

Министерство науки и высшего образования Российской Федерации
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО

Факультет ФТИИ

Дисциплина: «Сетевые технологии»

ПРАКТИЧЕСКАЯ РАБОТА № 2

**«Мониторинг сетевого трафика на хосте на примере работы с утилитами
диагностики и мониторинга сетевых соединений в Linux»**

Выполнил:

Тиганов Вадим Игоревич, студент группы J3212
ИСУ: 467701

Проверила:

Шиманская Галина Станиславовна

Санкт-Петербург
2025

Содержание

1 Цель работы	2
2 Артефакты выполнения	2
2.1 Часть 1. Настройка инфраструктуры	2
2.2 Часть 2. Диагностика соединения	2
2.3 Часть 3. Работа с Wireshark	2
2.4 Часть 4. Определение маршрута прохождения пакета	4
2.5 Часть 5. Текущий мониторинг сетевых интерфейсов	4
2.6 Часть 6. Диагностика работы приложений через сеть	5
3 Ответы на вопросы и задания	7
4 Использование GAI	9
5 Рефлексия	9
6 Список использованных источников	10

1 Цель работы

Получить практические навыки мониторинга сетевого трафика и диагностирования сетевых соединений на хосте Linux с использованием консольных утилит: `ss`, `lsof`, `tcpdump/tshark`, `iftop/nload/bmon`, а также базовых средств `ip` и `ethtool`.

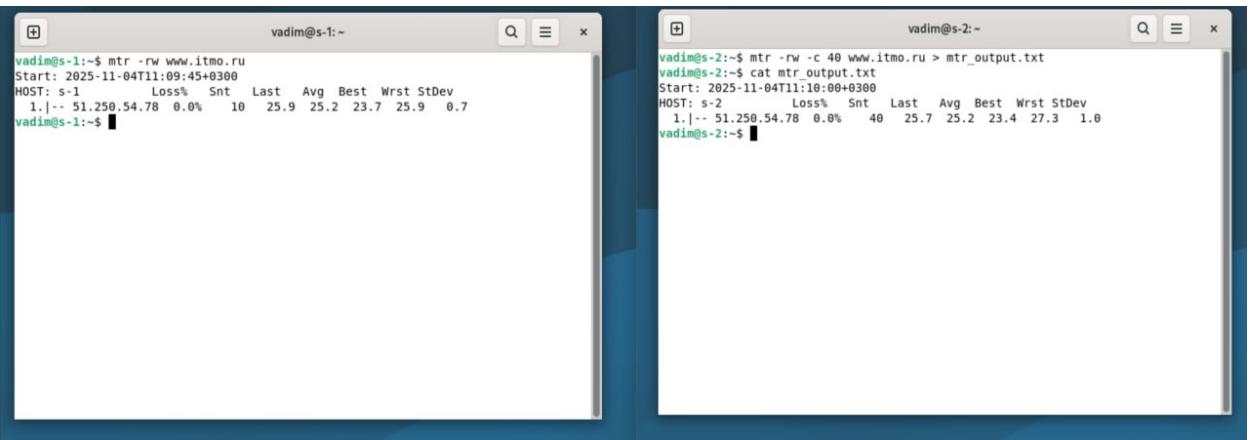
2 Артефакты выполнения

2.1 Часть 1. Настройка инфраструктуры

ВМ c7-1/c7-2, NAT в VirtualBox, DHCP-адреса, проверка ping внешней сети, установка утилит (bmon/nload/iftop, nethogs, mtr, traceroute, vnstat, nc)

2.2 Часть 2. Диагностика соединения

Команды и выводы по ping, запуск `mtr` к `www.itmo.ru`, сохранение расширенной статистики на 40 пакетов:



The image shows two terminal windows side-by-side. The left window is titled 'vadim@s-1:~' and displays the command 'mtr -rw www.itmo.ru' followed by its output. The output includes the start time (2025-11-04T11:09:45+0300), host information (HOST: s-1), and a table of statistics. The right window is titled 'vadim@s-2:~' and shows the command 'mtr -rw -c 40 www.itmo.ru > mtr_output.txt' followed by the command 'cat mtr_output.txt'. This output is identical to the one in the left window, showing the start time (2025-11-04T11:18:00+0300), host information (HOST: s-2), and a table of statistics.

