

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

**Факультет безопасности информационных технологий**

**Дисциплина:**  
«Компьютерные сети»

**ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №3**  
«Основы администрирования маршрутизируемых компьютерных сетей»

**Выполнили:**

Чу Ван Доан, студент группы N3347



---

(подпись)

**Проверил:**

Есипов Дмитрий Андреевич

---

(отметка о выполнении)

---

(подпись)

Санкт-Петербург  
2024 г.

## СОДЕРЖАНИЕ

1 ХОД РАБОТЫ.....	4
1.1 Настройка IPv4.....	4
1.2 Работа с утилитой nc.....	6
1.3 Настройка iptables.....	7
1.4 Проверка работоспособности правил iptables.....	8

## ВВЕДЕНИЕ

**Цель работы** – изучение основных методов настройки маршрутизируемых компьютерных сетей на примере сети, состоящей из компьютеров под управлением ОС Linux.

**Для достижения поставленной цели необходимо решить следующие задачи:**

1. Провести теоретический анализ сетевого уровня модели OSI, включая его основные функции и протоколы, применяемые в маршрутизируемых сетях.
2. Выполнить базовую настройку сетевых интерфейсов и связности между компьютерами в сети, чтобы обеспечить возможность обмена данными.
3. Исследовать и настроить таблицы маршрутизации для корректной передачи пакетов в сети, включая маршрутизацию для IPv4 и IPv6.
4. Использовать утилиту **tcpdump** для наблюдения за сетевым трафиком, анализируя проходящие пакеты и их внутреннюю структуру, а также изучить применение технологии NAT.

# 1 ХОД РАБОТЫ

Выбор варианта:

Меня зовут Чу Ван (5 букв)

$$V1 = 1 + (N \bmod 5) = 1 + (5 \bmod 5) = 1$$

$$V2 = 6 + (N \bmod 5) = 6 + (5 \bmod 5) = 6$$

Я выбираю операционную систему Ubuntu версии 14.04.6 Server.

VirtualBox с установленными машинами и их сетевыми настройками будет выглядеть так:

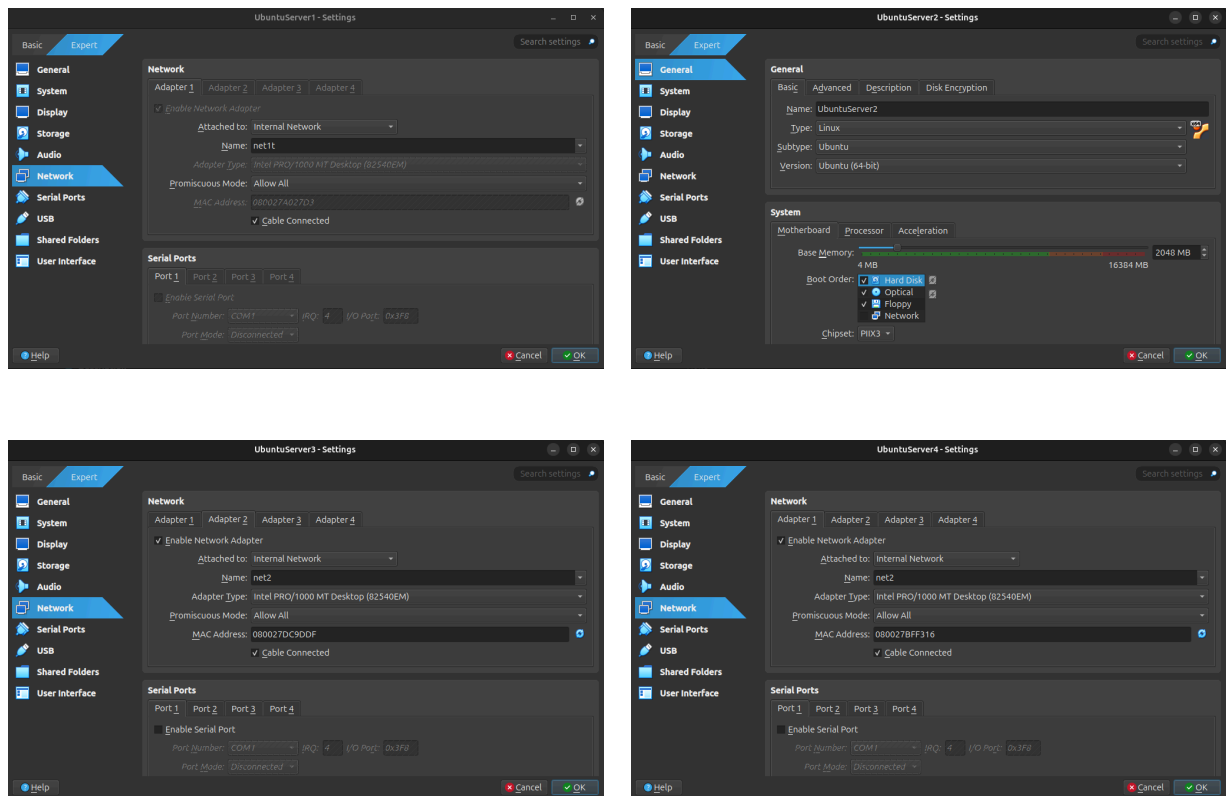


Рисунок 1 – Установленные машины и их сетевые настройки

## 1.1 Настройка IPv4

Настроим IPv4-адреса на всех компьютерах сети. В качестве доказательства настройки приведем результаты выполнения команды `ifconfig` на всех машинах

```
UbuntuServer1 [Running] - Oracle VirtualBox
File Machine View Input Devices Help
root@vbox:~# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1
    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: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 08:00:27:a0:27:d3 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.1/24 brd 192.168.1.255 scope global eth0
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fea9:27d3/64 scope link
        valid_lft forever preferred_lft forever
root@vbox:~# ip route show
192.168.1.0/24 dev eth0 proto kernel scope link src 192.168.1.1
192.168.2.4 via 192.168.1.3 dev eth0
root@vbox:~#

UbuntuServer2 [Running] - Oracle VirtualBox
File Machine View Input Devices Help
root@vbox:~# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1
    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: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 08:00:27:04:fd:b6 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.2/24 brd 192.168.1.255 scope global eth0
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fe84:fd86/64 scope link
        valid_lft forever preferred_lft forever
root@vbox:~# ip route show
192.168.1.0/24 dev eth0 proto kernel scope link src 192.168.1.2
192.168.2.4 via 192.168.1.3 dev eth0
root@vbox:~#

UbuntuServer3 [Running] - Oracle VirtualBox
File Machine View Input Devices Help
root@vbox:~# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1
    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: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 08:00:27:d1:64:c1 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.3/24 brd 192.168.1.255 scope global eth0
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fed1:64c1/64 scope link
        valid_lft forever preferred_lft forever
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 08:00:27:dc:9d:df brd ff:ff:ff:ff:ff:ff
    inet 192.168.2.3/24 brd 192.168.2.255 scope global eth1
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fedc:9ddf/64 scope link
        valid_lft forever preferred_lft forever
root@vbox:~# ip route show
192.168.1.0/24 dev eth0 proto kernel scope link src 192.168.1.3
192.168.1.2 dev eth0 scope link
192.168.2.0/24 dev eth1 proto kernel scope link src 192.168.2.3
root@vbox:~#

UbuntuServer4 [Running] - Oracle VirtualBox
File Machine View Input Devices Help
root@vbox:~# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1
    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: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 08:00:27:bf:f3:16 brd ff:ff:ff:ff:ff:ff
    inet 192.168.2.4/24 brd 192.168.2.255 scope global eth0
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:feb3:f316/64 scope link
        valid_lft forever preferred_lft forever
root@vbox:~# ip route show
192.168.1.0/24 via 192.168.2.3 dev eth0
192.168.1.2 via 192.168.2.3 dev eth0
192.168.2.0/24 dev eth0 proto kernel scope link src 192.168.2.4
root@vbox:~#
```

Рисунок 2 – Результат выполнения команды «ip a» и «ip route show» на всех машинах  
Была получена следующая топология с поднятыми интерфейсами и IP-адресами:

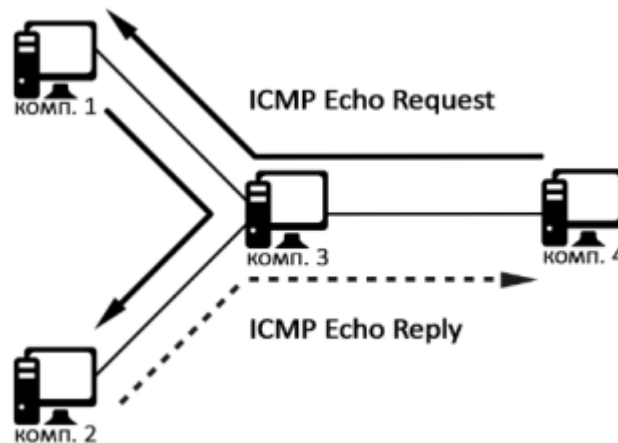
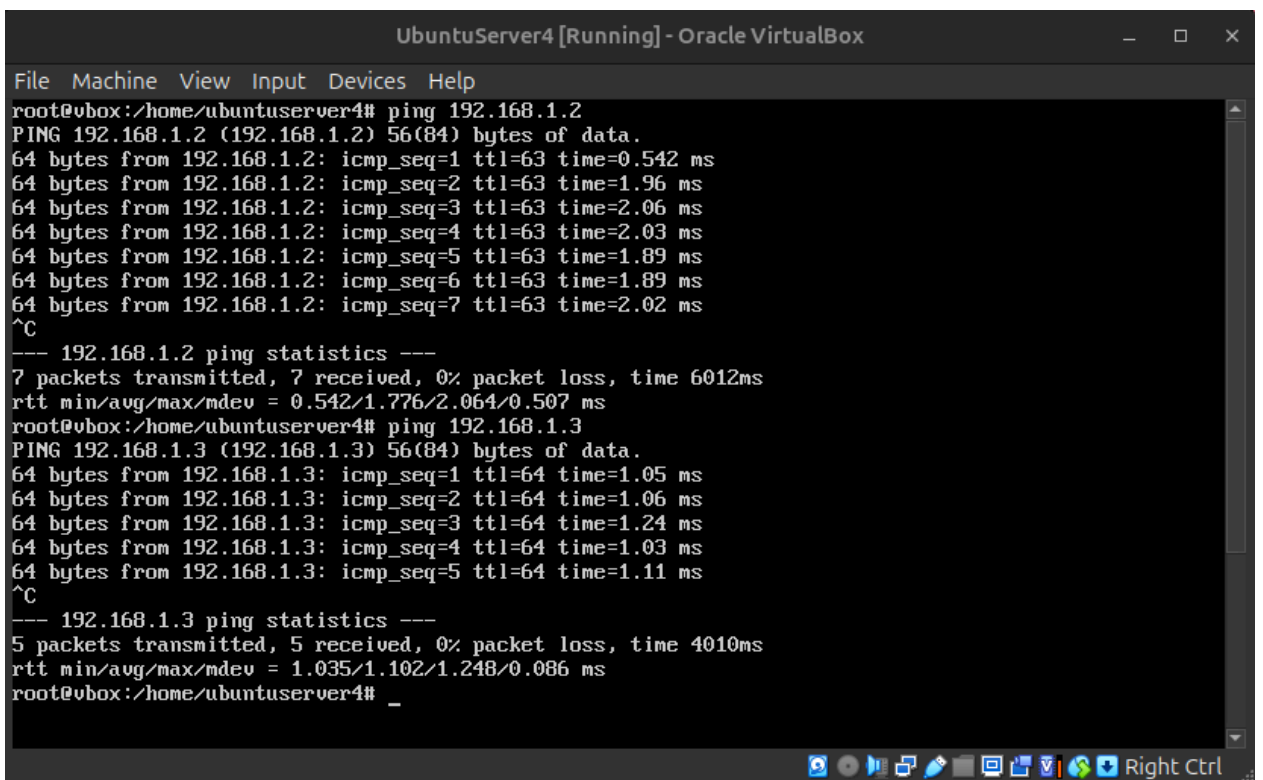


Рисунок 3 – Полученная топология

Теперь продемонстрируем, исходная машина может «пинговать» целевую машину:

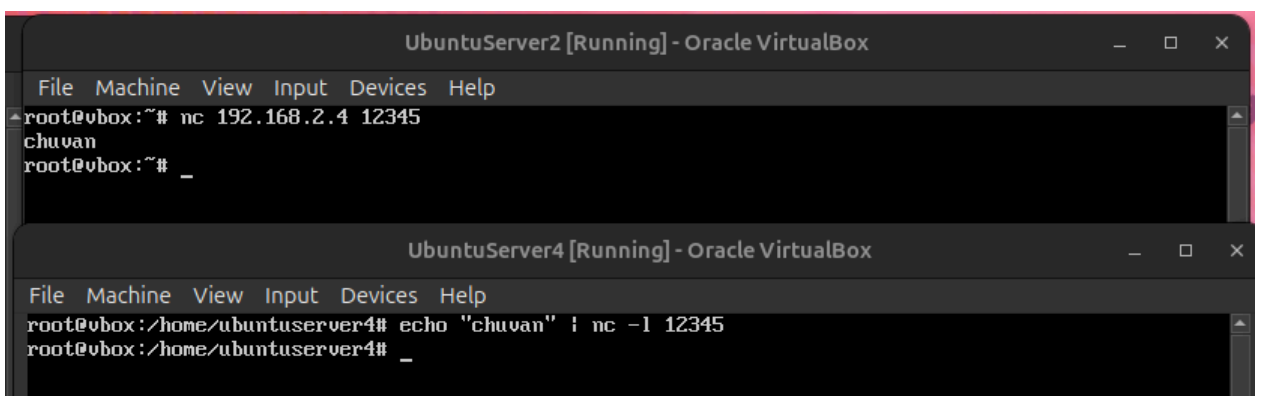


```
File Machine View Input Devices Help
root@vbox:/home/ubuntuuserver4# ping 192.168.1.2
PING 192.168.1.2 (192.168.1.2) 56(84) bytes of data.
64 bytes from 192.168.1.2: icmp_seq=1 ttl=63 time=0.542 ms
64 bytes from 192.168.1.2: icmp_seq=2 ttl=63 time=1.96 ms
64 bytes from 192.168.1.2: icmp_seq=3 ttl=63 time=2.06 ms
64 bytes from 192.168.1.2: icmp_seq=4 ttl=63 time=2.03 ms
64 bytes from 192.168.1.2: icmp_seq=5 ttl=63 time=1.89 ms
64 bytes from 192.168.1.2: icmp_seq=6 ttl=63 time=1.89 ms
64 bytes from 192.168.1.2: icmp_seq=7 ttl=63 time=2.02 ms
^C
--- 192.168.1.2 ping statistics ---
7 packets transmitted, 7 received, 0% packet loss, time 6012ms
rtt min/avg/max/mdev = 0.542/1.776/2.064/0.507 ms
root@vbox:/home/ubuntuuserver4# ping 192.168.1.3
PING 192.168.1.3 (192.168.1.3) 56(84) bytes of data.
64 bytes from 192.168.1.3: icmp_seq=1 ttl=64 time=1.05 ms
64 bytes from 192.168.1.3: icmp_seq=2 ttl=64 time=1.06 ms
64 bytes from 192.168.1.3: icmp_seq=3 ttl=64 time=1.24 ms
64 bytes from 192.168.1.3: icmp_seq=4 ttl=64 time=1.03 ms
64 bytes from 192.168.1.3: icmp_seq=5 ttl=64 time=1.11 ms
^C
--- 192.168.1.3 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4010ms
rtt min/avg/max/mdev = 1.035/1.102/1.248/0.086 ms
root@vbox:/home/ubuntuuserver4# _
```

Рисунок 4 – С машины 4 выполните команду **ping** на машину 2

## 1.2 Работа с утилитой nc

Для тестирования работы утилиты nc выберем comr4 и comr2 (в нашей нотации) как самые удаленные. Итак, проверим ее работу:



```
File Machine View Input Devices Help
root@vbox:~# nc 192.168.2.4 12345
chuvan
root@vbox:~# _

File Machine View Input Devices Help
root@vbox:/home/ubuntuuserver4# echo "chuvan" | nc -l 12345
root@vbox:/home/ubuntuuserver4# _
```

Рисунок 5 - Выполнили передачу сообщения с помощью утилиты nc

## **1.3 Настройка iptables**

### **1.3.1 Запрет передачи TCP на порт Netcat**

Предполагаемый порт Netcat — 12345.

Команда на машине А или В:

```
sudo iptables -A OUTPUT -p tcp --dport 12345 -j DROP
```

### **1.3.2 Запрет приема UDP с порта Netcat**

Команда на машине А или В:

```
sudo iptables -A INPUT -p udp --sport 12345 -j DROP
```

### **1.3.3 Запрет передачи пакетов с IP машины А**

Предположим, IP машины А — 192.168.1.2

Команда на машине В:

```
sudo iptables -A OUTPUT -s 192.168.1.2 -j DROP
```

### **1.3.4 Запрет приема пакетов на IP машины В**

Предположим, IP машины В — 192.168.2.4.

