МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

НИЖЕГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ

УНИВЕРСИТЕТ им. Р.Е.АЛЕКСЕЕВА

Институт радиоэлектроники и информационных технологий

Кафедра информатики и систем управления

ОТЧЕТ

По лабораторной работе №4  
 «сети и телекоммуникации»

РУКОВОДИТЕЛЬ:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Гай В.Е.

(подпись) (фамилия, и.,о.)

СТУДЕНТ:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Зиятдинов И.И.

(подпись) (фамилия, и.,о.)

19-АС

(шифр группы)

Работа защищена «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_

С оценкой \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Нижний Новгород 2021

**Задание на лабораторную работу:**

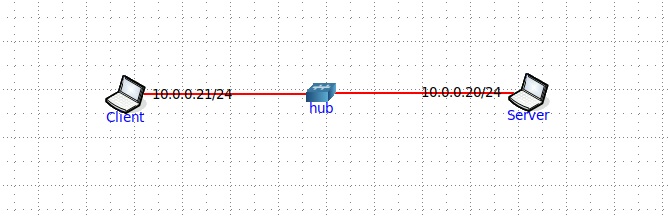
1. Подключиться к UDP серверу с помощью TCP клиента. Объяснить полученные пакеты в WireShark

2. Запустить TCP клиент, сервер, передать данные, затем прервать соединение (Ctrl + C) на стороне сервера. Объяснить полученные пакеты в WireShark

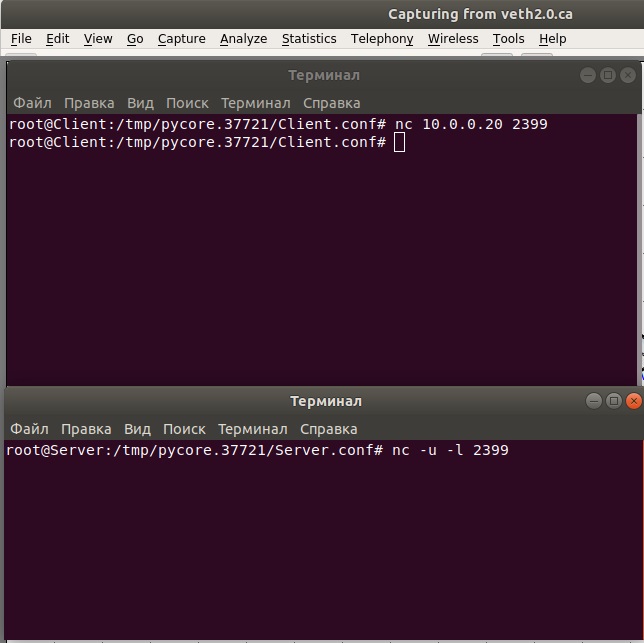
3. Запустить TCP клиент, сервер, передать данные, затем прервать соединение (Ctrl + C) на стороне клиента. Объяснить полученные пакеты в WireShark

