

# Сетевые технологии

Простые сети в GNS3. Анализ трафика (Лабораторная работа №5)

---

Заур Мустафаев

29 октября 2025

Российский университет дружбы народов, Москва, Россия

## Цель лабораторной работы

---

Построить простейшие модели сети в **GNS3**, используя:  
- Коммутатор **Ethernet** -  
Маршрутизаторы **FRRouting (FRR)** и **VyOS** - Средства анализа **Wireshark**

## Выполнение лабораторной работы

---

# Создание топологии в GNS3

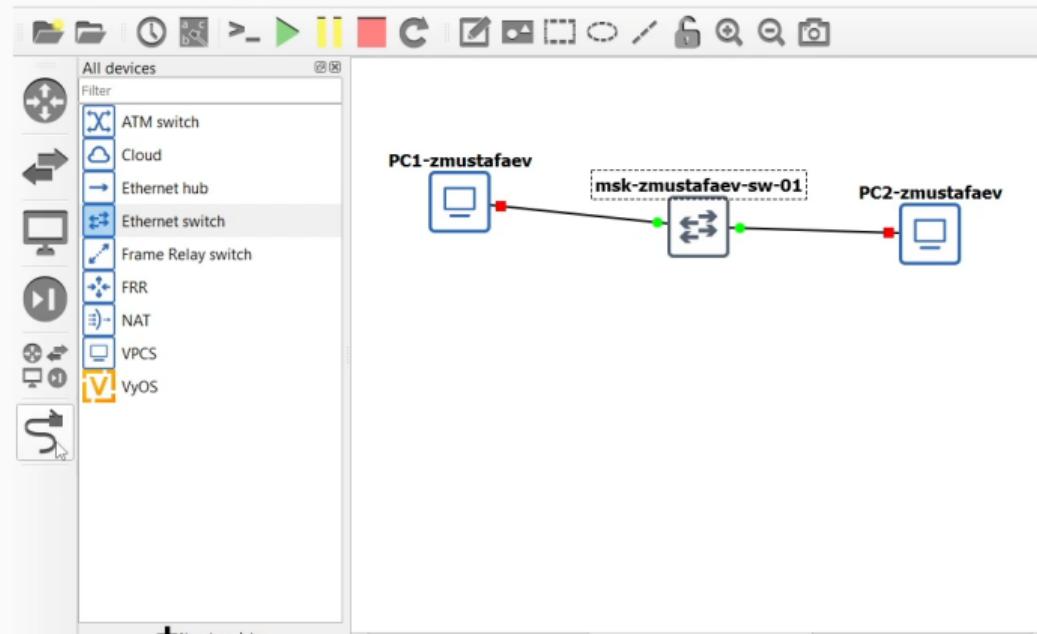
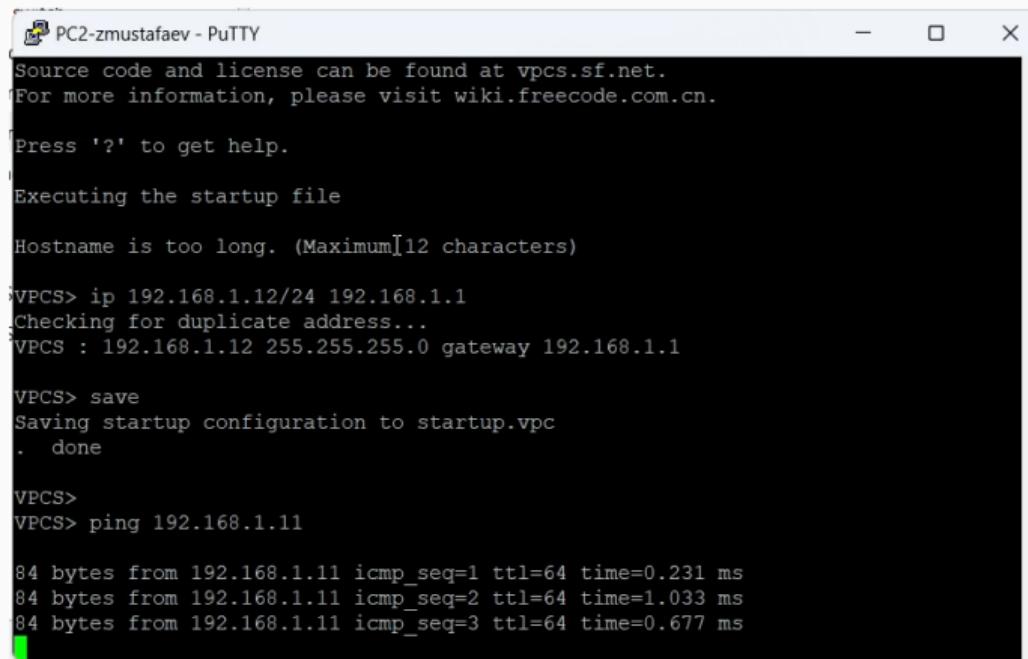