Команда на машине В:

```
sudo iptables -A INPUT -d 192.168.2.4 -j DROP
```

### **1.3.5 Запрет ICMP-пакетов с размером >1000 байт и TTL <10**

Команда на машине А или В:

```
sudo iptables -A INPUT -p icmp -m length --length 1001: -m ttl --ttl-lt 10 -j DROP
```

```
sudo iptables -A OUTPUT -p icmp -m length --length 1001: -m ttl --ttl-lt 10 -j DROP
```

## 1.4 Проверка работоспособности правил iptables

### 1.4.1 Запрет передачи TCP на порт 12345

Машина 2 работает как сервер Netcat на порту 12345. Машина 4 пытается установить TCP-соединение с машиной 2, но iptables на машине 4 блокирует это соединение

На машине 2 (Server): Запустите Netcat для прослушивания на порту 12345:

```
nc -l 12345
```

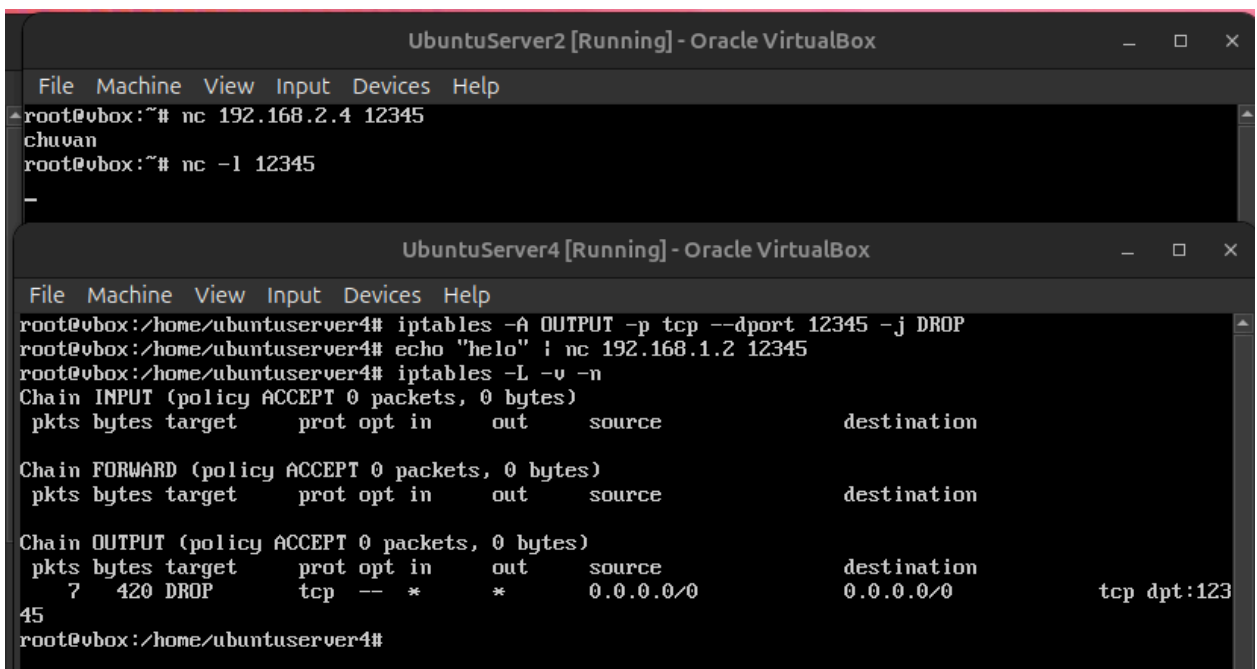
На машине 4 (Client): Добавьте правило для блокировки TCP на порт 12345:

```
sudo iptables -A OUTPUT -p tcp --dport 12345 -j DROP
```

На машине 4 (Client): Попробуйте отправить сообщение на машину 2:

```
echo "hello" | nc 192.168.1.2 12345
```

