Санкт-Петербургский национальный исследовательский УНИВЕРСИТЕТ информационных технологий, механики и оптики

**Факультет программной инженерии и компьютерной техники**

**ЛАБОРАТОРНАЯ РАБОТА № 4**

**ПО ДИСЦИПЛИНЕ «СЕТИ ЭВМ И ТЕЛЕКОММУНИКАЦИИ»**

Выполнили: Нгу Фыонг Ань

Группа: P3310

Преподаватель: Шинкарук Д.Н.

Санкт-Петербург

2020

# Цели работы

Изучение основных методов настройки маршрутизируемых компьютерных сетей на примере сети, состоящей из компьютеров под управлением ОС Linux.

**Вариант**: V1 = 3

# Общая часть

# 

## Настройка сети

Для реализации вполне неоднозначного и запутанного задания была создана сеть из четырёх хостов, состоящих в одной внутренней сети.

**Настраиваем сеть:**

**На комп.3:**

# Up interfaces.

ip link set eth1 up

ip link set eth2 up

ip link set eth3 up

# Flush old ip addresses from all interfaces.

ip a flush eth1

ip a flush eth2

ip a flush eth3

# Set new ip addresses for each interface.

ip a add 3.3.1.3/24 dev eth1

ip a add 3.3.2.3/24 dev eth2

ip a add 3.3.4.3/24 dev eth3

# Enable ip routing.

echo 1 > /proc/sys/net/ipv4/ip\_forward

**На комп.4**

ip link set eth1 up

ip a flush eth1

ip a add 3.3.4.4/24 dev eth1

route add default gw 3.3.4.3

**На комп.2**

ip link set eth1 up

ip a flush eth1

ip a add 3.3.2.2/24 dev eth1

route add default gw 3.3.2.3

**На комп.1**

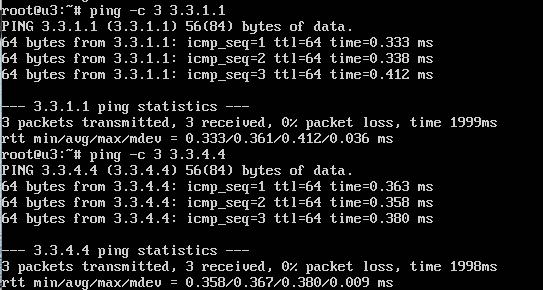
ip link set eth1 up

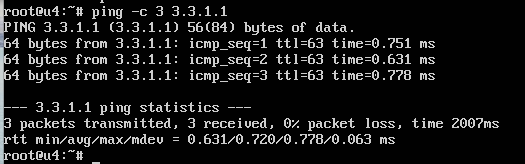
ip a flush eth1

ip a add 3.3.1.1/24 dev eth0

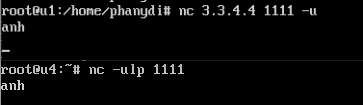
route add default gw 3.3.1.3

**1, ping**





**2, nc**



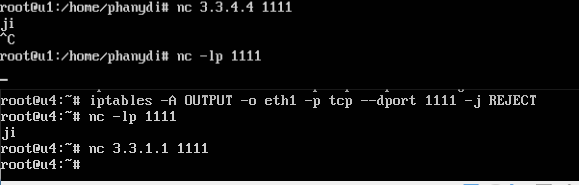
**3, iptables**

1. **Запретить передачу только тех пакетов, которые отправлены на TCP-порт, заданный в настройках утилиты nc**

