

Министерство цифрового развития, связи и  
массовых коммуникаций Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего  
образования «Сибирский государственный университет телекоммуникаций и  
информатики» (СибГУТИ)

Кафедра вычислительных систем

**ОТЧЕТ**  
по практической работе 2

по дисциплине «**Сети ЭВМ и телекоммуникации**»

Выполнил:  
студент гр. ИС-142  
«\_\_» июня 2023 г.

\_\_\_\_\_

/Григорьев Ю.В./

Проверил:  
«\_\_» июня 2023 г.

\_\_\_\_\_

/Перышкова Е.Н./

Оценка « \_\_\_\_\_ »

Новосибирск 2023

## **ОГЛАВЛЕНИЕ**

<b>ПОСТАНОВКА ЗАДАЧИ .....</b>	<b>3</b>
<b>ВЫПОЛНЕНИЕ РАБОТЫ .....</b>	<b>6</b>

## ПОСТАНОВКА ЗАДАЧИ

1. Соберите конфигурацию сети, представленной на рисунке 1. Коммутатор на рисунке – это виртуальный коммутатор VirtualBox, работающий в режиме Host-only network.

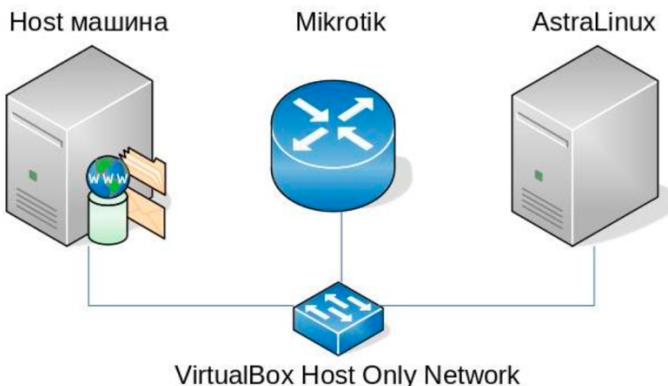


Рисунок 1 – Конфигурация сети для первой части практического занятия

2. Определите MAC адреса всех сетевых интерфейсов маршрутизатора Mikrotik, AstraLinux, хост-машины.
3. Вам выделено адресное пространство  $10.N.0.0/16$ , где  $N$  – это Ваш порядковый номер в журнале (по списку преподавателя). Спроектируйте выделенное Вам адресное пространство таким образом, чтобы разделить его на 4 равные по количеству адресов подсети (количество адресов в каждой подсети должно быть максимально возможным). Запишите диапазоны адресов для каждой полученной подсети, указав какой адрес будет называться адресом сети и какой адрес будет использоваться для широковещательной передачи данных.
4. Выберите один из рассчитанных в п. 3 диапазонов адресов и сконфигурируйте соответствующим образом сетевые интерфейсы хост-машины, astrarlinux и mikrotik. Адреса для узлов из назначенного диапазона выбираются произвольно.
5. Запустите на host-машине сетевой анализатор Wireshark и проверьте связность узлов с AstraLinux и Mikrotik с использованием протокола ICMP и режима ping-pong. В захваченном потоке пакетов покажите этапы работы протокола разрешения сетевых адресов (ARP) и протокола контроля сетевого подключения (ICMP). Покажите содержание таблицы MAC адресов на host-машине и покажите соответствие пакетам, захваченным для протокола ARP и используемых при работе протокола ICMP.
6. Запустите на host-машине сетевой анализатор Wireshark. Запустите ping с astrarlinux до microtik. Проанализируйте полученный поток пакетов.
7. Перезапустите виртуальную машину с astrarlinux в режиме записи потока сетевых пакетов1. Повторите действия пункта 6. Покажите, что в захваченном потоке пакетов присутствуют все пакеты, отправляемые и получаемые виртуальной машиной. Объясните, почему поток захваченных пакетов в п.6 отличается от потока, полученного в текущем пункте.

8. Запустите на host-машине сетевой анализатор Wireshark. С хост-машины из назначенного диапазона адресов попробуйте «пропинговать» адрес сети и широковещательный адрес.

1 [https://www.virtualbox.org/wiki/Network\\_tips](https://www.virtualbox.org/wiki/Network_tips)

Что поменяется в последовательности пакетов в сравнении с пингом простых адресов? В каком случае будет использоваться протокол ARP и почему? Могут ли в сети оказаться узлы, на интерфейсах которых назначены такие адреса и будет работать протокол ARP (если могут, то продемонстрируйте это)?

9. На машине AstraLinux установите пакет для конфигурирования сетевых интерфейсов с использованием APIPA2.

10. Запустите AstraLinux в режиме захвата пакетов. Настройте сетевой интерфейс так, чтобы для его для конфигурирования использовалась APIPA. Запустите процесс конфигурации интерфейса. В захваченном потоке пакетов покажите какие действия выполнялись при конфигурации интерфейса.

11. Используя web-интерфейс mikrotik установите на нем DHCP сервер. Адреса в DHCP должны выдаваться из иного диапазона, чем Вы выбрали в п. 4. В настройке DHCP укажите, что все пакеты протокола должны передаваться в широковещательном режиме. Если в Mikrotik запущен DHCP-client, то он должен быть остановлен.

12. Запустите на host-машине сетевой анализатор Wireshark. В astralinux создайте виртуальный интерфейс (например eth0:1), который настройте на получение сетевых настроек вавтоматическом режиме. Переведите интерфейс в рабочее состояние, определите какой адрес был получен для назначения на сетевой интерфейс. На хост-машине отфильтруйте захваченный поток пакетов так, чтобы отображались только пакеты протокола DHCP. Какие типы пакетов были захвачены? Подождите время, на которое была выдана аренда адреса (допускается в настройках DHCP сервера это время уменьшить) и покажите какие пакеты были отправлены клиентом и сервером? Остановите DHCP сервер и покажите, что будет делать клиент в этом случае? Будут ли все эти пакеты видны на host-машине? Если необходимо, то перезапустите astralinux в режиме захвата пакетов и покажите полный перечень пакетов, которые появляются в этом случае?