Машина 2 не получит сообщение, так как TCP-пакет будет заблокирован



```
UbuntuServer2 [Running] - Oracle VirtualBox
File Machine View Input Devices Help
root@vbox:~# nc 192.168.2.4 12345
chuvan
root@vbox:~# nc -l 12345
-

UbuntuServer4 [Running] - Oracle VirtualBox
File Machine View Input Devices Help
root@vbox:/home/ubuntuuserver4# iptables -A OUTPUT -p tcp --dport 12345 -j DROP
root@vbox:/home/ubuntuuserver4# echo "helo" | nc 192.168.1.2 12345
root@vbox:/home/ubuntuuserver4# iptables -L -v -n
Chain INPUT (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target    prot opt in     out     source                   destination

Chain FORWARD (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target    prot opt in     out     source                   destination

Chain OUTPUT (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target    prot opt in     out     source                   destination
  7    420 DROP      tcp  --  *      *        0.0.0.0/0                0.0.0.0/0                tcp dpt:12345
root@vbox:/home/ubuntuuserver4#
```

Рисунок 6 - Пример запрета передачи TCP на порт 12345

### 1.4.2 Запрет приема UDP с порта 12345

Машина 4 работает как сервер Netcat и принимает данные UDP от машины 2, но iptables на машине 4 блокирует входящие UDP-пакеты с порта 12345.



На машине 4 (Server): Запустите Netcat для прослушивания UDP на порту 12345:

```
nc -u -l 12345
```

На машине 4 (Server): Добавьте правило для блокировки UDP-пакетов с порта 12345:

```
sudo iptables -A INPUT -p udp --sport 12345 -j DROP
```

На машине 2 (Client): Отправьте сообщение UDP на машину 4:

```
echo "hi" | nc -u 192.168.2.4 12345
```

Машина 4 не получит сообщение от машины 2, так как UDP-пакет будет заблокирован.

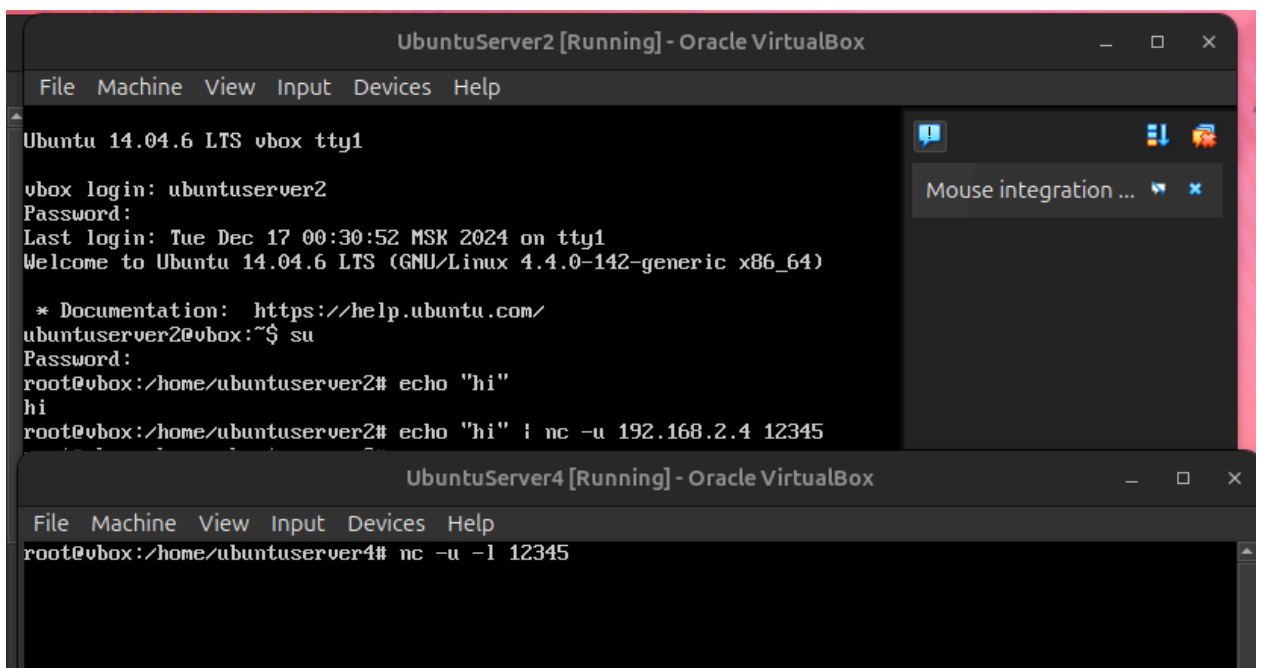


Рисунок 7 - Пример запрета передачи UDP на порт 12345

### 1.4.3 Запрет передачи пакетов от IP машины А (192.168.1.2)

Машине 2 (192.168.1.2) запрещается отправлять любые пакеты на машину 4.

