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 №1**

Выполнил: студент группы 18 В-1

Петров М.А.

Проверил: Гай В.Е.

Нижний Новгород 2021

**Задание на работу:**

**Работа с анализатором протоколов tcpdump**

1. Запустить tcpdump в режиме захвата всех пакетов, проходящих по сети. Количество захватываемых пакетов ограничить 10. Результаты протоколировать в файл.

2. Запустить tcpdump в режиме перехвата широковещательного трафика (фильтр по MAC-адресу). Количество захватываемых пакетов ограничить 5. Включить распечатку пакета в шестнадцатеричной системе (включая заголовок канального уровня).

3. Запустить tcpdump так, чтобы он перехватывал только пакеты протокола ICMP, отправленные на определенный IP-адрес. При этом включить распечатку пакета в шестнадцатеричной системе и ASCII-формате (включая заголовок канального уровня). Количество захватываемых пакетов ограничить 3. Для генерирования пакетов возпользоваться утилитой ping.

4. Запустить tcpdump в режиме сохранения данных в двоичном режиме так, чтобы он перехватывал пакеты, созданные утилитой traceroute для определения маршрута к заданному в варианте узлу. Включить распечатку пакета в шестнадцатеричной системе и ASCII-формате (включая заголовок канального уровня). Количество захватываемых пакетов ограничить 7. Результат работы программы писать в файл.

5. Прочесть программой tcpdump созданный в предыдущем пункте файл.

6. Придумать три задания для фильтрации пакетов на основе протоколов ARP, TCP, UDP, ICMP

**Работа с анализатором протоколов wireshark**

1. Захватить 5-7 пакетов широковещательного трафика (фильтр по IP-адресу). Результат сохранить в текстовый файл.

2. Захватить 3-4 пакета ICMP, полученных от определенного узла. Для генерирования пакетов воспользоваться утилитой ping. Результат сохранить в текстовый файл.

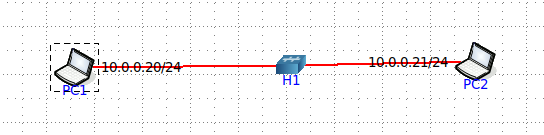
3. Перехватить пакеты, созданные утилитой traceroute для определения маршрута к заданному в варианте узлу. По результатам построить диаграмму Flow Graph. Диаграмму сохранить либо в виде текстового файла либо в виде изображения.

4. Прочесть файл, созданный программой tcpdump. Сравнить с тем, что было получено утилитой wireshark.

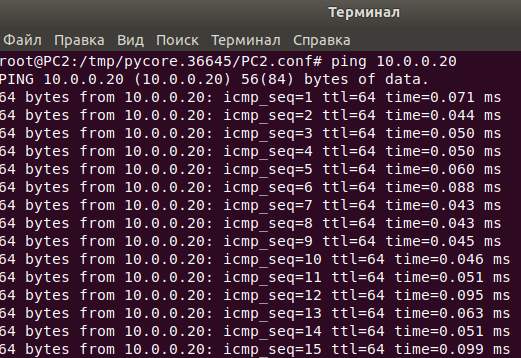
**Работа с анализатором протоколов tcpdump**

**Задание 1:** запустить tcpdump в режиме захвата всех пакетов, проходящих по сети. Количество захватываемых пакетов ограничить 10. Результаты протоколировать в файл.

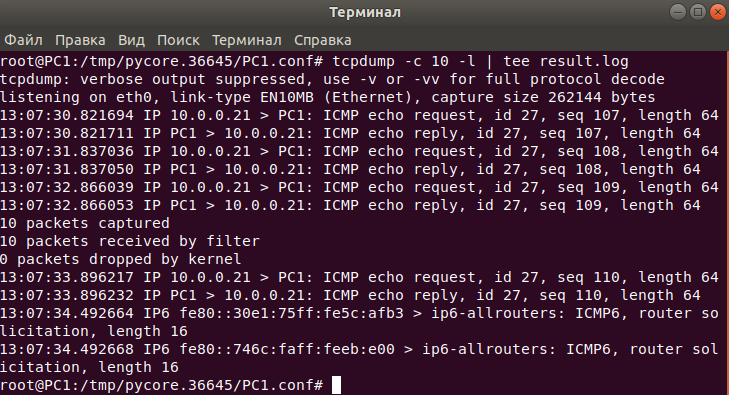
Структура сети:

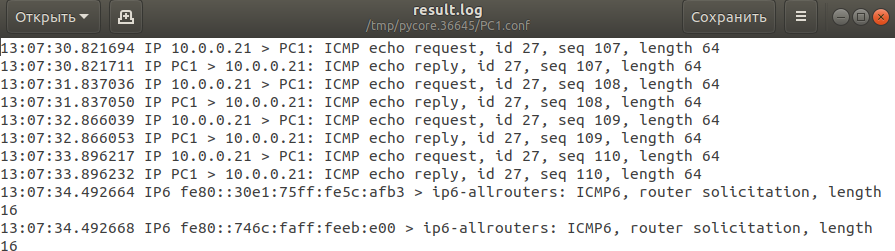
****

Начинаем передачу пакетов с компьютера PC2 на PC1:



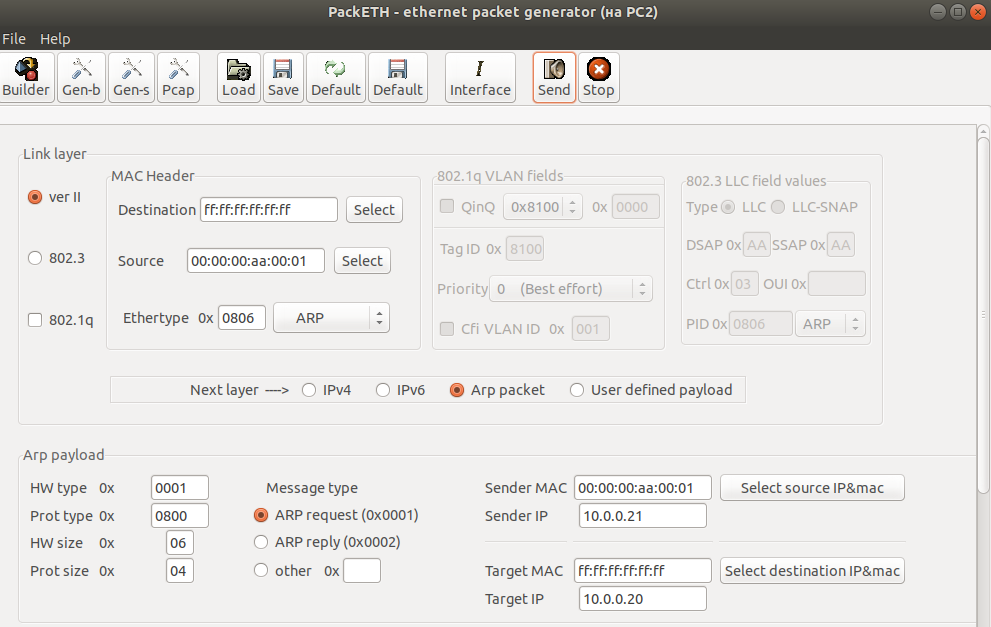
На PC1 тем временем запускаем **tcpdump** с нужные параметрами:

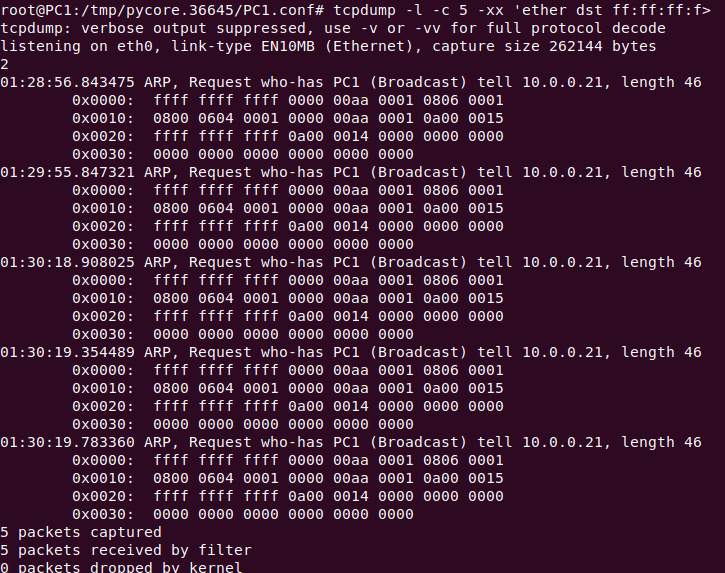




**Задание 2:** запустить tcpdump в режиме перехвата широковещательного трафика (фильтр по MAC-адресу). Количество захватываемых пакетов ограничить 5. Включить распечатку пакета в шестнадцатеричной системе (включая заголовок канального уровня).