13. Измените конфигурацию сети в соответствие с представленной на рисунке 2. Коммутаторы на рисунке – это виртуальные коммутаторы VirtualBox, работающие в режиме Host-only network.

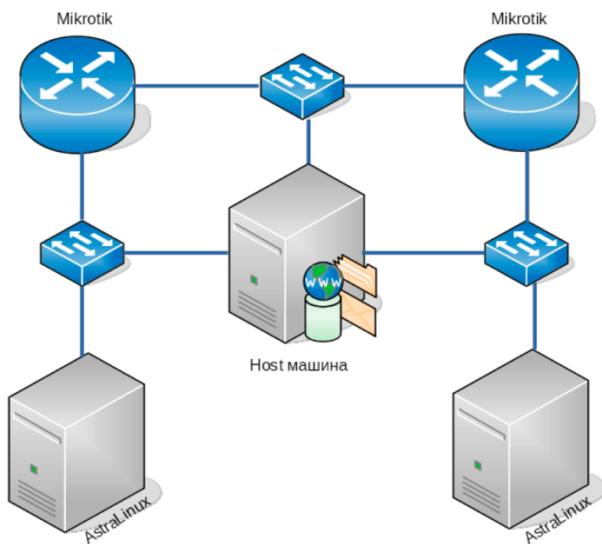


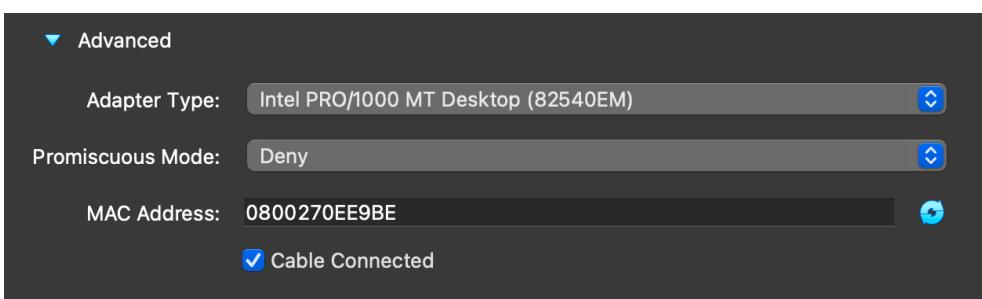
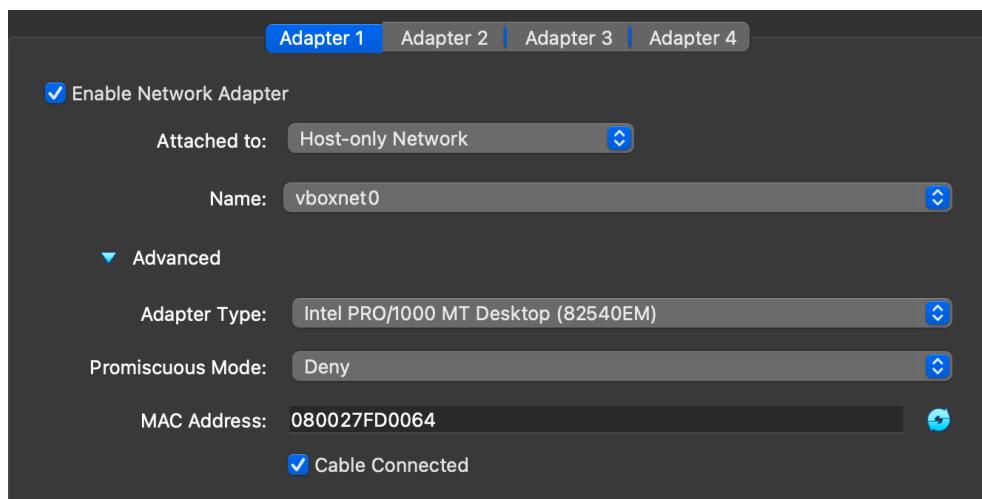
Рисунок 2 – Конфигурация сети для второй части практического занятия

14. На второй виртуальной машине Astralinux сконфигурируйте интерфейс так, чтобы он получал настройки автоматически. На маршрутизаторе mikrotik установите второй DHCP сервер, чтобы он выдавал адреса из свободного диапазона, рассчитанного в п. 3. Попробуйте получить сетевые настройки. Объясните почему не получилось?
15. Настройте сеть между маршрутизаторами Mikrotik в соответствии со свободным диапазоном из п. 3. Настройте на втором mikrotik dhcp-relay агент.
16. Запустите на host-машине сетевой анализатор Wireshark. Получите сетевые настройки для интерфейса второй виртуальной машины astralinux. Какие пакеты «видны» на хост машине? Почему? Перезапустите виртуальную машину astralinux в режиме захвата пакетов. Получите сетевые настройки. Какие типы пакетов DHCP были отправлены?
17. Перезапустите маршрутизаторы mikrotik в режиме захвата пакетов. Получите сетевые настройки на второй виртуальной машине. Покажите какие DHCP пакеты передавались между маршрутизаторами?

## ВЫПОЛНЕНИЕ РАБОТЫ

При выполнении работы было сделано следующее:

1. Конфигурация, представленная на рисунке 1, была выполнена в первой практической работе. Для выполнения текущей мной был отключен DHCP-сервер в настройках VirtualBox.
2. MAC-адреса были определены через VirtualBox.



