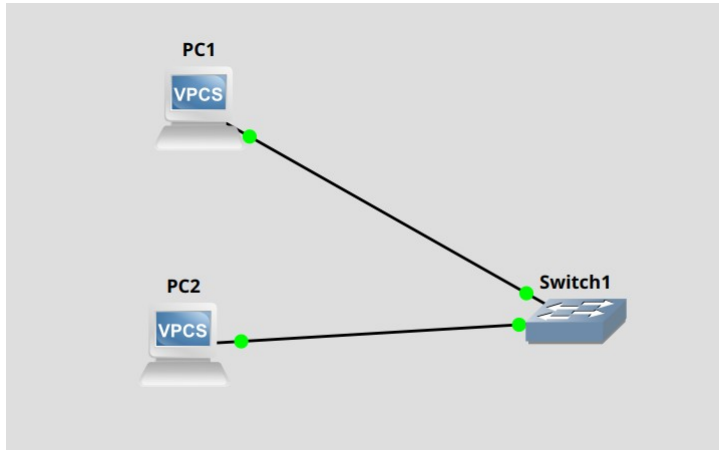


1) Установить и настроить эмулятор GNS3

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



На PC1 ip 192.168.3.2 255.255.255.0

На PC2 ip 192.168.3.3 255.255.255.0

3) Запустить симуляцию, выполнить команду ping с одного из компьютеров, используя ip адрес второго компьютера

ping 192.168.3.3

```
PC1> ping 192.168.3.3

84 bytes from 192.168.3.3 icmp_seq=1 ttl=64 time=0.168 ms
84 bytes from 192.168.3.3 icmp_seq=2 ttl=64 time=0.137 ms
84 bytes from 192.168.3.3 icmp_seq=3 ttl=64 time=0.181 ms
84 bytes from 192.168.3.3 icmp_seq=4 ttl=64 time=0.207 ms
84 bytes from 192.168.3.3 icmp_seq=5 ttl=64 time=0.146 ms

PC1> 
```

4) 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

Пингуем с PC2 (192.168.3.3) PC1 (192.168.3.2)

ARP запрос (кто имеет ip 192.168.3.2).

Dest mac широковещательный.

Target ip: ip адрес для которого нужно узнать mac адрес

No.	Time	Source	Destination	Protocol	Length	Info
15	65.180922	00:50:79:66:68:00	Broadcast	ARP	64	Who has 192.168.3.2? Tell 192.168.3.3
16	65.180309	00:50:79:66:68:01	00:50:79:66:68:00	ARP	64	192.168.3.2 is at 00:50:79:66:68:01


```

> Frame 15: 64 bytes on wire (512 bits), 64 bytes captured on interface eth0
Ethernet II, Src: Realtek-USB-Ethernet (00:50:79:66:68:00), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
    Destination: Broadcast (ff:ff:ff:ff:ff:ff)
    Source: 00:50:79:66:68:00 (00:50:79:66:68:00)
        Type: ARP (0x0806)
        Padding: 000000000000000000000000000000000000000000000000
        Frame check sequence: 0x00000000 [unverified]
        [FCS Status: Unverified]
> Address Resolution Protocol (request)
    Hardware type: Ethernet (1)
    Protocol type: IPv4 (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: request (1)
    Sender MAC address: 00:50:79:66:68:00 (00:50:79:66:68:00)
    Sender IP address: 192.168.3.3
    Target MAC address: Broadcast (ff:ff:ff:ff:ff:ff)
    Target IP address: 192.168.3.2
  
```

PC1 отвечает:

[illegible]

dest mac: mac адрес PC2 который был в ARP запросе

Opcode : 2 — ответ

Target mac и target ip — адреса PC2

[illegible]

15	65.179985	00:50:79:66:68:00	Broadcast	ARP	64 Who has 192.168.3.2? Tell 192.168.3.3
16	65.180292	00:50:79:66:68:01	00:50:79:66:68:00	ARP	64 192.168.3.2 is at 00:50:79:66:68:01