**На комп.4**

iptables -F

iptables -A OUTPUT -o eth1 -p tcp --dport 1111 -j REJECT

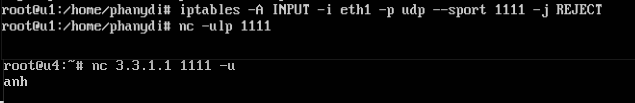


**2. Запретить приём только тех пакетов, которые отправлены с UDP- порта утилиты nc.**

**На комп.1**

iptables -F

iptables -A INPUT -i eth1 -p udp --sport 1111 -j REJECT

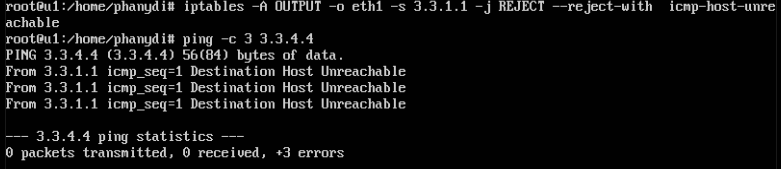


**3. Запретить передачу только тех пакетов, которые отправлены с IP- адреса компьютера Б.**

**На комп.1**

iptables -F

iptables -A OUTPUT -o eth1 -s 3.3.1.1 -j REJECT --reject-with icmp-host-unreachable



**4. Запретить приём только тех пакетов, которые отправлены на IP- адрес компьютера А.**

**На комп.4**

iptables -F

iptables -A INPUT -i eth1 -d 3.3.4.4 -j REJECT --reject-with icmp-host-unreachable



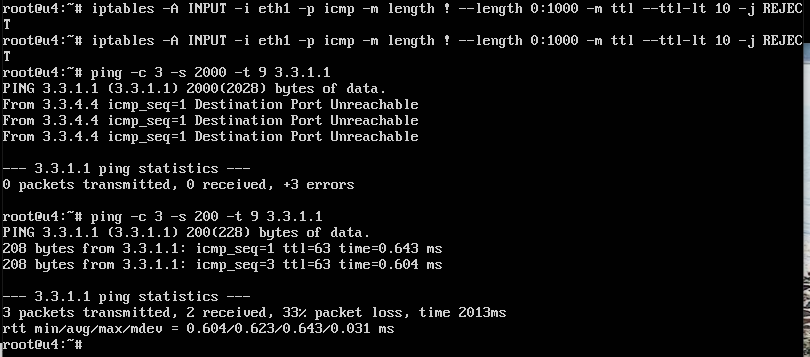
**5. Запретить приём и передачу ICMP-пакетов, размер которых превышает 1000 байт, а поле TTL при этом меньше 10.**

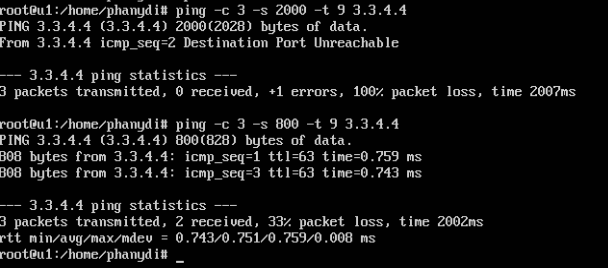
**На комп.4**

iptables -F

iptables -A INPUT -i eth1 -p icmp -m length ! --length 0:1000 -m ttl --ttl-lt 10 -j REJECT

iptables -A OUTPUT -o eth1 -p icmp -m length ! --length 0:1000 -m ttl --ttl-lt 10 -j REJECT

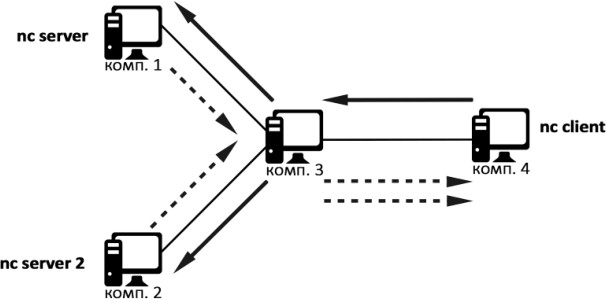




# V1: вариант 3

На компьютерах 1 и 2 запущены два nc-сервера. На компьютере 4 запущен nc-клиент, который осу- ществляет подключение к IP-адресу, который не представлен в сети. На компьютее 3 настроен DNAT и дублирование трафика таким образом, чтобы оба nc-сервера получали запросу от nc-клиента. Ответные сообщение от серверов приходят отбратно к клиенту. Клиент и серверы работают по протоколу UDP.

