

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение высшего
образования

Санкт-Петербургский национальный исследовательский университет ИТМО

Мегафакультет трансляционных информационных технологий

Факультет информационных технологий и программирования

**Практическая работа №1. Консольные утилиты настройки сетевых
компонентов Linux**

По дисциплине «Телекоммуникационные системы и технологии»

Выполнил:

студент группы №М3306

Тимофеев Вячеслав

Проверил:

Самигуллин



УНИВЕРСИТЕТ ИТМО

Санкт-Петербург
2025

Цель работы: получить практические навыки по конфигурированию сетевых интерфейсов (на примере протокола IPv4) в операционных системах Linux, ознакомиться с утилитами командной строки, освоить современные сетевые менеджеры Linux

Артефакты выполнения:

Часть 1

Скрипт ч.1-п.2:

```
#!/bin/bash

get_device() {
    ip link show | grep -E "^[0-9]+: enp.*state UP" | awk -F: '{print $2}' | head
-1 | sed 's/^ *//'
}

get_net_info() {
    network_iface_name=$device
    mac_address=$(cat /sys/class/net/$device/address 2>/dev/null)
    speed=$(ethtool $device 2>/dev/null | grep "Speed:" | awk '{print $2}')
    duplex=$(ethtool $device 2>/dev/null | grep "Duplex:" | awk '{print $2}')
    link=$(ethtool $device 2>/dev/null | grep "Link detected:" | awk '{print
$3}')

    echo "Модель сетевой карты: $network_iface_name
        Канальная скорость: $speed
        Режим работы Duplex: $duplex
        Наличие физического подключения: $link
        MAC-адрес: $mac_address"
}

get_ipv4_config() {
    ip_addr=$(ip addr show $device | grep "inet " | awk '{print $2}')
    gateway=$(ip route | grep "default via" | grep $device | awk '{print $3}')
    dns=$(grep nameserver /etc/resolv.conf 2>/dev/null | head -1 | awk '{print
$2}')

    echo "IP4.ADDRESS: $ip_addr"
    echo "IP4.GATEWAY: $gateway"
    echo "IP4.DNS:      $dns"
}

setup_static_addressing() {
    sudo ip addr flush dev $device 2>/dev/null
    sudo ip route flush dev $device 2>/dev/null
    sudo ip addr add 10.100.0.2/24 dev $device
    sudo ip link set $device up
    sudo ip route add default via 10.100.0.1 dev $device

    echo "Успешно установлен статический адрес"
```

```

    get_ipv4_config
}

setup_dynamic_addressing() {
    sudo ip addr flush dev $device 2>/dev/null
    sudo ip route flush dev $device 2>/dev/null
    sudo dhclient $device 2>/dev/null &
    sleep 2

    echo "Успешно установлен динамический адрес"
    get_ipv4_config
}

help="СПРАВКА
a) Узнать модель сетевой карты, канальную скорость, режим работы (Full
или Half Duplex). Наличие физического подключения(линк), MAC адрес
b) Получить информацию о текущей конфигурации IPv4 (ip, mask, gate, dns)
c) Настроить сетевой интерфейс - Статическая адресация IP=10.100.0.2,
MASK=255.255.255.0, GATE=10.100.0.1, DNS = 8.8.8.8
d) Настроить сетевой интерфейс - Динамическая адресация
e) Закрыть скрипт
h) Показать справку
"

device=$(get_device) || { echo "Не удалось определить девайс"; exit 1; }
echo "$help"
while true;
do
    read -r input
    case $input in
        a) get_net_info ;;
        b) get_ipv4_config ;;
        c) setup_static_addressing ;;
        d) setup_dynamic_addressing ;;
        e) echo "Завершение..."; break ;;
        h) echo "$help" ;;
        *) echo "Нет такой опции" ;;
    esac
done

```

СПРАВКА

a) Узнать модель сетевой карты, канальную скорость, режим работы (Full или Half Duplex). Наличие физического подключения(линк), MAC адрес
b) Получить информацию о текущей конфигурации IPv4 (ip, mask, gate, dns)
c) Настроить сетевой интерфейс - Статическая адресация IP=10.100.0.2, MASK=255.255.255.0, GATE=10.100.0.1, DNS = 8.8.8.8
d) Настроить сетевой интерфейс - Динамическая адресация
e) Закрыть скрипт
h) Показать справку

```
Модель сетевой карты: enp0s3
Канальная скорость: 1000Mb/s
Режим работы Duplex: Full
Наличие физического подключения: yes
MAC-адрес: 08:00:27:c3:04:a5
```