<ul style="list-style-type: none"> ▶ Frame 16: 64 bytes on wire (512 bits), 64 bytes captured on interface 0 ▼ Ethernet II, Src: 00:50:79:66:68:01 (00:50:79:66:68:01), Dst: 00:50:79:66:68:00 (00:50:79:66:68:00) ▶ Destination: 00:50:79:66:68:00 (00:50:79:66:68:00) ▶ Source: 00:50:79:66:68:01 (00:50:79:66:68:01) Type: ARP (0x0806) Padding: 00000000000000000000000000000000 Frame check sequence: 0x00000000 [unverified] [FCS Status: Unverified] ▼ Address Resolution Protocol (reply) Hardware type: Ethernet (1) Protocol type: IPv4 (0x0800) Hardware size: 6 Protocol size: 4 Opcode: reply (2) Sender MAC address: 00:50:79:66:68:01 (00:50:79:66:68:01) Sender IP address: 192.168.3.2 Target MAC address: 00:50:79:66:68:00 (00:50:79:66:68:00) Target IP address: 192.168.3.3 	<pre> 0000 00 50 79 66 68 00 50 79 66 68 01 08 06 00 01 0010 08 00 06 04 00 02 00 50 79 66 68 01 c0 a8 03 02 0020 00 50 79 66 68 00 c0 a8 03 03 00 00 00 00 00 0030 00 00 00 00 00 00 00 00 00 00 00 00 00 00 Pyfh..P yfh....P yfh.... Pyfh..... </pre>
--	--

Еще хосты при включении рассылают *Gratuitous ARP*:

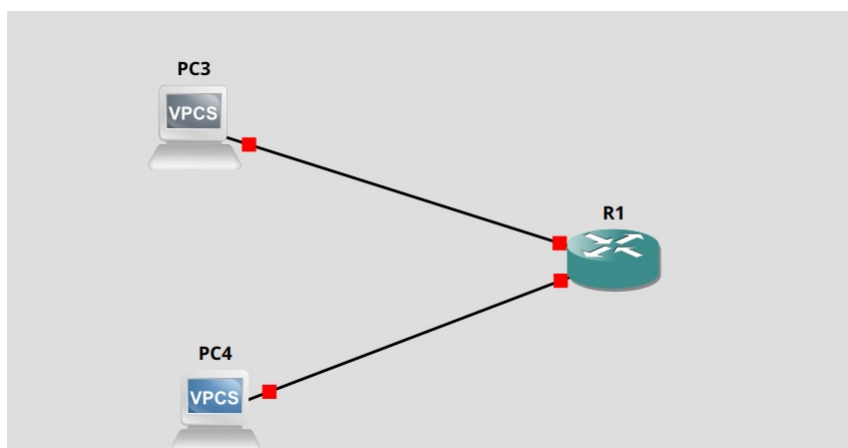
11	16.529029	00:50:79:66:68:01	Broadcast	ARP	64 Gratuitous ARP for 192.168.3.2 (Request)
12	16.574941	00:50:79:66:68:00	Broadcast	ARP	64 Gratuitous ARP for 192.168.3.3 (Request)
13	17.529891	00:50:79:66:68:01	Broadcast	ARP	64 Gratuitous ARP for 192.168.3.2 (Request)
14	17.575644	00:50:79:66:68:00	Broadcast	ARP	64 Gratuitous ARP for 192.168.3.3 (Request)
15	65.180022	00:50:79:66:68:00	Broadcast	ARP	64 Who has 192.168.3.2? Tell 192.168.3.3
16	65.180309	00:50:79:66:68:01	00:50:79:66:68:00	ARP	64 192.168.3.2 is at 00:50:79:66:68:01


```

> Frame 12: 64 bytes on wire (512 bits), 64 bytes captured on interface 0
> Ethernet II, Src: 00:50:79:66:68:00 (00:50:79:66:68:00), Dst: ff:ff:ff:ff:ff:ff
  > Destination: Broadcast (ff:ff:ff:ff:ff:ff)
  > Source: 00:50:79:66:68:00 (00:50:79:66:68:00)
    Type: ARP (0x0806)
    Padding: 00000000000000000000000000000000
    Frame check sequence: 0x00000000 [unverified]
    [FCS Status: Unverified]
  > Address Resolution Protocol (request/gratuitous ARP)
    Hardware type: Ethernet (1)
    Protocol type: IPv4 (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: request (1)
    [Is gratuitous: True]
    Sender MAC address: 00:50:79:66:68:00 (00:50:79:66:68:00)
    Sender IP address: 192.168.3.3
    Target MAC address: Broadcast (ff:ff:ff:ff:ff:ff)
    Target IP address: 192.168.3.3
  
```

В них ip отправителя и получателя совпадают.

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



PC3 : ip 192.168.3.2 192.168.3.1 24

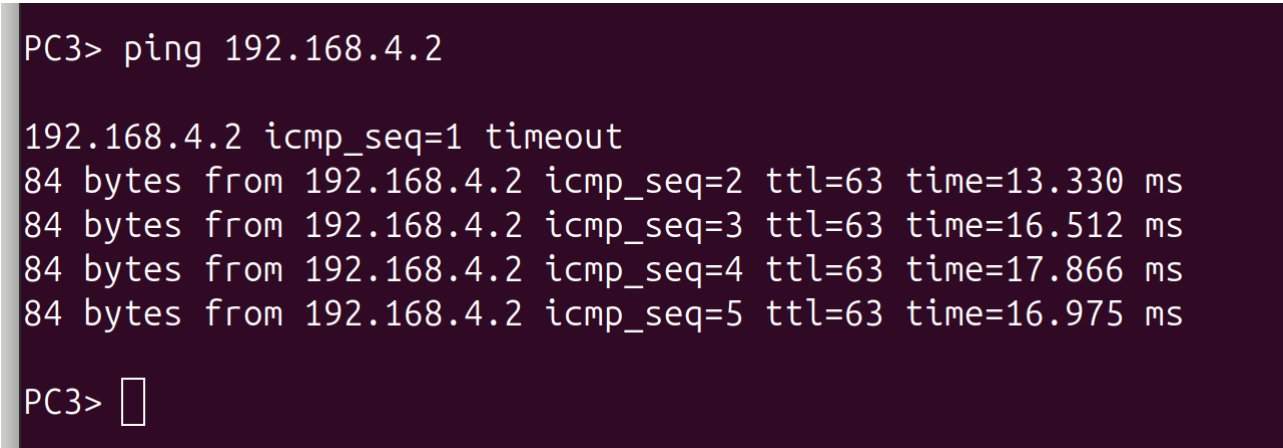
PC4: ip 192.168.4.2 192.168.4.1 24

На маршрутизаторе задаем адреса для интерфейсов и включаем их.

```
conf t
interface FastEthernet0/0
ip address 192.168.3.1 255.255.255.0
exit
interface FastEthernet1/0
ip address 192.168.4.1 255.255.255.0
end
copy running-config startup-config
```

6) Запустить симуляцию, выполнить команду ping с одного из компьютеров, используя ip адрес второго компьютера

```
ping 192.168.4.2
```



```
PC3> ping 192.168.4.2

192.168.4.2 icmp_seq=1 timeout
84 bytes from 192.168.4.2 icmp_seq=2 ttl=63 time=13.330 ms
84 bytes from 192.168.4.2 icmp_seq=3 ttl=63 time=16.512 ms
84 bytes from 192.168.4.2 icmp_seq=4 ttl=63 time=17.866 ms
84 bytes from 192.168.4.2 icmp_seq=5 ttl=63 time=16.975 ms

PC3> 
```

7) 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

2. Маршрутизатор генерирует agr ответ:

3. Далее РСЗ отправляет істр сообщение:

```

Frame 25: 98 bytes on wire (784 bits), 98 bytes captured (784 bi
Ethernet II, Src: 00:50:79:66:68:03 (00:50:79:66:68:03), Dst: cc
  Destination: cc:01:76:0b:00:00 (cc:01:76:0b:00:00)
    Source: 00:50:79:66:68:03 (00:50:79:66:68:03)
      Type: IPv4 (0x0800)
Internet Protocol Version 4, Src: 192.168.3.2, Dst: 192.168.4.2
  Internet Control Message Protocol
    Type: 8 (Echo (ping) request)
    Code: 0
    Checksum: 0xf9a1 [correct]
    [Checksum Status: Good]
    Identifier (BE): 9832 (0x2668)
    Identifier (LE): 26682 (0x6826)
    Sequence Number (BE): 2 (0x0002)
    Sequence Number (LE): 512 (0x0200)
    [Response frame: 26]
  Data (56 bytes)

```

Он должен выяснить тас адрес этого узла (192.168.4.2).

5. На другом линке: маршрутизатор генерирует arg запрос на широковещательный mac адрес.

```

Frame 22: 60 bytes on wire (480 bits) 60 bytes captured (480 bi 0000 ff ff ff ff ff ff ff cc 01 76 0b 00 10 08 06 00 01 ..... v .....
Ethernet II, Src: cc:01:76:0b:00:10 (cc:01:76:0b:00:10), Dst: Br 0010 00 00 06 04 00 01 cc 01 76 0b 00 10 c0 a8 04 01 ..... v .....
  Destination: Broadcast (ff:ff:ff:ff:ff:ff) 0020 00 00 00 00 00 00 c0 a8 04 02 00 00 00 00 00 00 ..... v .....
  Source: cc:01:76:0b:00:10 (cc:01:76:0b:00:10) 0030 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ..... v .....
  Type: ARP (0x0806)
  Padding: 00000000000000000000000000000000000000000000
Address Resolution Protocol (request)
  Hardware type: Ethernet (1)
  Protocol type: IPv4 (0x0800)
  Hardware size: 6
  Protocol size: 4
  Opcode: request (1)
  Sender MAC address: cc:01:76:0b:00:10 (cc:01:76:0b:00:10)
  Sender IP address: 192.168.4.1
  Target MAC address: 00:00:00:00:00:00 (00:00:00:00:00:00)
  Target IP address: 192.168.4.2