Рис. 1: Топология сети

## Проверка связи между узлами



Source code and license can be found at vpcs.sf.net.  
For more information, please visit wiki.freecode.com.cn.

Press '?' to get help.

Executing the startup file

Hostname is too long. (Maximum[12 characters])

VPCS> ip 192.168.1.12/24 192.168.1.1  
Checking for duplicate address...  
VPCS : 192.168.1.12 255.255.255.0 gateway 192.168.1.1

VPCS> save  
Saving startup configuration to startup.vpc  
. done

VPCS>  
VPCS> ping 192.168.1.11

84 bytes from 192.168.1.11 icmp\_seq=1 ttl=64 time=0.231 ms  
84 bytes from 192.168.1.11 icmp\_seq=2 ttl=64 time=1.033 ms  
84 bytes from 192.168.1.11 icmp\_seq=3 ttl=64 time=0.677 ms

Рис. 2: Ping между узлами

# Анализ ARP-трафика

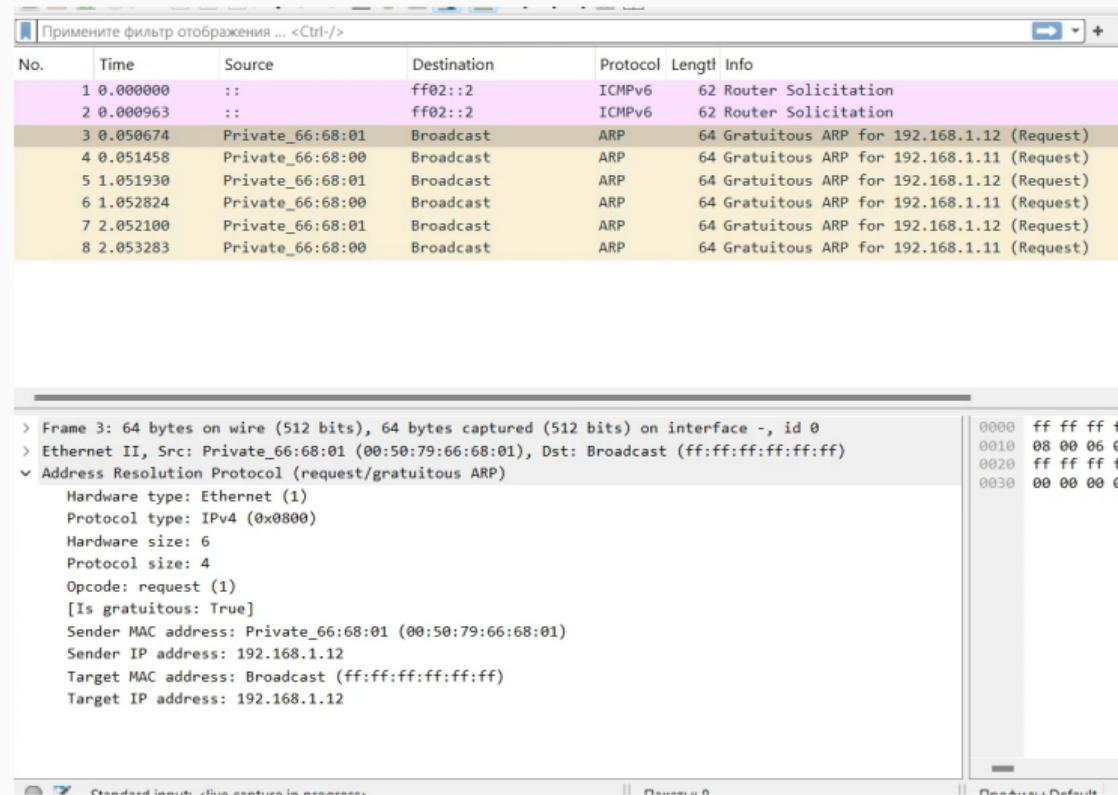


Рис. 3: ARP-пакеты

# Анализ ICMP, UDP и TCP

No.	Time	Source	Destination	Protocol	Length	Info
11	109.470659	192.168.1.12	192.168.1.11	ICMP	98	Echo (ping) request id=0x8b23, seq=1/256,
12	109.470882	192.168.1.11	192.168.1.12	ICMP	98	Echo (ping) reply id=0x8b23, seq=1/256,
13	125.589410	192.168.1.12	192.168.1.11	ECHO	98	Request
14	125.589631	192.168.1.11	192.168.1.12	ECHO	98	Response
15	133.717388	192.168.1.12	192.168.1.11	TCP	74	51909 → 7 [SYN] Seq=0 Win=2920 Len=0 MSS=1460
16	133.717720	192.168.1.11	192.168.1.12	TCP	54	7 → 51909 [SYN, ACK] Seq=0 Ack=1 Win=2920 Len=0