3. Для разделения адресного пространства на 4 равные подпространства нужно посчитать количество адресов для одного подпространства — так как максимальный адрес равен 10.3.255.255, а минимальный 10.3.0.0, делим на 4 — получается шаг в 64 => первое подпространства — с 10.3.0.0 до 10.3.63.255, второе — с 10.3.64.0 до 10.3.127.255, третье — с 10.3.128.255 до 10.3.191.255, четвёртое — с 10.3.192.0 до 10.3.255.255, при этом 10.3.\*.0 будут адресами сети, а 10.3.\*.255 — адресами broadcast. Маска сети — 255.255.192.0 или /18.

4. DHCP-сервер для сети отключен с помощью команды 'VBoxManage dhcpserver stop ...' в терминале

```
VBoxManage dhcpserver remove <--network=netname | --interface=ifname>
VBoxManage dhcpserver start <--network=netname | --interface=ifname>
VBoxManage dhcpserver restart <--network=netname | --interface=ifname>
VBoxManage dhcpserver stop <--network=netname | --interface=ifname>
VBoxManage dhcpserver findlease <--network=netname | --interface=ifname>
    <--mac-address=mac>
allenvox@MacBook-Pro-Yuriy ~ % VBoxManage dhcpserver stop --network=HostInterfaceNetworking-vboxnet0
allenvox@MacBook-Pro-Yuriy ~ %
```

## 5. IP-адрес для astralinux настроен в файле /etc/network/interfaces.d/eth0

```
GNU nano 2.7.4                                Файл: /etc/network/interfaces.d/eth0

auto eth0
iface eth0 inet static
    address 10.3.0.2
    netmask 255.255.192.0
```

После перезапуска интерфейса eth0 машина имеет нужный IP-адрес

```
[#11 18:33:34]user@astralinux:~$ sudo ifdown eth0
ifdown: interface eth0 not configured
[#12 18:33:39]user@astralinux:~$ sudo ifup eth0
[#13 18:34:11]user@astralinux:~$ ip -c a show dev eth0
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default
  link/ether 08:00:27:0e:e9:be brd ff:ff:ff:ff:ff:ff
    inet 10.3.0.2/16 brd 10.3.255.255 scope global eth0
      valid_lft forever preferred_lft forever
    inet 10.3.0.2/18 brd 10.3.63.255 scope global eth0
      valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fe0e:e9be/64 scope link
      valid_lft forever preferred_lft forever
[#14 18:34:18]user@astralinux:~$ _
```

## 6. Проделана похожая процедура на роутере MikroTik: командой ip address add добавлен IP адрес для интерфейса ether1, после чего он был выведен на экран командой ip address print.

```
[admin@mt-01] > ip address add address=10.3.0.3/18 interface=ether1
[admin@mt-01] > ip address print
Flags: D - DYNAMIC
Columns: ADDRESS, NETWORK, INTERFACE
#  ADDRESS      NETWORK      INTERFACE
0  D 10.3.0.3/16  10.3.0.0  ether1
1  10.3.0.3/18  10.3.0.0  ether1
[admin@mt-01] > _
```

## 7. Пробуем «пинговать» виртуальные машины с хоста — они получают ARP-пакет, буквально значащий «Кто имеет такой IP – отзовись», и после получения этих данных шлёт echo ping на машину.

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	fe80::a00:27ff:fe0e:e9be	ff02::2	ICMP...	70	Router Solicitation from 08:00:27:0e:e9:be
2	38.528290	8e:85:90:29:7a:64	Broadcast	ARP	42	Who has 10.3.0.2? Tell 10.3.0.1
3	38.528858	PcsCompu_0e:e9:be	8e:85:90:29:7a:64	ARP	60	10.3.0.2 is at 08:00:27:0e:e9:be
4	38.528925	10.3.0.1	10.3.0.2	ICMP	98	Echo (ping) request id=0xe91d, seq=0/0, ttl=64
5	38.529444	10.3.0.2	10.3.0.1	ICMP	98	Echo (ping) reply id=0xe91d, seq=0/0, ttl=64

No.	Time	Source	Destination	Protocol	Length	Info
5	2.009136	10.3.0.1	10.3.0.3	ICMP	98	Echo (ping) request id=0xf51d, seq=2/512, ttl=64
6	2.009764	10.3.0.3	10.3.0.1	ICMP	98	Echo (ping) reply id=0xf51d, seq=2/512, ttl=64
7	3.012105	10.3.0.1	10.3.0.3	ICMP	98	Echo (ping) request id=0xf51d, seq=3/768, ttl=64
8	3.012924	10.3.0.3	10.3.0.1	ICMP	98	Echo (ping) reply id=0xf51d, seq=3/768, ttl=64
9	4.017116	10.3.0.1	10.3.0.3	ICMP	98	Echo (ping) request id=0xf51d, seq=4/1024, ttl=64
10	4.018619	10.3.0.3	10.3.0.1	ICMP	98	Echo (ping) reply id=0xf51d, seq=4/1024, ttl=64
11	5.022237	10.3.0.1	10.3.0.3	ICMP	98	Echo (ping) request id=0xf51d, seq=5/1280, ttl=64
12	5.023007	10.3.0.3	10.3.0.1	ICMP	98	Echo (ping) reply id=0xf51d, seq=5/1280, ttl=64
13	5.058643	PcsCompu_fd:00:64	8e:85:90:29:7a:64	ARP	60	Who has 10.3.0.1? Tell 10.3.0.3
14	5.058699	8e:85:90:29:7a:64	PcsCompu_fd:00:64	ARP	42	10.3.0.1 is at 8e:85:90:29:7a:64

```
[allenvox@MacBook-Pro-Yuriy ~ % ping 10.3.0.2
PING 10.3.0.2 (10.3.0.2): 56 data bytes
64 bytes from 10.3.0.2: icmp_seq=0 ttl=64 time=1.291 ms
64 bytes from 10.3.0.2: icmp_seq=1 ttl=64 time=0.538 ms
64 bytes from 10.3.0.2: icmp_seq=2 ttl=64 time=0.760 ms
64 bytes from 10.3.0.2: icmp_seq=3 ttl=64 time=0.789 ms
^C
--- 10.3.0.2 ping statistics ---
4 packets transmitted, 4 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 0.538/0.845/1.291/0.275 ms
[allenvox@MacBook-Pro-Yuriy ~ % ping 10.3.0.3
PING 10.3.0.3 (10.3.0.3): 56 data bytes
64 bytes from 10.3.0.3: icmp_seq=0 ttl=64 time=0.903 ms
64 bytes from 10.3.0.3: icmp_seq=1 ttl=64 time=1.242 ms
64 bytes from 10.3.0.3: icmp_seq=2 ttl=64 time=1.053 ms
64 bytes from 10.3.0.3: icmp_seq=3 ttl=64 time=0.949 ms
^C
--- 10.3.0.3 ping statistics ---
4 packets transmitted, 4 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 0.903/1.037/1.242/0.130 ms
[allenvox@MacBook-Pro-Yuriy ~ % ping 10.3.0.3
PING 10.3.0.3 (10.3.0.3): 56 data bytes
64 bytes from 10.3.0.3: icmp_seq=0 ttl=64 time=1.101 ms
64 bytes from 10.3.0.3: icmp_seq=1 ttl=64 time=1.191 ms
64 bytes from 10.3.0.3: icmp_seq=2 ttl=64 time=0.869 ms
64 bytes from 10.3.0.3: icmp_seq=3 ttl=64 time=1.030 ms
64 bytes from 10.3.0.3: icmp_seq=4 ttl=64 time=1.733 ms
64 bytes from 10.3.0.3: icmp_seq=5 ttl=64 time=1.058 ms
^C
--- 10.3.0.3 ping statistics ---
6 packets transmitted, 6 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 0.869/1.164/1.733/0.272 ms
allenvox@MacBook-Pro-Yuriy ~ % ]
```