```

6. И получает `arg` ответ от хоста:

```

Frame 23: 60 bytes on wire (480 bits), 60 bytes captured (480 bi
Ethernet II, Src: 00:50:79:66:68:00 (00:50:79:66:68:00), Dst: cc
  Destination: cc:01:76:0b:00:10 (cc:01:76:0b:00:10)
  Source: 00:50:79:66:68:00 (00:50:79:66:68:00)
    Type: ARP (0x0806)
      Padding: 00000000000000000000000000000000000000000000
Address Resolution Protocol (reply)
  Hardware type: Ethernet (1)
  Protocol type: IPv4 (0x0800)
  Hardware size: 6
  Protocol size: 4
  Opcode: reply (2)
  Sender MAC address: 00:50:79:66:68:00 (00:50:79:66:68:00)
  Sender IP address: 192.168.4.2
  Target MAC address: cc:01:76:0b:00:10 (cc:01:76:0b:00:10)
  Target IP address: 192.168.4.1

```


7. После чего маршрутизатор посылает icmp запрос к нужному хосту (PC4):
ttl пакета уменьшился на 1

24	136.850944	192.168.3.2	192.168.4.2	ICMP	98 Echo (ping) request	id=0x2668, seq=2/512, ttl=63 (reply in 25)
25	136.850993	192.168.4.2	192.168.3.2	ICMP	98 Echo (ping) reply	id=0x2668, seq=2/512, ttl=64 (request in 24)
26	137.868446	192.168.3.2	192.168.4.2	ICMP	98 Echo (ping) request	id=0x2768, seq=3/768, ttl=63 (reply in 27)
27	137.868527	192.168.4.2	192.168.3.2	ICMP	98 Echo (ping) reply	id=0x2768, seq=3/768, ttl=64 (request in 26)
28	138.886923	192.168.3.2	192.168.4.2	ICMP	98 Echo (ping) request	id=0x2868, seq=4/1024, ttl=63 (reply in 29)
29	138.887194	192.168.4.2	192.168.3.2	ICMP	98 Echo (ping) reply	id=0x2868, seq=4/1024, ttl=64 (request in 28)

Frame 24: 98 bytes on wire (784 bits), 98 bytes captured (784 b) on interface 0, Ethernet II, Src: cc:01:76:0b:00:10 (cc:01:76:0b:00:10), Dst: cc:01:76:0b:00:10 (cc:01:76:0b:00:10), Protocol: ICMP, Length: 8, Type: Echo (ping) request, TTL: 63, Window: 0, Checksum: 0x9a1, [Checksum Status: Good], Identifier (BE): 9832 (0x2668), Sequence Number (BE): 2 (0x0002), Sequence Number (LE): 512 (0x0200), [Request frame: 24], [Response time: 0,049 ms], Data (56 bytes)

0000 00 50 79 66 68 00 cc 01 76 0b 00 10 08 00 45 00 Pyfh... v...E
0010 00 54 68 25 00 00 3f 01 8b 2f c0 a8 03 02 c0 a8 Th%..?.. /.....
0020 04 02 08 00 f9 a1 26 68 00 02 08 09 0a 0b 0c 0d&h.....
0030 0e 0f 10 11 12 13 14 15 16 17 18 19 1a 1b 1c 1d .. !"#%&'()*+,-.
0040 1e 1f 20 21 22 23 24 25 26 27 28 29 2a 2b 2c 2d .. !"#%&'()*+,-.
0050 2e 2f 30 31 32 33 34 35 36 37 38 39 3a 3b 3c 3d ./012345 6789;<=
0060 3e 3f >?

8. PC4 генерирует icmp ответ и отправляет обратно. Агр запросов уже не происходит, потому что все заполнили свои таблицы.

