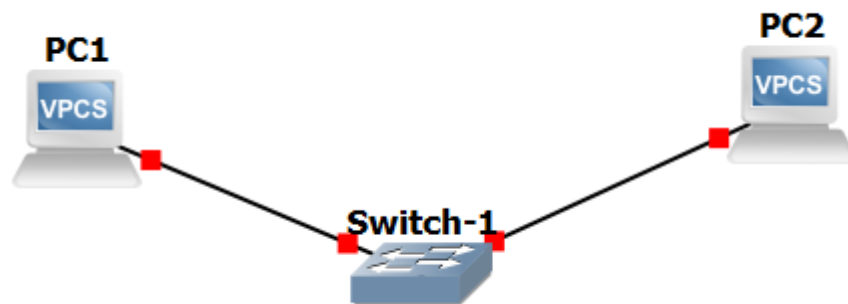


## Практика №1

### Тема: Освоение инструментария для выполнения работ, построение простой сети

- 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



Назначаем ip-адреса из одной сети:

PC1: ip 192.168.1.1/24

PC2: ip 192.168.1.2/24

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

Пинг PC1 -> PC2

```
PC1> ping 192.168.1.2

84 bytes from 192.168.1.2 icmp_seq=1 ttl=64 time=4.832 ms
84 bytes from 192.168.1.2 icmp_seq=2 ttl=64 time=0.649 ms
84 bytes from 192.168.1.2 icmp_seq=3 ttl=64 time=5.953 ms
84 bytes from 192.168.1.2 icmp_seq=4 ttl=64 time=0.656 ms
84 bytes from 192.168.1.2 icmp_seq=5 ttl=64 time=8.242 ms
```

- 4) Перехватить трафик протокола arp на всех линках, задокументировать и проанализировать заголовки пакетов в программе Wireshark, для фильтрации трафика, относящегося к указанному протоколу использовать фильтры Wireshark

Захват из Standard input [PC2 Ethernet0 to Switch-1 Ethernet1]

Файл Правка Вид Запуск Захват Анализ Статистика Телефония Беспроводная связь Инструменты Справка

arp

No.	Time	Source	Destination	Protocol	Length	Info
4	4.613126	Private_66:68:01	Broadcast	ARP	64	Who has 192.168.1.2? Tell 192.168.1.1
5	4.613214	Private_66:68:00	Private_66:68:01	ARP	64	192.168.1.2 is at 00:50:79:66:68:00

- 1) По широковещательному адресу PC1 спрашивает у всех устройств в сети кто имеет IP-адрес 192.168.1.2
- 2) Далее PC1 получает ответ, в котором содержится MAC-адрес устройства с IP-адресом 192.168.1.2

### Первый пакет:

```
> Frame 4: 64 bytes on wire (512 bits), 64 bytes captured (512 bits) on interface -, id 0
✓ Ethernet II, Src: Private_66:68:01 (00:50:79:66:68:01), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
  > Destination: Broadcast (ff:ff:ff:ff:ff:ff)
  > Source: Private_66:68:01 (00:50:79:66:68:01)
    Type: ARP (0x0806)
    [Stream index: 1]
    Padding: 00000000000000000000000000000000
    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: Private_66:68:01 (00:50:79:66:68:01)
    Sender IP address: 192.168.1.1
    Target MAC address: Broadcast (ff:ff:ff:ff:ff:ff)
    Target IP address: 192.168.1.2
```

Source: устройство PC1 (192.168.1.1) (00:50:79:66:68:01)

Destination: Широковещательный адрес (все устройства) (ff:ff:ff:ff:ff:ff)

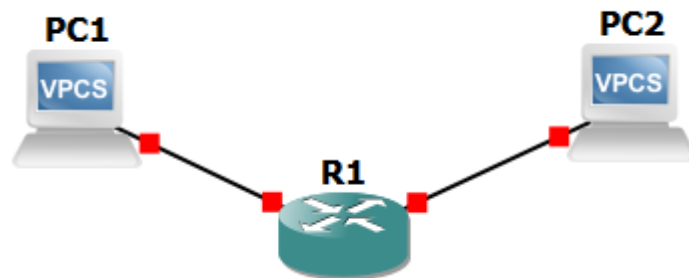
### Второй пакет:

```
> Frame 5: 64 bytes on wire (512 bits), 64 bytes captured (512 bits) on interface -, id 0
✓ Ethernet II, Src: Private_66:68:00 (00:50:79:66:68:00), Dst: Private_66:68:01 (00:50:79:66:68:01)
  > Destination: Private_66:68:01 (00:50:79:66:68:01)
  > Source: Private_66:68:00 (00:50:79:66:68:00)
    Type: ARP (0x0806)
    [Stream index: 2]
    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: Private_66:68:00 (00:50:79:66:68:00)
    Sender IP address: 192.168.1.2
    Target MAC address: Private_66:68:01 (00:50:79:66:68:01)
    Target IP address: 192.168.1.1
```

Source: устройство PC2 (192.168.1.2) (00:50:79:66:68:00)  
Destination: PC1 (192.168.1.1) (00:50:79:66:68:01)

Ответ от PC2 содержит его MAC-адрес

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



Выполняем команды:

```
configure terminal
```

```
interface FastEthernet 0/0
```

```
ip address 192.168.1.1 255.255.255.0
```

```
no shutdown
```

```
exit
```

```
int FastEthernet 1/0
```

```
ip address 192.168.2.1 255.255.255.0
```

```
no shutdown
```

```
exit
```

Таким образом, интерфейсам к которым подключены PC1 и PC2 были назначены адреса.

Назначаем ip-адрес для PC1 в первой подсети:

```
ip 192.168.1.11 192.168.1.1
```

Назначаем ip-адрес для PC1 во второй подсети:

```
ip 192.168.2.22 192.168.2.1
```

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

Выполняем команду ping 192.168.2.22 на устройстве PC1 (Пинг PC1 -> PC2)

Выполняем команду ping 192.168.1.11 на устройстве PC2 (Пинг PC2 -> PC1)

7) Перехватить трафик протокола arp и icmp на всех линках, задокументировать и проанализировать заголовки пакетов в программе Wireshark, для фильтрации трафика, относящегося к указанному протоколу использовать фильтры Wireshark

Пинг PC1 -> PC2:

Захват из Standard input [PC1 Ethernet0 to R1 FastEthernet0/0]

No.	Time	Source	Destination	Protocol	Length	Info
2	9.168151	Private_66:68:01	Broadcast	ARP	64	Who has 192.168.1.1? Tell 192.168.1.11
3	9.177453	cc:01:5c:08:00:00	Private_66:68:01	ARP	60	192.168.1.1 is at cc:01:5c:08:00:00
4	9.177692	192.168.1.11	192.168.2.22	ICMP	98	Echo (ping) request id=0x8b2f, seq=1/256, ttl=64 (no response found!)
5	11.178826	192.168.1.11	192.168.2.22	ICMP	98	Echo (ping) request id=0x8d2f, seq=2/512, ttl=64 (reply in 6)
6	11.189699	192.168.2.22	192.168.1.11	ICMP	98	Echo (ping) reply id=0x8d2f, seq=2/512, ttl=63 (request in 5)
7	12.190393	192.168.1.11	192.168.2.22	ICMP	98	Echo (ping) request id=0x8e2f, seq=3/768, ttl=64 (reply in 8)
8	12.205406	192.168.2.22	192.168.1.11	ICMP	98	Echo (ping) reply id=0x8e2f, seq=3/768, ttl=63 (request in 7)
9	13.206326	192.168.1.11	192.168.2.22	ICMP	98	Echo (ping) request id=0x8f2f, seq=4/1024, ttl=64 (reply in 10)
10	13.221252	192.168.2.22	192.168.1.11	ICMP	98	Echo (ping) reply id=0x8f2f, seq=4/1024, ttl=63 (request in 9)
11	14.222331	192.168.1.11	192.168.2.22	ICMP	98	Echo (ping) request id=0x902f, seq=5/1280, ttl=64 (reply in 12)
12	14.237414	192.168.2.22	192.168.1.11	ICMP	98	Echo (ping) reply id=0x902f, seq=5/1280, ttl=63 (request in 11)

Пинг PC2 -> PC1:

Захват из Standard input [PC2 Ethernet0 to R1 FastEthernet1/0]

No.	Time	Source	Destination	Protocol	Length	Info
2	16.709093	Private_66:68:00	Broadcast	ARP	64	Who has 192.168.2.1? Tell 192.168.2.22
3	16.712155	cc:01:5c:08:00:10	Private_66:68:00	ARP	60	192.168.2.1 is at cc:01:5c:08:00:10
4	16.712247	192.168.2.22	192.168.1.11	ICMP	98	Echo (ping) request id=0x6b30, seq=1/256, ttl=64 (reply in 5)
5	16.742080	192.168.1.11	192.168.2.22	ICMP	98	Echo (ping) reply id=0x6b30, seq=1/256, ttl=63 (request in 4)
6	17.743099	192.168.2.22	192.168.1.11	ICMP	98	Echo (ping) request id=0x6c30, seq=2/512, ttl=64 (reply in 7)
7	17.757930	192.168.1.11	192.168.2.22	ICMP	98	Echo (ping) reply id=0x6c30, seq=2/512, ttl=63 (request in 6)
8	18.759094	192.168.2.22	192.168.1.11	ICMP	98	Echo (ping) request id=0x6d30, seq=3/768, ttl=64 (reply in 9)
9	18.774702	192.168.1.11	192.168.2.22	ICMP	98	Echo (ping) reply id=0x6d30, seq=3/768, ttl=63 (request in 8)
10	19.775126	192.168.2.22	192.168.1.11	ICMP	98	Echo (ping) request id=0x6e30, seq=4/1024, ttl=64 (reply in 11)
11	19.791675	192.168.1.11	192.168.2.22	ICMP	98	Echo (ping) reply id=0x6e30, seq=4/1024, ttl=63 (request in 10)
12	20.792095	192.168.2.22	192.168.1.11	ICMP	98	Echo (ping) request id=0x6f30, seq=5/1280, ttl=64 (reply in 13)
13	20.807753	192.168.1.11	192.168.2.22	ICMP	98	Echo (ping) reply id=0x6f30, seq=5/1280, ttl=63 (request in 12)