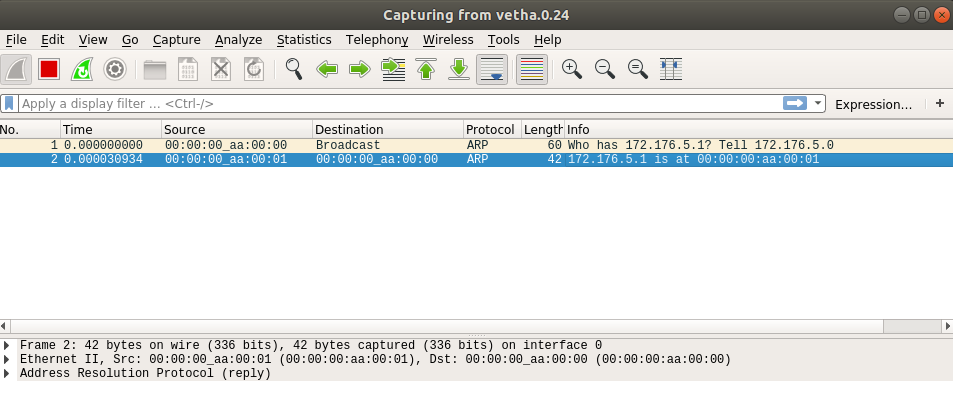


Производим захват пакетов при помощи **Wireshark** у всех PC, нажимаем **send** (отправляем **ARP-запрос**) в **PackEth** и получаем **ARP-ответ**:



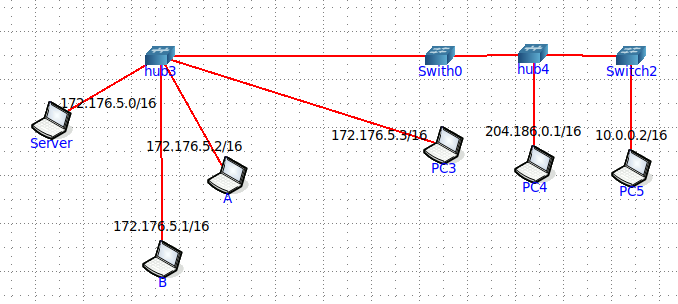




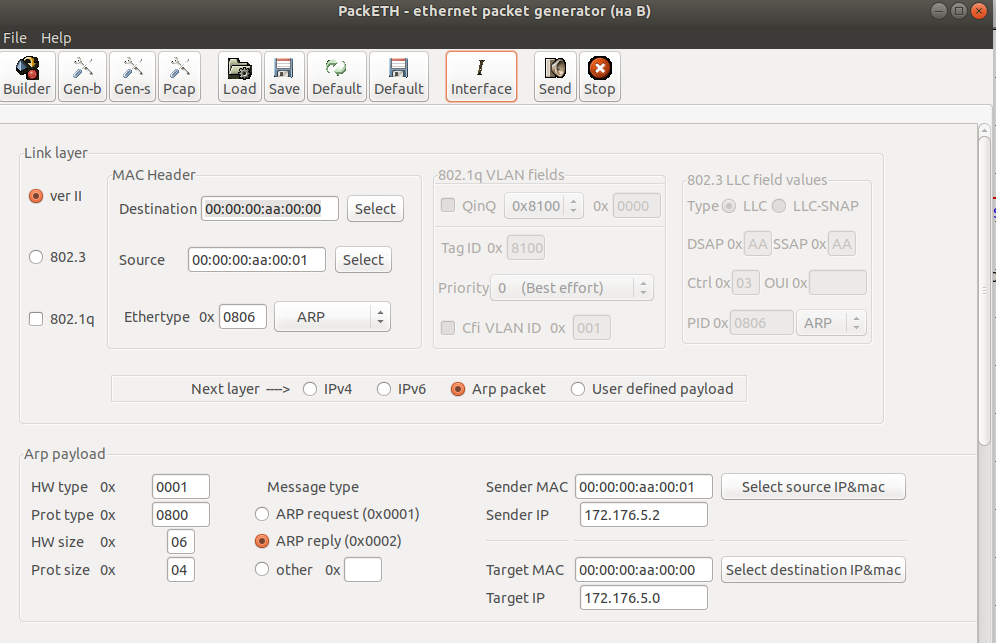


**Часть 2**. **ARP-спуфинг**

Обозначаем на схеме три компьютера: **A, B, Сервер**:



Подготавливаем кадр **ARP-ответа** хоста **B** с помощью программы **PackEth**:



**Destination** – MAC-адрес Сервера

**Source** – MAC-адрес хоста B

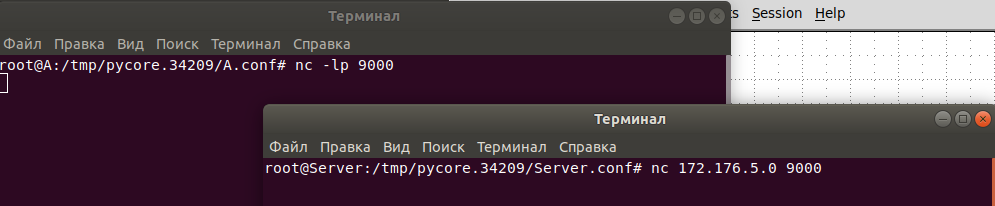
**Sender MAC –** MAC-адрес хоста B

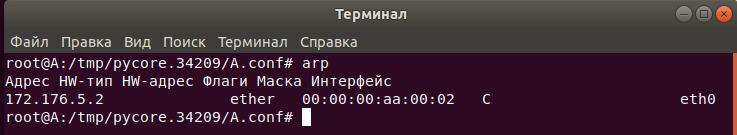
**Sender IP –** ip-адрес хоста A

**Target MAC –** MAC-адрес Сервера

**Target IP –** ip-адрес Сервера

Устанавливаем соединение между **Сервером** и хостом **A** при помощи утилиты netcat:





Отправляем(**send**) сформированный пакет от **Сервера** хосту **А** в **PackETH**:

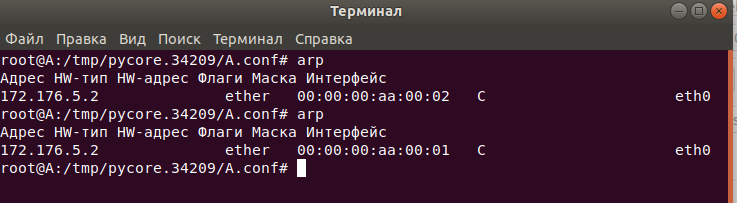
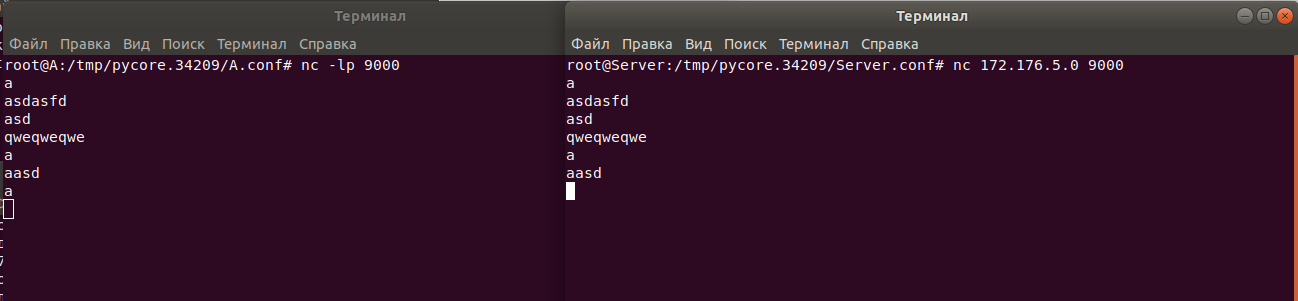
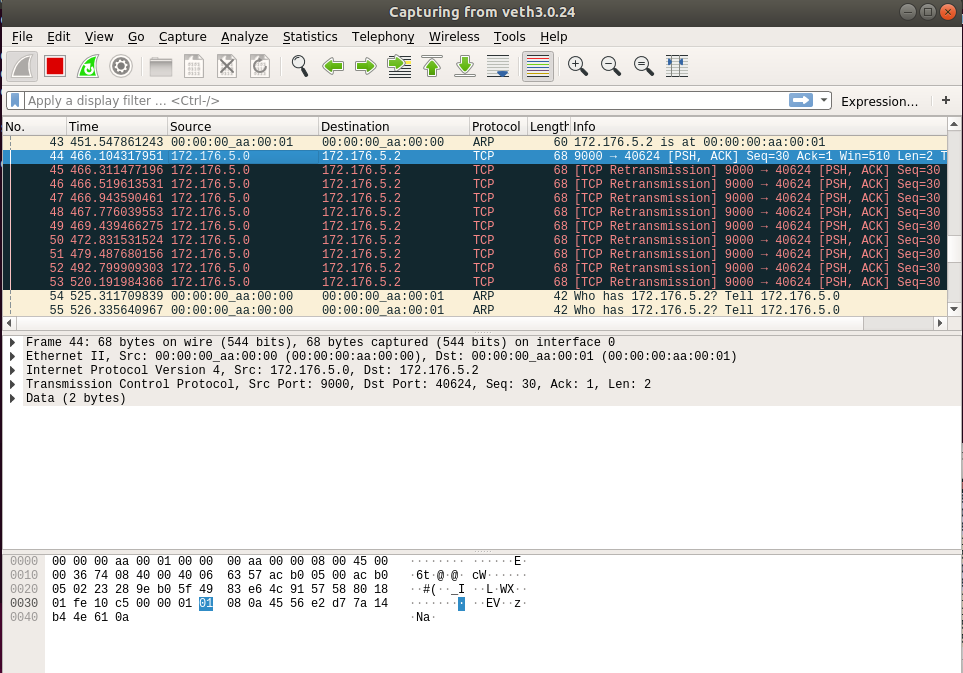
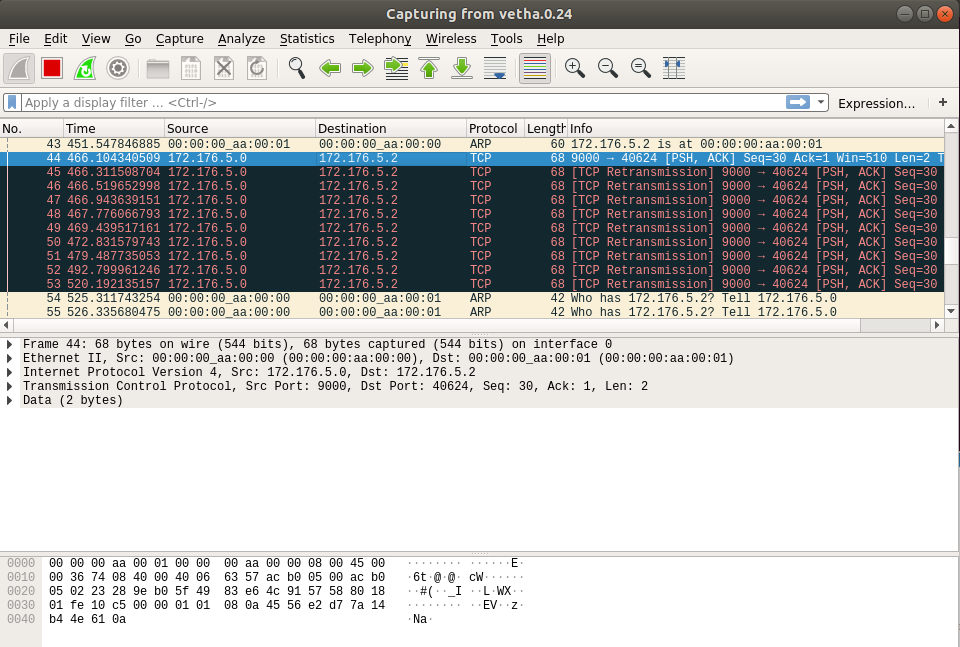