## Топология сети



**Рис. 1.** Топология сети и схема прохождения трафика

## Настройка IPv4

Комп.1

ip link set eth1 up

ip a flush dev eth1

ip a add 3.3.1.2/30 dev eth1

route add default gw 3.3.1.1

Комп. 2.

ip link set eth1 up

ip a flush dev eth1

ip a add 3.3.2.2/30 dev eth1

route add default gw 3.3.2.1 iptables -t nat -F

iptables -t nat -A PREROUTING -p udp -d 3.3.12.12 -j DNAT --to-dest 3.3.2.2

Комп. 3.

ip link set eth1 up

ip link set eth2 up

ip link set eth3 up

ip a flush dev eth1

ip a flush dev eth2

ip a flush dev eth3

ip a add 3.3.1.1/30 dev eth1

ip a add 3.3.2.1/30 dev eth2

ip a add 3.3.3.1/30 dev eth3

echo 1 > /proc/sys/net/ipv4/ip\_forward

iptables -t nat -F

iptables -t nat -A PREROUTING -i eth3 -d 3.3.12.12 -p udp

-j DNAT --to-destination 3.3.1.2 iptables -t nat -A POSTROUTING -d 3.3.3.2 -p udp

-j SNAT --to-source 3.3.12.12

iptables -t mangle -F

iptables -t mangle -A PREROUTING -i eth3 -d 3.3.12.12 -p udp

-j TEE --gateway 3.3.2.2

iptables -t raw -F

iptables -t raw -A PREROUTING -s 3.3.12.12 -p udp -j NOTRACK

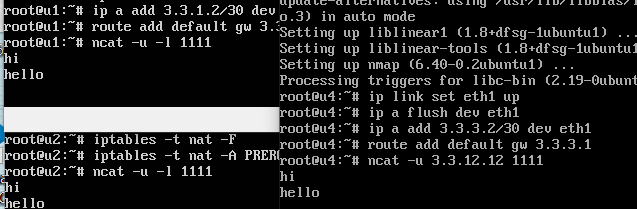
Комп 4.

ip link set eth1 up

ip a flush dev eth1

ip a add 3.3.3.2/30 dev eth1

route add default gw 3.3.3.1



## Настройка IPv6

Комп 1.

ip link set eth1 up

ip -6 a add ::ffff:3:3:1:2/126 dev eth1

route -A inet6 add default gw ::ffff:3:3:1:1

Комп 2.

ip link set eth1 up

ip -6 a add ::ffff:3:3:2:2/126 dev eth1

route -A inet6 add default gw ::ffff:3:3:2:1 ip6tables -t nat -F

ip6tables -t nat -A PREROUTING -d ::ffff:3:3:12:12/128 -i eth1 -p udp

-j DNAT --to-destination ::ffff:3:3:2:2

Комп 3.

ip link set eth1up

ip link set eth2 up

ip link set eth3 up

ip -6 a add ::ffff:3:3:1:1/126 dev eth1

ip -6 a add ::ffff:3:3:2:1/126 dev eth2

ip -6 a add ::ffff:3:3:3:1/126 dev eth3

echo 1 | tee /proc/sys/net/ipv6/conf/\*/forwarding > /dev/null

echo 0 | tee /proc/sys/net/ipv6/conf/\*/accept\* > /dev/null

ip6tables -t nat -F

ip6tables -t nat -A PREROUTING -d ::ffff:3:3:12:12/128 -i eth3 -p udp

-j DNAT --to-destination ::ffff:3:3:1:2

ip6tables -t nat -A POSTROUTING -d ::ffff:3:3:3:2/128 -p udp

-j SNAT --to-source ::ffff:3:3:12:12

ip6tables -t mangle -F

ip6tables -t mangle -A PREROUTING -d ::ffff:3:3:12:12/128 -i eth3 -p udp

-j TEE --gateway ::ffff:3:3:2:2

ip6tables -t raw -F

ip6tables -t raw -A PREROUTING -s ::ffff:3:3:12:12 -p udp -j NOTRACK

Комп 4.

ip link set eth1 up

ip -6 a add ::ffff:3:3:3:2/126 dev eth1

route -A inet6 add default gw ::ffff:3:3:3:1

# Вывод

В ходе лабораторной работы были изучены основы администрирования маршрутизируемых компьютерных сетей в Ubuntu server, а так же познак bb омился с установкой виртуальных машин и их настройкой.

.