17	133.718814	192.168.1.12	192.168.1.11	TCP	66	51909 → 7 [ACK] Seq=1 Ack=1 Win=2920 Len=0
18	133.718894	192.168.1.12	192.168.1.11	ECHO	122	Request
19	133.719408	192.168.1.11	192.168.1.12	TCP	54	7 → 51909 [ACK] Seq=1 Ack=57 Win=2920 Len=0
20	133.720635	192.168.1.12	192.168.1.11	TCP	66	51909 → 7 [FIN, PSH, ACK] Seq=57 Ack=1 Win=0
21	133.720738	192.168.1.11	192.168.1.12	TCP	54	7 → 51909 [ACK] Seq=1 Ack=58 Win=2920 Len=0
22	133.720749	192.168.1.11	192.168.1.12	TCP	54	7 → 51909 [FIN, ACK] Seq=1 Ack=58 Win=2920
23	133.723289	192.168.1.12	192.168.1.11	TCP	66	51909 → 7 [ACK] Seq=58 Ack=2 Win=2920 Len=0

```
> Frame 13: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface -, id 0
> Ethernet II, Src: Private_66:68:01 (00:50:79:66:68:01), Dst: Private_66:68:00 (00:50:79:66:68:00)
> Internet Protocol Version 4, Src: 192.168.1.12, Dst: 192.168.1.11
< User Datagram Protocol, Src Port: 32551, Dst Port: 7
    Source Port: 32551
    Destination Port: 7
    Length: 64
    Checksum: 0x614d [unverified]
        [Checksum Status: Unverified]
        [Stream index: 0]
        [Stream Packet Number: 1]
    > [Timestamps]
        UDP payload (56 bytes)
< Echo
    Echo data: 0050796668010e0f101112131415161718191a1b1c1d1e1f202122232425262728292a2b2c2d2e2f3031323...
```

0000	00 50 79 6
0010	00 54 23 9
0020	01 0b 7f 2
0030	0e 0f 10 1
0040	1e 1f 20 2
0050	2e 2f 30 3
0060	3e 3f