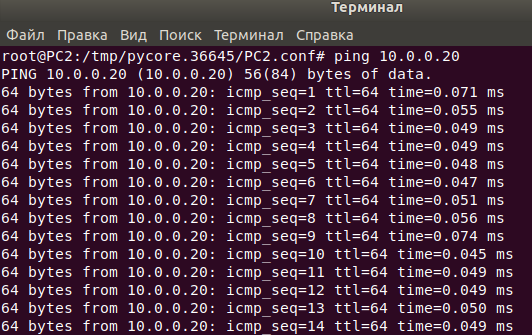
Составляем широковещательный MAC-пакет на хосте n2 с помощью PackETH:

****

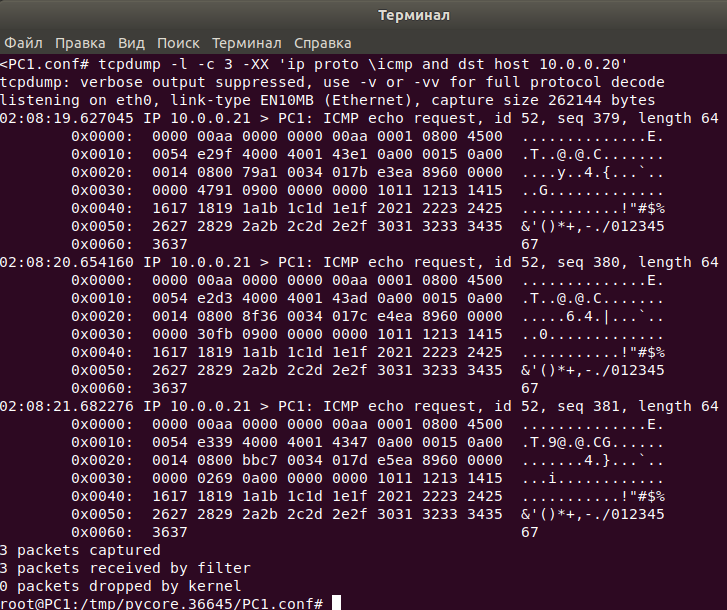


**Задание 3**: запустить tcpdump так, чтобы он перехватывал только пакеты протокола ICMP, отправленные на определенный IP-адрес. При этом включить распечатку пакета в шестнадцатеричной системе и ASCII-формате (включая заголовок канального уровня). Количество захватываемых пакетов ограничить 3. Для генерирования пакетов воспользоваться утилитой ping.

Снова начнём с передачи пакетов с компьютера PC2 на PC1:

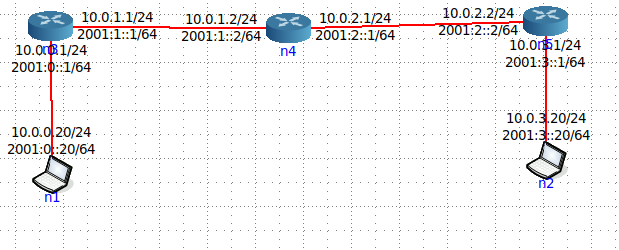


Параллельно с этим запускаем tcpdump на 1-ом компьютере и пропишем вместе с ним нужные параметры и фильтры, которые помогут перехватывать только пакеты ICMP.

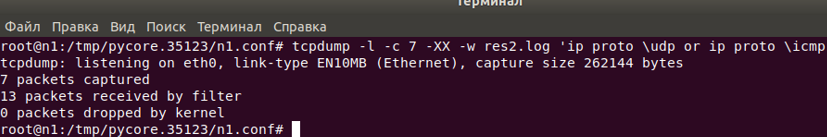


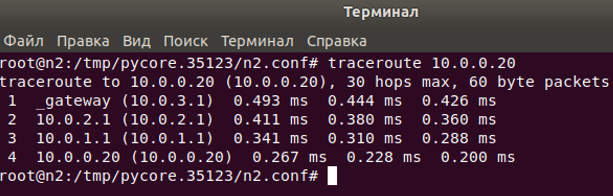
**Задание 4**: запустить tcpdump в режиме сохранения данных в двоичном режиме так, чтобы он перехватывал пакеты, созданные утилитой traceroute для определения маршрута к заданному в варианте узлу. Включить распечатку пакета в шестнадцатеричной системе и ASCII-формате (включая заголовок канального уровня). Количество захватываемых пакетов ограничить 7. Результат работы программы писать в файл.