Статический адрес:

```
Успешно установлен статический адрес
IP4.ADDRESS: 10.100.0.2/24
IP4.GATEWAY: 10.100.0.1
IP4.DNS: 1.1.1.1
```

Динамический адрес:

```
Успешно установлен динамический адрес
IP4.ADDRESS: 10.0.2.15/24
IP4.GATEWAY: 10.0.2.2
IP4.DNS: 1.1.1.1
```

Часть 2

Настройка сетевого интерфейса:

```
[root@localhost ~]# nmcli con mod enp0s3 ipv4.dns 8.8.8.8
[root@localhost ~]# nmcli con mod enp0s3 ipv4.addresses 10.100.0.2/24
[root@localhost ~]# nmcli con mod enp0s3 ipv4.gateway 10.100.0.1
[root@localhost ~]# nmcli con mod enp0s3 ipv4.method manual
[root@localhost ~]# nmcli con up enp0s3
Подключение успешно активировано (активный путь D-Bus: /org/freedesktop/NetworkManager/ActiveConnection/4)
[root@localhost ~]# nmcli con show
NAME      UUID                                  TYPE      DEVICE
enp0s3    880a7420-0768-32d1-a9fd-cef142b4fd49 ethernet  enp0s3
lo        31a436bc-95e8-437a-82ad-37558d29c6a6 loopback   lo
```

Проверка:

```
[root@localhost ~]# nmcli device show enp0s3
GENERAL.DEVICE: enp0s3
GENERAL.TYPE: ethernet
GENERAL.HWADDR: 08:00:27:9A:A4:EF
GENERAL.MTU: 1500
GENERAL.STATE: 100 (подключено)
GENERAL.CONNECTION: enp0s3
GENERAL.CON-PATH: /org/freedesktop/NetworkManager/ActiveConnection/5
WIRED-PROPERTIES.CARRIER: вкл
IP4.ADDRESS[1]: 10.100.0.2/24
IP4.GATEWAY: 10.100.0.1
IP4.ROUTE[1]: dst = 10.100.0.0/24, nh = 0.0.0.0, mt = 100
IP4.ROUTE[2]: dst = 0.0.0.0/0, nh = 10.100.0.1, mt = 100
IP4.DNS[1]: 8.8.8.8
IP6.ADDRESS[1]: fe80::a00:27ff:fe9a:a4ef/64
IP6.GATEWAY: --
IP6.ROUTE[1]: dst = fe80::/64, nh = ::, mt = 1024
```

Создание и активация виртуального сетевого интерфейса (ч.2-п.2):

```

[root@localhost ~]# nmcli con add type bridge ifname br0 con-name br0 ip4 10.100.0.3/24
[ 2383.625986] bridge: filtering via arp/ip6tables is no longer available by default. Update your scripts to load br_netfilter if you need this.
Подключение «br0» (51b48052-86b1-4322-ba00-2e24db5fe996) успешно добавлено.
[root@localhost ~]# nmcli connection up br0
Подключение успешно активировано (controller waiting for ports) (активный путь D-Bus: /org/freedesktop/NetworkManager/ActiveConnection/6)

```

Проверка связи между реальным и виртуальным сетевыми интерфейсами:

```

[root@localhost ~]# ping -c 3 10.100.0.3
PING 10.100.0.3 (10.100.0.3) 56(84) bytes of data.
64 bytes from 10.100.0.3: icmp_seq=1 ttl=64 time=0.093 ms
64 bytes from 10.100.0.3: icmp_seq=2 ttl=64 time=0.095 ms
64 bytes from 10.100.0.3: icmp_seq=3 ttl=64 time=0.107 ms

--- 10.100.0.3 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2088ms
rtt min/avg/max/mdev = 0.093/0.098/0.107/0.006 ms

