ЛАБОРАТОРНАЯ РАБОТА №5

Студент: Талебу Тенке Франк Устон

Группа: НФИбд-02-23

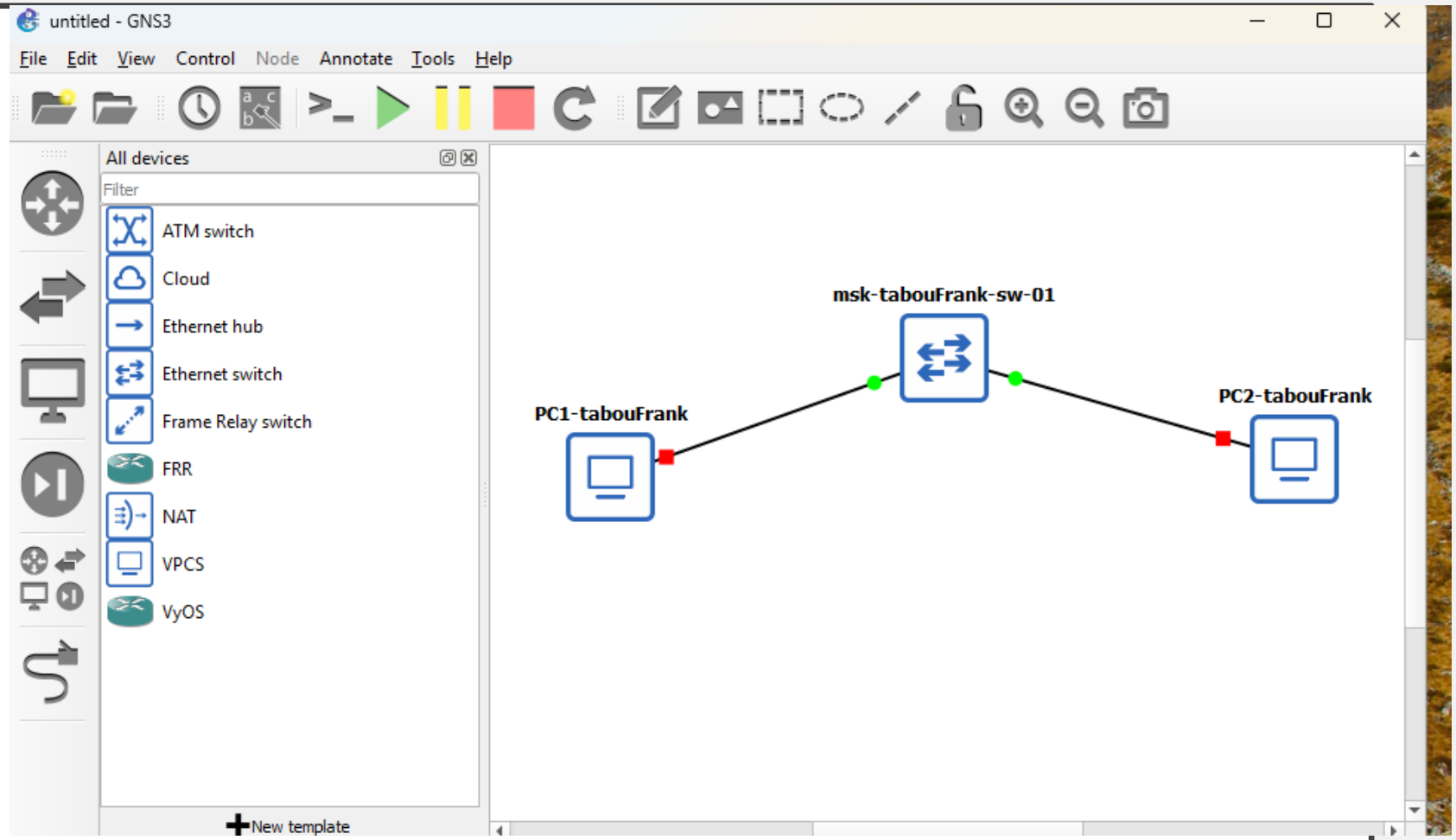
ЦЕЛЬ РАБОТА

Построить простейшие модели сетей на базе коммутатора и маршрутизаторов FRR и VyOS в GNS3, проанализировать трафик посредством Wireshark.

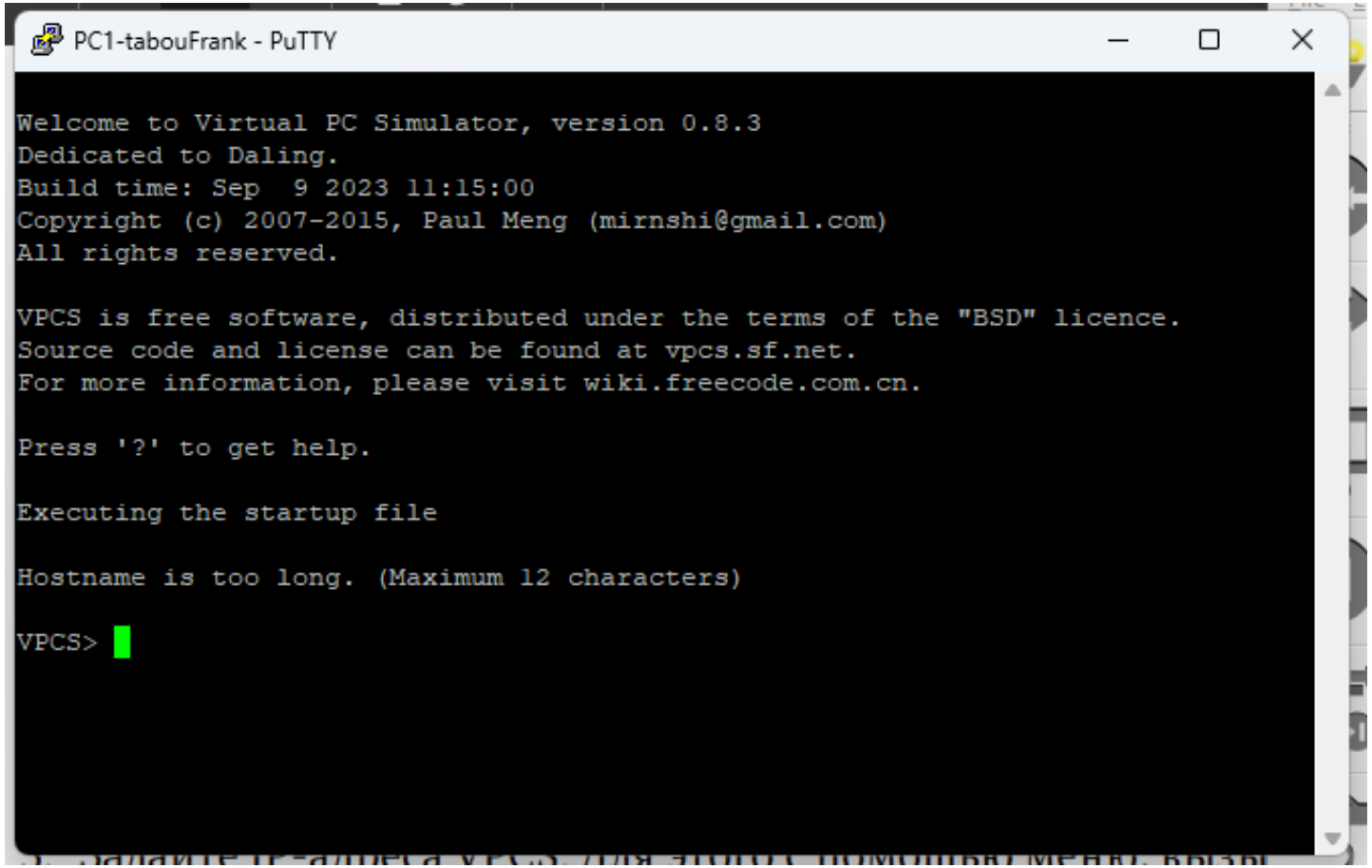
ЗАДАНИЕ

- Смоделировать простейшую сеть на базе коммутатора в GNS3;
- Проанализировать трафик в GNS3 посредством Wireshark;
- Смоделировать простейшую сеть на базе маршрутизатора FRR в GNS3;
- Смоделировать простейшую сеть на базе маршрутизатора VyOS в GNS3.