Увеличим структуру сети:

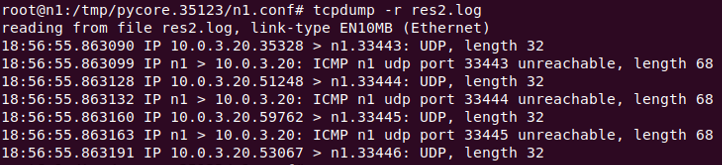


Запустим tcpdump на компьютере n1 с нужными параметрами, фильтрами и одновременно передадим пакеты с компьютера n2 с помощью утилиты traceroute:





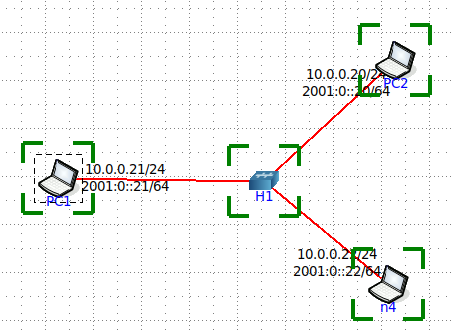
**Задание 5:** прочесть программой tcpdump созданный в предыдущем пункте файл.



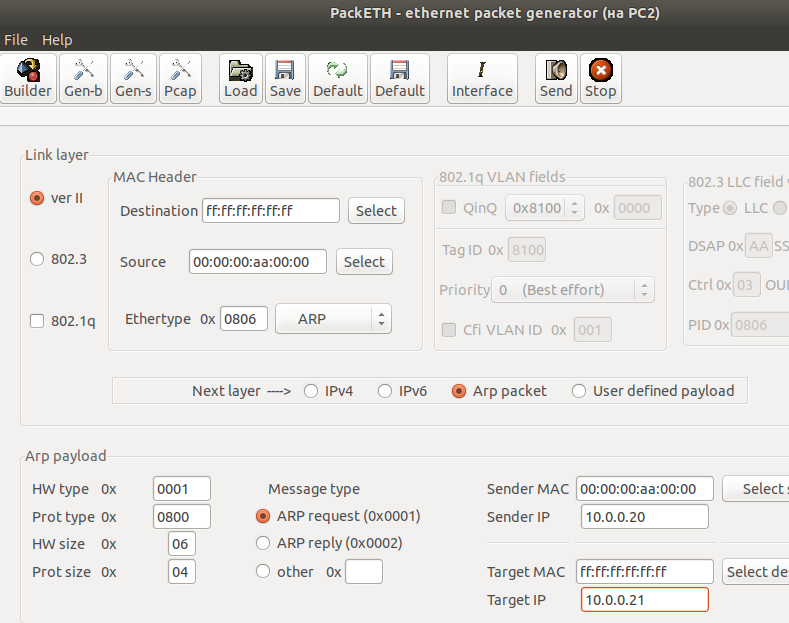
**Задание 6**: придумать три задания для фильтрации пакетов на основе протоколов ARP, TCP, UDP, ICMP

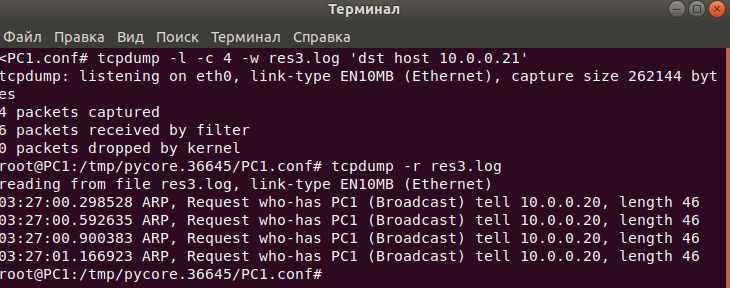
1) Запустить tcpdump в режиме перехвата широковещательного трафика (с фильтром ip-адреса получателя). Сохранять данные в двоичном формате. Количество захватываемых пакетов ограничить до 4-х. Широковещательную рассылку совершать с помощью PackETH.