```

MAC-адрес виртуального интерфейса:

```

[root@localhost lab-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
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:9a:a4:ef brd ff:ff:ff:ff:ff:ff
    inet 10.100.0.2/24 brd 10.100.0.255 scope global noprefixroute enp0s3
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fe9a:a4ef/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
3: br0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue state DOWN group default qlen 1000
    link/ether 1a:0a:49:20:05:7b brd ff:ff:ff:ff:ff:ff
    inet 10.100.0.3/24 brd 10.100.0.255 scope global noprefixroute br0
        valid_lft forever preferred_lft forever

```

Часть 3

Изменил тип подключения у Адаптера 1 на “Внутренняя сеть”

Отредактировал `/etc/netplan/01-netcfg.yaml` (ч.3-п.3):

```

network:
  version: 2
  renderer: networkd
  ethernets:
    enp0s3:
      addresses:
        - 10.100.0.4/24
        - 10.100.0.5/24
      routes:
        - to: 0.0.0.0/0
          via: 10.100.0.3

```

Принимаем конфигурацию (`sudo netplan apply`) и получаем:


```

root@d12:~/lab-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: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:c3:04:a5 brd ff:ff:ff:ff:ff:ff
    inet 10.100.0.4/24 brd 10.100.0.255 scope global enp0s3
        valid_lft forever preferred_lft forever
    inet 10.100.0.5/24 brd 10.100.0.255 scope global secondary enp0s3
        valid_lft forever preferred_lft forever
    inet6 fd17:625c:f037:2:a00:27ff:fec3:4a5/64 scope global dynamic mngtmpaddr noprefixroute
        valid_lft 86364sec preferred_lft 14364sec
    inet6 fd17:625c:f037:2:70d7:f31c:f313:f545/64 scope global dynamic noprefixroute
        valid_lft 86365sec preferred_lft 14365sec
    inet6 fe80::a00:27ff:fec3:4a5/64 scope link
        valid_lft forever preferred_lft forever
    inet6 fe80::acfa:334d:bd72:975f/64 scope link noprefixroute
        valid_lft forever preferred_lft forever

```

Ping's (4.3-п.5):

```

root@d12:~# ping -c 3 10.100.0.2
PING 10.100.0.2 (10.100.0.2) 56(84) bytes of data.
64 bytes from 10.100.0.2: icmp_seq=1 ttl=64 time=0.739 ms
64 bytes from 10.100.0.2: icmp_seq=2 ttl=64 time=0.639 ms
64 bytes from 10.100.0.2: icmp_seq=3 ttl=64 time=1.14 ms

--- 10.100.0.2 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2048ms
rtt min/avg/max/mdev = 0.639/0.840/1.143/0.217 ms

```

```

root@d12:~# ping -c 3 10.100.0.3
PING 10.100.0.3 (10.100.0.3) 56(84) bytes of data.
64 bytes from 10.100.0.3: icmp_seq=1 ttl=64 time=1.66 ms
64 bytes from 10.100.0.3: icmp_seq=2 ttl=64 time=0.779 ms
64 bytes from 10.100.0.3: icmp_seq=3 ttl=64 time=0.684 ms

--- 10.100.0.3 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2026ms
rtt min/avg/max/mdev = 0.684/1.040/1.658/0.438 ms

```

```

root@d12:~# ping -c 3 10.100.0.4
PING 10.100.0.4 (10.100.0.4) 56(84) bytes of data.
64 bytes from 10.100.0.4: icmp_seq=1 ttl=64 time=0.076 ms
64 bytes from 10.100.0.4: icmp_seq=2 ttl=64 time=0.034 ms
64 bytes from 10.100.0.4: icmp_seq=3 ttl=64 time=0.032 ms

--- 10.100.0.4 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2030ms
rtt min/avg/max/mdev = 0.032/0.047/0.076/0.020 ms

```

```

root@d12:~# ping -c 3 10.100.0.5
PING 10.100.0.5 (10.100.0.5) 56(84) bytes of data.
64 bytes from 10.100.0.5: icmp_seq=1 ttl=64 time=0.116 ms
64 bytes from 10.100.0.5: icmp_seq=2 ttl=64 time=0.056 ms
64 bytes from 10.100.0.5: icmp_seq=3 ttl=64 time=0.057 ms

--- 10.100.0.5 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2047ms
rtt min/avg/max/mdev = 0.056/0.076/0.116/0.028 ms