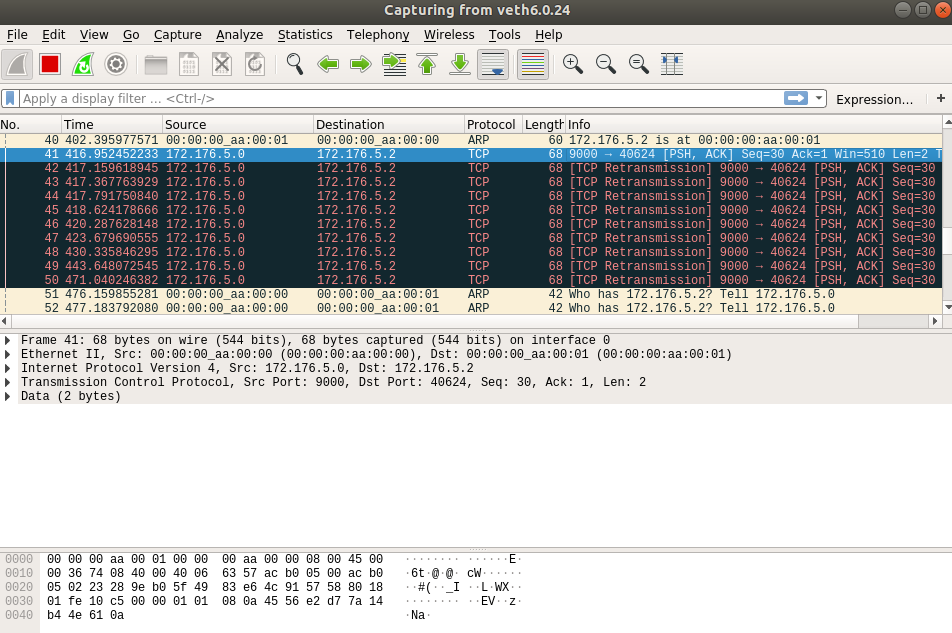
Таблица MAC-адресов у **Сервера** поменялась. Отправим какие-нибудь пакеты данных через **netcat**:



Заметим, до хоста **A** они не сразу доходят. Посмотрим на захват пакетов при помощи **WireShark**. Хосту **B** передаются пакеты, которые не должны были передаваться:







В итоге пакеты доходят до хоста **A**:

