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

Лабораторная работа №3

ОТЧЕТ

по лабораторной работе

по дисциплине

Сети и телекоммуникации

РУКОВОДИТЕЛЬ:

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

(подпись) (фамилия, и.,о.)

СТУДЕНТ:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Ганин Н. Н.

(подпись) (фамилия, и.,о.)

19-АС

(шифр группы)

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

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

Нижний Новгород 2021

**Задание:**

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.

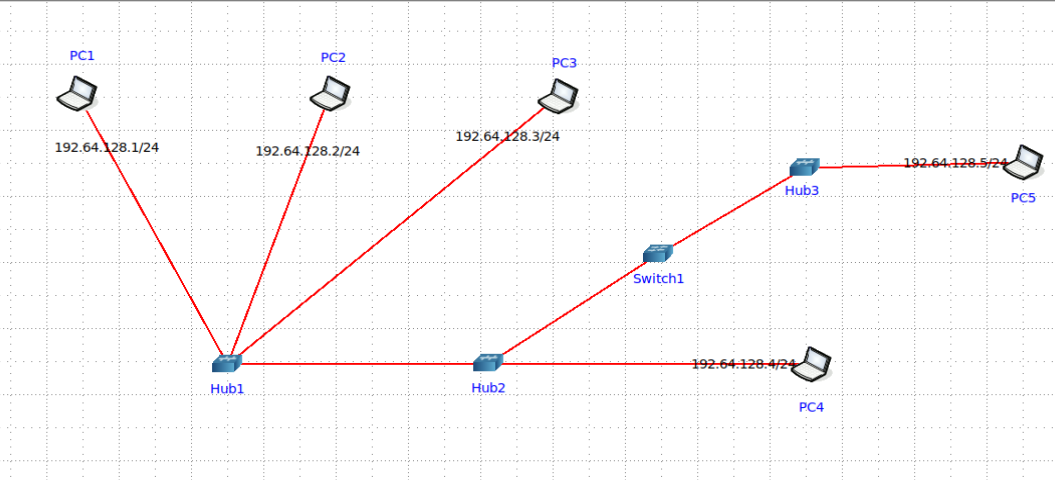
4. Попытаться установить соединение между хостом А и хостом В с помощью программы netcat (А отправляет сообщения В). Убедиться, что запросы от хоста A, направленные хосту В поступают на Сервер.

5. Прекратить захват пакетов.

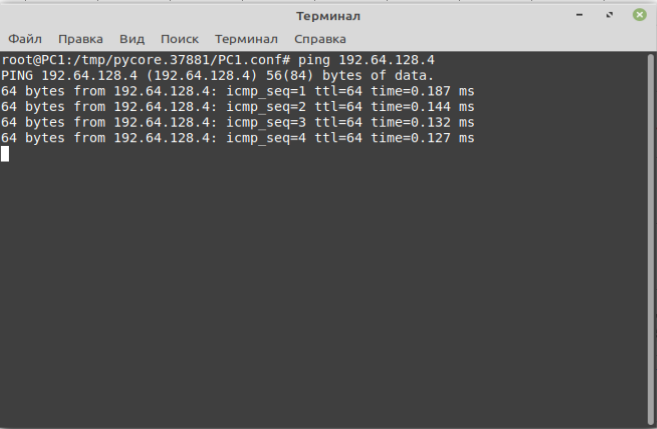
6. Сохранить для отчета отправленный кадр ARP-ответа и несколько перехваченных пакетов, переданных на Сервер, arp таблицу хоста А.

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

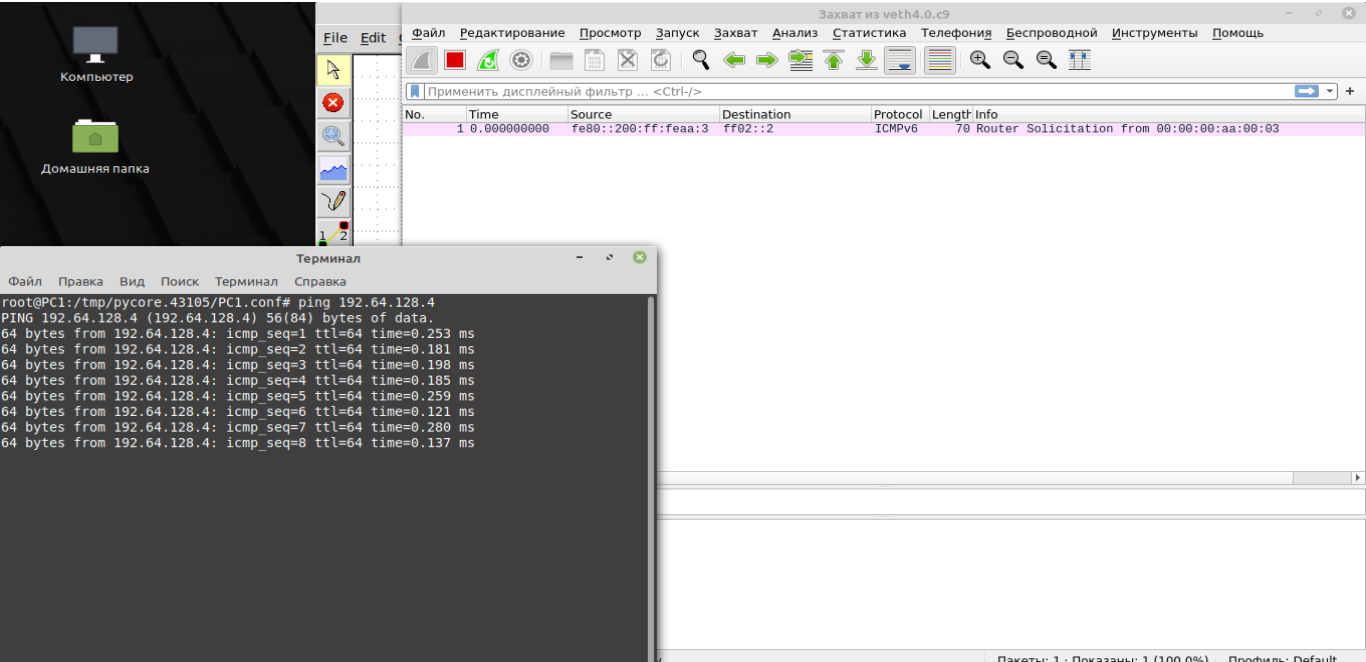
Собрал сеть:



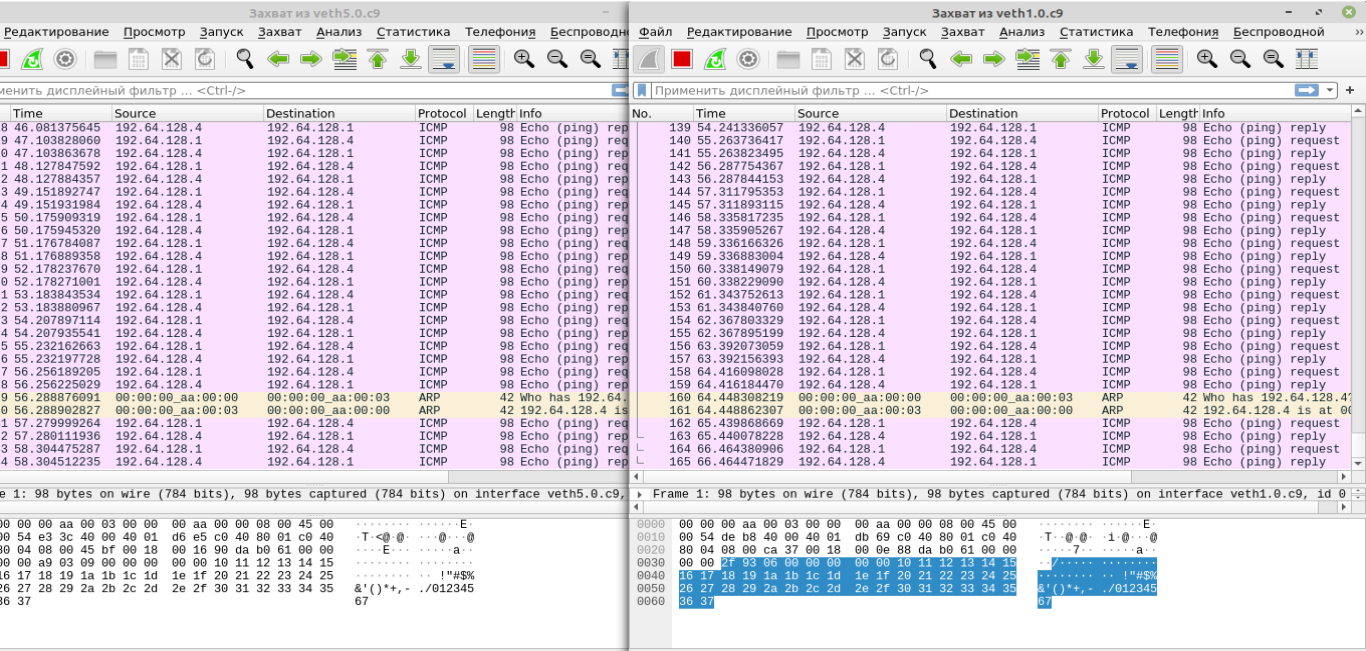
Проверил работу сети:



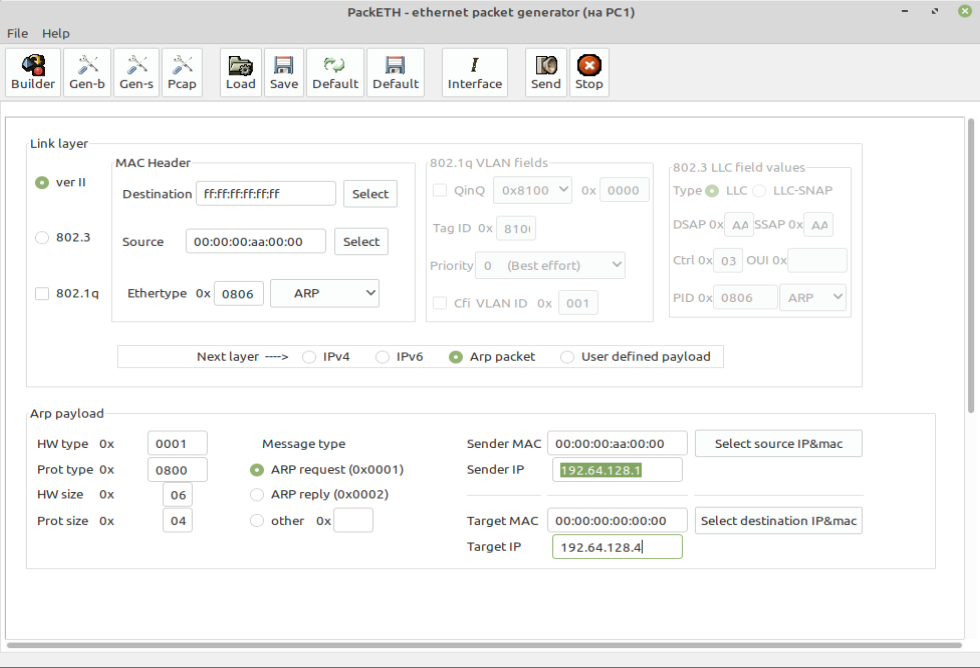
Проверил, идут ли пакеты в PC 5 (нужно для выбора сервера, который будет захватывать пакеты при ARP-спуфинге):