ВЫПОЛНЕНИЕ ЛАБОРАТОРНОЙ РАБОТЫ



ВЫПОЛНЕНИЕ ЛАБОРАТОРНОЙ РАБОТЫ



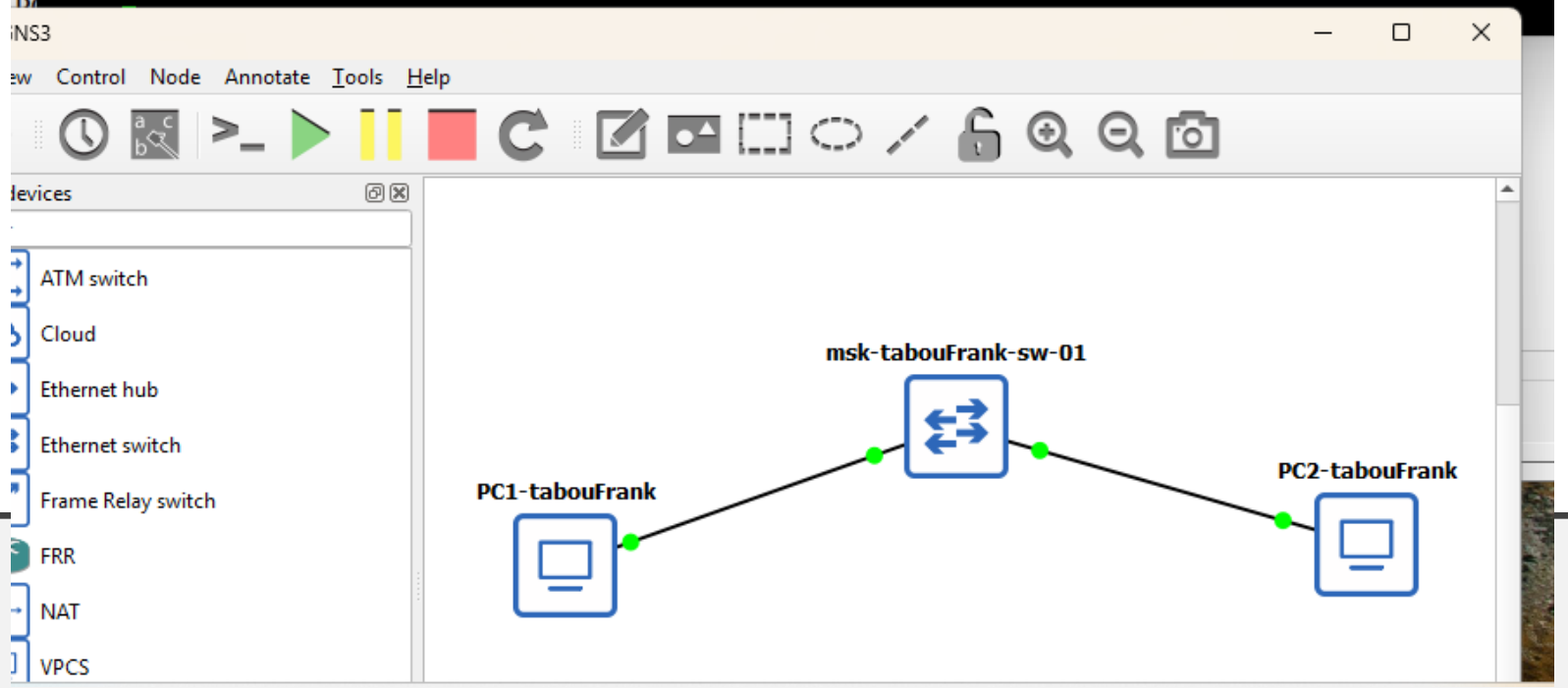
ВЫПОЛНЕНИЕ ЛАБОРАТОРНОЙ РАБОТЫ

ВЫПОЛНЕНИЕ ЛАБОРАТОРНОЙ РАБОТЫ

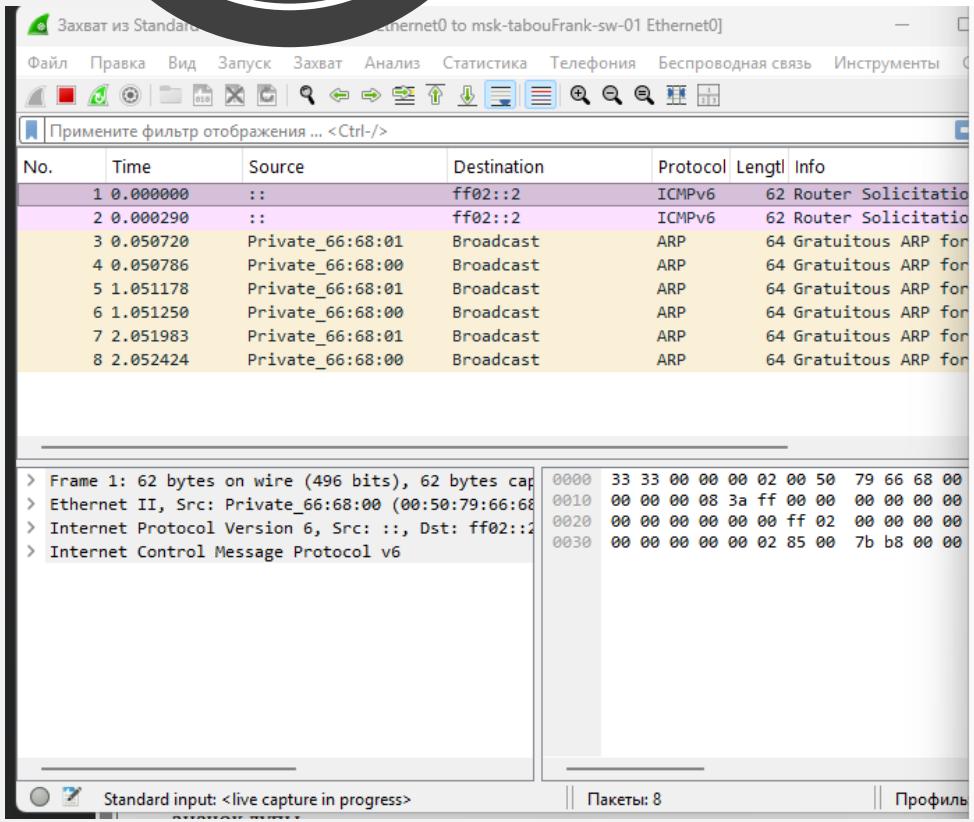
```
VPCS> /?

?                       Print help
arp                       Shortcut for: show arp. Show arp table
clear ARG               Clear IPv4/IPv6, arp/neighbor cache, command history
dhcp [OPTION]           Shortcut for: ip dhcp. Get IPv4 address via DHCP
disconnect              Exit the telnet session (daemon mode)
echo TEXT               Display TEXT in output. See also  set echo ?
help                    Print help
history                 Shortcut for: show history. List the command history
ip ARG ... [OPTION]     Configure the current VPC's IP settings. See ip ?
load [FILENAME]         Load the configuration/script from the file FILENAME
ping HOST [OPTION ...]  Ping HOST with ICMP (default) or TCP/UDP. See ping ?
quit                    Quit program
relay ARG ...           Configure packet relay between UDP ports. See relay ?
rlogin [ip] port        Telnet to port on host at ip (relative to host PC)
save [FILENAME]         Save the configuration to the file FILENAME
set ARG ...             Set VPC name and other options. Try set ?
show [ARG ...]          Print the information of VPCs (default). See show ?
sleep [seconds] [TEXT]  Print TEXT and pause running script for seconds
trace HOST [OPTION ...] Print the path packets take to network HOST
version                 Shortcut for: show version

To get command syntax help, please enter '?' as an argument of the command.

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>
```