25	136.850993	192.168.4.2	192.168.3.2	ICMP	98 Echo (ping) reply	id=0x2668, seq=2/512, ttl=64 (request in 24)
26	137.868446	192.168.3.2	192.168.4.2	ICMP	98 Echo (ping) request	id=0x2768, seq=3/768, ttl=63 (reply in 27)
27	137.868527	192.168.4.2	192.168.3.2	ICMP	98 Echo (ping) reply	id=0x2768, seq=3/768, ttl=64 (request in 26)
28	138.886923	192.168.3.2	192.168.4.2	ICMP	98 Echo (ping) request	id=0x2868, seq=4/1024, ttl=63 (reply in 29)
29	138.887194	192.168.4.2	192.168.3.2	ICMP	98 Echo (ping) reply	id=0x2868, seq=4/1024, ttl=64 (request in 28)

Frame 25: 98 bytes on wire (784 bits), 98 bytes captured (784 b) on interface 0, Ethernet II, Src: 00:50:79:66:68:00 (00:50:79:66:68:00), Dst: cc:01:76:0b:00:10 (cc:01:76:0b:00:10), Protocol: ICMP, Length: 8, Type: Echo (ping) reply, TTL: 64, Window: 0, Checksum: 0x01a2, [Checksum Status: Good], Identifier (BE): 9832 (0x2668), Identifier (LE): 26662 (0x6826), Sequence Number (BE): 2 (0x0002), Sequence Number (LE): 512 (0x0200), [Request frame: 24], [Response time: 0,049 ms], Data (56 bytes)

0000 cc 01 76 0b 00 10 00 50 79 66 68 00 08 00 45 00 ..v...P yfh...E
0010 00 54 68 25 00 00 40 01 8a 2f c0 a8 04 02 c0 a8 Th%..@. 2f.....
0020 03 02 00 00 01 a2 26 68 00 02 08 09 0a 0b 0c 0d&h.....
0030 0e 0f 10 11 12 13 14 15 16 17 18 19 1a 1b 1c 1d .. !"#%&'()*+,-.
0040 1e 1f 20 21 22 23 24 25 26 27 28 29 2a 2b 2c 2d .. !"#%&'()*+,-.
0050 2e 2f 30 31 32 33 34 35 36 37 38 39 3a 3b 3c 3d ./012345 6789;<=
0060 3e 3f >?

26	136.861034	192.168.4.2	192.168.3.2	ICMP	98 Echo (ping) reply	id=0x2668, seq=2/512, ttl=63 (request in 25)
27	137.862066	192.168.3.2	192.168.4.2	ICMP	98 Echo (ping) request	id=0x2768, seq=3/768, ttl=64 (reply in 28)
28	137.878526	192.168.4.2	192.168.3.2	ICMP	98 Echo (ping) reply	id=0x2768, seq=3/768, ttl=63 (request in 27)
29	138.879185	192.168.3.2	192.168.4.2	ICMP	98 Echo (ping) request	id=0x2868, seq=4/1024, ttl=64 (reply in 30)
30	138.896989	192.168.4.2	192.168.3.2	ICMP	98 Echo (ping) reply	id=0x2868, seq=4/1024, ttl=63 (request in 29)
31	138.937550	cc:01:76:0b:00:00	cc:01:76:0b:00:00	LOOP	60 Reply	

Frame 26: 98 bytes on wire (784 bits), 98 bytes captured (784 b) on interface 0, Ethernet II, Src: cc:01:76:0b:00:00 (cc:01:76:0b:00:00), Dst: 00:50:79:66:68:03 (00:50:79:66:68:03), Protocol: ICMP, Length: 8, Type: Echo (ping) reply, TTL: 63, Window: 0, Checksum: 0x01a2, [Checksum Status: Good], Identifier (BE): 9832 (0x2668), Identifier (LE): 26662 (0x6826), Sequence Number (BE): 2 (0x0002), Sequence Number (LE): 512 (0x0200), [Request frame: 25], [Response time: 13,278 ms], Data (56 bytes)

0000 00 50 79 66 68 03 cc 01 76 0b 00 00 08 00 45 00 Pyfh... v...E
0010 00 54 68 25 00 00 3f 01 8b 2f c0 a8 04 02 c0 a8 Th%..?.. /.....
0020 03 02 00 00 01 a2 26 68 00 02 08 09 0a 0b 0c 0d&h.....
0030 0e 0f 10 11 12 13 14 15 16 17 18 19 1a 1b 1c 1d .. !"#%&'()*+,-.
0040 1e 1f 20 21 22 23 24 25 26 27 28 29 2a 2b 2c 2d .. !"#%&'()*+,-.
0050 2e 2f 30 31 32 33 34 35 36 37 38 39 3a 3b 3c 3d ./012345 6789;<=
0060 3e 3f >?