**На машине 4:** Добавьте правило для блокировки пакетов от машины 2:

```
sudo iptables -A INPUT -s 192.168.1.2 -j DROP
```

**На машине 2:** Отправьте пинг на машину 4:

```
ping 192.168.2.4
```

Машина 4 не ответит на пинг от машины 2.

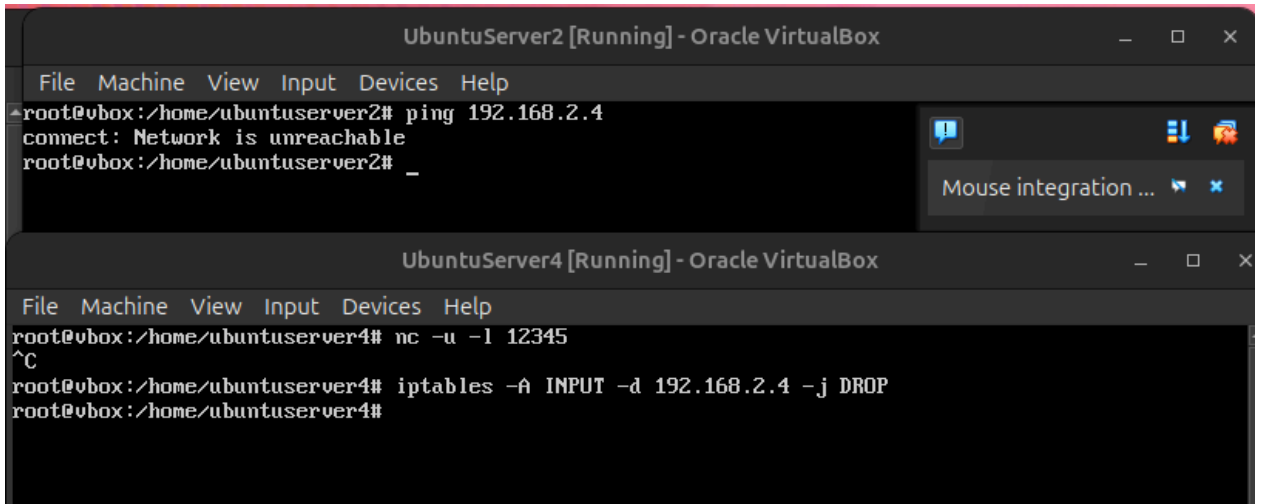


Рисунок 8 - Пример запрета передачи пакетов от IP машины А (192.168.1.2)

#### 1.4.4 Запрет приема пакетов на IP машины В (192.168.2.4)

Запрещаются все входящие пакеты на IP машины 4.