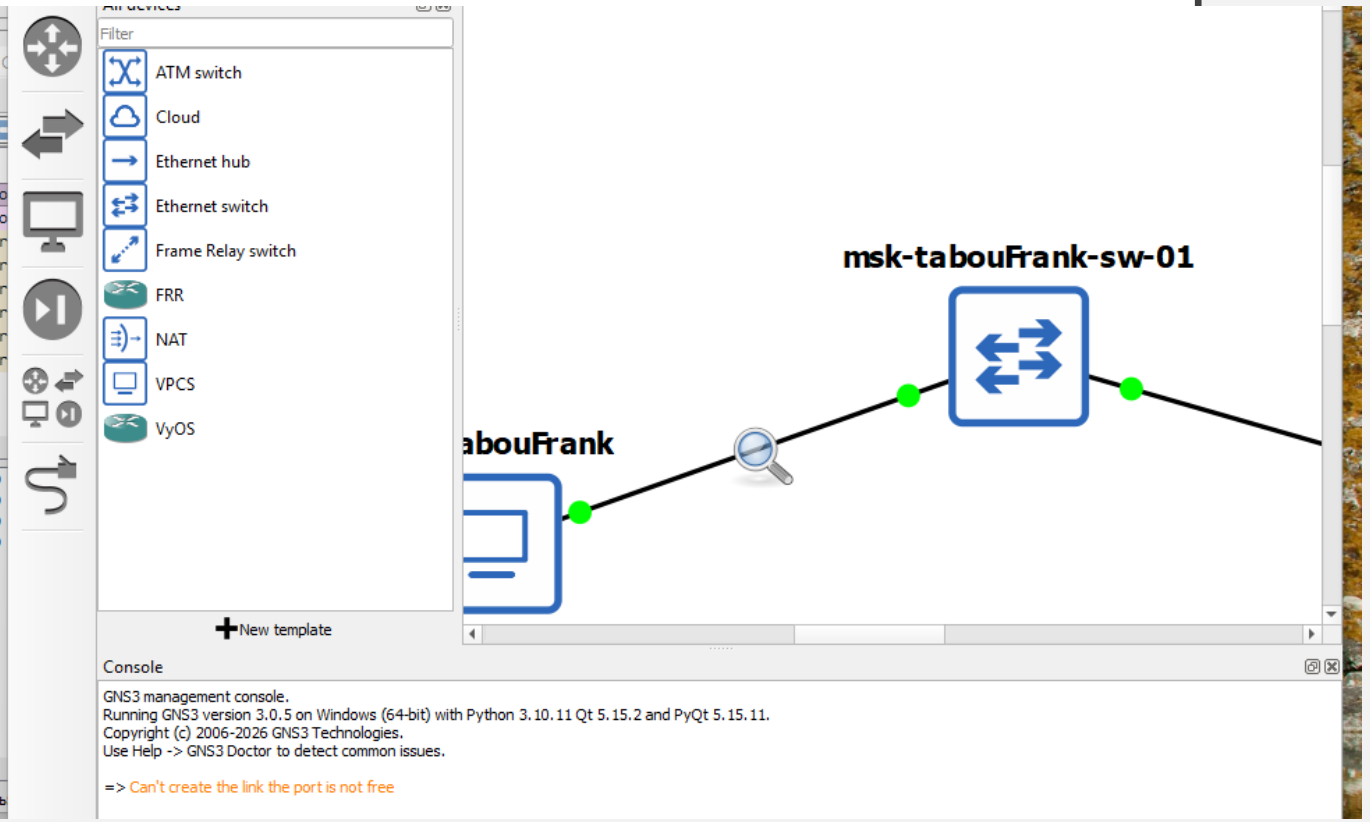
ВЫПОЛНЕНИЕ ЛАБОРАТОРНОЙ РАБОТЫ

```
VPCS> ping

ping HOST [OPTION ...]
  Ping the network HOST. HOST can be an ip address or name
    Options:
     -1             ICMP mode, default
     -2             UDP mode
     -3             TCP mode
     -c count       Packet count, default 5
     -D             Set the Don't Fragment bit
     -f FLAG        Tcp header FLAG |C|E|U|A|P|R|S|F|
                                bits |7 6 5 4 3 2 1 0|
     -i ms          Wait ms milliseconds between sending each packet
     -l size        Data size
     -P protocol    Use IP protocol in ping packets
                      1 - ICMP (default), 17 - UDP, 6 - TCP
     -p port        Destination port
     -s port        Source port
     -T ttl         Set ttl, default 64
     -t             Send packets until interrupted by Ctrl+C
     -w ms          Wait ms milliseconds to receive the response

  Notes: 1. Using names requires DNS to be set.
         2. Use Ctrl+C to stop the command.
```