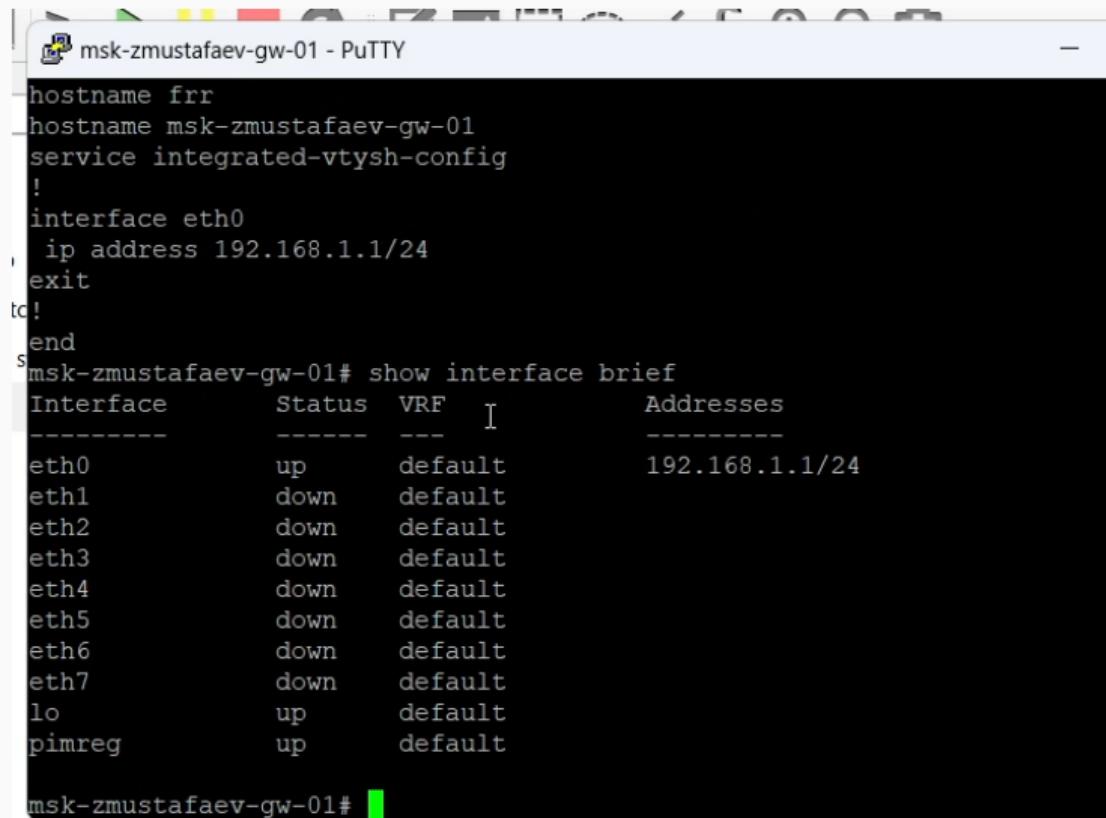


Standard input: &lt;live capture in progress&gt;

Пакеты: 23

Профиль: Default

## Настройка FRR



The screenshot shows a PuTTY terminal window titled "msk-zmustafaev-gw-01 - PuTTY". The session has a color scheme with green, yellow, red, and grey. The terminal displays the following configuration and output:

```
hostname frr
hostname msk-zmustafaev-gw-01
service integrated-vtysh-config
!
interface eth0
 ip address 192.168.1.1/24
exit
tc!
end
msk-zmustafaev-gw-01# show interface brief
Interface      Status   VRF    I      Addresses
-----        -----   ---    --      -----
eth0          up       default  192.168.1.1/24
eth1          down     default
eth2          down     default
eth3          down     default
eth4          down     default
eth5          down     default
eth6          down     default
eth7          down     default
lo            up       default
pimreg        up       default
msk-zmustafaev-gw-01#
```

# Анализ ICMP FRR

No.	Time	Source	Destination	Protocol	Length	Info
17	230.456497	192.168.1.10	192.168.1.1	ICMP	98	Echo (ping) request id=0x8125, seq=1/256,
18	230.458452	192.168.1.1	192.168.1.10	ICMP	98	Echo (ping) reply id=0x8125, seq=1/256,
19	231.459625	192.168.1.10	192.168.1.1	ICMP	98	Echo (ping) request id=0x8225, seq=2/512,
20	231.460535	192.168.1.1	192.168.1.10	ICMP	98	Echo (ping) reply id=0x8225, seq=2/512,
21	232.462738	192.168.1.10	192.168.1.1	ICMP	98	Echo (ping) request id=0x8325, seq=3/768,
22	232.464103	192.168.1.1	192.168.1.10	ICMP	98	Echo (ping) reply id=0x8325, seq=3/768,
23	233.466019	192.168.1.10	192.168.1.1	ICMP	98	Echo (ping) request id=0x8425, seq=4/1024,
24	233.466750	192.168.1.10	192.168.1.10	ICMP	98	Echo (ping) reply id=0x8425, seq=4/1024,
→	25	234.468902	192.168.1.10	192.168.1.1	ICMP	98 Echo (ping) request id=0x8525, seq=5/1280,
←	26	234.470519	192.168.1.1	192.168.1.10	ICMP	98 Echo (ping) reply id=0x8525, seq=5/1280,
27	235.498572	0c:a8:c1:ff:00:00	Private_66:68:00	ARP	60	Who has 192.168.1.10? Tell 192.168.1.1
28	235.499272	Private_66:68:00	0c:a8:c1:ff:00:00	ARP	60	192.168.1.10 is at 00:50:79:66:68:00