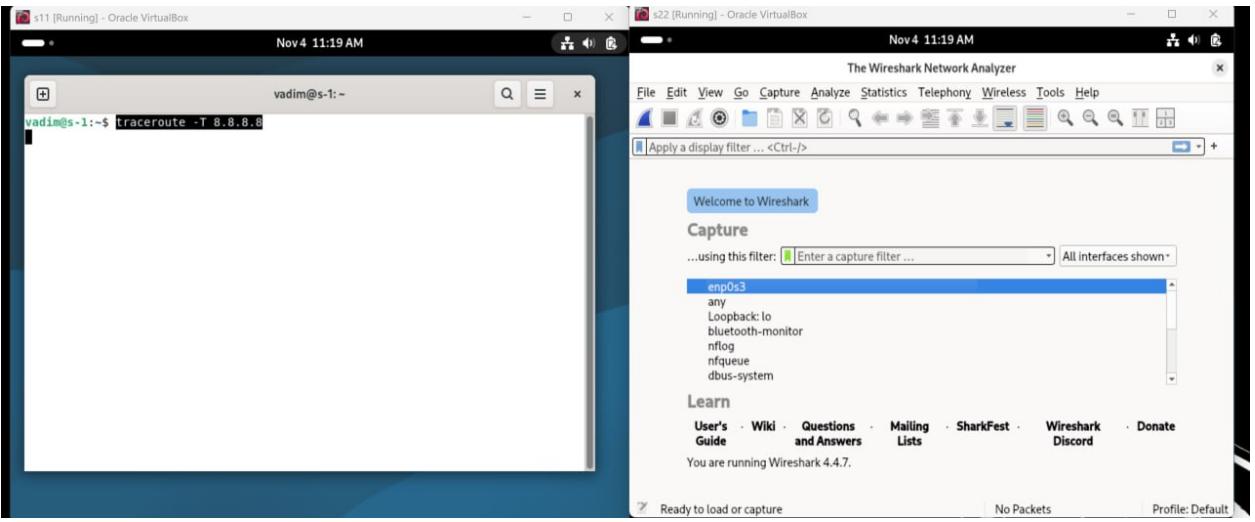
```
vadim@s-1:~$ mtr -rw www.itmo.ru
Start: 2025-11-04T11:09:45+0300
HOST: s-1          Loss% Snt Last Avg Best Wrst StDev
1.|-- 51.250.54.78 0.0%   10  25.9  25.2  23.7  25.9  0.7
vadim@s-1:~$ 

vadim@s-2:~$ mtr -rw -c 40 www.itmo.ru > mtr_output.txt
vadim@s-2:~$ cat mtr_output.txt
Start: 2025-11-04T11:18:00+0300
HOST: s-2          Loss% Snt Last Avg Best Wrst StDev
1.|-- 51.250.54.78 0.0%   40  25.7  25.2  23.4  27.3  1.0
vadim@s-2:~$
```