Для начала создадим новую сеть, состоящую из 1 хаба и 3 компьютеров:



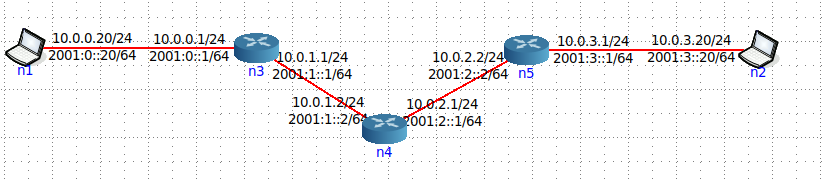
Через PackETH осуществляем широковещательную рассылку с PC2 на PC1. На первом же компьютере тем временем запускаем tcpdump и наблюдаем за результатом. После 4-ех нажатий клавиши «send» выведется инфа об отправки.



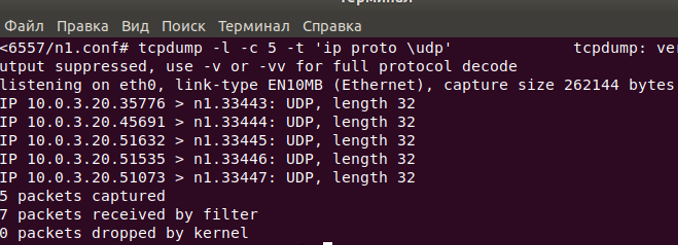


2) Запустить tcpdump, перехватывающий пакеты с фильтром протокола UDP. Не отображать метку времени в каждой строке. Количество захватываемых пакетов ограничить до 5. Для передачи пакетов воспользоваться утилитой traceroute.

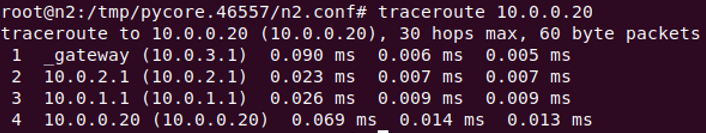
Снова создаём новую сеть:



И осуществляем передачу информации с компьютера n1 на n2

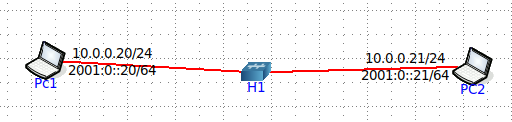


Затем запускаем tcpdump и перехватываем пакеты с информацияей, используя фильтр протокола

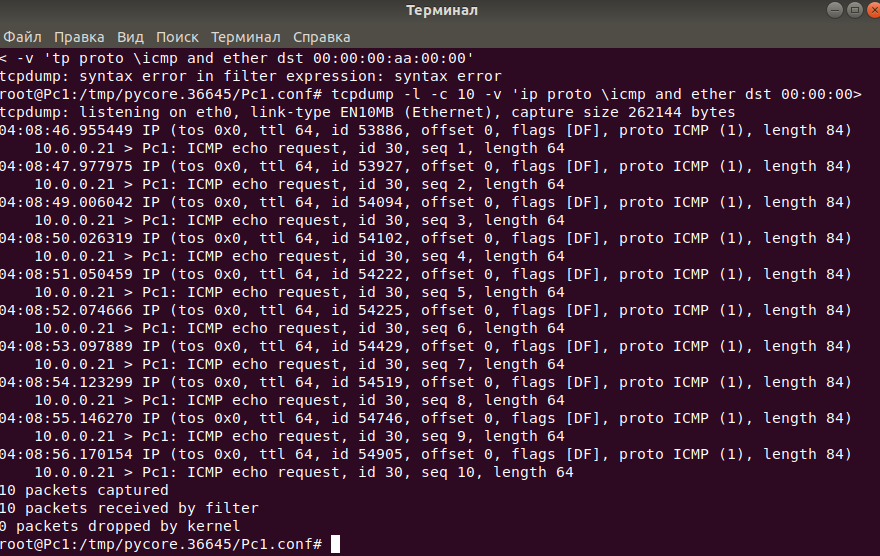


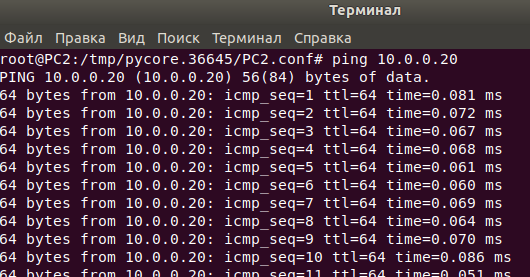
3) Запустить tcpdump, перехватывающий пакеты только протокола ICMP (с фильтром MAC-адреса получателя). Вывод подробной информации. Количество захватываемых пакетов ограничить до 10. Для передачи пакетов воспользоваться утилитой ping.