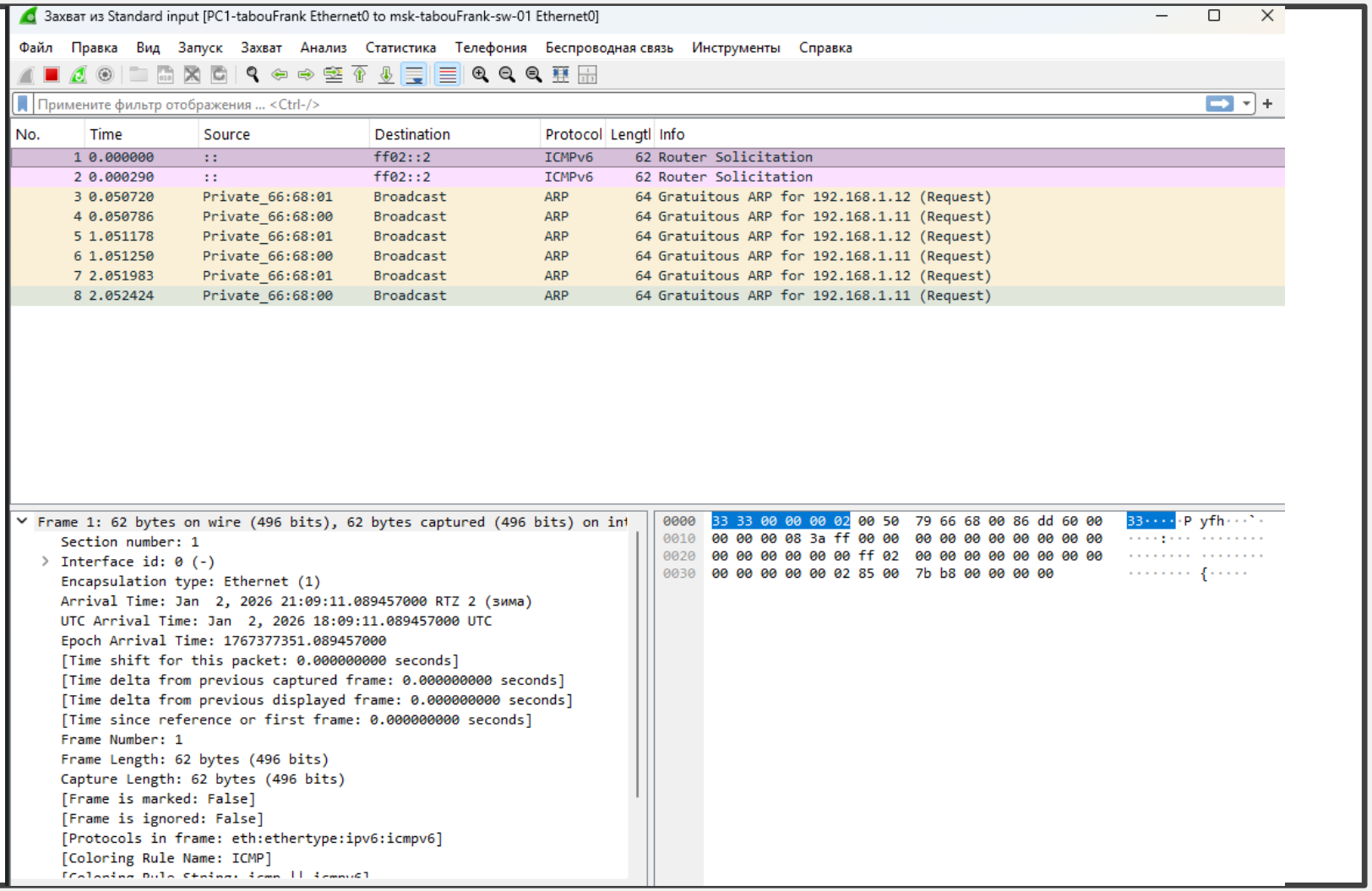


ВЫПОЛНЕНИЕ ЛАБОРАТОРНОЙ РАБОТЫ

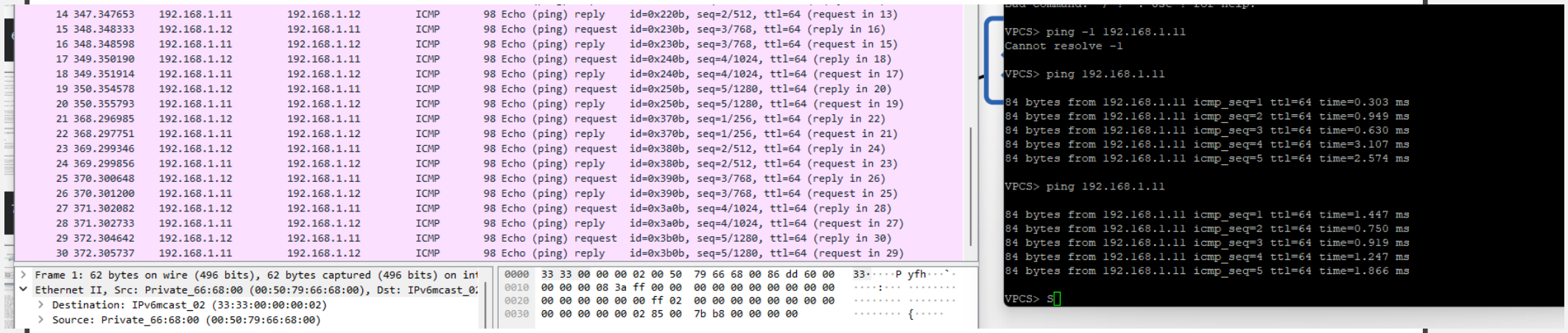




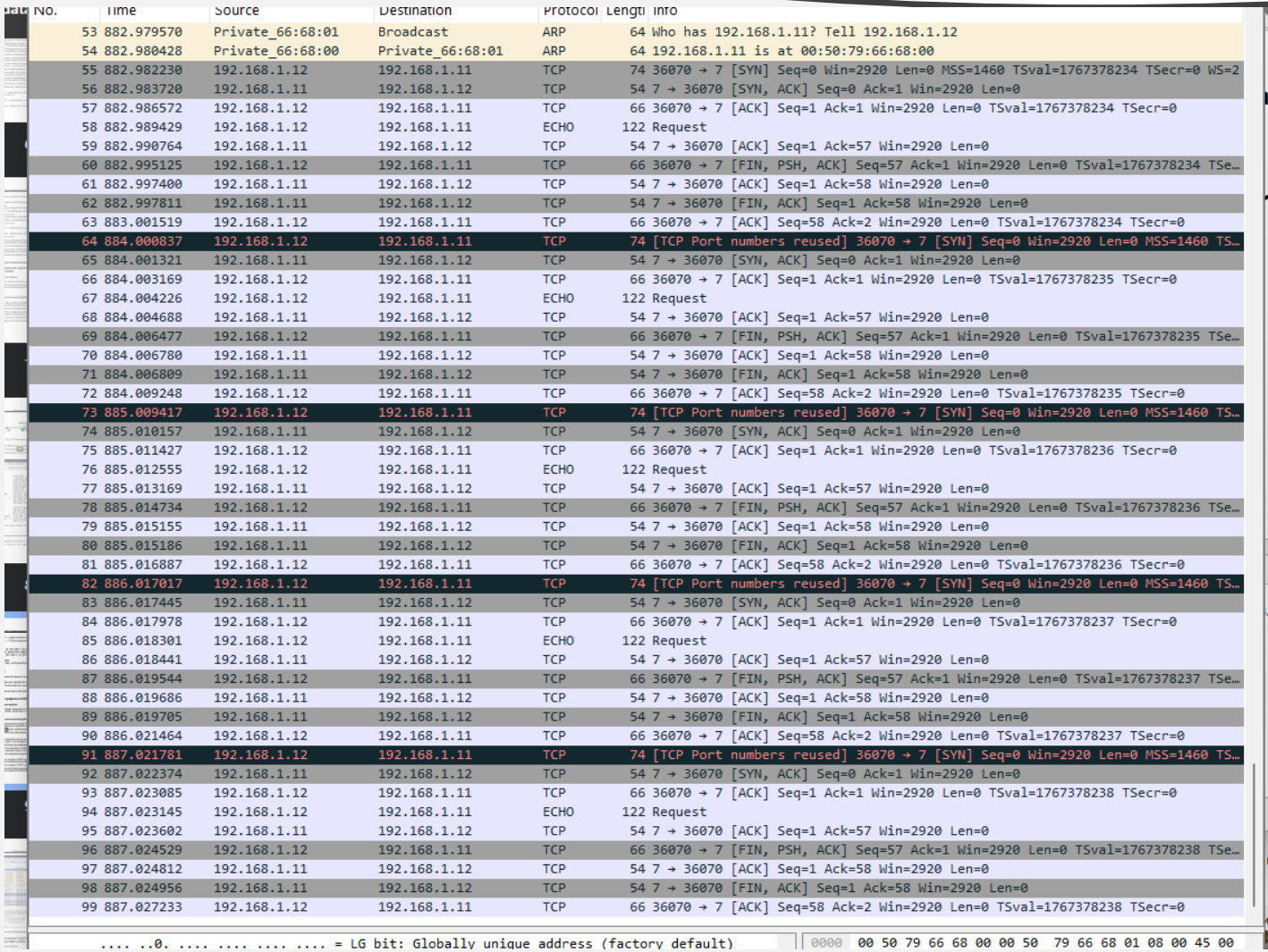
ВЫПОЛНЕНИЕ ЛАБОРАТОРНОЙ РАБОТЫ



ВЫПОЛНЕНИЕ ЛАБОРАТОРНОЙ РАБОТЫ



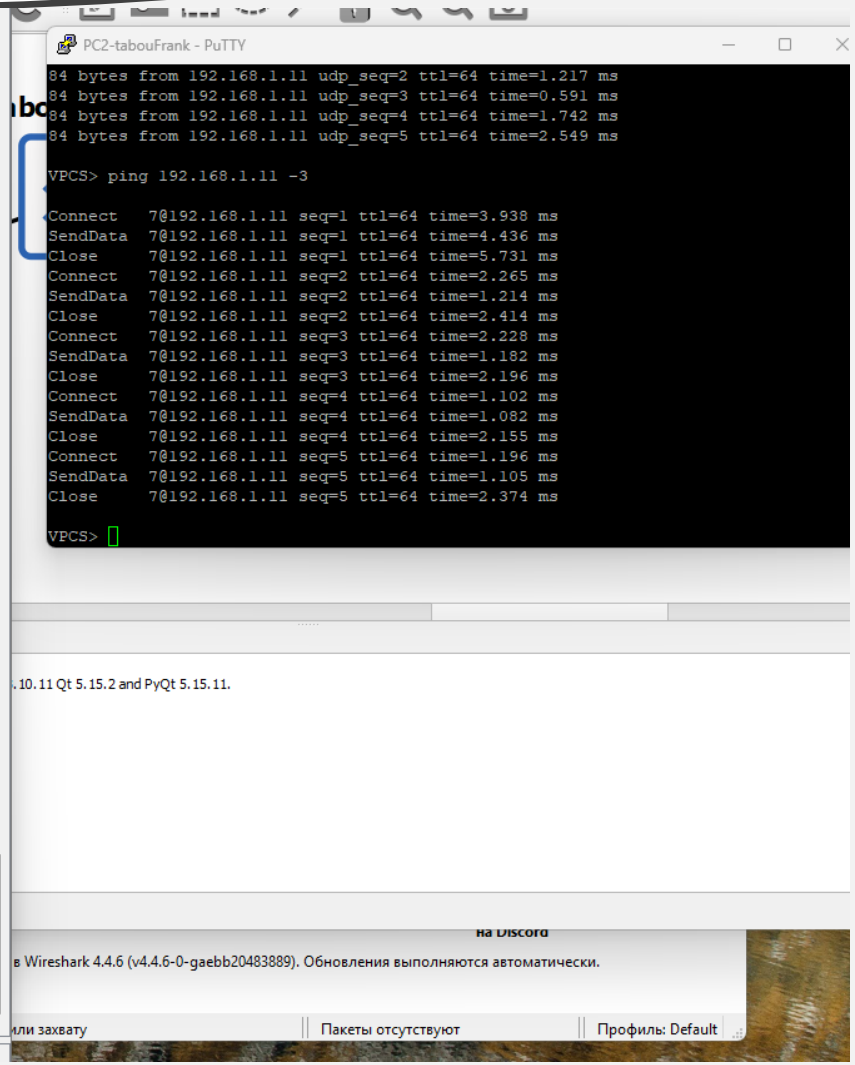
ВЫПОЛНЕНИЕ ЛАБОРАТОРНОЙ РАБОТЫ



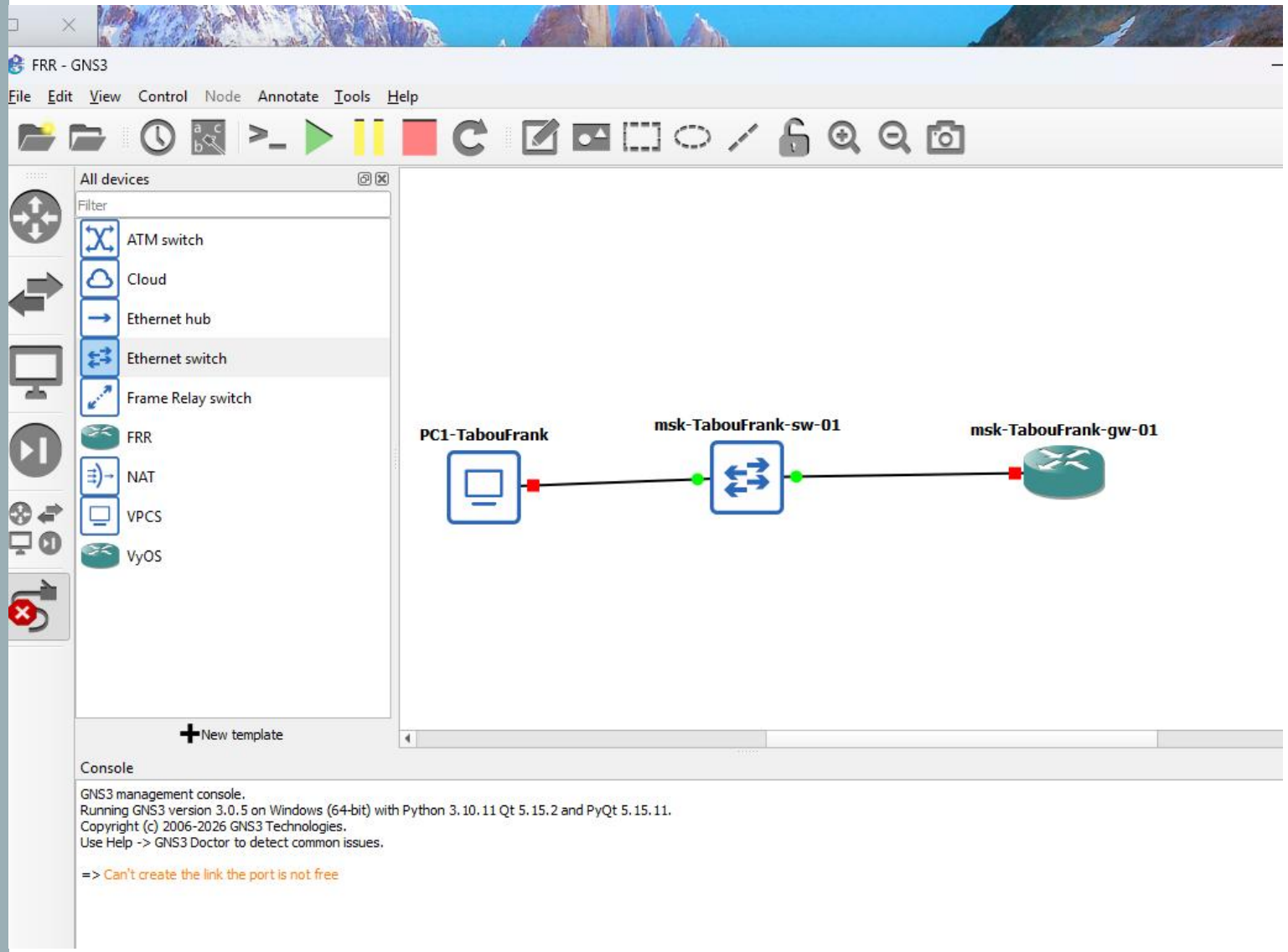
No.	Time	Source	Destination	Protocol	Length	Info
53	882.979570	Private_66:68:01	Broadcast	ARP	64	Who has 192.168.1.11? Tell 192.168.1.12
54	882.980428	Private_66:68:00	Private_66:68:01	ARP	64	192.168.1.11 is at 00:50:79:66:68:00
55	882.982230	192.168.1.12	192.168.1.11	TCP	74	36070 → 7 [SYN] Seq=0 Win=2920 Len=0 MSS=1460 TSval=1767378234 TSecr=0 WS=2
56	882.983720	192.168.1.11	192.168.1.12	TCP	54	7 → 36070 [SYN, ACK] Seq=0 Ack=1 Win=2920 Len=0
57	882.986572	192.168.1.12	192.168.1.11	TCP	66	36070 → 7 [ACK] Seq=1 Ack=1 Win=2920 Len=0 TSval=1767378234 TSecr=0
58	882.989429	192.168.1.12	192.168.1.11	ECHO	122	Request
59	882.990764	192.168.1.11	192.168.1.12	TCP	54	7 → 36070 [ACK] Seq=1 Ack=57 Win=2920 Len=0
60	882.995125	192.168.1.12	192.168.1.11	TCP	66	36070 → 7 [FIN, PSH, ACK] Seq=57 Ack=1 Win=2920 Len=0 TSval=1767378234 TSe…
61	882.997400	192.168.1.11	192.168.1.12	TCP	54	7 → 36070 [ACK] Seq=1 Ack=58 Win=2920 Len=0
62	882.997811	192.168.1.11	192.168.1.12	TCP	54	7 → 36070 [FIN, ACK] Seq=1 Ack=58 Win=2920 Len=0
63	883.001519	192.168.1.12	192.168.1.11	TCP	66	36070 → 7 [ACK] Seq=58 Ack=2 Win=2920 Len=0 TSval=1767378234 TSecr=0
64	884.000837	192.168.1.12	192.168.1.11	TCP	74	[TCP Port numbers reused] 36070 → 7 [SYN] Seq=0 Win=2920 Len=0 MSS=1460 TS…
65	884.001321	192.168.1.11	192.168.1.12	TCP	54	7 → 36070 [SYN, ACK] Seq=0 Ack=1 Win=2920 Len=0
66	884.003169	192.168.1.12	192.168.1.11	TCP	66	36070 → 7 [ACK] Seq=1 Ack=1 Win=2920 Len=0 TSval=1767378235 TSecr=0
67	884.004226	192.168.1.12	192.168.1.11	ECHO	122	Request
68	884.004688	192.168.1.11	192.168.1.12	TCP	54	7 → 36070 [ACK] Seq=1 Ack=57 Win=2920 Len=0
69	884.006477	192.168.1.12	192.168.1.11	TCP	66	36070 → 7 [FIN, PSH, ACK] Seq=57 Ack=1 Win=2920 Len=0 TSval=1767378235 TSe…
70	884.006780	192.168.1.11	192.168.1.12	TCP	54	7 → 36070 [ACK] Seq=1 Ack=58 Win=2920 Len=0
71	884.006809	192.168.1.11	192.168.1.12	TCP	54	7 → 36070 [FIN, ACK] Seq=1 Ack=58 Win=2920 Len=0
72	884.009248	192.168.1.12	192.168.1.11	TCP	66	36070 → 7 [ACK] Seq=58 Ack=2 Win=2920 Len=0 TSval=1767378235 TSecr=0
73	885.009417	192.168.1.12	192.168.1.11	TCP	74	[TCP Port numbers reused] 36070 → 7 [SYN] Seq=0 Win=2920 Len=0 MSS=1460 TS…