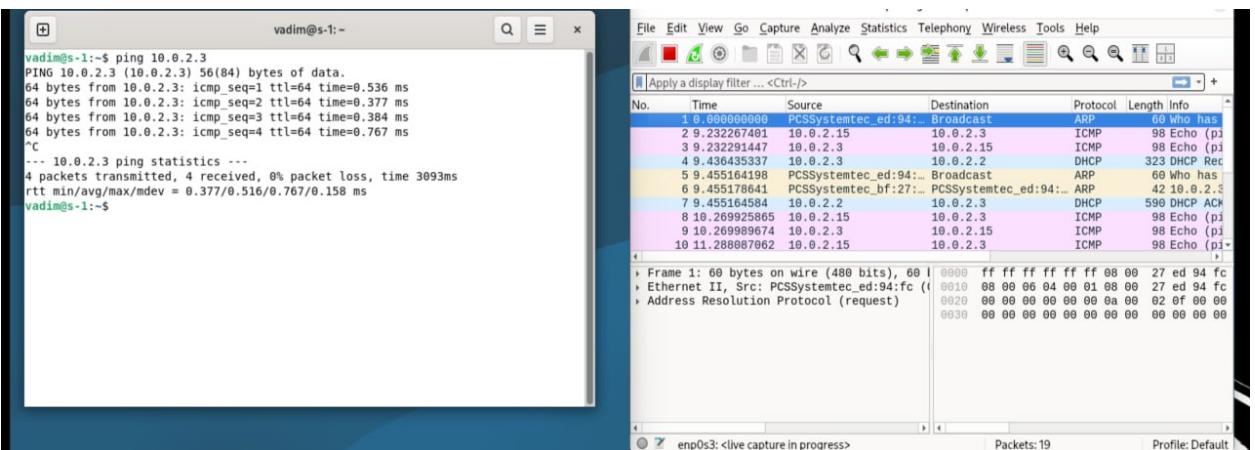
Рис. 1: Вывод команды ping

2.3 Часть 3. Работа с Wireshark

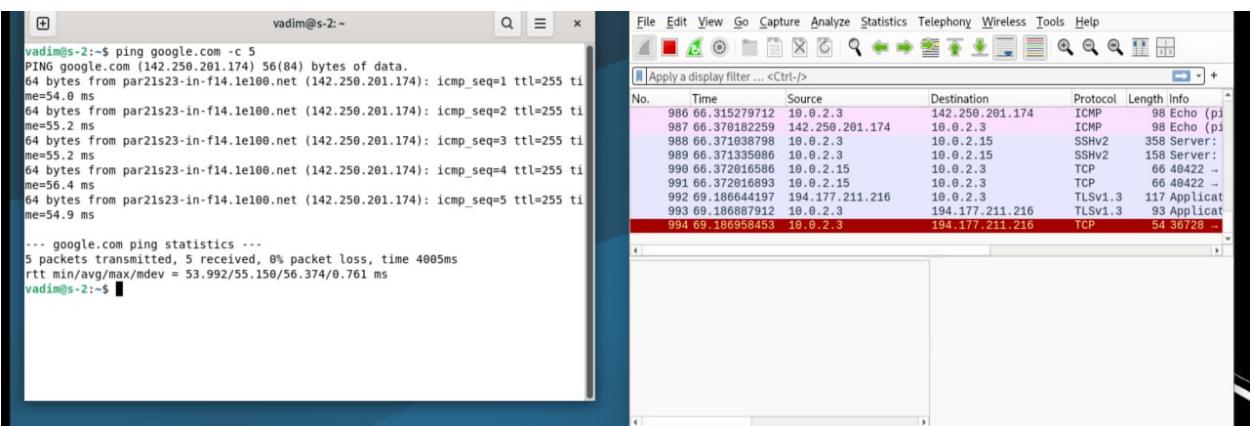
Настройка захвата (ограничение 5 МБ), статистика — самый активный узел, широковещательный «говорун», самый активный TCP-порт; графики Io Graphs (TCP/UDP вместе); Flow Graph по HTTPS; фильтры отображения для DNS/кадров хоста/широковещаний:



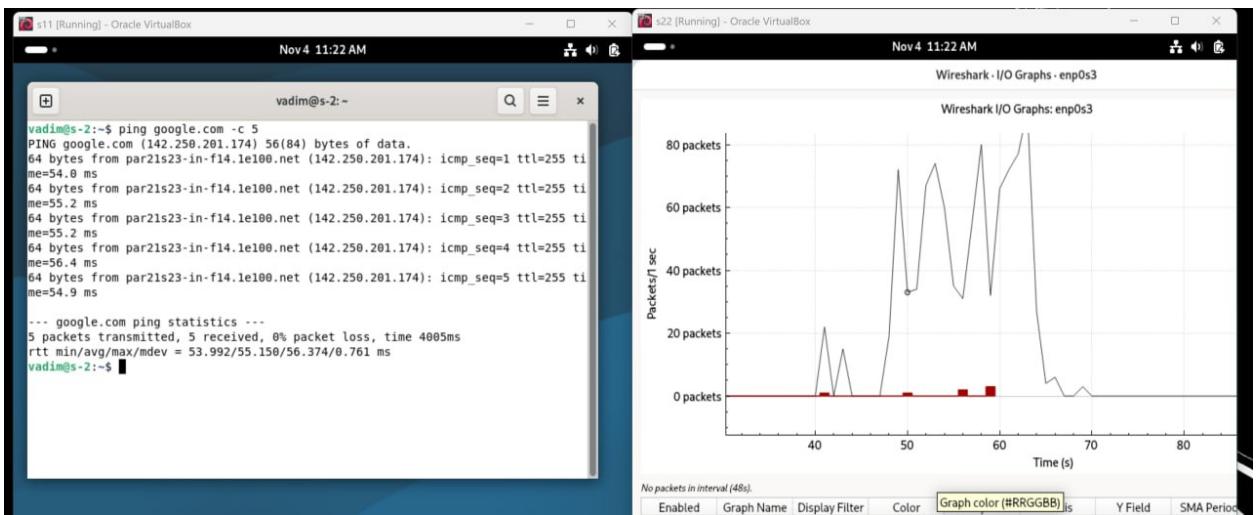
wireshark 1



wireshark 2



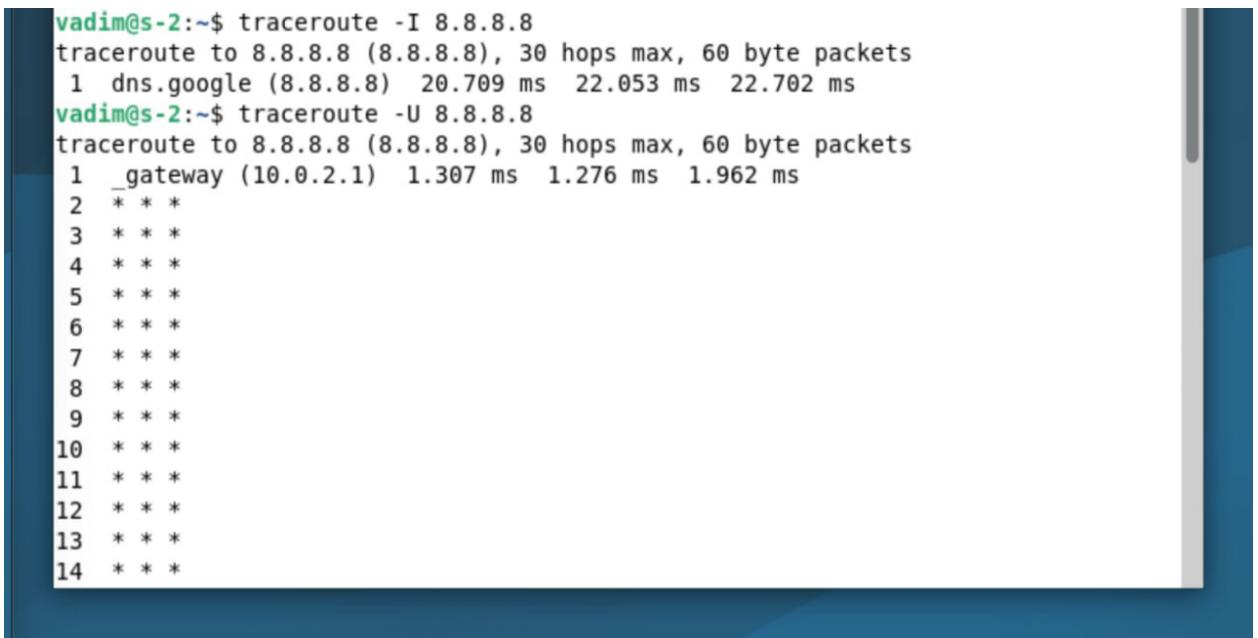
wireshark 3



wireshark 4

2.4 Часть 4. Определение маршрута прохождения пакета

Место для: команды *traceroute* с ICMP/UDP/TCP до 8.8.8.8 и проверка фрагментации IPv4 — команды и выводы



traceroute 8.8.8.8

2.5 Часть 5. Текущий мониторинг сетевых интерфейсов

Место для: ping -f с c7-2 на внутренний интерфейс c7-1;

```

vadim@s-1:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
    qlen 1000
        link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
        inet 127.0.0.1/8 scope host lo
            valid_lft forever preferred_lft forever
            inet6 ::1/128 scope host noprefixroute
                valid_lft forever preferred_lft forever
2: enp0s3: <NOARP,BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default
    qlen 1000
        link/ether 08:00:27:3e:8f:1b brd ff:ff:ff:ff:ff:ff
        altname enx0800273e8f1b
        inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic noprefixroute enp0s3
            valid_lft 589sec preferred_lft 589sec
            inet6 fd17:625c:f037:2:95aa:581b:eb61:f254/64 scope global temporary dynamic
                valid_lft 86189sec preferred_lft 14189sec
                inet6 fd17:625c:f037:2:a00:27ff:fe3e:8f1b/64 scope global dynamic mngtmpaddr
                    valid_lft 86189sec preferred_lft 14189sec
                    inet6 fe80::a00:27ff:fe3e:8f1b/64 scope link noprefixroute
                        valid_lft forever preferred_lft forever
vadim@s-1:~$ 

vadim@s-2:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
    qlen 1000
        link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
        inet 127.0.0.1/8 scope host lo
            valid_lft forever preferred_lft forever
            inet6 ::1/128 scope host noprefixroute
                valid_lft forever preferred_lft forever
2: enp0s3: <NOARP,BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default
    qlen 1000
        link/ether 08:00:27:b7:2f:7d brd ff:ff:ff:ff:ff:ff
        altname enx080027b72f7d
        inet 10.0.2.3/24 brd 10.0.2.255 scope global dynamic noprefixroute enp0s3
            valid_lft 595sec preferred_lft 595sec
            inet6 fe80::a00:27ff:feb:277d/64 scope link noprefixroute
                valid_lft forever preferred_lft forever
vadim@s-2:~$ 