```

ARP-таблица:

```

root@d12:~/lab-1# ip neighbor show
10.100.0.3 dev enp0s3 lladdr 08:00:27:9a:a4:ef REACHABLE
10.100.0.2 dev enp0s3 lladdr 08:00:27:9a:a4:ef REACHABLE

```

Часть 4 [Debian]

Включил Адаптер 2 (*enp0s8*), переключил обоих на NUT

```

root@d12:~# nmcli con

```

NAME	UUID	TYPE	DEVICE
enp0s8	81c16286-c4e1-30b0-97ec-82a343635072	ethernet	enp0s8
enp0s3	d6df5981-18d5-4187-984b-276c50b732d0	ethernet	enp0s3

Установил модуль bonding:

```

root@d12:~# sudo modprobe bonding
root@d12:~# lsmod | grep bonding
bonding                221184  0
tls                     135168  1 bonding

```

Остановил и удалил подключения *enp0s3* и *enp0s8*, чтобы только *bond007* получал IP через DHCP и чередование пакетов между интерфейсами корректно работало:

```

root@d12:~# nmcli con down enp0s3
Подключение «enp0s3» успешно отключено (активный путь D-Bus: /org/freedesktop/NetworkManager/ActiveConnection/3)
root@d12:~# nmcli con down enp0s8
Подключение «enp0s8» успешно отключено (активный путь D-Bus: /org/freedesktop/NetworkManager/ActiveConnection/2)
root@d12:~# nmcli con delete enp0s3
Подключение «enp0s3» (d6df5981-18d5-4187-984b-276c50b732d0) успешно удалено.
root@d12:~# nmcli con delete enp0s8
Подключение «enp0s8» (81c16286-c4e1-30b0-97ec-82a343635072) успешно удалено.

```

Активировал bonding через netplan:

```

network:
  version: 2
  renderer: networkd
  ethernets:
    enp0s3:
      dhcp4: no
    enp0s8:
      dhcp4: no
  bonds:
    bond007:
      interfaces: [enp0s3, enp0s8]
      parameters:
        mode: balance-rr
        dhcp4: true

```

```
root@d12:~# sudo netplan apply
```

Получил IPv4-адрес для интерфейса bond007 через DHCP:

```
root@d12:~# sudo dhclient -4 bond007
```

Информация о IP-адресах всех интерфейсов:

```

root@d12:~# 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: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 1500 qdisc fq_codel master bond007 state UP group default qlen 1000
    link/ether ca:c5:df:35:3c:4d brd ff:ff:ff:ff:ff:ff permaddr 08:00:27:c3:04:a5
3: enp0s8: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 1500 qdisc fq_codel master bond007 state UP group default qlen 1000
    link/ether ca:c5:df:35:3c:4d brd ff:ff:ff:ff:ff:ff permaddr 08:00:27:84:6c:6e
4: bond007: <BROADCAST,MULTICAST,MASTER,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default qlen 1000
    link/ether ca:c5:df:35:3c:4d brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.16/24 brd 10.0.2.255 scope global dynamic bond007
        valid_lft 85949sec preferred_lft 85949sec
    inet6 fd17:625c:f037:2:c8c5:dfff:fe35:3c4d/64 scope global dynamic mngtmpaddr noprefixroute
        valid_lft 86360sec preferred_lft 14360sec
    inet6 fd17:625c:f037:3:c8c5:dfff:fe35:3c4d/64 scope global dynamic mngtmpaddr noprefixroute
        valid_lft 86253sec preferred_lft 14253sec
    inet6 fe80::c8c5:dfff:fe35:3c4d/64 scope link
        valid_lft forever preferred_lft forever