8. Пробую «пинговать» виртуальные машины друг через друга. В Wireshark видно только ARP пакеты, echo ping не видно, хотя через машину видно, что пакеты доходят.

```
[admin@mt-011 > ping 10.3.0.2
SEQ HOST                               SIZE TTL TIME      STATUS
 0 10.3.0.2                           56  64 1ms755us
 1 10.3.0.2                           56  64 1ms868us
 2 10.3.0.2                           56  64 1ms218us
 3 10.3.0.2                           56  64 840us
 4 10.3.0.2                           56  64 2ms779us
 5 10.3.0.2                           56  64 1ms441us
sent=6 received=6 packet-loss=0% min-rtt=840us avg-rtt=1ms650us
max-rtt=2ms779us
```

303	5774.540390	host-3-0-2.211.ru	PcsCompu_fd:00:64	ARP	60	Who has 10.3.0.3? Tell 10.3.0.2
304	5774.540756	PcsCompu_fd:00:64	host-3-0-2.211.ru	ARP	60	10.3.0.3 is at 08:00:27:fd:00:64

9. Для исправления проблемы и чтобы посмотреть, получает ли виртуальная машина наши пакеты, выключаю машину astralinux и прописываю следующую команду для включения выводения пакетов с машины в лог astra-packages.rcap. Всё отлично работает и видны пакеты ICMP. Можем включить функцию логгирования, чтобы не заполнять диск лишними файлами.

```
[allenvox@MacBook-Pro-Yuriy ~ % vboxmanage modifyvm "astralinux" --nictrace1
on --nictracefile1 /Users/allenvox/Documents/Workspace/mikrotik-networking
/astra-packages.pcap
allenvox@MacBook-Pro-Yuriy ~ % ]
```

299	5772.540472	10.3.0.3	host-3-0-2.211.ru	ICMP	70	Echo (ping) request id=0xc700
300	5772.540820	host-3-0-2.211.ru	10.3.0.3	ICMP	70	Echo (ping) reply id=0xc700,
301	5773.545217	10.3.0.3	host-3-0-2.211.ru	ICMP	70	Echo (ping) request id=0xc700
302	5773.546296	host-3-0-2.211.ru	10.3.0.3	ICMP	70	Echo (ping) reply id=0xc700,
303	5774.540390	host-3-0-2.211.ru	PcsCompu_fd:00:64	ARP	60	Who has 10.3.0.3? Tell 10.3.0.2
304	5774.540756	PcsCompu_fd:00:64	host-3-0-2.211.ru	ARP	60	10.3.0.3 is at 08:00:27:fd:00:64
305	5774.552162	10.3.0.3	host-3-0-2.211.ru	ICMP	70	Echo (ping) request id=0xc700
306	5774.553090	host-3-0-2.211.ru	10.3.0.3	ICMP	70	Echo (ping) reply id=0xc700,

```
[allenvox@MacBook-Pro-Yuriy ~ % vboxmanage modifyvm "astralinux" --nictrace1
off
allenvox@MacBook-Pro-Yuriy ~ % ]
```

10. Попробуем «пропинговать» адрес сети 10.3.0.0 : видим ARP-пакеты, на которые нет ответа. Терминал показывает потерю пакетов в 100%.

```
[allenvox@MacBook-Pro-Yuriy ~ % ping 10.3.0.0
PING 10.3.0.0 (10.3.0.0): 56 data bytes
Request timeout for icmp_seq 0
Request timeout for icmp_seq 1
Request timeout for icmp_seq 2
Request timeout for icmp_seq 3
Request timeout for icmp_seq 4
Request timeout for icmp_seq 5
Request timeout for icmp_seq 6
Request timeout for icmp_seq 7
Request timeout for icmp_seq 8
Request timeout for icmp_seq 9
Request timeout for icmp_seq 10
Request timeout for icmp_seq 11
Request timeout for icmp_seq 12
Request timeout for icmp_seq 13
Request timeout for icmp_seq 14
^C
--- 10.3.0.0 ping statistics ---
16 packets transmitted, 0 packets received, 100.0% packet loss
allenvox@MacBook-Pro-Yuriy ~ % ]
```