```

Проверим ip адреса

```

vadim@s-1:~$ ping 10.0.2.3 -c 2
PING 10.0.2.3 (10.0.2.3) 56(84) bytes of data.
64 bytes from 10.0.2.3: icmp_seq=1 ttl=64 time=0.479 ms
64 bytes from 10.0.2.3: icmp_seq=2 ttl=64 time=0.705 ms
--- 10.0.2.3 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1030ms
rtt min/avg/max/mdev = 0.479/0.592/0.705/0.113 ms
vadim@s-1:~$ 

vadim@s-2:~$ ping 10.0.2.15 -c 2
PING 10.0.2.15 (10.0.2.15) 56(84) bytes of data.
64 bytes from 10.0.2.15: icmp_seq=1 ttl=64 time=0.530 ms
64 bytes from 10.0.2.15: icmp_seq=2 ttl=64 time=0.448 ms
--- 10.0.2.15 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1026ms
rtt min/avg/max/mdev = 0.448/0.489/0.530/0.041 ms
vadim@s-2:~$ 

```

Пингуем для проверки соединения

2.6 Часть 6. Диагностика работы приложений через сеть

SSH-сессии; вывод `netstat/ss` — прослушиваемые порты и установленные соединения; скрипт агрегации соединений по порту (по умолчанию 22); закрытие сессий — вывод;

```

vadim@s-1:~$ ping 10.0.2.3
PING 10.0.2.3 (10.0.2.3) 56(84) bytes of data.
64 bytes from 10.0.2.3: icmp_seq=1 ttl=64 time=2.02 ms
64 bytes from 10.0.2.3: icmp_seq=2 ttl=64 time=2.33 ms
64 bytes from 10.0.2.3: icmp_seq=3 ttl=64 time=0.797 ms
64 bytes from 10.0.2.3: icmp_seq=4 ttl=64 time=1.09 ms
64 bytes from 10.0.2.3: icmp_seq=5 ttl=64 time=0.794 ms
64 bytes from 10.0.2.3: icmp_seq=6 ttl=64 time=2.92 ms
64 bytes from 10.0.2.3: icmp_seq=7 ttl=64 time=1.23 ms
64 bytes from 10.0.2.3: icmp_seq=8 ttl=64 time=0.499 ms
64 bytes from 10.0.2.3: icmp_seq=9 ttl=64 time=1.22 ms
64 bytes from 10.0.2.3: icmp_seq=10 ttl=64 time=0.877 ms
64 bytes from 10.0.2.3: icmp_seq=11 ttl=64 time=0.736 ms
64 bytes from 10.0.2.3: icmp_seq=12 ttl=64 time=0.871 ms
64 bytes from 10.0.2.3: icmp_seq=13 ttl=64 time=0.375 ms
64 bytes from 10.0.2.3: icmp_seq=14 ttl=64 time=0.334 ms
64 bytes from 10.0.2.3: icmp_seq=15 ttl=64 time=0.967 ms
64 bytes from 10.0.2.3: icmp_seq=16 ttl=64 time=0.658 ms
64 bytes from 10.0.2.3: icmp_seq=17 ttl=64 time=2.09 ms
64 bytes from 10.0.2.3: icmp_seq=18 ttl=64 time=1.38 ms
64 bytes from 10.0.2.3: icmp_seq=19 ttl=64 time=0.894 ms
64 bytes from 10.0.2.3: icmp_seq=20 ttl=64 time=0.802 ms
64 bytes from 10.0.2.3: icmp_seq=21 ttl=64 time=1.06 ms
64 bytes from 10.0.2.3: icmp_seq=22 ttl=64 time=0.465 ms

vadim@s-2:~$ sudo tcpdump icmp
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on enp0s3, link-type EN10MB (Ethernet), snapshot length 262144 bytes
10:49:45.886724 IP 10.0.2.15 > s-2: ICMP echo request, id 5, seq 338, length 64
10:49:45.886798 IP s-2 > 10.0.2.15: ICMP echo reply, id 5, seq 338, length 64
10:49:46.890873 IP 10.0.2.15 > s-2: ICMP echo request, id 5, seq 339, length 64
10:49:46.890917 IP s-2 > 10.0.2.15: ICMP echo reply, id 5, seq 339, length 64
10:49:47.893690 IP 10.0.2.15 > s-2: ICMP echo request, id 5, seq 340, length 64
10:49:47.893704 IP s-2 > 10.0.2.15: ICMP echo reply, id 5, seq 340, length 64
10:49:49.037557 IP 10.0.2.15 > s-2: ICMP echo request, id 5, seq 341, length 64
10:49:49.037571 IP s-2 > 10.0.2.15: ICMP echo reply, id 5, seq 341, length 64
10:49:50.055622 IP 10.0.2.15 > s-2: ICMP echo request, id 5, seq 342, length 64
10:49:50.055645 IP s-2 > 10.0.2.15: ICMP echo reply, id 5, seq 342, length 64
10:49:51.065089 IP 10.0.2.15 > s-2: ICMP echo request, id 5, seq 343, length 64
10:49:51.065105 IP s-2 > 10.0.2.15: ICMP echo reply, id 5, seq 343, length 64
10:49:52.123518 IP 10.0.2.15 > s-2: ICMP echo request, id 5, seq 344, length 64
10:49:52.123526 IP s-2 > 10.0.2.15: ICMP echo reply, id 5, seq 344, length 64
10:49:53.171303 IP 10.0.2.15 > s-2: ICMP echo request, id 5, seq 345, length 64
10:49:54.174199 IP 10.0.2.15 > s-2: ICMP echo reply, id 5, seq 346, length 64
10:49:55.174253 IP s-2 > 10.0.2.15: ICMP echo reply, id 5, seq 346, length 64
10:49:55.234245 IP 10.0.2.15 > s-2: ICMP echo request, id 5, seq 347, length 64
10:49:55.234269 IP s-2 > 10.0.2.15: ICMP echo reply, id 5, seq 347, length 64
10:49:56.258230 IP 10.0.2.15 > s-2: ICMP echo request, id 5, seq 348, length 64