```

Итог: *enp0s3/enp0s8* (SLAVE's) используют MAC-адрес *bond007* (MASTER) и имеют собственные. То есть все три интерфейса используют один виртуальный MAC-адрес (*ca:c5:df:35:3c:4d*)

```

root@d12:~# ip link show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN mode DEFAULT group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: enp0s3: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 1500 qdisc fq_codel master bond007 state UP mode DEFAULT group default qlen 1000
    link/ether ca:c5:df:35:3c:4d brd ff:ff:ff:ff:ff:ff permaddr 08:00:27:c3:04:a5
3: enp0s8: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 1500 qdisc fq_codel master bond007 state UP mode DEFAULT group default qlen 1000
    link/ether ca:c5:df:35:3c:4d brd ff:ff:ff:ff:ff:ff permaddr 08:00:27:84:6c:6e
4: bond007: <BROADCAST,MULTICAST,MASTER,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP mode DEFAULT group default qlen 1000
    link/ether ca:c5:df:35:3c:4d brd ff:ff:ff:ff:ff:ff

```

Вывод (ч.4-п.5):


```

root@d12:~# cat /proc/net/bonding/bond007
Ethernet Channel Bonding Driver: v6.1.0-25-amd64

Bonding Mode: load balancing (round-robin)
MII Status: up
MII Polling Interval (ms): 0
Up Delay (ms): 0
Down Delay (ms): 0
Peer Notification Delay (ms): 0

Slave Interface: enp0s8
MII Status: up
Speed: 1000 Mbps
Duplex: full
Link Failure Count: 0
Permanent HW addr: 08:00:27:84:6c:6e
Slave queue ID: 0

Slave Interface: enp0s3
MII Status: up
Speed: 1000 Mbps
Duplex: full
Link Failure Count: 0
Permanent HW addr: 08:00:27:c3:04:a5
Slave queue ID: 0

```

Имеем нужный режим работы: *load balancing (round-robin)* – чередование пакетов между интерфейсами

Статистика сетевых интерфейсов:

```

root@d12:~# cat /proc/net/dev

```

Inter-	Receive								Transmit							
face	bytes	packets	errs	drop	fifo	frame	compressed	multicast	bytes	packets	errs	drop	fifo	colls	carrier	compressed
lo:	16275	162	0	0	0	0	0	0	16275	162	0	0	0	0	0	0
enp0s3:	51151	134	0	1	0	0	0	24	64393	421	0	0	0	0	0	0
enp0s8:	53130	248	0	0	0	0	0	26	56666	401	0	0	0	0	0	0
bond007:	89731	323	0	0	0	0	0	38	81173	500	0	39	0	0	0	0

Скрипт (ч.4-п.7):

```

#!/bin/bash

date
cat /proc/net/dev | tail -n +3 | awk '{print $1, $3, $11}'

```

Результаты с пингом:

```

root@d12:~# cd lab-1
root@d12:~/lab-1# ./p4task7.sh
C6 04 окт 2025 04:58:39 MSK
lo: 162 162
enp0s3: 138 431
enp0s8: 254 410
bond007: 333 519
root@d12:~/lab-1# ./p4task7.sh
C6 04 окт 2025 04:58:53 MSK
lo: 162 162
enp0s3: 146 442
enp0s8: 261 422
bond007: 348 542
root@d12:~/lab-1# ./p4task7.sh
C6 04 окт 2025 04:59:02 MSK
lo: 162 162
enp0s3: 150 446
enp0s8: 265 426
bond007: 356 550
root@d12:~/lab-1#
root@d12:~# ping -I bond007 8.8.8.8
PING 8.8.8.8 (8.8.8.8) from 10.0.2.16 bond007: 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=6 ttl=255 time=34.9 ms
64 bytes from 8.8.8.8: icmp_seq=7 ttl=255 time=34.2 ms
64 bytes from 8.8.8.8: icmp_seq=8 ttl=255 time=34.7 ms
64 bytes from 8.8.8.8: icmp_seq=11 ttl=255 time=34.3 ms
64 bytes from 8.8.8.8: icmp_seq=12 ttl=255 time=35.0 ms
64 bytes from 8.8.8.8: icmp_seq=13 ttl=255 time=35.0 ms
64 bytes from 8.8.8.8: icmp_seq=14 ttl=255 time=34.2 ms
64 bytes from 8.8.8.8: icmp_seq=15 ttl=255 time=34.3 ms
64 bytes from 8.8.8.8: icmp_seq=16 ttl=255 time=33.9 ms
64 bytes from 8.8.8.8: icmp_seq=17 ttl=255 time=34.8 ms
64 bytes from 8.8.8.8: icmp_seq=18 ttl=255 time=35.2 ms
64 bytes from 8.8.8.8: icmp_seq=19 ttl=255 time=33.7 ms
64 bytes from 8.8.8.8: icmp_seq=20 ttl=255 time=34.5 ms
64 bytes from 8.8.8.8: icmp_seq=21 ttl=255 time=34.2 ms
64 bytes from 8.8.8.8: icmp_seq=22 ttl=255 time=34.2 ms
64 bytes from 8.8.8.8: icmp_seq=23 ttl=255 time=33.8 ms
64 bytes from 8.8.8.8: icmp_seq=24 ttl=255 time=34.9 ms
64 bytes from 8.8.8.8: icmp_seq=25 ttl=255 time=33.6 ms
64 bytes from 8.8.8.8: icmp_seq=26 ttl=255 time=34.8 ms
64 bytes from 8.8.8.8: icmp_seq=27 ttl=255 time=34.2 ms
64 bytes from 8.8.8.8: icmp_seq=28 ttl=255 time=37.5 ms