11. Меняем IP-адрес astralinux на 10.3.0.0 и всё снова работает — пакеты доходят, адрес сети работает на устройстве.

```
[#1 21:49:05]user@astral1:~$ ip -c a show dev eth0
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast
    link/ether 08:00:27:0e:e9:be brd ff:ff:ff:ff:ff:ff
    inet 10.3.0.0/18 brd 10.3.63.255 scope global eth0
        valid_lft forever preferred_lft forever
        inet6 fe80::a00:27ff:fe0e:e9be/64 scope link
            valid_lft forever preferred_lft forever
[#2 21:49:12]user@astral1:~$ _
```

13	8.007979	host-3-0-1.211.ru	10.3.0.0	ICMP	98	Echo (ping) request
14	9.008433	host-3-0-1.211.ru	10.3.0.0	ICMP	98	Echo (ping) request
15	10.010413	host-3-0-1.211.ru	10.3.0.0	ICMP	98	Echo (ping) request
16	11.011549	host-3-0-1.211.ru	10.3.0.0	ICMP	98	Echo (ping) request
17	12.016413	host-3-0-1.211.ru	10.3.0.0	ICMP	98	Echo (ping) request
18	13.020749	host-3-0-1.211.ru	10.3.0.0	ICMP	98	Echo (ping) request

12. Пробуем «пропинговать» адрес broadcast – 10.3.63.255 : пакеты ни до кого не доходят, видим в Wireshark, что наш хост спрашивает у каждого устройства об этом адресе.

```
[allenvox@MacBook-Pro-Yuriy ~ % ping 10.3.63.255
PING 10.3.63.255 (10.3.63.255): 56 data bytes
Request timeout for icmp_seq 0
Request timeout for icmp_seq 1
^C
--- 10.3.63.255 ping statistics ---
3 packets transmitted, 0 packets received, 100.0% packet loss
allenvox@MacBook-Pro-Yuriy ~ %
```

382	4558.055954	MacBook-Pro-Yuriy.local	Broadcast	ARP	42	Who has 10.3.63.255? Tell 10.3.0.1
383	4559.059462	MacBook-Pro-Yuriy.local	Broadcast	ARP	42	Who has 10.3.63.255? Tell 10.3.0.1
384	4560.064339	MacBook-Pro-Yuriy.local	Broadcast	ARP	42	Who has 10.3.63.255? Tell 10.3.0.1
385	4561.068913	MacBook-Pro-Yuriy.local	Broadcast	ARP	42	Who has 10.3.63.255? Tell 10.3.0.1
386	4562.074037	MacBook-Pro-Yuriy.local	Broadcast	ARP	42	Who has 10.3.63.255? Tell 10.3.0.1

13. Присваиваем устройству astralinux адрес 10.3.63.255 : astralinux посылает ARP-пакет с запросом, но не получает ответа. Вывод – присвоение адреса сети возможно, присвоение адреса broadcast может привести к печальным последствиям.

```
[#1 21:51:38]user@astralinux:~$ ip -c a show dev eth0
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast
    link/ether 08:00:27:0e:e9:be brd ff:ff:ff:ff:ff:ff
    inet 10.3.63.255/18 brd 10.3.63.255 scope global eth0
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fe0e:e9be/64 scope link
        valid_lft forever preferred_lft forever
[#2 21:51:41]user@astralinux:~$
```

```
27 70.718536 host-3-0-1.211.ru Broadcast ARP 42 Who has 10.3.63.255? Tell 10.3.0.1
```

14. Выключаем astralinux, меняем сетевой адаптер в нём на NAT для выхода во внешнюю сеть. Ставим в /etc/network/interfaces.d/eth0 получение IP через dhcp. Загружаем пакет avahi-autoipd и переводим устройство обратно в режим HostOnlyNetwork.

```
GNU nano 2.7.4
auto eth0
iface eth0 inet dhcp #static
    #address 10.3.63.255
    #netmask 255.255.192.0
```

→

```
GNU nano 2.7.4
auto eth0
iface eth0 inet static
    address 10.3.0.2
    netmask 255.255.192.0
```

15. Для работы с Avahi меняем конфигурацию интерфейса eth0 на получение Zeroconf (IPv4 link-local) адреса.

```
GNU nano 2.7.4
#auto eth0
iface eth0 inet ipv4ll
    #address 10.3.0.2
    #netmask 255.255.192.0
```

Теперь astralinux получает случайный IP из диапазона 169.254.\*.\* и опрашивает сеть с помощью ARP на доступность этого адреса (ARP Probe). Если он доступен – Avahi делает анонс также с помощью ARP (ARP Announcement).

4	0.778507	PcsCompu_0e:e9:be	Broadcast	ARP	60	Who has 169.254.8.211? (ARP Probe)
5	1.380678	fe80::a00:27ff:fe0e:e9be	ff02::16	ICMPv6	90	Multicast Listener Report Message v2
6	1.381406	fe80::a00:27ff:fe0e:e9be	ff02::2	ICMPv6	70	Router Solicitation from 08:00:27:0e:e9:be
7	2.342469	fe80::a00:27ff:fe0e:e9be	ff02::16	ICMPv6	90	Multicast Listener Report Message v2
8	2.723517	PcsCompu_0e:e9:be	Broadcast	ARP	60	Who has 169.254.8.211? (ARP Probe)
9	4.538953	PcsCompu_0e:e9:be	Broadcast	ARP	60	Who has 169.254.8.211? (ARP Probe)
...	5.678806	fe80::a00:27ff:fe0e:e9be	ff02::2	ICMPv6	70	Router Solicitation from 08:00:27:0e:e9:be
11	6.545301	PcsCompu_0e:e9:be	Broadcast	ARP	60	ARP Announcement for 169.254.8.211
12	8.550005	PcsCompu_0e:e9:be	Broadcast	ARP	60	ARP Announcement for 169.254.8.211

Так как взаимодействие с APIPA завершено, возвращаем astralinux в первоначальное состояние со статичным IP-адресом 10.3.0.2 .