```

tcpdump

s11 [Running] - Oracle VirtualBox

vadim@s-1:~\$ ping 10.0.2.3

PING 10.0.2.3 (10.0.2.3) 56(84) bytes of data.

64 bytes from 10.0.2.3: icmp_seq=1 ttl=64 time=0.409 ms
 64 bytes from 10.0.2.3: icmp_seq=2 ttl=64 time=1.33 ms
 64 bytes from 10.0.2.3: icmp_seq=3 ttl=64 time=0.752 ms
 64 bytes from 10.0.2.3: icmp_seq=4 ttl=64 time=0.959 ms
 64 bytes from 10.0.2.3: icmp_seq=5 ttl=64 time=0.612 ms
 64 bytes from 10.0.2.3: icmp_seq=6 ttl=64 time=0.730 ms
 64 bytes from 10.0.2.3: icmp_seq=7 ttl=64 time=0.916 ms
 64 bytes from 10.0.2.3: icmp_seq=8 ttl=64 time=0.611 ms
 64 bytes from 10.0.2.3: icmp_seq=9 ttl=64 time=1.02 ms
 64 bytes from 10.0.2.3: icmp_seq=10 ttl=64 time=0.789 ms
 64 bytes from 10.0.2.3: icmp_seq=11 ttl=64 time=1.08 ms
 64 bytes from 10.0.2.3: icmp_seq=12 ttl=64 time=0.845 ms
 64 bytes from 10.0.2.3: icmp_seq=13 ttl=64 time=0.790 ms
 64 bytes from 10.0.2.3: icmp_seq=14 ttl=64 time=0.394 ms
 64 bytes from 10.0.2.3: icmp_seq=15 ttl=64 time=0.497 ms
 64 bytes from 10.0.2.3: icmp_seq=16 ttl=64 time=0.777 ms
 64 bytes from 10.0.2.3: icmp_seq=17 ttl=64 time=1.85 ms
 64 bytes from 10.0.2.3: icmp_seq=18 ttl=64 time=0.577 ms
 64 bytes from 10.0.2.3: icmp_seq=19 ttl=64 time=2.16 ms
 64 bytes from 10.0.2.3: icmp_seq=20 ttl=64 time=2.58 ms
 64 bytes from 10.0.2.3: icmp_seq=21 ttl=64 time=0.345 ms
 64 bytes from 10.0.2.3: icmp_seq=22 ttl=64 time=0.646 ms

s22 [Running] - Oracle VirtualBox

vadim@s-2:~\$

Device enp0s3 [10.0.2.3] (1/2):

Incoming:

Curr: 1.23 kBit/s
 Avg: 888.00 Bit/s
 Min: 0.00 Bit/s
 Max: 1.23 kBit/s
 Ttl: 59.17 MByte

Outgoing:

Curr: 1.23 kBit/s
 Avg: 888.00 Bit/s
 Min: 0.00 Bit/s
 Max: 1.23 kBit/s
 Ttl: 485.45 KByte

netstat

vadim@s-2:~\$

Interfaces	RX bps	pps	%	TX bps	pps	%
>lo	0	0		0	0	
qdisc none (noqueue)	0	0		0	0	
enp0s3	98B	1		98B	1	
qdisc none (fq_codel)	0	0		98B	1	

Increase screen height to see graphical statistics
 Increase screen height to see detailed statistics
 Increase screen height to see additional information

