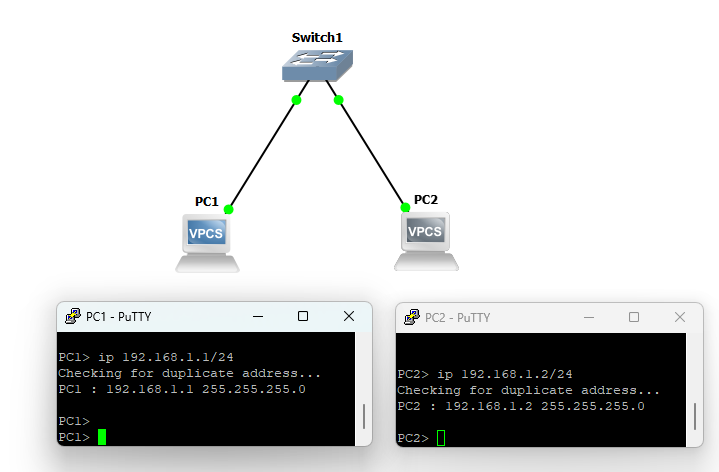
Лабораторная работа №1, выполнил Грищенко Александр Дмитриевич.

1.Создать простейшую сеть, состоящую из 1 коммутатора и 2 компьютеров, назначить им произвольные ip адреса из одной сети

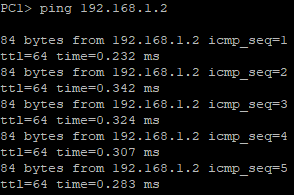
С помощью команды «ip…» задаём адрес обоим ПК, так, чтобы они были в одной сети.

ip 192.168.1.1/24

ip 192.168.1.2/24

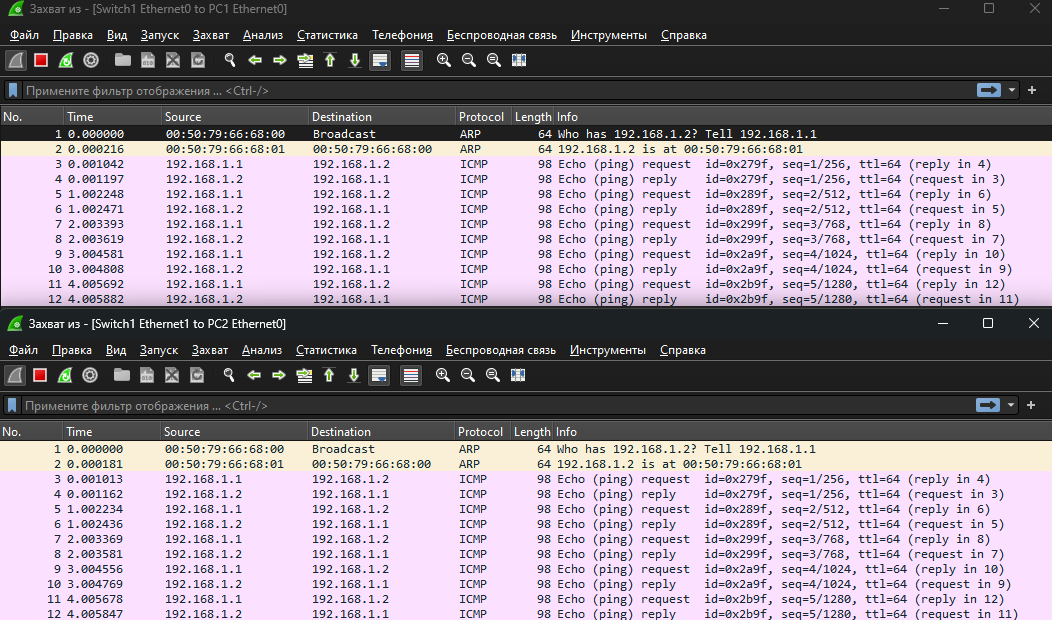


2. Запустить симуляцию, выполнить команду ping с одного из компьютеров, используя ip адрес второго компьютера