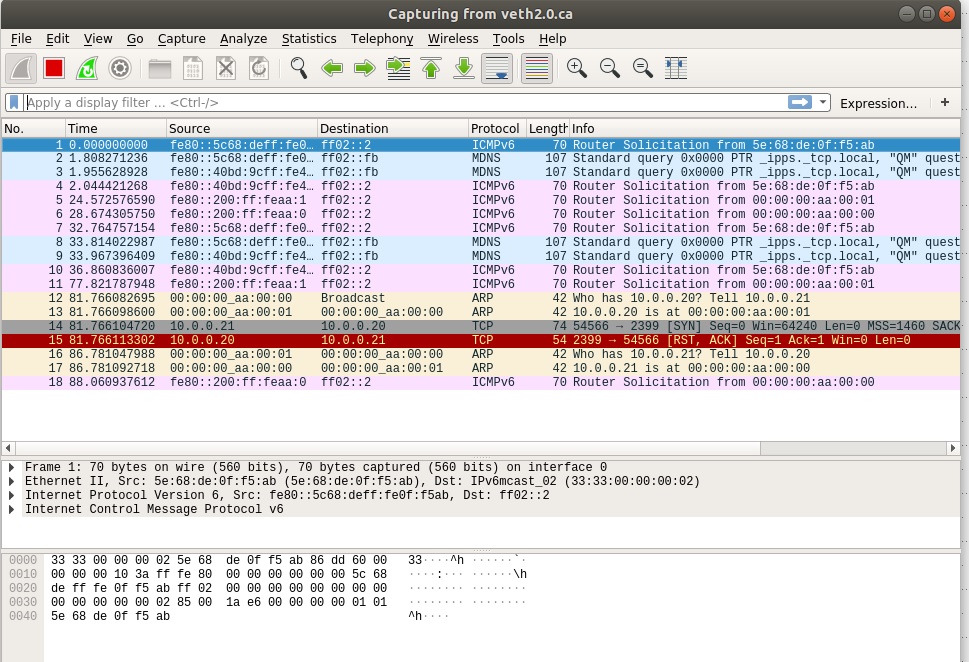
**Ход работы:**

1. Создать сеть в Core, состоящую из двух компьютеров.

****

Запустить UDP сервер на одном из компьютеров. Подключиться к UDP серверу с помощью TCP клиента. Объяснить полученные пакеты в WireShark.