**На машине 4:** Добавьте правило:

```
sudo iptables -A INPUT -d 192.168.2.4 -j DROP
```

**На машине 2:** Отправьте пинг на машину 4:

```
ping 192.168.2.4
```

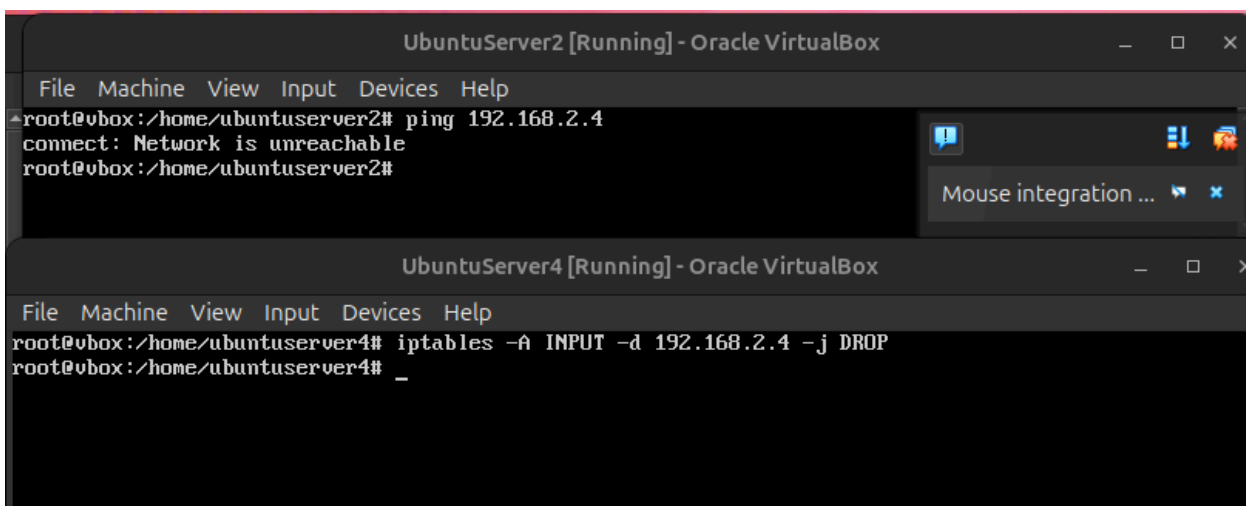


Рисунок 9 -Запрет приема пакетов на IP машины В (192.168.2.4)

#### 1.4.5 Запрет ICMP-пакетов больше 1000 байт и TTL меньше 10

Запрещаются ICMP-пакеты размером >1000 байт и TTL <10.

**На машине 4:** Добавьте правило:

```
sudo iptables -A INPUT -p icmp -m length --length 1001: -m ttl --ttl-lt 10 -j DROP
```

```
sudo iptables -A OUTPUT -p icmp -m length --length 1001: -m ttl --ttl-lt 10 -j DROP
```

**На машине 2:** Отправьте большой ICMP-пакет:

```
ping 192.168.2.4 -s 1200 -t 9
```

Машина 4 проигнорирует этот ICMP-пакет.

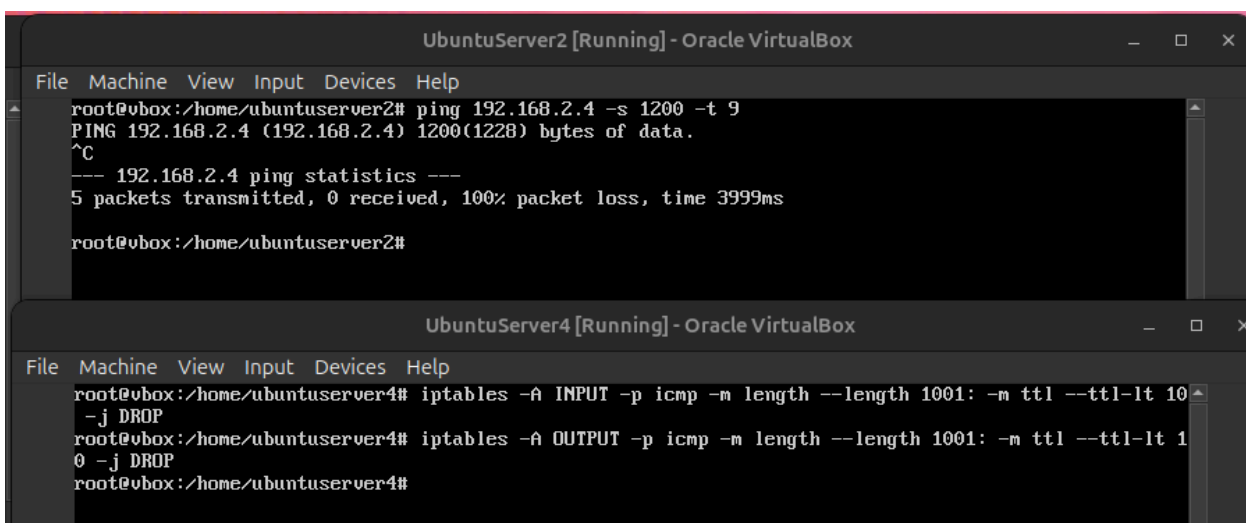


Рисунок 10 - Запрет ICMP-пакетов больше 1000 байт и TTL меньше 10

## **ЗАКЛЮЧЕНИЕ**

В ходе работы были изучены основные аспекты сетевого уровня модели OSI, включая его функции и протоколы, применяемые в маршрутизируемых сетях. Выполнена базовая настройка сетевых интерфейсов для обеспечения связности между компьютерами. Настроены таблицы маршрутизации для корректной передачи пакетов как в сетях IPv4, так и IPv6. Исследован и применен анализ сетевого трафика с помощью утилиты tcpdump, а также изучена технология NAT и её роль в маршрутизации пакетов.