Tue Nov 4 10:54:01 2025 Press ? for help

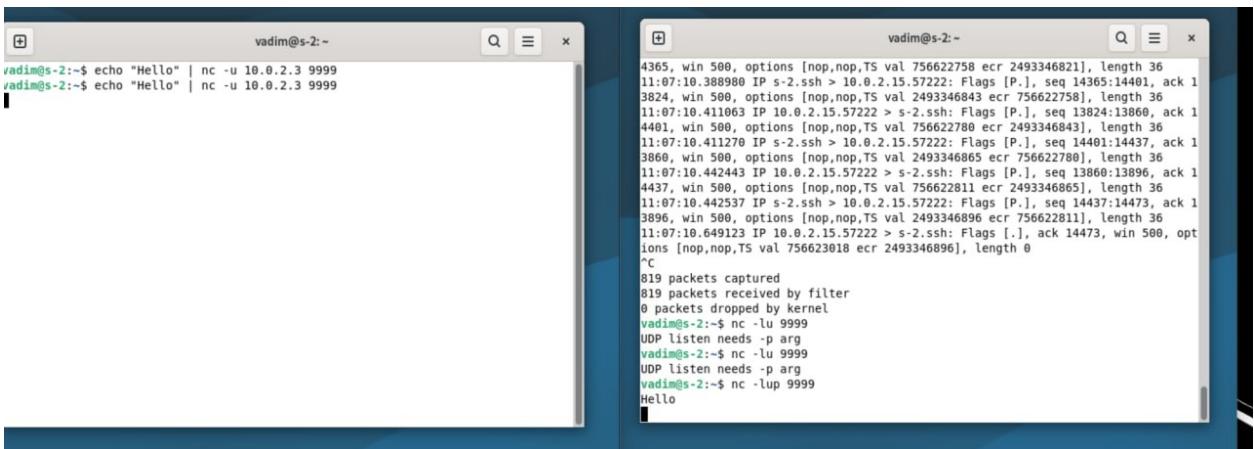
bmon

vadim@s-2:~

	12.5Kb	25.0Kb	37.5Kb	50.0Kb	62.5Kb
s-2	=> 10.0.2.15		336b	420b	420b
	<=		336b	420b	420b
s-2	=> router.asus.com		0b	138b	138b
	<=		0b	167b	167b

TX:	cum:	558B	peak:	1.20Kb	rates:	336b	558b	558b
RX:		587B		1.30Kb		336b	587b	587b
TOTAL:		1.12KB		2.50Kb		672b	1.12Kb	1.12Kb

Дополнительный мониторинг



```
vadim@s-2:~$ echo "Hello" | nc -u 10.0.2.3 9999
vadim@s-2:~$ echo "Hello" | nc -u 10.0.2.3 9999
```

```
4365, win 500, options [nop,nop,TS val 756622758 ecr 2493346821], length 36
11:07:10.388980 IP s-2.ssh > 10.0.2.15.57222: Flags [.], seq 14365:14401, ack 1
3824, win 500, options [nop,nop,TS val 2493346843 ecr 756622758], length 36
11:07:10.411063 IP 10.0.2.15.57222 > s-2.ssh: Flags [.], seq 13824:13860, ack 1
4481, win 500, options [nop,nop,TS val 756622788 ecr 2493346843], length 36
11:07:10.411279 IP s-2.ssh > 10.0.2.15.57222: Flags [.], seq 14401:14437, ack 1
3860, win 500, options [nop,nop,TS val 2493346865 ecr 756622780], length 36
11:07:10.442443 IP 10.0.2.15.57222 > s-2.ssh: Flags [.], seq 13860:13896, ack 1
4437, win 500, options [nop,nop,TS val 756622811 ecr 2493346865], length 36
11:07:10.442537 IP s-2.ssh > 10.0.2.15.57222: Flags [.], seq 14437:14473, ack 1
3896, win 500, options [nop,nop,TS val 2493346896 ecr 756622811], length 36
11:07:10.649123 IP 10.0.2.15.57222 > s-2.ssh: Flags [.], ack 14473, win 500, options [nop,nop,TS val 756623018 ecr 2493346896], length 0
^C
819 packets captured
819 packets received by filter
0 packets dropped by kernel
vadim@s-2:~$ nc -lU 9999
UDP listen needs -p arg
vadim@s-2:~$ nc -lU 9999
UDP listen needs -p arg
vadim@s-2:~$ nc -lU 9999
Hello
```

Отправим Hello World по SSH

3 Ответы на вопросы и задания

- По какому протоколу работает утилита mtr? Как это можно определить?

По умолчанию mtr в Linux использует **ICMP Echo** (Echo Request/Reply) аналогично traceroute -I. Определить можно по захвату tcpdump/wireshark (видны ICMP Echo), либо явно переключая режимы -u (UDP) и -T (TCP) и наблюдая изменение типа пакетов.