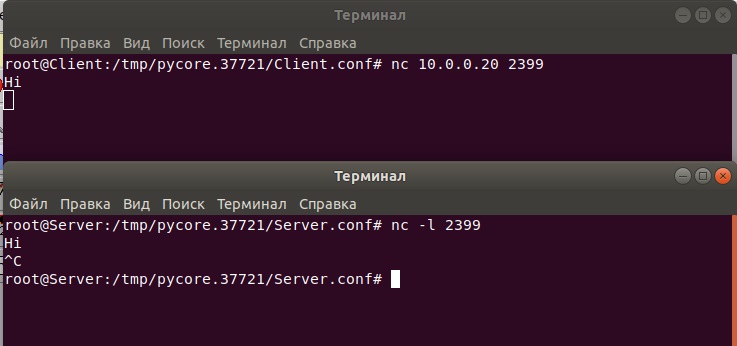
****

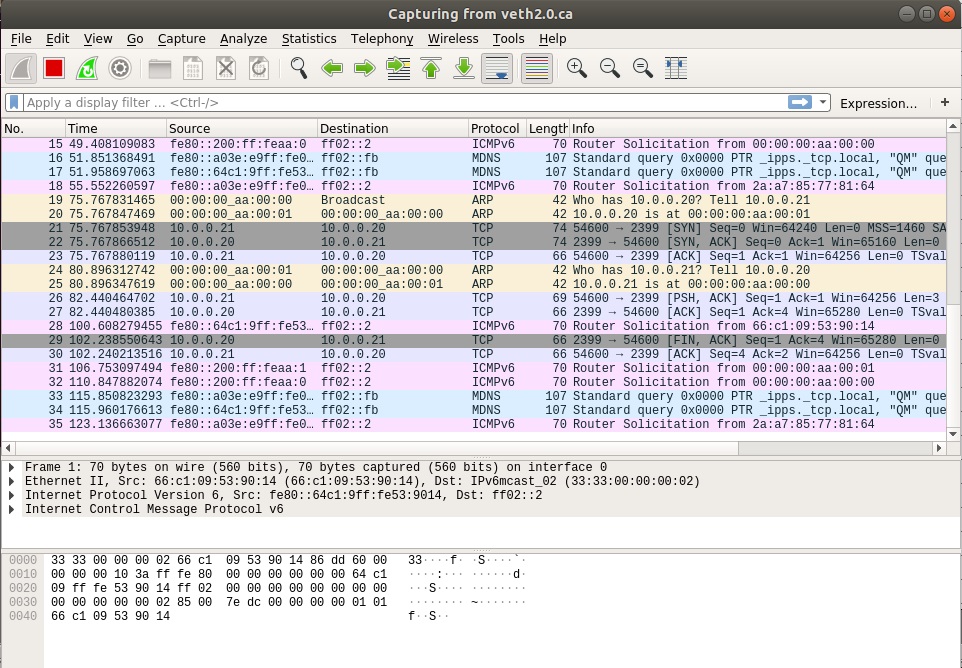
****

* (14) SYN. Начинает соединение и синхронизирует порядковые номера. Первый пакет, отправленный с каждой стороны, должен в обязательном порядке иметь установленным этот флаг. Так как TCP только клиент – он и отправляет этот пакет.
* (15) RST: сброс данного соединения. Отправкой пакета RST одна из сторон сообщает о немедленном разрыве соединения. При этом соединение обрывается, а буфер очищается. Самые распространенные причины отправки пакета с установленным флагом RST — ответ на пакет, полученный для закрытого сокета; пользователь сам прервал соединение (например, закрыв браузер, не дожидаясь ответа); соединение не было нормально закрыто, но находится в неактивном состоянии некоторое время.

ACK. Устанавливается, когда пакет содержит значение номера подтверждения в поле подтверждения. Все пакеты после стартового пакета SYN будут иметь установленный флаг ACK.

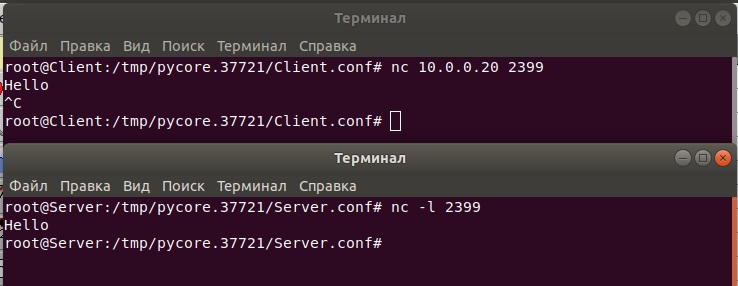
2. Запустить TCP клиент, сервер, передать данные, затем прервать соединение (Ctrl + C) на стороне сервера. Объяснить полученные пакеты в WireShark.