74	885.010157	192.168.1.11	192.168.1.12	TCP	54	7 → 36070 [SYN, ACK] Seq=0 Ack=1 Win=2920 Len=0
75	885.011427	192.168.1.12	192.168.1.11	TCP	66	36070 → 7 [ACK] Seq=1 Ack=1 Win=2920 Len=0 TSval=1767378236 TSecr=0
76	885.012555	192.168.1.12	192.168.1.11	ECHO	122	Request
77	885.013169	192.168.1.11	192.168.1.12	TCP	54	7 → 36070 [ACK] Seq=1 Ack=57 Win=2920 Len=0
78	885.014734	192.168.1.12	192.168.1.11	TCP	66	36070 → 7 [FIN, PSH, ACK] Seq=57 Ack=1 Win=2920 Len=0 TSval=1767378236 TSe…
79	885.015155	192.168.1.11	192.168.1.12	TCP	54	7 → 36070 [ACK] Seq=1 Ack=58 Win=2920 Len=0
80	885.015186	192.168.1.11	192.168.1.12	TCP	54	7 → 36070 [FIN, ACK] Seq=1 Ack=58 Win=2920 Len=0
81	885.016887	192.168.1.12	192.168.1.11	TCP	66	36070 → 7 [ACK] Seq=58 Ack=2 Win=2920 Len=0 TSval=1767378236 TSecr=0
82	886.017017	192.168.1.12	192.168.1.11	TCP	74	[TCP Port numbers reused] 36070 → 7 [SYN] Seq=0 Win=2920 Len=0 MSS=1460 TS…
83	886.017445	192.168.1.11	192.168.1.12	TCP	54	7 → 36070 [SYN, ACK] Seq=0 Ack=1 Win=2920 Len=0
84	886.017978	192.168.1.12	192.168.1.11	TCP	66	36070 → 7 [ACK] Seq=1 Ack=1 Win=2920 Len=0 TSval=1767378237 TSecr=0
85	886.018301	192.168.1.12	192.168.1.11	ECHO	122	Request
86	886.018441	192.168.1.11	192.168.1.12	TCP	54	7 → 36070 [ACK] Seq=1 Ack=57 Win=2920 Len=0
87	886.019544	192.168.1.12	192.168.1.11	TCP	66	36070 → 7 [FIN, PSH, ACK] Seq=57 Ack=1 Win=2920 Len=0 TSval=1767378237 TSe…
88	886.019686	192.168.1.11	192.168.1.12	TCP	54	7 → 36070 [ACK] Seq=1 Ack=58 Win=2920 Len=0
89	886.019705	192.168.1.11	192.168.1.12	TCP	54	7 → 36070 [FIN, ACK] Seq=1 Ack=58 Win=2920 Len=0
90	886.021464	192.168.1.12	192.168.1.11	TCP	66	36070 → 7 [ACK] Seq=58 Ack=2 Win=2920 Len=0 TSval=1767378237 TSecr=0
91	887.021781	192.168.1.12	192.168.1.11	TCP	74	[TCP Port numbers reused] 36070 → 7 [SYN] Seq=0 Win=2920 Len=0 MSS=1460 TS…
92	887.022374	192.168.1.11	192.168.1.12	TCP	54	7 → 36070 [SYN, ACK] Seq=0 Ack=1 Win=2920 Len=0
93	887.023085	192.168.1.12	192.168.1.11	TCP	66	36070 → 7 [ACK] Seq=1 Ack=1 Win=2920 Len=0 TSval=1767378238 TSecr=0
94	887.023145	192.168.1.12	192.168.1.11	ECHO	122	Request
95	887.023602	192.168.1.11	192.168.1.12	TCP	54	7 → 36070 [ACK] Seq=1 Ack=57 Win=2920 Len=0
96	887.024529	192.168.1.12	192.168.1.11	TCP	66	36070 → 7 [FIN, PSH, ACK] Seq=57 Ack=1 Win=2920 Len=0 TSval=1767378238 TSe…
97	887.024812	192.168.1.11	192.168.1.12	TCP	54	7 → 36070 [ACK] Seq=1 Ack=58 Win=2920 Len=0
98	887.024956	192.168.1.11	192.168.1.12	TCP	54	7 → 36070 [FIN, ACK] Seq=1 Ack=58 Win=2920 Len=0
99	887.027233	192.168.1.12	192.168.1.11	TCP	66	36070 → 7 [ACK] Seq=58 Ack=2 Win=2920 Len=0 TSval=1767378238 TSecr=0