```

Итого:

- Запуск 1 -> 2:

- Recieve enp0s3/enp0s8= +8/+7; bond007=+15
- Transmit enp0s3/enp0s8= +11/+12; bond007=+23

- Запуск 2 -> 3:

- Recieve enp0s3/enp0s8= +4/+4; bond007=+8
- Transmit enp0s3/enp0s8= +4/+4; bond007=+8

- Запуск 1 -> 3:

- Recieve enp0s3/enp0s8= +12/+11; bond007=+23
- Transmit enp0s3/enp0s8= +15/+16; bond007=+31

Вопросы:

1. Как с помощью команды ip:

- a.** назначить новый IPv4 адрес?

```
ip addr add <IP адрес/маска> dev <имя интерфейса>
```

- b.** назначить новый MAC адрес?

Нужно сначала отключить интерфейс: `ip link set dev <имя интерфейса> down`

Назначить новый MAC адрес: `ip link set dev <имя интерфейса> address <MAC-адрес>`

Включить интерфейс: `ip link set dev <имя интерфейса> up`

- c.** назначить новый gateway?

```
ip route replace default via <IP адрес шлюза> dev <имя интерфейса>
```

или

```
ip route add <IP адрес/маска> via <IP адрес шлюза> dev <имя интерфейса>
```

- d.** вывести информацию об ARP кэше?

```
ip n / ip neighbor show / arp -a (из пакета net-tools)
```

- e.** очистить arp кэш?

```
ip n flush dev <имя интерфейса>
```

- f.** включить интерфейс?

```
ip link set dev <имя интерфейса> up
```

- g.** выключить интерфейс?

```
ip link set dev <имя интерфейса> down
```

2. Как с помощью nmcli назначить на интерфейс статический IP адрес, маску и настроить default gateway?

```
nmcli con mod <имя интерфейса> ipv4.addresses <IP адрес/маска> / ipv4.gateway <IP адрес шлюза> (10.100.0.1 из ТЗ; 192.168.(1/0).1 чаще всего в домашних сетях)
```

3. Как с помощью netplan назначить на интерфейс статический IP адрес, маску и настроить default gateway?

В конфигурационном *yaml* файле Netplan'a изменить для нужного сетевого интерфейса поля `addresses` и `gateway4`; сделать `netplan apply`

4. Какие режимы bonding стандартно существуют в Linux? Опишите их назначение, возможности по отказоустойчивости и необходимость поддержки со стороны оборудования.