2. Опишите значения столбцов статистики, выводимой утилитой mtr. Какие еще статистики доступны в mtr кроме основных?

Базовые столбцы: *Loss%* — потери по хопу; *Snt* — число зондов; *Last* — время последнего ответа; *Avg* — средняя RTT; *Best* — минимальная RTT; *Wrst* — максимальная RTT; *StDev* — стандартное отклонение. Дополнительно доступны отчеты (*-report*, *-report-cycles*), форматы *-json/-xml*, отображение *-show-ips/AS/GeoIP*, а также *Jitter* в некоторых сборках.

3. Какие типы кадров Ethernet бывают, в чем их отличия?

Ethernet II (поле EtherType, наиболее распространен); *IEEE 802.3 LLC* (длина + LLC заголовок); *802.3 SNAP* (LLC+SNAP для индикации протокола); *802.1Q VLAN/QinQ* (теги VLAN); служебные кадры типа *PAUSE 802.3x*, *LLDP*. Отличаются форматом полей заголовка и наличием тегов/LLC.

4. На какие адреса сетевого уровня осуществляются широковещательные рассылки?

В IPv4: **255.255.255.255** (ограниченный широковещательный) и **направленный широковещательный** адрес сети (например, 192.168.1.255/24). В IPv6 широковещания нет — используется многоадресная рассылка (**ff00::/8**).

5. На какой канальный адрес осуществляются широковещательные рассылки?

На MAC-адрес **ff:ff:ff:ff:ff:ff**.

6. Для чего применяются перехваченные широковещательные рассылки в Части 3?

Примеры: *ARP Request* (разрешение IP→MAC), *DHCP Discover/Offer* (получение параметров IP-сети), *mDNS/LLMNR/NBNS* (локальное разрешение имён), а также сервисные рассылки (например, *STP/LLDP*). Требовалось определить назначение минимум трёх таких рассылок.

7. Как изменяется загрузка интерфейса в Части 5. п. 3? Почему?

При увеличении размера ICMP-пакета (100→60100 байт) **доля заголовочного оверхеда уменьшается**, а **битовая загрузка** интерфейса при flood растёт до упора в пропускную способность/ограничения ядра; частота пакетов (pps) падает. Итог: использование канала увеличивается примерно пропорционально полезной нагрузке до достижения пределов CPU/линейной скорости.

8. На каком уровне модели OSI работает vnstat?

vnstat собирает счётчики интерфейсов из ядра (*/proc/net/dev*), то есть оперирует **на канальном уровне (L2)** независимо от протоколов L3/L4.

4 Использование GAI

Был ли использован в ходе выполнения практической работы GAI (ChatGPT, YandexGPT и др.)?

Да, для помощи в структуре отчета, формулировках теоретических ответов и оформления LaTeX.

Цели использования:

- Подготовка шаблона разделов и заглушек под артефакты
- Вычитка и сжатие теоретических формулировок

Оценка качества ответов моделей:

Качество ответов ChatGPT было высоким. Модель:

- Помогла структурировать материал и сделать листинг более читаемым
- Дала корректные примеры фильтров и ключей для утилит (особенно для wireshark)

5 Рефлексия

Что вы узнали нового из работы? Как, по-вашему, эти знания или навыки могут пригодиться в будущей профессиональной деятельности?

В ходе выполнения данной лабораторной работы я приобрел следующие знания и навыки:

Новые знания:

- Различия между инвентаризацией сокетов (`ss/lssof`) и пакетным анализом (`tcpdump/tshark`)
- Интерпретация состояний TCP и счетчиков интерфейса
- Подходы к фильтрации трафика с помощью BPF-выражений

Практические навыки:

- Быстрая диагностика «что слушает порт» и «кто держит соединение»
- Прицельный захват и разбор трафика, запись/чтение PCAP
- Анализ пропускной способности и выявление узких мест по интерфейсам

Применение в профессиональной деятельности:

- **Системное администрирование:** оперативная диагностика сетевых инцидентов
- **DevOps/SRE:** воспроизведение проблем, профилирование сетевых зависимостей сервисов
- **Безопасность:** сетевой Threat Hunting и анализ аномалий

6 Список использованных источников

1. man-pages: `ss(8)`, `tcpdump(8)`, `tshark(1)`, `lsof(8)`, `ip-link(8)`, `ethtool(8)`
2. <https://www.tcpdump.org/manpages/tcpdump.1.html>
3. https://www.wireshark.org/docs/wsug_html_chunked/
4. <https://www.kernel.org/doc/Documentation/networking/>