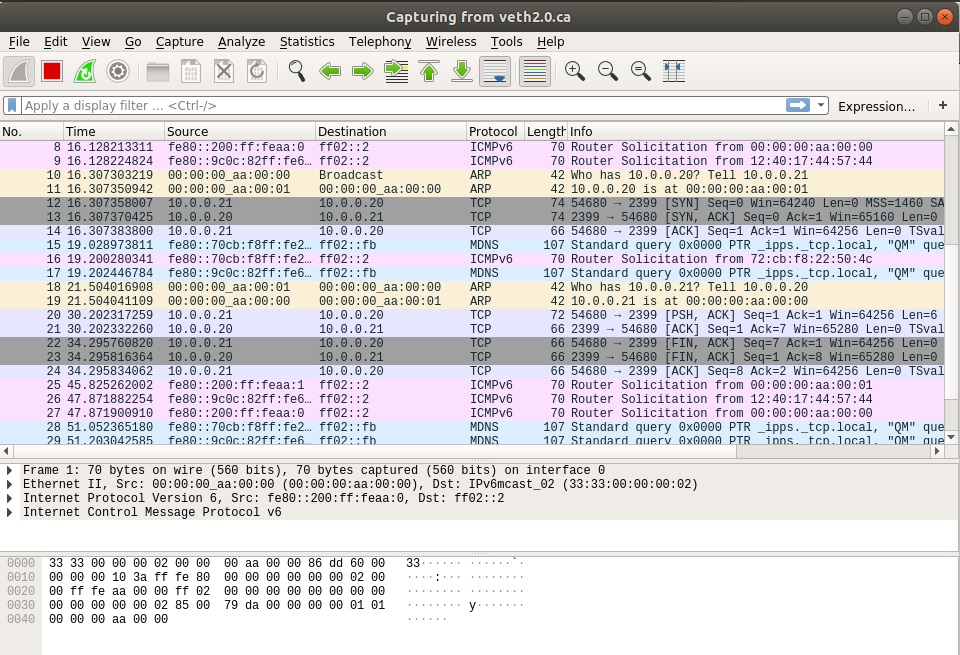




* (21) (22) SYN. Начинает соединение и синхронизирует порядковые номера. Первый пакет, отправленный с каждой стороны, должен в обязательном порядке иметь установленным этот флаг. Так как TCP сервер и клиент – они отправляют пакет с этим флагом друг-другу.
* (23) Пакет от клиента с подтверждением.
* (26) PSH. Делает этот пакет пакетом PUSH (проталкивания). При нормальном потоке передачи данных система получателя не будет подтверждать получение каждого пакета сразу же после его получения. Вместо этого система получателя в течении некоторого времени будет собирать и хранить полученные данные в буфере, пока не передаст их приложению пользователя. Пакет PUSH инструктирует систему получателя немедленно передать все полученные ранее данные из буфера в приложение пользователя и сразу же отправить сообщение с подтверждением. Клиент отправляет серверу сообщение.
* (27) Подтверждение получения сообщения от сервера.
* (29) FIN. Одна из конечных точек отправляет пакет с установленным флагом FIN для другой конечной точки, чтобы сообщить, что все пакеты были отправлены, и соединение пора завершить. Сервер завершает сессию.
* (30) Клиент подтверждает завершение сессии.

3. Запустить TCP клиент, сервер, передать данные, затем прервать соединение (Ctrl + C) на стороне клиента. Объяснить полученные пакеты в WireShark.





* (12) (13) SYN. Начинает соединение и синхронизирует порядковые номера. Первый пакет, отправленный с каждой стороны, должен в обязательном порядке иметь установленным этот флаг. Так как TCP сервер и клиент – они отправляют пакет с этим флагом друг-другу.
* (14) Пакет от клиента с подтверждением.
* (20) PSH. Делает этот пакет пакетом PUSH (проталкивания). При нормальном потоке передачи данных система получателя не будет подтверждать получение каждого пакета сразу же после его получения. Вместо этого система получателя в течении некоторого времени будет собирать и хранить полученные данные в буфере, пока не передаст их приложению пользователя. Пакет PUSH инструктирует систему получателя немедленно передать все полученные ранее данные из буфера в приложение пользователя и сразу же отправить сообщение с подтверждением. Клиент отправляет серверу сообщение.
* (21) Подтверждение получения сообщения от сервера.
* (22) (23) FIN. Одна из конечных точек отправляет пакет с установленным флагом FIN для другой конечной точки, чтобы сообщить, что все пакеты были отправлены, и соединение пора завершить. Клиент отправляет пакет с флагом о завершении сессии, сервер отправляет такой же пакет в ответ.
* (24) Клиент подтверждает завершение сессии.

**Вывод:**

Протокол TCP, за счет подтверждения получения данных, повторной отправки в случае необходимости имеет более высокую надежность.

TCP гарантирует передачу пакетов данных именно в том порядке, в котором они были отправлены. В случае с UDP такой порядок не соблюдается.

UDP имеет более высокую скорость передачи, так как более тяжеловесному TCP-протоколу будет требоваться больше времени для установки соединения, подтверждения получения, повторной отправки данных и т.д.

TCP применяется там, где требуется точная и подтверждаемая передача данных – например, отправка фотографий, или переписка между пользователями. UDP, в свою очередь, нужен для общения в голосовом формате, или при передаче потокового видео, например, с веб-камер или IP-камер.