.... ..0. .... = LG bit: Globally unique address (factory default)

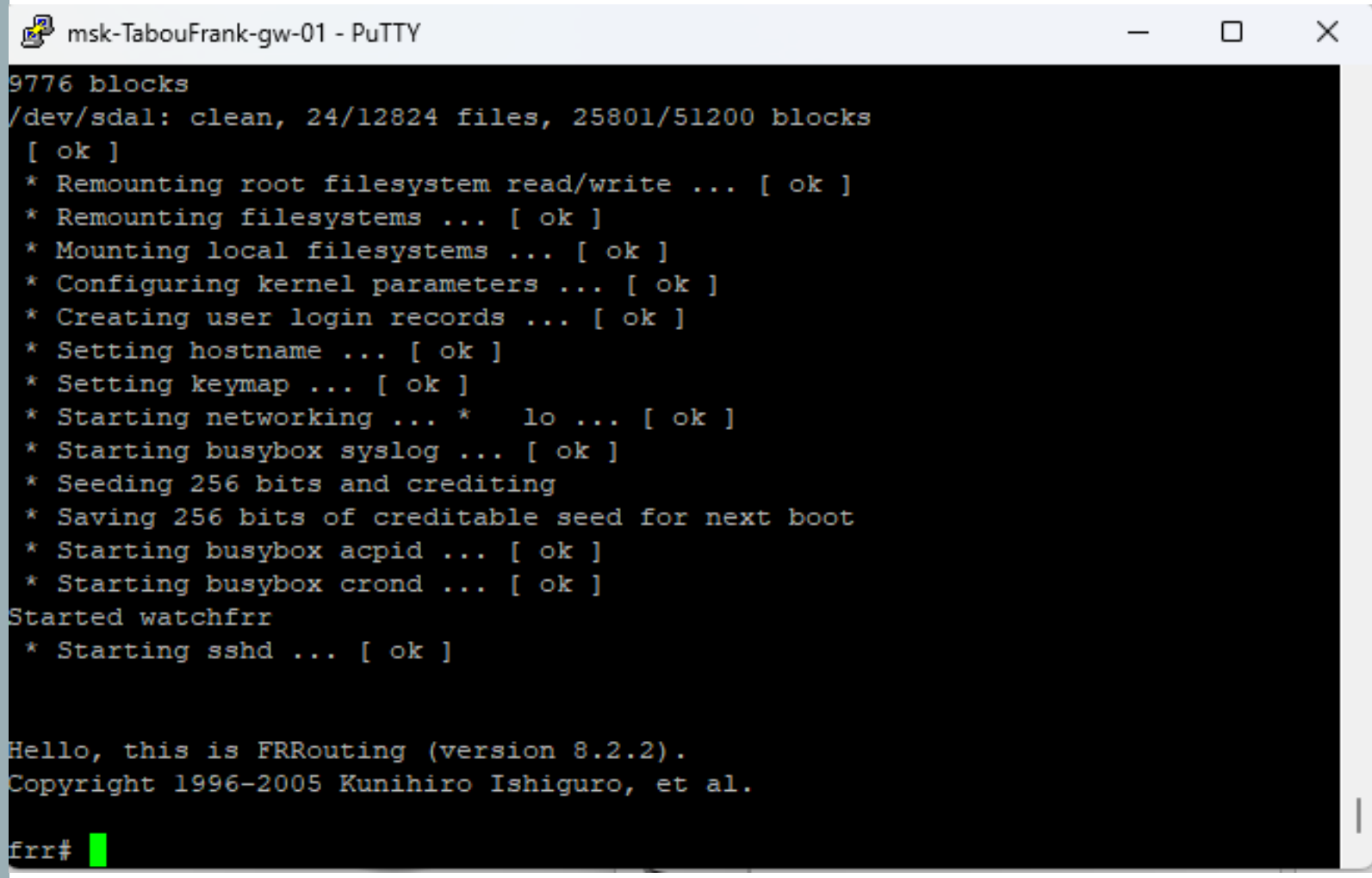
0000 00 50 79 66 68 00 00 50 79 66 68 01 08 00 45 00