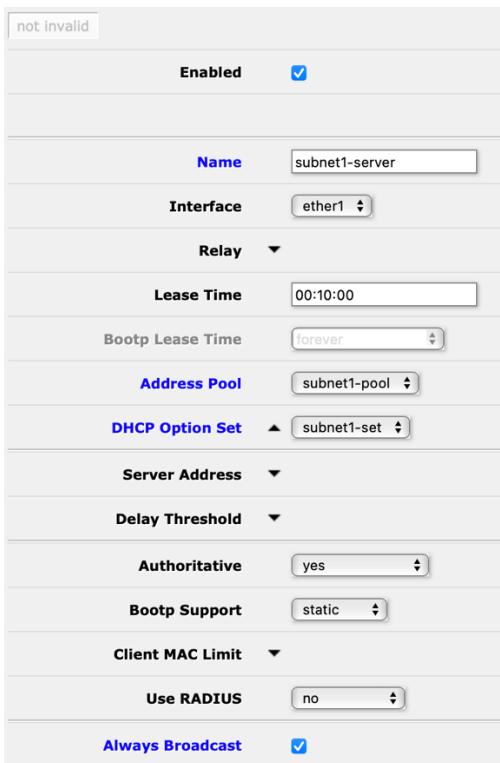
16. Заходим в WebFig MikroTik-роутера с помощью браузера, переходим в IP —> Pool. Задаём пул IP-адресов для статических и динамических подключений (2-й из четырёх).

The screenshot shows the RouterOS v7.7 interface with several windows open:

- Left Sidebar:** Contains icons for CAPsMAN, Wireless, Interfaces, WireGuard, PPP, Bridge, Mesh, IP, MPLS, IPv6, Routing, System, Queues, Dot1X, Files, Log, RADIUS, Tools, Make Supout.rif, Undo, Redo, Hide Passwords, Safe Mode, Design Skin, WinBox, Graphs, and End-User License.
- Top Bar:** Shows "RouterOS v7.7 (stable)" and tabs for Quick Set, WebFig, Terminal, and a file icon.
- Interface List Window:** Shows a table with one item for interface "ether1" (Type: Ethernet, MTU: 1500, Tx: 9.2 kbps, Rx: 5.1 kbps).
- IP Pool Configuration Window:** Set to "Address" 10.3.0.0/18, "Gateway" 255.255.192.0, "Netmask" 255.255.192.0, and "DNS Servers" (disabled). It also includes sections for Domain, WINS Servers, NTP Servers, CAPS Managers, Next Server, Boot File Name, and DHCP Options (set to subnet1-set).
- DHCP Client Configuration Window:** Set to "Name" subnet1-pool, "Addresses" 10.3.64.1-10.3.127.255, and "Interface" ether1. It includes fields for Use Peer DNS (yes), Add Default Route (yes), IP Address (10.3.0.3/16), and Expires After (22:48:31).

В меню IP —> DHCP Client отключаем клиент для ether1, в IP —> DHCP Server —> Option Sets создаём пустое множество настроек, которое далее назначаем при создании новой сети в IP —> DHCP Server —> Networks.

Создаём новый DHCP-сервер на интерфейсе ether1, в созданном нами пуле адресов и множестве настроек. Время аренды адреса стандартное (10 минут), включен широковещательный режим Always Broadcast.



17. Настраиваем astralinux для работы с новым DHCP-сервером: в файле /etc/network/interfaces.d/eth0 прописываем новый интерфейс eth0:1 для работы с dhcp,

```
GNU nano 2.7.4
auto eth0
iface eth0 inet static
    address 10.3.0.2
    netmask 255.255.192.0

auto eth0:1
iface eth0:1 inet dhcp
```

который к нему и подключаем. Перезапускаем виртуальную машину и ловим DHCP-пакеты: получаем ACK на предоставление и ещё через 10 минут (время аренды) пакет DHCP REQUEST (запрос на продление аренды). IP-адрес принадлежит нашему пулу адресов, значит всё сделано верно. При выключении роутера MikroTik и вместе с ним нашего DHCP-сервера при конце аренды IP-адреса astralinux начинает снова его запрашивать и искать роутер по ARP-протоколу.

2	0.158404	0.0.0.0	255.255.255.255	DHCP	342	DHCP Discover - Transaction ID 0x82039619
3	0.197415	10.3.0.1	10.3.0.2	DHCP	342	DHCP Offer - Transaction ID 0x82039619
4	0.198888	0.0.0.0	255.255.255.255	DHCP	342	DHCP Request - Transaction ID 0x82039619
5	0.200583	10.3.0.1	10.3.0.2	DHCP	342	DHCP ACK - Transaction ID 0x82039619
6	0.207705	...	...	DHCP	00	...
250.679298	8e:85:90:29:7a:64	Broadcast		ARP	42	Who has 10.3.0.3? Tell 10.3.0.1
250.679552	PcsCompu_fd:00:64	8e:85:90:29:7a:64		ARP	60	10.3.0.3 is at 08:00:27:fd:00:64

18. Для создания новой конфигурации сети, создам вторую машину astralinux (склонирую через VirtualBox) и изменю её аппаратное имя командой ‘hostnamectl set-hostname astra2’ вместе с именем хоста в файле /etc/hosts. Также в VirtualBox создам новые подсети vboxnet1 и vboxnet2 (для 3 и 4 секции из п.3).