> Frame 26: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface -, id 0	0000 00 50 79 60
> Ethernet II, Src: 0c:a8:c1:ff:00:00 (0c:a8:c1:ff:00:00), Dst: Private_66:68:00 (00:50:79:66:68:00)	0010 00 54 eb 1b
> Internet Protocol Version 4, Src: 192.168.1.1, Dst: 192.168.1.10	0020 01 0a 00 00
Internet Control Message Protocol	0030 0e 0f 10 11
Type: 0 (Echo (ping) reply)	0040 1e 1f 20 21
Code: 0	0050 2e 2f 30 31
Checksum: 0xa2e1 [correct]	0060 3e 3f
[Checksum Status: Good]	
Identifier (BE): 34085 (0x8525)	
Identifier (LE): 9605 (0x2585)	
Sequence Number (BE): 5 (0x0005)	
Sequence Number (LE): 1280 (0x0500)	
<u>[Request frame: 25]</u>	
[Response time: 1,617 ms]	
Data (56 bytes)	

Рис. 6: ICMP FRR

# Настройка VyOS

```
msk-zmustafaev-gw-01 - PuTTY
[edit]
vyos@vyos# commit
[edit]
vyos@vyos# save
Saving configuration to '/config/config.boot'...
Done
[edit]
vyos@vyos# show interfaces
 ethernet eth0 {
     address 192.168.1.1/24
     hw-id 0c:77:69:cf:00:00
 }
 ethernet eth1 {
     hw-id 0c:77:69:cf:00:01
 }
 ethernet eth2 {
     hw-id 0c:77:69:cf:00:02
 }
 loopback lo {
 }
[edit]
vyos@vyos# exit
exit
vyos@vyos:~$
```

## Анализ ICMP и ARP

No.	Time	Source	Destination	Protocol	Length	Info
26	217.570334	192.168.1.10	192.168.1.1	ICMP	98	Echo (ping) request id=0xc026, seq=1/256,
27	217.573379	192.168.1.1	192.168.1.10	ICMP	98	Echo (ping) reply id=0xc026, seq=1/256,
28	218.577156	192.168.1.10	192.168.1.1	ICMP	98	Echo (ping) request id=0xc126, seq=2/512,
29	218.578906	192.168.1.1	192.168.1.10	ICMP	98	Echo (ping) reply id=0xc126, seq=2/512,
30	219.580463	192.168.1.10	192.168.1.1	ICMP	98	Echo (ping) request id=0xc226, seq=3/768,
31	219.582168	192.168.1.1	192.168.1.10	ICMP	98	Echo (ping) reply id=0xc226, seq=3/768,
32	220.584325	192.168.1.10	192.168.1.1	ICMP	98	Echo (ping) request id=0xc326, seq=4/1024,
33	220.585093	192.168.1.1	192.168.1.10	ICMP	98	Echo (ping) reply id=0xc326, seq=4/1024,
→	34 221.586548	192.168.1.10	192.168.1.1	ICMP	98	Echo (ping) request id=0xc426, seq=5/1280,
←	35 221.587954	192.168.1.1	192.168.1.10	ICMP	98	Echo (ping) reply id=0xc426, seq=5/1280,
36	222.939655	0c:77:69:cf:00:00	Private_66:68:00	ARP	60	Who has 192.168.1.10? Tell 192.168.1.1
37	222.940529	Private_66:68:00	0c:77:69:cf:00:00	ARP	60	192.168.1.10 is at 00:50:79:66:68:00

> Frame 35: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface -, id 0	0000 00 50 79 6
> Ethernet II, Src: 0c:77:69:cf:00:00 (0c:77:69:cf:00:00), Dst: Private_66:68:00 (00:50:79:66:68:00)	0010 00 54 31 1
> Internet Protocol Version 4, Src: 192.168.1.1, Dst: 192.168.1.10	0020 01 0a 00 0
> Internet Control Message Protocol	0030 0e 0f 10 1
	0040 1e 1f 20 2
	0050 2e 2f 30 3
	0060 3e 3f

Рис. 8: Анализ трафика VyOS

## Выводы по работе

---

- Смоделированы сети на базе маршрутизаторов **FRR** и **VyOS**
- Настроены IP-сети и подтверждена их работоспособность
- Проведён анализ трафика с помощью **Wireshark**
- Проверена корректность протоколов **ARP**, **ICMP**, **UDP** и **TCP**
- Работа выполнена успешно