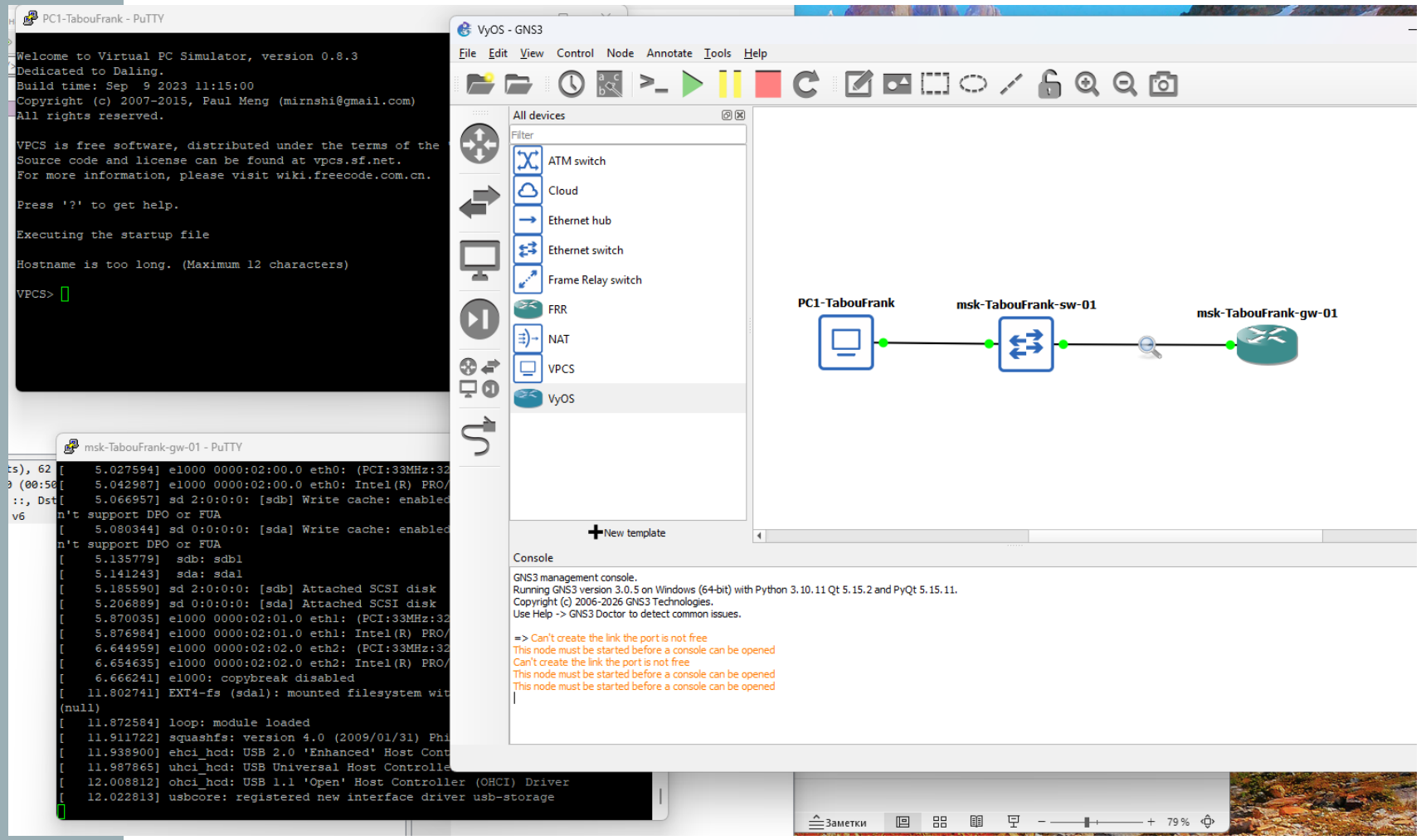
ВЫПОЛНЕНИЕ ЛАБОРАТОРНОЙ РАБОТЫ



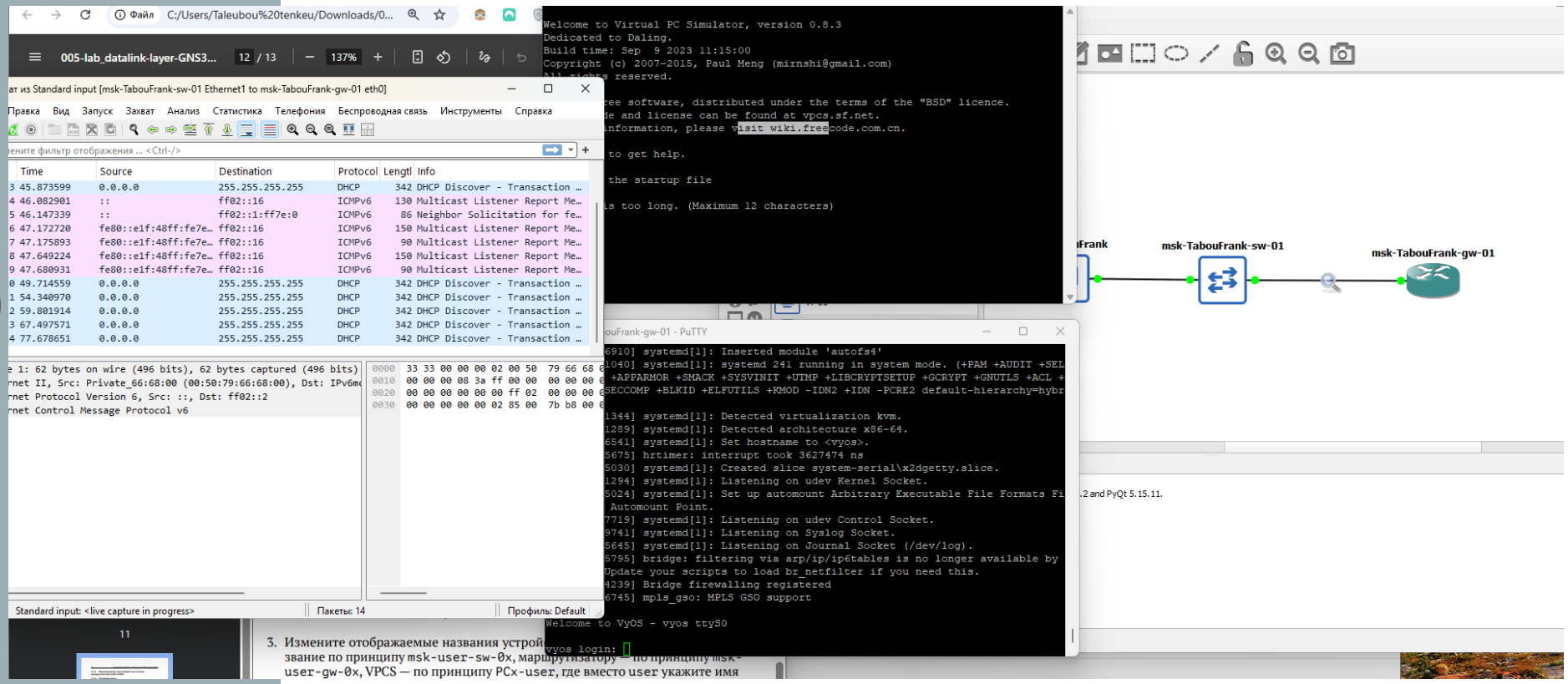
ВЫПОЛНЕНИЕ ЛАБОРАТОРНОЙ РАБОТЫ



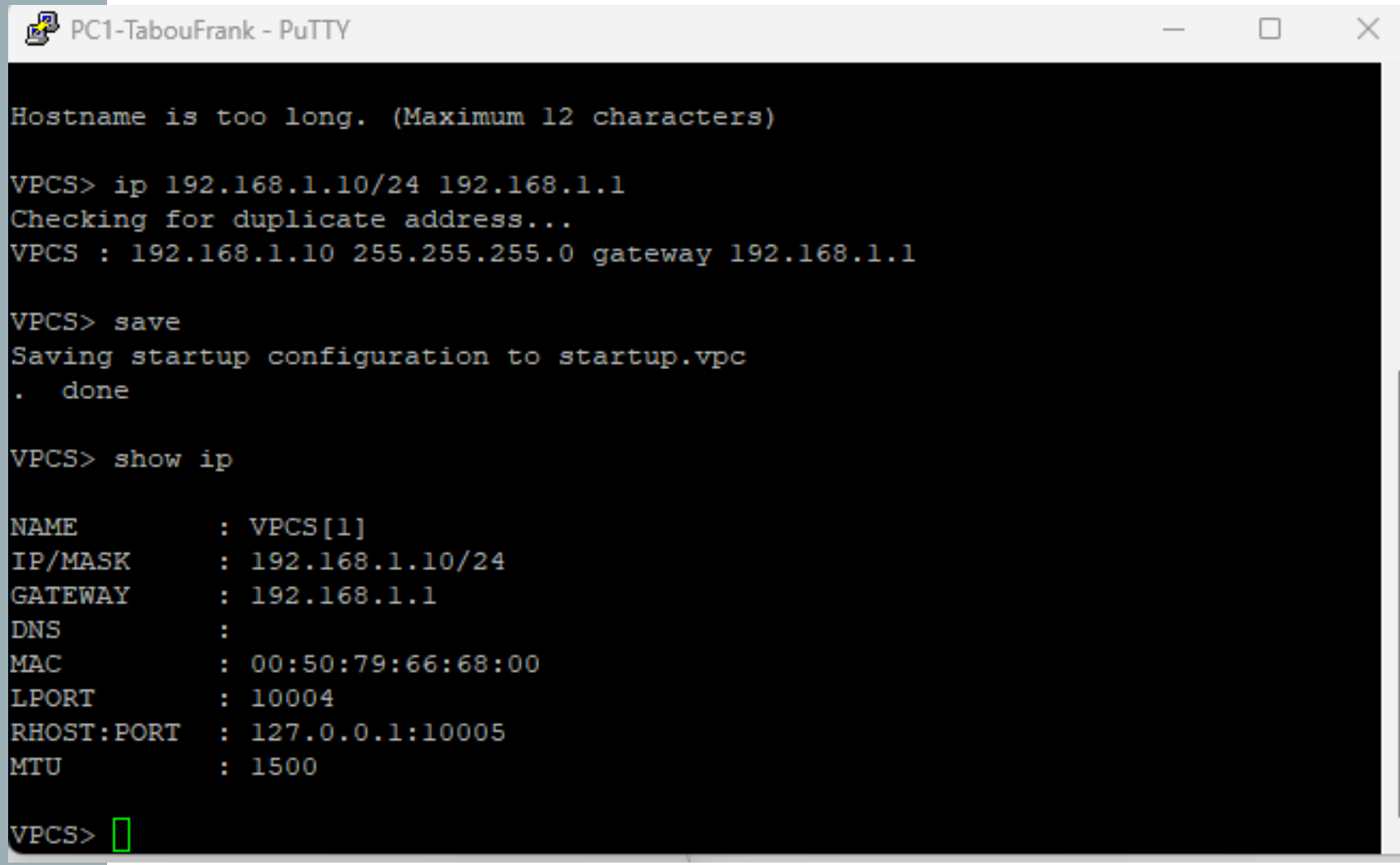
ВЫПОЛНЕНИЕ ЛАБОРАТОРНОЙ РАБОТЫ



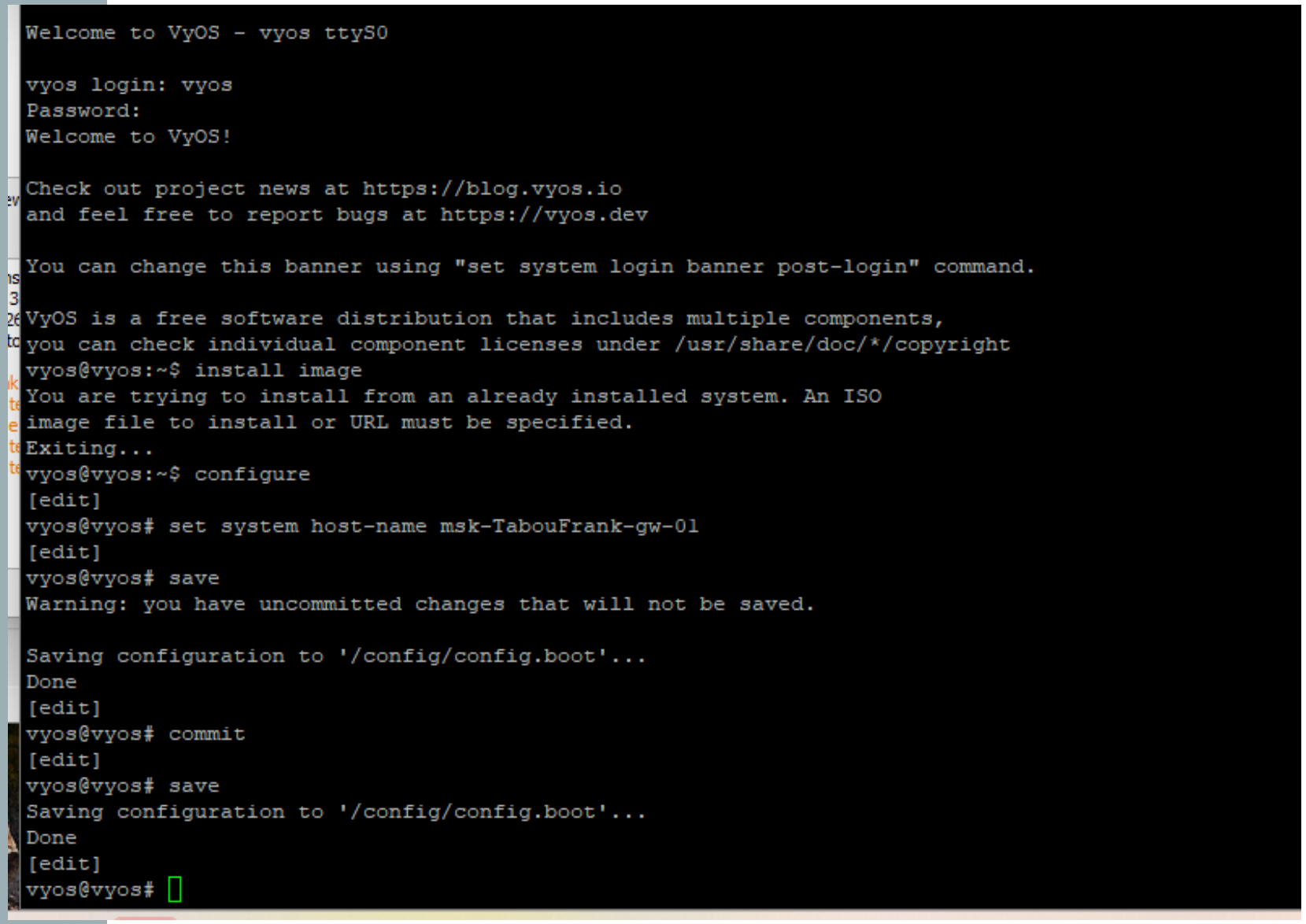
ВЫПОЛНЕНИЕ ЛАБОРАТОРНОЙ РАБОТЫ