Меняем сеть:



Делаем передачу пакетов с PC1 на другой комп и не забываем пинговать со второго – первый пк.

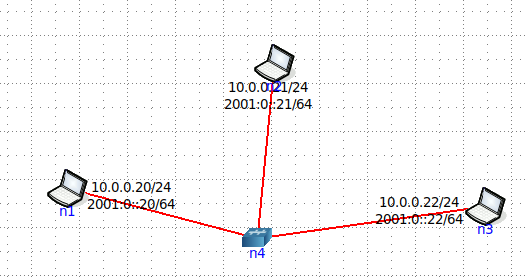




**Работа с анализатором протоколов Wireshark**

**Задание 1**: захватить 5-7 пакетов широковещательного трафика (фильтр по IP-адресу). Результат сохранить в текстовый файл.

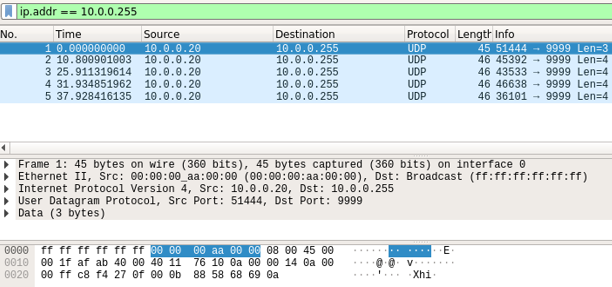
Структура сети:



Создаем широковещательный трафик на компьютере n1 и отправляем 5-7 раз сообщение:

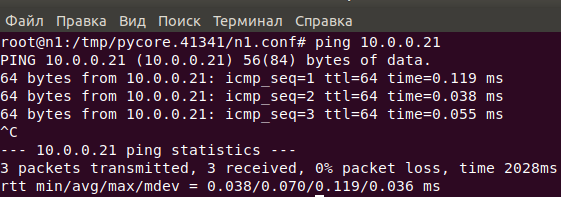


В Wireshark наблюдаем захват пакетов, используя фильтр по IP-адресу:

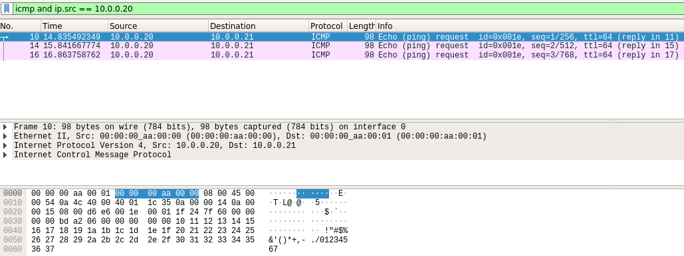


**Задание 2**: захватить 3 пакета ICMP, полученных от определенного узла. Для генерирования пакетов воспользоваться утилитой ping. Результат сохранить в текстовый файл.

Начинаем генерировать пакеты в компьютере n1, используя ping:

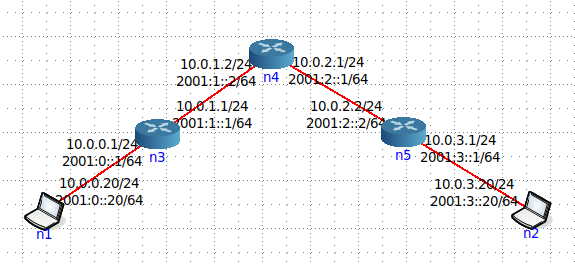


А также в Wireshark захватываем все эти пакеты на пк n2, фильтруя всё это по протоколу ICMP и IP-адресу:

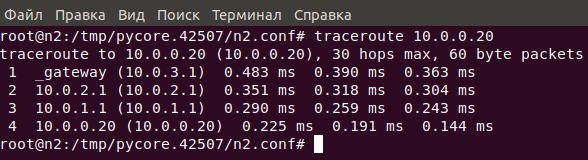
****

**Задание 3:** перехватить пакеты, созданные утилитой traceroute для определения маршрута к заданному в варианте узлу. По результатам построить диаграмму Flow Graph. Диаграмму сохранить либо в виде текстового файла, либо в виде изображения.