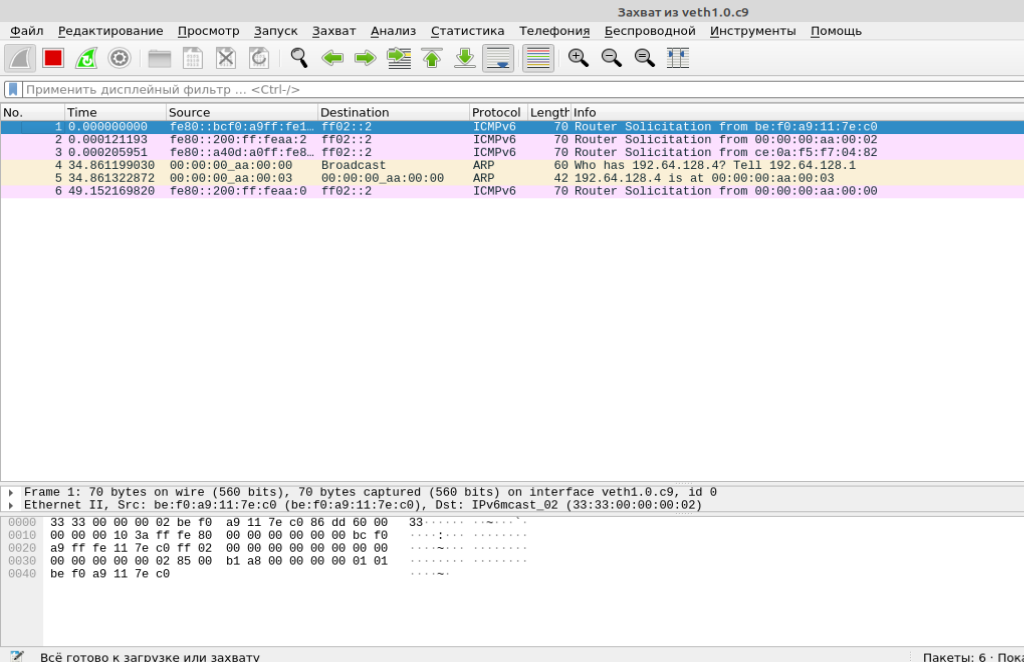
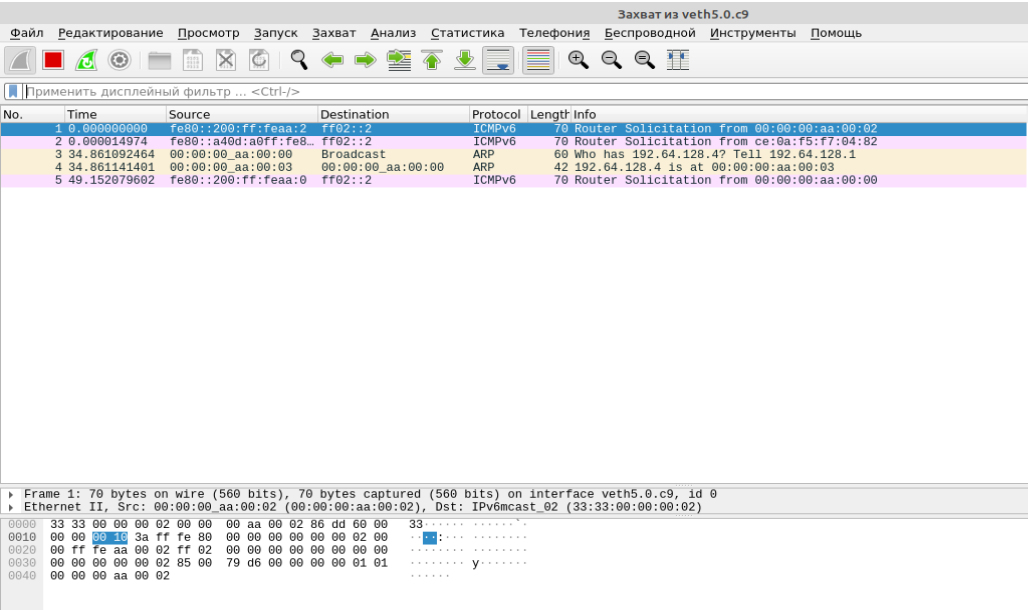