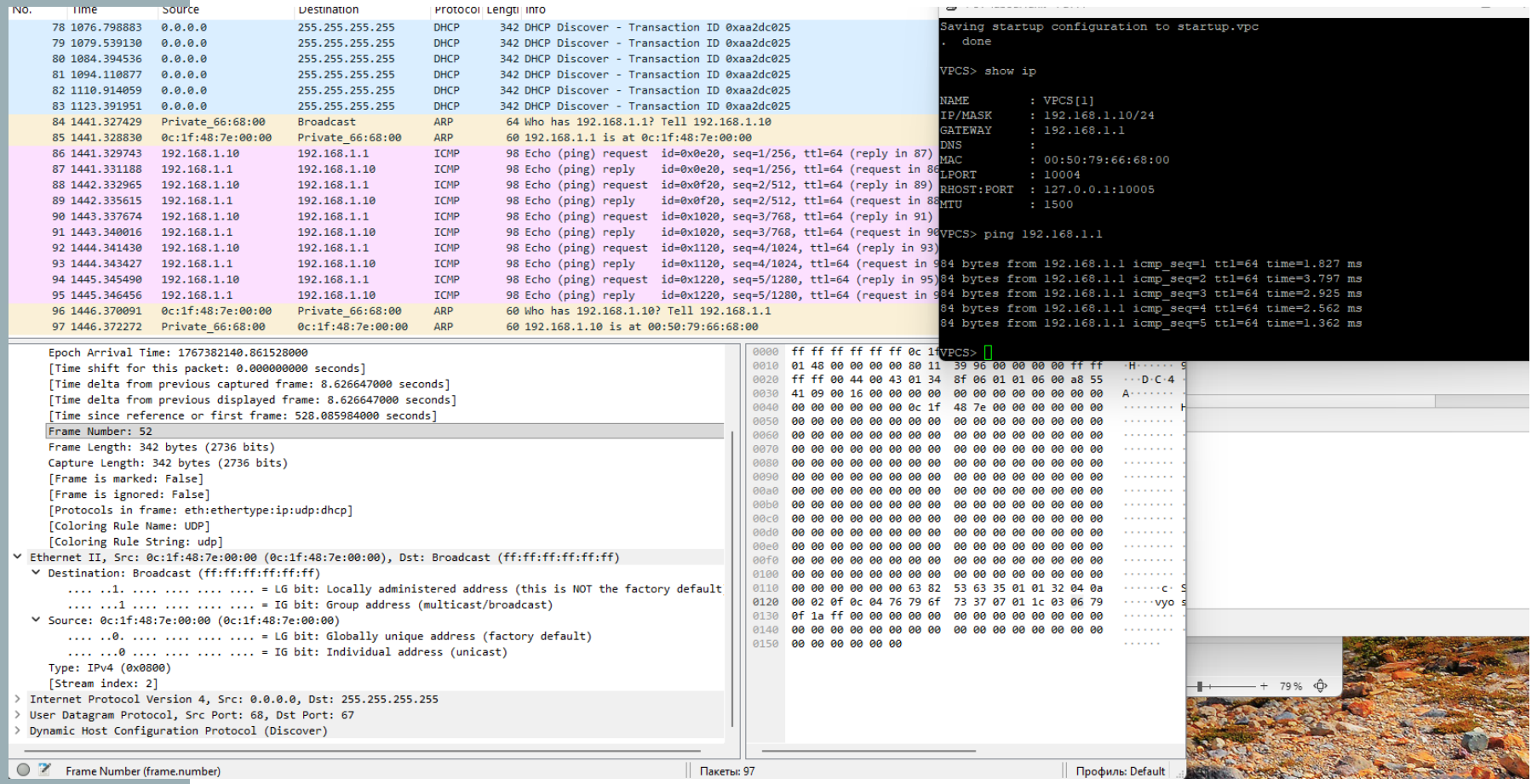
ВЫПОЛНЕНИЕ ЛАБОРАТОРНОЙ РАБОТЫ



ВЫПОЛНЕНИЕ ЛАБОРАТОРНОЙ РАБОТЫ



ВЫПОЛНЕНИЕ ЛАБОРАТОРНОЙ РАБОТЫ



ВЫВОДЫ

В процессе выполнения данной лабораторной работы я построил простейшие модели сети на базе коммутатора и маршрутизатора VyOS в GNS3, проанализировал трафик посредством Wireshark.