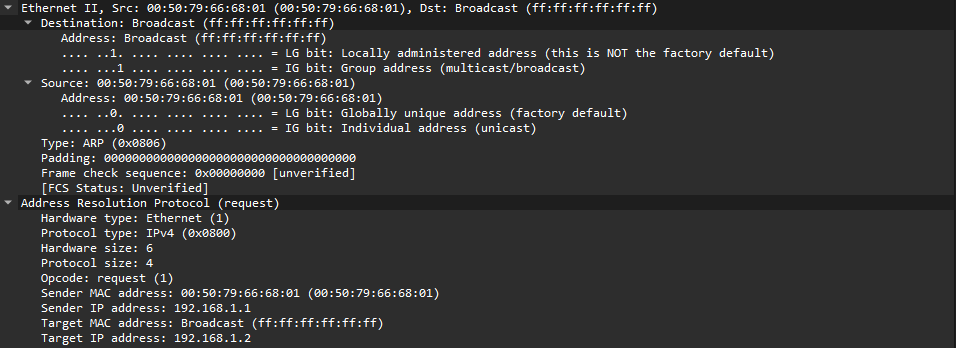


ping 192.168.1.2

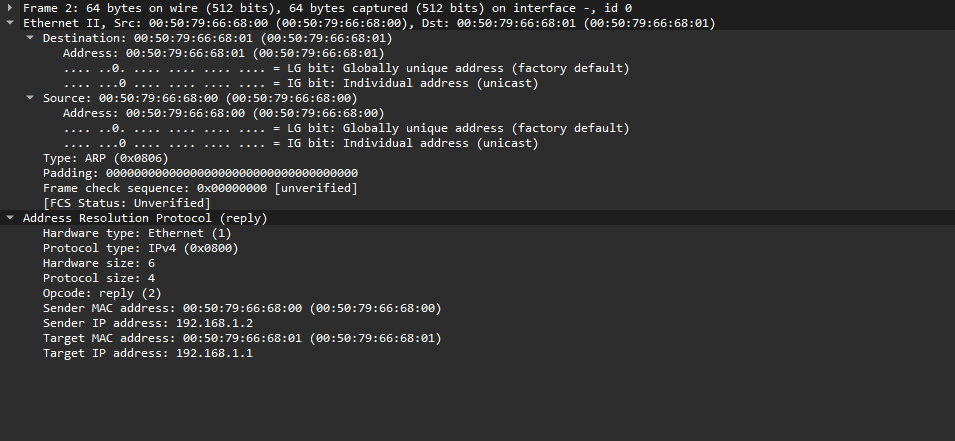
3. Перехватить трафик протокола arp на всех линках(nb!), задокументировать и проанализировать заголовки пакетов в программе Wireshark, для фильтрации трафика, относящегося к указанному протоколу использовать фильтры Wireshark.

При пинге с первого пк на обоих линках одинаковая «картина» т.к. это всё одна сеть, wireshark перехватывает два пакета протокола ARP:  


Первый пакет:



В заголовке ethernet первые два октета – MAC адреса назначения и источника, потом два байта протокола (ethertype) (ARP), в содержании пакета можно увидеть протокол сетевого уровня IPv4, код действия (запрос (request)), MAC и ip адреса отправителя и цели.

Второй пакет:  


То же самое, только теперь известен MAC адрес второго ПК, ну и отправителем является он, а получателем первый ПК, код операции (тип ARP) теперь ответ (reply).

4. Создать простейшую сеть, состоящую из 1 маршрутизатора и 2 компьютеров, назначить им произвольные ip адреса из разных сетей.

Назначаем ip адреса ПК, устанавливаем им адрес шлюза, после чего назначем интерфейсам роутера ip адреса (шлюзы):  
ПК1:

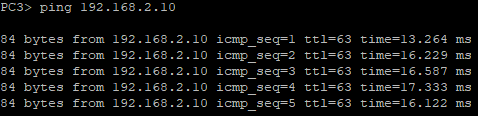
  
ПК2:

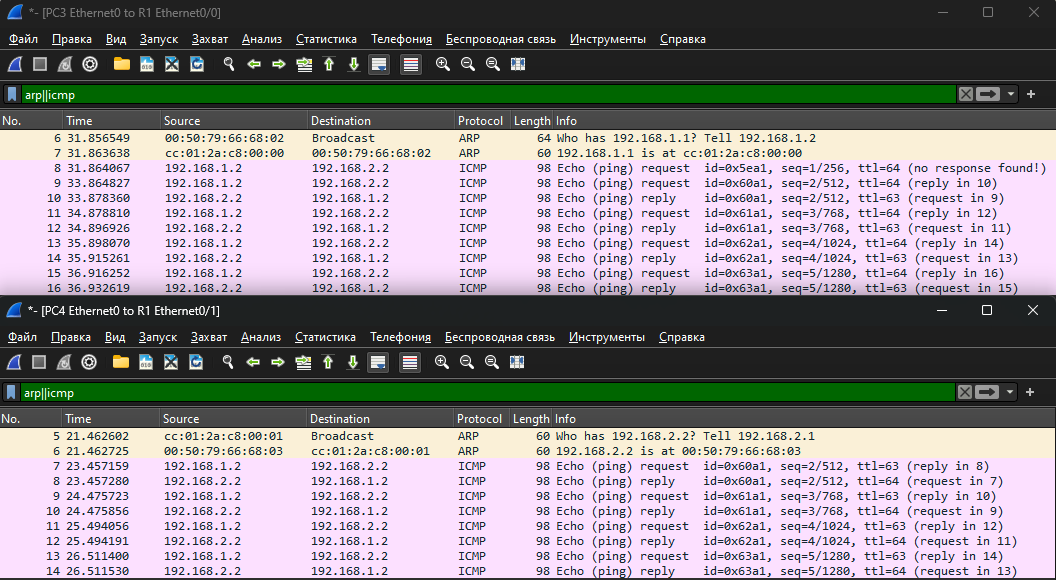


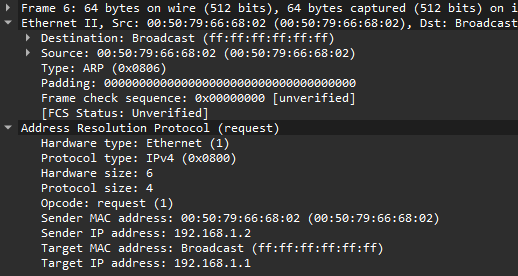
Роутер:

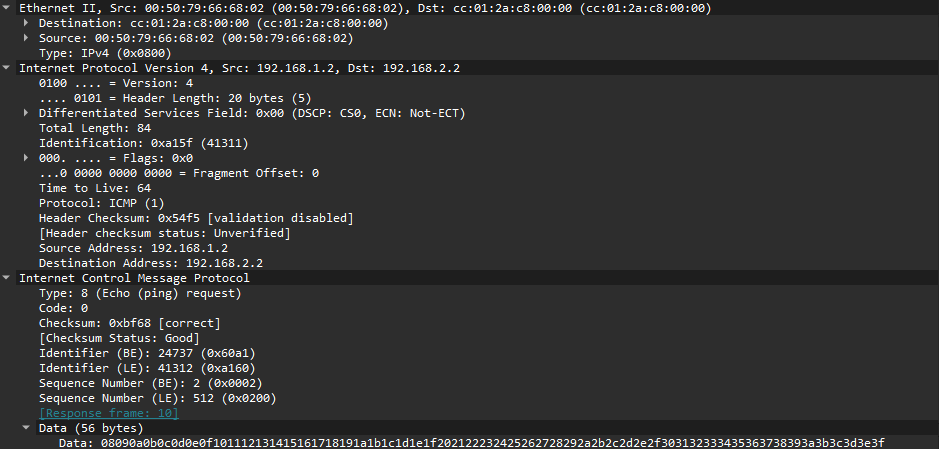
R1#configure terminal  
R1(config)#int e0/0  
R1(config-if)#ip address 192.168.1.1 255.255.255.0  
R1(config-if)#no shut  
R1(config-if)#no shutdown  
R1(config-if)#exit  
R1(config)#int e0/1  
R1(config-if)#ip address 192.168.2.1 255.255.255.0  
R1(config-if)#no shutdown  
R1(config-if)#exit

5. Запустить симуляцию, выполнить команду ping с одного из компьютеров, используя ip адрес второго компьютера (могу ошибаться, но по идее пинг надо делать только в следующем пункте, так как если сделать его сейчас, а захватывать пакеты только в следующем, то арп запросов уже не будет, так как они были здесь).



6. Перехватить трафик протокола arp и icmp на всех линках(nb!), задокументировать и проанализировать заголовки пакетов в программе Wireshark, для фильтрации трафика, относящегося к указанному протоколу использовать фильтры Wireshark.  
Перехваченные пакеты в этой ситуации отличаются, от ситуации с коммутатором и ПК в одной сети:   
Теперь действует такая логика: ПК1 понимает, что ПК2 в другой сети, тогда ему необходимо отправить пакеты на роутер, но ему надо узнать его мак адрес, тогда он делает arp запрос по широковещательному адресу по ip шлюза (первого интефейса роутера) (видно в info, 192.168.1.2 – ПК1, а 192.168.1.1 – первый интерфейс роутера), теперь он будет отправлять пинги по mac адресу роутера, но с ip адресом ПК2.  
Роутеру, в свою очередь, необходимо узнать mac адресс ПК2, он делает arp запрос в подсеть второго пк, получает его mac адрес и всё, теперь пинги идут без проблем от ПК1 через роутер к ПК2. **Только не понял, почему арп запрос от роутера к ПК2 происходит раньше, чем арп запрос от ПК1 к роутеру (если судить по времени захвата), хотя, вероятно, я что-то не так понял.**

Заголовки одного из arp запросов, просто там нет ничего нового, так что рассматривать все 4 в отдельности смысла нет (мне так кажется):  
  
Опять же, в заголовке ethernet: цель, источник, протокол, заполнение нулями, контрольная сумма. В заголовке arp: тип аппаратного обеспечения, тип протокола, mac и ip адреса отправителя и цели.

Заголовок ICMP:  
  
В заголовке ethernet уже ethertype – Ipv4, опять сурс и цель mac адрес. В заголовке IPv4 общая длина, TTL, инкапсулированный протокол (ICMP), сурс и цель IP адрес, в заголовке ICMP: тип – пинг, сообщение, длина сообщения.