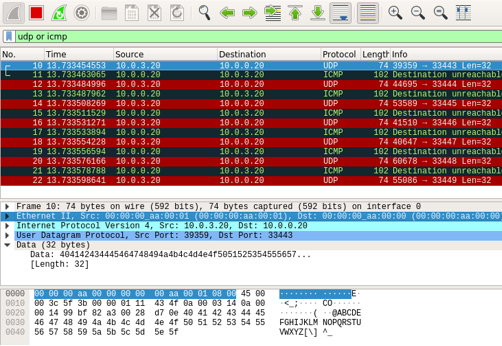
Делаем сеть из двух пк и 3 свитчей:



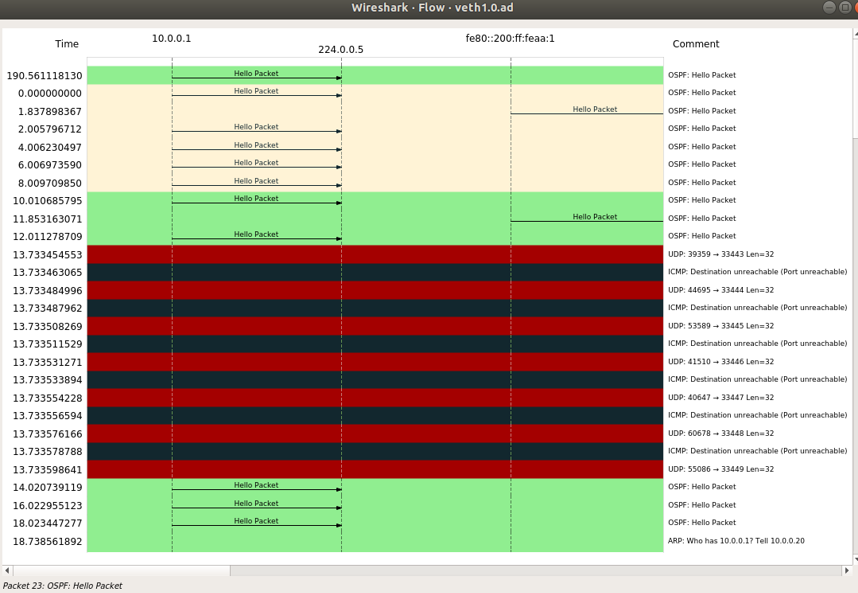
С помощью traceroute передаем пакеты с компьютера n2 на пк n1:



Затем производим захват пакетов в Wireshark на компьютере n1:

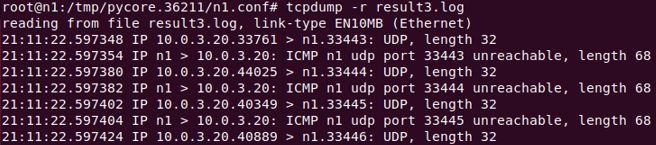


И по результатам строим диаграмму Flow Graph:

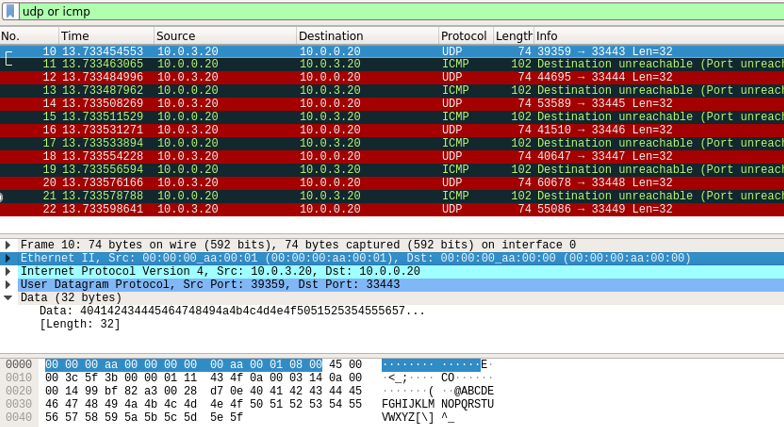


**Задание 4**: прочесть файл, созданный программой tcpdump. Сравнить с тем, что было получено утилитой wireshark.

Если сравнить чтение файла в терминале:



И чтение файла в wireshark:



То можно сделать вывод, что в Wireshark более подробно расписана вся необходимая информация о пересылаемых пакетах.