- I. **balance-rr** – режим по умолчанию, обеспечивает балансировку нагрузки и отказоустойчивость, пакеты отправляются по кругу от первого интерфейса к последнему, если один из них выходит из строя, пакеты отправляются дальше по кругу. При подключении портов к разным коммутаторам требуется их настройка
- II. **active-backup** – один интерфейс активен, остальные – ожидают (поломки текущего), требуются автономные порты и не требуется поддержка со стороны коммутатора
- III. **Balance-xor** – пакеты распределяются между объединенными интерфейсами по формуле $((\text{MAC-адрес источника}) \text{ XOR } (\text{MAC-адрес получателя})) \% \text{число интерфейсов}$. Требуется поддержка со стороны коммутатора
- IV. **Broadcast** – пакеты передаются во все объединенные интерфейсы
- V. **802.3ad** – динамическое объединение портов, которое способствует увеличению пропускной способности. Требуется поддержка от коммутатора и иногда его дополнительная настройка, а также согласованный по LACP (Link Aggregation Control Protocol) Etherchannel
- VI. **Balance-tlb** – адаптивная балансировка нагрузки: входящий трафик идет на активный интерфейс, а исходящий разделяется между оставшимися на основе хеша от IP-адреса пакета. Не требует специальной поддержки
- VII. **Balance-alb** – адаптивная балансировка нагрузки как входящего, так и исходящего трафика. Не нужна поддержка коммутатора, но требуется возможность изменять MAC-адрес устройства

5. Какие существуют и чем отличаются режимы работы адаптера (duplex)?

- I. **Полнодуплексный** режим. Передача данных может происходить одновременно в обоих направлениях без риска коллизий данных, так как используются отдельные провода
- II. **Полудуплексный** режим. Передача данных может происходить только в одном направлении за раз, есть риск коллизий данных в сети.
- III. **Симплексный** режим. Данные передаются только в одну сторону

6. Какой, по-вашему, практический смысл в возможности назначения нескольких IP адресов на один интерфейс?

- I. **Разделение сети** на логические участки для обеспечения безопасности
- II. **Поддержка** различных протоколов
- III. **Хостинг** нескольких сайтов со своими IP-адресами и SSL сертификатами

7. Какой, по-вашему, практический смысл в возможности создания виртуальных интерфейсов?

- I. **Разделение** трафика
- II. **Тестирование** и отладка сетевых приложений и конфигураций
- III. **VPN**

Понятийный минимум:

- 1. **IP адрес** - Уникальный "номер дома" устройства в сети (логический адрес)
- 2. **Gate (шлюз)** - "Выезд из района", маршрутизатор для связи с другими сетями
- 3. **Адрес DNS** - "Телефонная книга интернета", преобразует домены в IP
- 4. **DHCP клиент** - "Арендатор", автоматически получает настройки сети
- 5. **DHCP сервер** - "Арендодатель", раздает настройки сети устройствам
- 6. **Режим работы (duplex)** - "Дорога" передачи данных: Full = двустороннее движение, Half = одностороннее

7. Linux

- a. Имя интерфейса (enp0s3 или eth0 и т.п.) - Имя сетевой карты, которую используют для подключения к сети (eth0, Wi-Fi, br0, bond0)
- b. ifconfig (получение информации о адресах) – батя “ip”. Уже постарел (не поддерживает современные интерфейсы [VLAN, bridges..]), медленнее
- c. ip (контекст address, link, route):
 - address: показывает к каким интерфейсам принадлежит наш IP (IPv4, IPv6)
- d. dhclient – автоматический способ “прописки” в сети от DHCP серверов (роутер, офисный сервер, провайдер, VB..)
- e. /etc/resolv.conf – файл с адресами DNS серверов (“переводчик”, связывающий привычные названия сайтов (домены) с их реальными IP)
- f. Nmcli (включение, выключение, назначение адресов статически, динамически):
 - ВКЛ/ВЫКЛ: разрешение на обработку пакетов через интерфейс (con up <>)
 - Статический IP: “постоянная прописка” – ручное назначение адреса
 - Динамический IP: “временная аренда” – получение адреса от DHCP сервера (проверка неактивности через 24ч.)
- g. Netplan (конфигурирование, назначение адресов статически, динамически) – “чертеж” сети в YAML файлах
 - Статический IP: addresses: [10.100.0.4/24] ; gateway4: 10.100.0.1

- Динамический IP: `dhcp(4/6): true`

h. Bonding, общие принципы, файл текущей конфигурации в `/proc` :

- "Связывание" интерфейсов для надежности/скорости

- В конфиге: режим работы, статус, состояние "рабов"

i. Файл статистики интерфейсов в системе `/proc` - (`/proc/net/dev`) "Счетчик пакетов" для каждого интерфейса (Receive/Transmit)