В файле /etc/network/interfaces.d/eth0 машины astra2 поставлю получение IP автоматически по DHCP. Также склонирую роутер MikroTik и назначу ему имя mt-02. К нему будут подключены сети vboxnet1 и vboxnet2, к первому роутеру — vboxnet0 и vboxnet1.

```

GNU nano 2.7.4

auto eth0
iface eth0 inet dhcp

```

```

[admin@mt-01] >
[admin@mt-01] > system identity set name=mt-02
[admin@mt-02] >

```

Name	IPv4 Address/Mask
vboxnet0	10.3.0.1/18
vboxnet1	10.3.128.1/18
vboxnet2	10.3.192.1/18

Так как виртуальная машина astra2 подключена к новому роутеру без DHCP-сервера (mt-02), она не может обмениваться пакетами с другими устройствами. Это решаемо с помощью DHCP Relay (см. далее).

Чтобы связать роутеры 1 и 2, выдаю интерфейсу ether2 на mt-01 IP-адрес 10.3.128.2/18, а интерфейсу ether1 mt-02 адрес 10.3.128.3/18. Пробую «пропинговать» один роутер с другого, всё работает.

```

[admin@mt-01] > ip address print
Columns: ADDRESS, NETWORK, INTERFACE
# ADDRESS      NETWORK      INTERFACE
0 10.3.0.3/18  10.3.0.0    ether1
[admin@mt-01] > ip address add interface=ether2 address=10.3.128.2/18
[admin@mt-01] > ip address print
Columns: ADDRESS, NETWORK, INTERFACE
# ADDRESS      NETWORK      INTERFACE
0 10.3.0.3/18  10.3.0.0    ether1
1 10.3.128.2/18 10.3.128.0  ether2
[admin@mt-01] > ping 10.3.128.3/18
invalid value for argument address:
    invalid value of mac-address, mac address required
    invalid value for argument ipv6-address
    while resolving ip-address: could not get answer from dns server
[admin@mt-01] > ping 10.3.128.3
SEQ HOST                                SIZE TTL TIME      STATUS
  0 10.3.128.3                            56  64 4ms734us
  1 10.3.128.3                            56  64 1ms538us
  2 10.3.128.3                            56  64 1ms774us
    sent=3 received=3 packet-loss=0% min-rtt=1ms538us avg-rtt=2ms682us
    max-rtt=4ms734us

[admin@mt-01] > _

```

19. Чтобы связать второй роутер с сетью vboxnet2, выдаём интерфейсу ether2 на нём адрес 10.3.192.3/18 и удаляем DHCP-клиенты.

```

[admin@mt-02] > interface set 0 name="ether1"
[admin@mt-02] > interface print
Flags: R - RUNNING
Columns: NAME, TYPE, ACTUAL-MTU, MAC-ADDRESS
#  NAME      TYPE      ACTUAL-MTU  MAC-ADDRESS
0  R ether1   ether      1500    08:00:27:0B:2B:18
1  R ether3   ether      1500    08:00:27:98:BC:CA
[admin@mt-02] > interface set 1 name="ether2"
[admin@mt-02] > interface print
Flags: R - RUNNING
Columns: NAME, TYPE, ACTUAL-MTU, MAC-ADDRESS
#  NAME      TYPE      ACTUAL-MTU  MAC-ADDRESS
0  R ether1   ether      1500    08:00:27:0B:2B:18
1  R ether2   ether      1500    08:00:27:98:BC:CA
[admin@mt-02] > ip address add interface=ether1 address=10.3.128.3/18
[admin@mt-02] > ip address print
Columns: ADDRESS, NETWORK, INTERFACE
# ADDRESS          NETWORK        INTERFACE
0 10.3.128.3/18  10.3.128.0  ether1
[admin@mt-02] > ip dhcp-client remove 0
[admin@mt-02] > ip dhcp-client print

```

```

[admin@mt-02] > ip address add interface=ether2 address=10.3.192.3/18
failure: already have such address
[admin@mt-02] > ip address print
Flags: X, D - DYNAMIC
Columns: ADDRESS, NETWORK, INTERFACE
#  ADDRESS          NETWORK        INTERFACE
0  10.3.128.3/18  10.3.128.0  ether1
2  10.3.192.3/18  10.3.192.0  ether2
[admin@mt-02] >

```

20. Создаём DHCP-Relay в WebFig'e RouterOS. Задаём интерфейс ether2 для связи с другой сетью (vboxnet1) и её IP-адрес (адрес второго роутера). Чтобы запросы четвёртой подсети обрабатывались, необходимо добавить ещё один пул адресов в IP → Pool и DHCP-сеть для выдачи адресов четвёртой подсети. Также создадим новый DHCP-сервер для выдачи IP-адресов в этой подсети.

<b>Enabled</b>	<input checked="" type="checkbox"/>
<b>Name</b>	subnet4-relay
<b>Interface</b>	ether2
<b>DHCP Server</b>	10.3.128.2

<b>Name</b>	subnet4-pool
<b>Addresses</b>	10.3.192.1-10.3.255.254
<b>Address</b>	10.3.192.0/18
<b>Gateway</b>	
<b>Netmask</b>	255.255.192.0

<b>Enabled</b>	<input checked="" type="checkbox"/>
<b>Name</b>	subnet4-server
<b>Interface</b>	ether2
<b>Relay</b>	▲ 10.3.192.2
<b>Lease Time</b>	00:10:00
Bootp Lease Time	forever
<b>Address Pool</b>	subnet4-pool
<b>DHCP Option Set</b>	▲ subnet1-set

Запускаем все устройства в режиме захвата пакетов и смотрим на процедуры получения IP, которые предоставляют роутеры.

Получение IP astra1 от router1:

2	0.168688	0.0.0.0	255.255.255.255	DHCP	342	DHCP Discover -
3	0.175839	PcsCompu_b4:55:40	Broadcast	ARP	60	Who has 10.3.127.1?
4	0.217309	host-3-0-1.211.ru	10.3.0.7	DHCP	342	DHCP Offer - T
5	0.218491	0.0.0.0	255.255.255.255	DHCP	342	DHCP Request -
6	0.251301	host-3-0-1.211.ru	10.3.0.7	DHCP	342	DHCP ACK - T
7	0.329764	::	ff02::16	ICMPv6	90	Multicast Listener
8	0.683116	10.3.0.3	255.255.255.255	DHCP	342	DHCP Offer - T

Получение IP astra2 от router2:

5	21.358988	0.0.0.0	255.255.255.255	DHCP	342	DHCP Discover -
6	21.359945	::	ff02::16	ICMPv6	90	Multicast Listener
7	21.465075	10.3.192.1	10.3.192.2	DHCP	342	DHCP Offer - T
8	21.465793	0.0.0.0	255.255.255.255	DHCP	342	DHCP Request -
9	21.467034	10.3.192.1	10.3.192.2	DHCP	342	DHCP ACK - T

Пропуем «пропинговать» все связанные устройства.

От router2 к router1:

SEQ	HOST		SIZE	TTL	TIME	STATUS
0	10.3.128.2		56	64	2ms231us	
1	10.3.128.2		56	64	1ms339us	
2	10.3.128.2		56	64	726us	
3	10.3.128.2		56	64	676us	
<b>sent=4 received=4 packet-loss=0% min-rtt=676us avg-rtt=1ms243us</b>						
<b>max-rtt=2ms231us</b>						

От router1 к astra1:

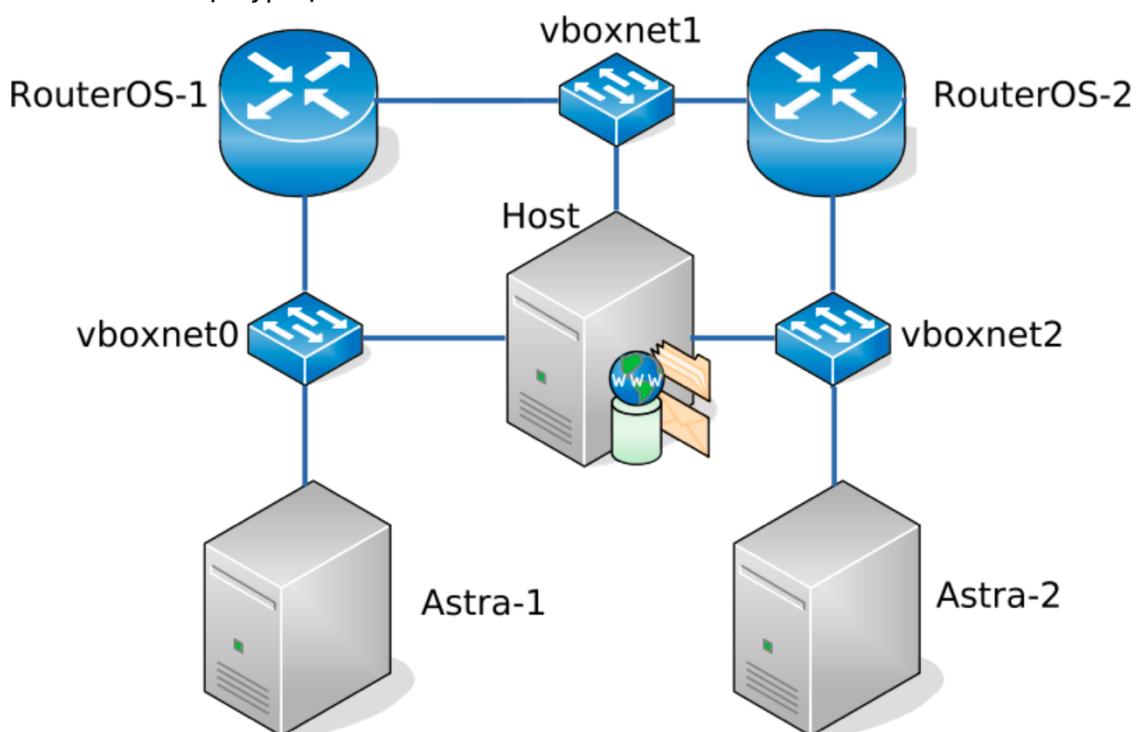
SEQ	HOST		SIZE	TTL	TIME	STATUS
0	10.3.0.2		56	64	1ms642us	
1	10.3.0.2		56	64	1ms267us	
2	10.3.0.2		56	64	1ms481us	
<b>sent=3 received=3 packet-loss=0% min-rtt=1ms267us avg-rtt=1ms463us</b>						
<b>max-rtt=1ms642us</b>						

От router2 к astra2:

```
Astra Linux CE 2.12.45 (orel) astra2 tty1
lo: 127.0.0.1astr้า
eth0: 10.3.192.2astr้า
eth1: astra2

root@router2 [Running]
# ip a
1: ADDRESS      NETWORK      INTERFACE
0 10.3.128.3/18 10.3.128.0  ether1
1 X 10.3.192.2/18 10.3.192.0  ether2
2 10.3.192.3/18 10.3.192.0  ether2
admin@mt-02] > reboot
bad command name reboot (line 1 column 1)
admin@mt-02] > ip address add interface=ether2 address=10.3.192.3/18
failure: already have such address
admin@mt-02] > ip address print
Flags: X, D - DYNAMIC
Columns: ADDRESS, NETWORK, INTERFACE
# ip a
0 10.3.128.3/18 10.3.128.0  ether1
1 X 10.3.192.2/18 10.3.192.0  ether2
2 10.3.192.3/18 10.3.192.0  ether2
admin@mt-02] > ping 10.3.192.2
SEQ HOST                               SIZE TTL TIME
0 10.3.192.2                           56 64 2ms401us
1 10.3.192.2                           56 64 1ms199us
2 10.3.192.2                           56 64 1ms109us
3 10.3.192.2                           56 64 927us
sent=4 received=4 packet-loss=0% min-rtt=927us avg-rtt=1ms409us
max-rtt=2ms401us
```

Финальная конфигурация:



Задание выполнено.

Связь между роутерами налажена, каждый роутер отвечает за свою локальную сеть.