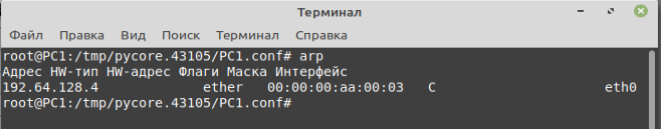
Посмотрел PC1 и PC4 через Wireshark



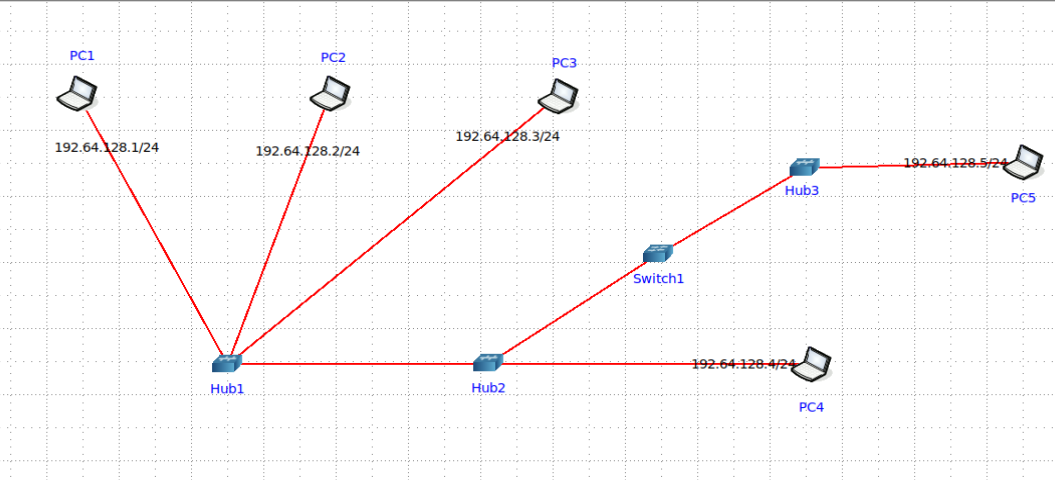
Сформировал кадр ARP-запроса с помощью утилиты PackeTH и отправил его в сеть:



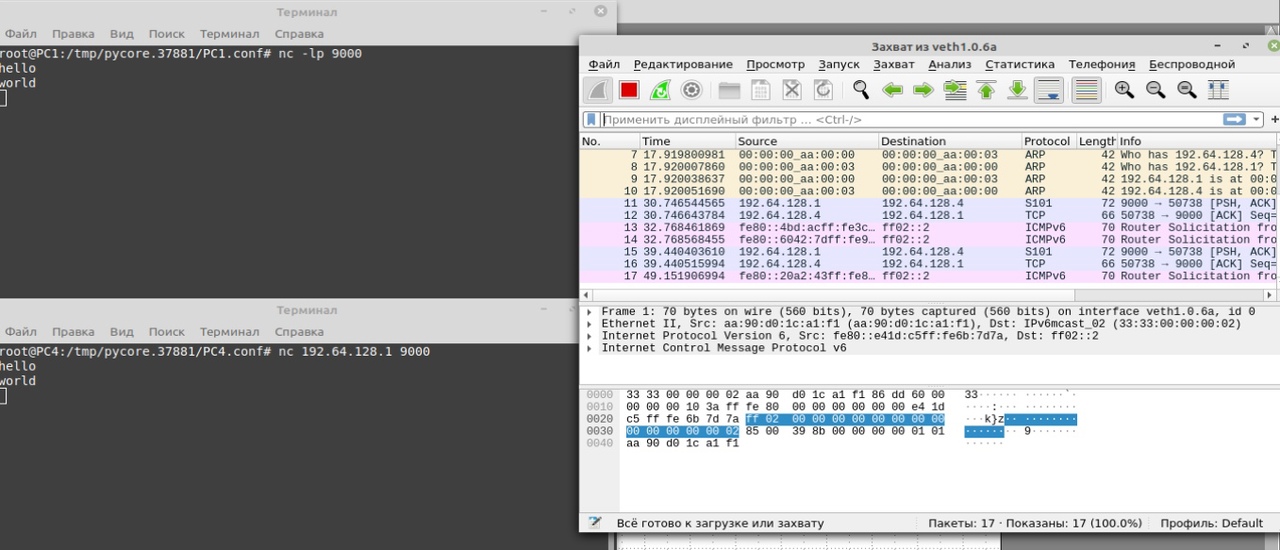


**Часть 2. ARP-спуфинг**

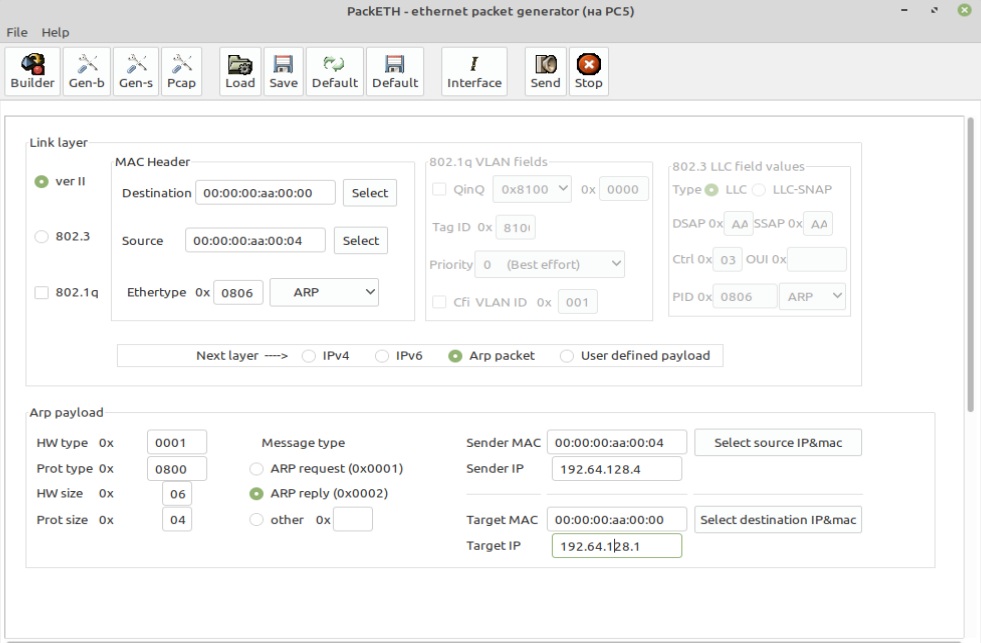


PC1 - A, PC4 – B, PC5 - Сервер

Запустил сервер на PC1, а с PC4 подключился к нему. Отправил сообщения и увидел их в Wireshark:



Подготовил кадр ARP-ответа, направляемый Сервером хосту А с помощью программы PackeTH и отправил сформированный пакет от Сервера хосту А:



Т.е. PC5 представится компьютеру PC1, как компьютер PC4 и будет пропускать через себя пакеты, которые между ними шли.

PC1 пытается отправить пакет PC4, но отправляет его PC5. Так как PC5 не дает подтверждения, то через некоторое время PC1 отправляет широковещательное сообщение, на которое ответит PC4 и пакет дойдет до PC4:

