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

**Факультет цифровых трансформаций**

**Дисциплина:**

«**Телекоммуникационные системы и технологии**»

**ПРАКТИЧЕСКАЯ РАБОТА №6**

«Трансляция адресов в ОС Linux»

**Выполнили:**

Гаджиев С. И., Васильков Д. A., Лавренов Д. А. M3304

*\_\_\_\_\_\_\_\_\_\_Pulpy\_\_\_\_\_\_\_\_*

(подпись)

**Проверила:**

Дяченко Екатерина Олеговна

*\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

(отметка о выполнении)

*\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

(подпись)

***Цель работы:***

Закрепить понимание принципов работы NAT и firewall, а также сформировать начальные навыки в конфигурировании NAT и Firewall на платформе и Linux;

***Требования:***

Установленная на компьютере среда виртуализации ORACLE Virtual Box с виртуальной машиной Linux Cent OS 7 (выполнять работу можно в любой ОС Linux, но все описания будут даваться для CentOS 7).

***Инструментальные средства:***

*Утилиты: sysctl systemctl ip ping tcpdump useradd ss netstat iptables iptables-save iptables-restore*

*Файлы: /etc/ssh/sshd\_config*

*Утилиты работы с текстом: echo, grep, sed*

*Редакторы: vi, nano*

***Порядок выполнения работы:***

***Примечание:*** вместо iptables можно выполнить работу на nftables.

***Часть 1. Подготовка и проверка конфигурации***

В VirtualBox:

1. Запустите виртуальную машину Linux. Удалите на хосте сервис firewalld.

Примечание: можно использовать утилиту systemctl. Для остановки сервиса используйте команду systemctl stop, для запуска systemctl start, для запрета автозапуска systemctl disable, для включения автозагрузки сервиса systemctl enable.

2. Установите iptables (пакет называется iptables-services), настройте автозапуск iptables.

3. Сделайте связанный клон виртуальной машины. Одну машину назовите c7-1, другой с7-2

4. Для виртуальной машины c7-1 добавьте второй сетевой интерфейс.

5. Подключите сетевой интерфейс c7-2 и новый сетевой интерфейс с7-1 к внутренней сети intnet.

6. Подключите исходный сетевой интерфейс с7-1 к NAT.

7. Для внутренней сети задайте для машин с7-1 и c7-2 адреса 10.0.0.1 и 10.0.0.2 с маской 255.255.255.0.

8. Для исходного интерфейса c7-1 оставьте получение адреса автоматически от dhcp сервара VirtualBox

9. Для обоих хостов отключите использование ipv6.

10. Задайте имена хостов, советующие именам виртуальных машин. Изменить имя хоста можно изменить с помощью утилиты hostnamectl

11. Проверьте доступность хостов по внутренней сети и доступность внешней сети на хосте c7-1.

12. Убедитесь, что на c7-2 в качестве шлюза по умолчанию задан адрес c7-1.

13. В качестве адреса DNS сервера на c7-2 указать адрес 8.8.8.8 и 77.88.8.1

14. Убедитесь, что на машине c7-1 параметры ядра позволяют передавать сетевые пакеты между сетевыми интерфейсами.

Должна получиться следующая схема:

Изображение выглядит как текст, снимок экрана, диаграмма, линия

Автоматически созданное описание

***Часть 2. Создание пользователей и настройка OpenSSH Server (sshd).***

1. На хосте c7-2 создайте пользователя с именем FIOuser, где FIO – ваши инициалы.

2. Редактируя файл /etc/ssh/sshd\_config, настройте ssh сервер так, чтобы (!):

a. Пользователю root нельзя было бы входить по ssh

b. Количество попыток ввода неверного пароля = 2

c. Время ожидания авторизации = 30 секундам.

d. Отключить определение имен хостов по DNS

3. После изменения конфигурации перезапустите сервис sshd.

4. С машины с7-1 подключитесь к с7-2 по ssh, используя новую учетную запись.

***Часть 3. Настройка NAT на шлюзе***

1. На хосте c7-1 разрешите передачу IP пакетов между интерфейсами.

2. Настройте на хосте клиентский NAT (действие SNAT или MASQUERADE), так чтобы внешняя сеть стала доступна из внутренней сети.

3. Настройте публикацию порта tcp\22 на хосте с7-2 на порту tcp\55022 на внешнем сетевом интерфейсе c7-1.

4. Используя утилиту iptables-save выведите автоматически созданные правила в текстовый файл /etc/sysconfig/iptables . Определите назначение каждой строки.

5. Подключитесь к ssh серверу на c7-2 с вашей реальной операционной системы (предварительно настройте публикацию портов в NAT в VirtualBox).

6. Проверьте командой ping с хоста c7-2 доступность любого работающего сервиса в Интернет (например адреса 8.8.8.8 или 77.88.8.1). Если хост недоступен, а подключение в п.5 удалось установить, то отредактируйте файл /etc/sysconfig/iptables, изменив правила так, чтобы запросы утилиты ping проходили. Для применения правил можно просто перезапустить сервис (systemctl reload или restart). Корректнее использовать iptables-restore (текущие соединения не сбрасываются).

***Часть 4. Установка дополнительного ПО***

1. На хосте c7-1 установите консольный броузер (lynx или links) и утилиту nmap.

2. На хосте c7-2 установите Web-сервер lighttpd, запустите его и разрешите автоматический запуск. Определите на каком сокете запускается сервер. Если по умолчанию он стартует на сокете ipv6, то измените конфигурационный файл Web-сервера, так, чтобы сервер запускался на ipv4.

3. С хоста c7-1 с помощью утилиты nmap проверьте какие порты открыты на хосте c7-2 (!).

4. На хосте c7-1 с помощью консольного броузера попробуйте открыть сайт на 10.0.0.2. Если сайт не отрывается, отредактируйте правила iptables, так, чтобы доступ к web-серверу был разрешен. Проверьте, что доступ появился.

***Часть 5. Исследование соединений***

1. На хосте c7-2 с помощью команд ss, netstat и lsof (любой из команд) выведите на консоль информацию о (!):

a. Открытых соединениях.

b. Открытых сетевых сокетах, ждущих подключение.

2. На машине c7-1 с помощью утилиты tcpdump выведите на разных консолях трафик с внутреннего и внешнего интерфейса, так чтобы отображались адреса отправителя и получателя, номера портов отправителя и получателя,

3. Запустите с хоста c7-2 передачу 5 TCP сегментов до хоста ya.ru с помощью утилиты mtr.

4. Наблюдая за консольными выводами tcpdump определите, как были изменены исходящие сообщения при трансляции адресов (!).

5. Закройте все ssh сессии с машиной c7-2

6. На машине c7-2 запустите с помощью утилиты tcpdump выведите консоль трафик, так чтобы отображались адреса отправителя и получателя, номера портов отправителя и получателя и флаги tcp (!).

7. Подключитесь с основной операционной системы к хосту c7-2 по ssh.

8. Определите какие флаги использовались при установлении соединения, как менялось значение полей ack и syn после начала передачи данных (!).

*Примечание*: значения флагов в выводе tcpdump следующие [.] - ACK (Acknowledgment), [S] - SYN (Start Connection); [P] - PSH (Push Data); [F] - FIN (Finish Connection); [R] - RST (Reset Connection); [S.] - SYN-ACK (SynAcK Packet)

***Часть 6. Настройка шлюза***

1. Задайте политики по умолчанию для цепочек INPUT и FORWARD – запрет передачи.

2. Добавьте правила, которые бы

a. Разрешали подключение к опубликованному порту ssh сервера c7-2 из IP сети реального хоста

b. Разрешили подключение из внутренней сети к DNS только на 8.8.8.8 и 77.88.8.1

c. Разрешали доступ из внутренней сети к протоколам POP3 (tcp 110), Web (tcp 80, 443, 8080), ssh (tcp 22).

d. Разрешили доступ к сервисам SMTP (tcp 25) на любом хосте сети вашего основного компьютера.

e. Запрещают любой трафик с хостов 192.56.0.11 и с подсети 14.12.44.0/18 как непосредственно на машину c7-1, так и во внутреннюю сеть.

f. Запрещают доступ к ssh серверу на c7-1 из внешней сети.

g. Разрешает доступ к ssh серверу на c7-1 из внутренней сети.

h. Разрешает icmp эхо запросы из внутренней сети наружу только на хост 8.8.8.8

i. Запрещает хосту c7-1 давать icmp эхо ответы, но при этом сохраняет возможность с самого хоста c7-1 делать icmp это запросы и получать на них ответы.

***Часть 7. Доступ через ssh к защищенным сервисам***

Используя возможности протокола ssh сделайте так, чтобы на основном компьютере Web-сервер с хоста c7-2 был доступен по адресу 127.0.0.80:8888.

***Артефакты:***

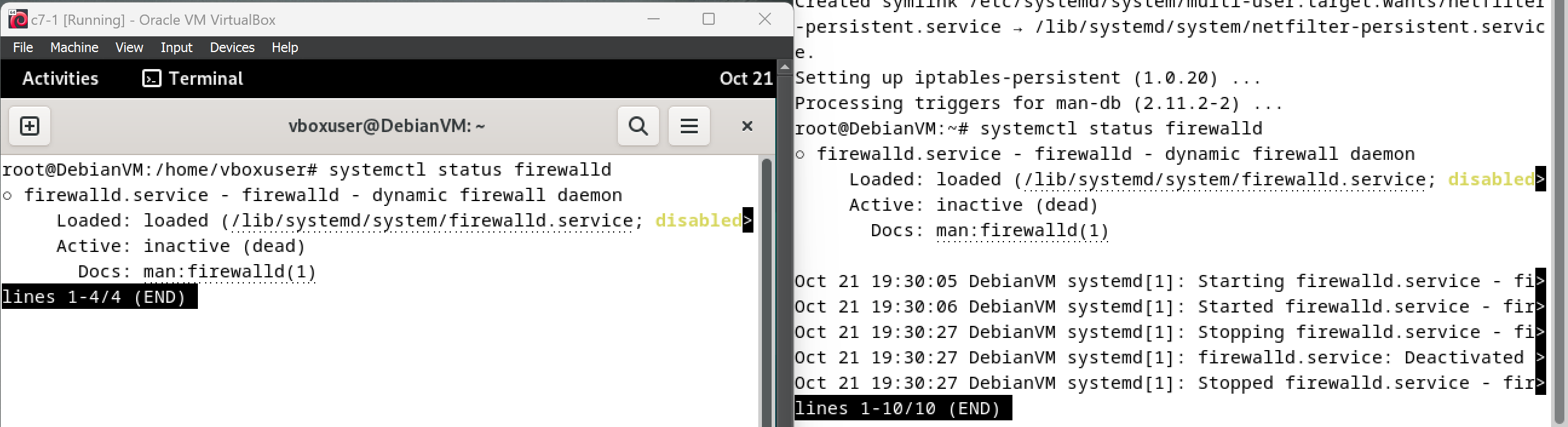
1. Измененные параметры sshd из Части 2.

Часть №1

Отключили на с7-1 и c7-2 firewalld:

*systemctl stop firewalld*

*systemctl disable firewalld*



Установили на с7-1 и c7-2 iptables и настроили автозапуск:

apt install iptables iptables-persistent -y

systemctl enable netfilter-persistent

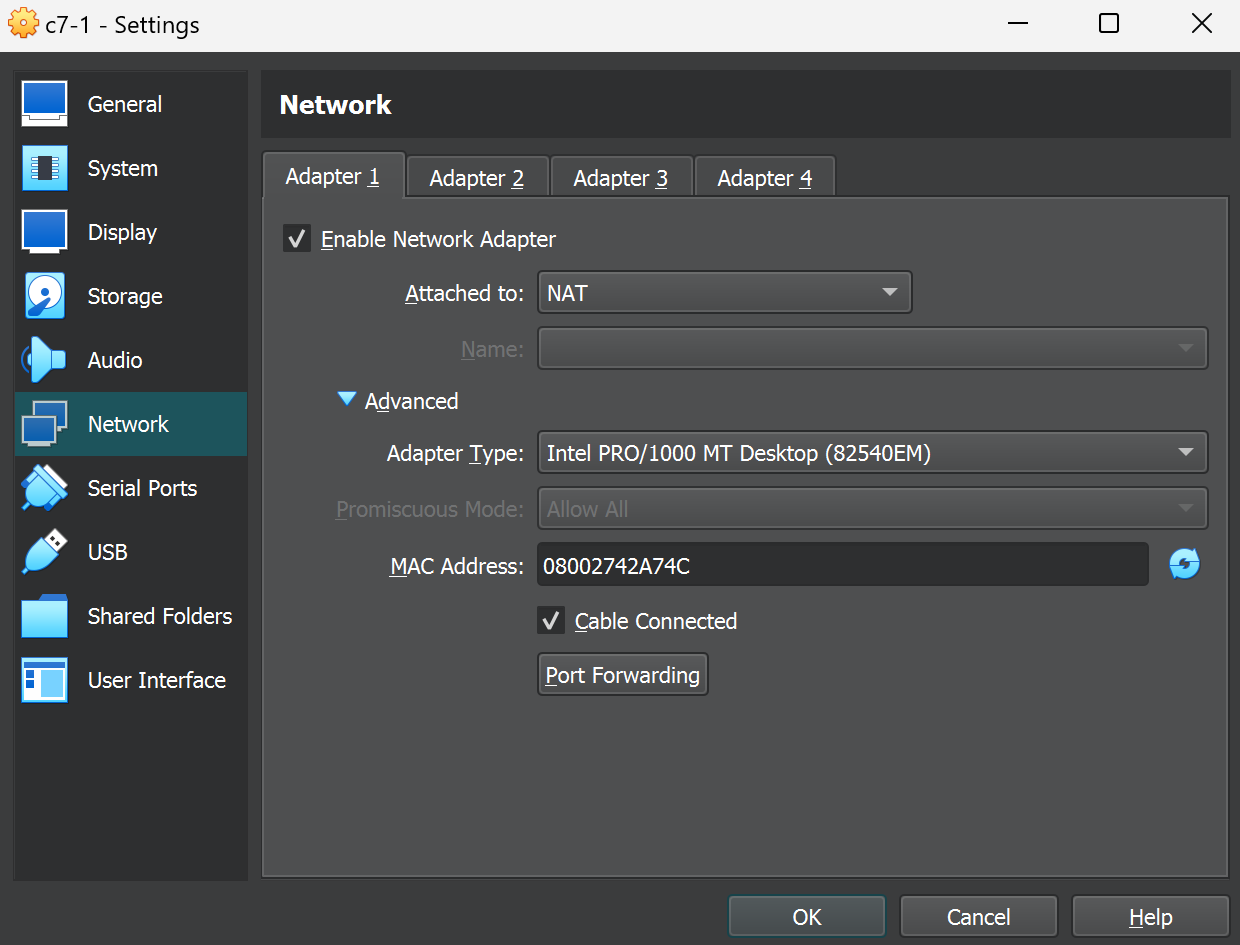
systemctl start netfilter-persistent

Изображение выглядит как текст, снимок экрана, Шрифт, документ

Автоматически созданное описание

Настройка сетевых интерфейсов для c7-1 и с7-2:

Для c7-1:



Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Автоматически созданное описание

Для с7-2:

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Автоматически созданное описание

Для внутренней сети зададим для машин с7-1 и c7-2 адреса 10.0.0.1 и 10.0.0.2:

с7-1

Изображение выглядит как текст, Шрифт, белый, алгебра

Автоматически созданное описаниеИзображение выглядит как текст, Шрифт, снимок экрана, белый

Автоматически созданное описание

с7-2

Изображение выглядит как текст, Шрифт, снимок экрана, белый

Автоматически созданное описание

Отключили IPv6 на обеих машинах:

sudo sysctl -w net.ipv6.conf.all.disable\_ipv6=1

sudo sysctl -w net.ipv6.conf.default.disable\_ipv6=1

Изменение имени хоста:

C7-1:

sudo hostnamectl set-hostname c7-1

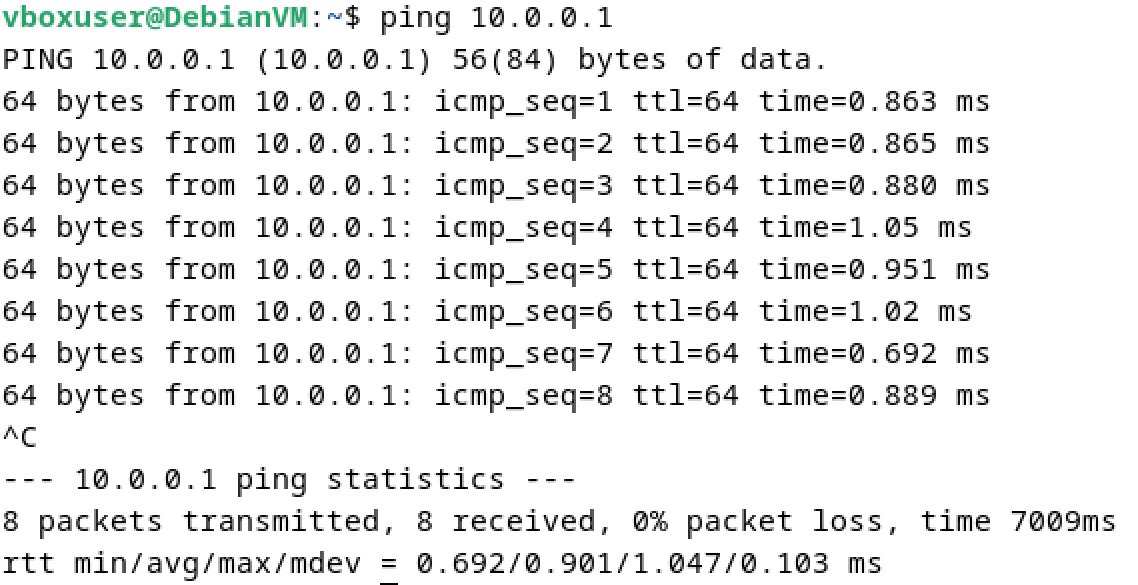
C7-2:

sudo hostnamectl set-hostname c7-2

Проверка доступности по внутренней сети:

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание



Настройка шлюза по умолчанию на c7-2:

sudo ip route add default via 10.0.0.1

Настройка DNS серверов на c7-2:

sudo nano /etc/resolv.conf

nameserver 8.8.8.8

nameserver 77.88.8.1

Разрешение передачи пакетов между интерфейсами на c7-1:

sudo sysctl -w net.ipv4.ip\_forward=1

Часть 2

Создание пользователя на c7-2:

sudo useradd FIOuser

sudo passwd FIOuser (пароль 239)

Настройка SSH на c7-2:

sudo nano /etc/ssh/sshd\_config

Внесли изменения:

**Запрет для root**:

PermitRootLogin no

**Количество попыток ввода пароля**:

MaxAuthTries 2

**Время ожидания авторизации**:

LoginGraceTime 30

**Отключение определения имен хостов по DNS**:

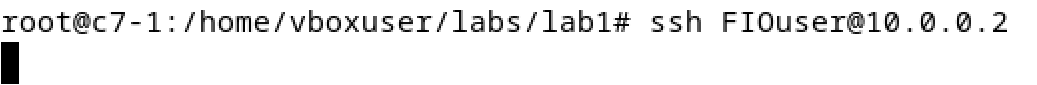
UseDNS no

Перезапуск службы SSH: (ДОДЕЛАТЬ)

sudo systemctl restart sshd

Подключение по SSH с c7-1 на c7-2:

ssh [FIOuser@10.0.0.2](mailto:FIOuser@10.0.0.2)



2. Итоговые файлы /etc/sysconfig/iptables с хостов c7-1 и с7-2

Часть №3

Разрешим пересылку пакетов на c7-1:

sudo sysctl -w net.ipv4.ip\_forward=1

Настройка SNAT/MASQUERADE на c7-1:

sudo iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE

Публикация порта SSH:

Настройка переадресации порта:

sudo iptables -t nat -A PREROUTING -p tcp --dport 55022 -j DNAT --to-destination 10.0.0.2:22

Сохранение iptables в файл:

sudo iptables-save > /home/vboxuser/labs/sixlabrestable

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Подключение к c7-2 через c7-1 по SSH с реальной ОС:

ssh -p 55022 FIOuser@localhost

Проверка доступа в Интернет:

ping 8.8.8.8

Изображение выглядит как текст, Шрифт, снимок экрана, белый

Автоматически созданное описание

3. Команду и консольный вывод из Части 4 п.3

Часть №4

Установка lynx и nmap на c7-1:

sudo apt install lynx nmap

Установка Web-сервера на c7-2:

sudo apt install lighttpd

Запуск и включение автозапуска:

sudo systemctl start lighttpd

sudo systemctl enable lighttpd

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Сканирование портов с помощью nmap на c7-1:

sudo nmap 10.0.0.2

Изображение выглядит как текст, Шрифт, белый, снимок экрана

Автоматически созданное описание

Открытие сайта на c7-2 с c7-1:

lynx 10.0.0.2

Изображение выглядит как текст, снимок экрана, Шрифт, Графика

Автоматически созданное описание

4. Команды и существенные части консольного вывода Части 5, п. 1,4,6,8

Часть №5

Вывод информации о соединениях на c7-2:

Открытые соединения:

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Открытые сетевые сокеты, ожидающие подключений:

Изображение выглядит как текст, чек, Шрифт, белый

Автоматически созданное описание

Дополнительная информация о сокетах:

Изображение выглядит как текст, Шрифт, снимок экрана, число

Автоматически созданное описание

Перехват трафика с помощью tcpdump на c7-1:

Открыли 2 терминала на c7-1.

В первом терминале начните прослушивание трафика на внешнем интерфейсе:

sudo tcpdump -i enp0s3 -nn

Изображение выглядит как текст, Шрифт, снимок экрана, черно-белый

Автоматически созданное описание

Во втором терминале прослушивайте трафик на внутреннем интерфейсе:

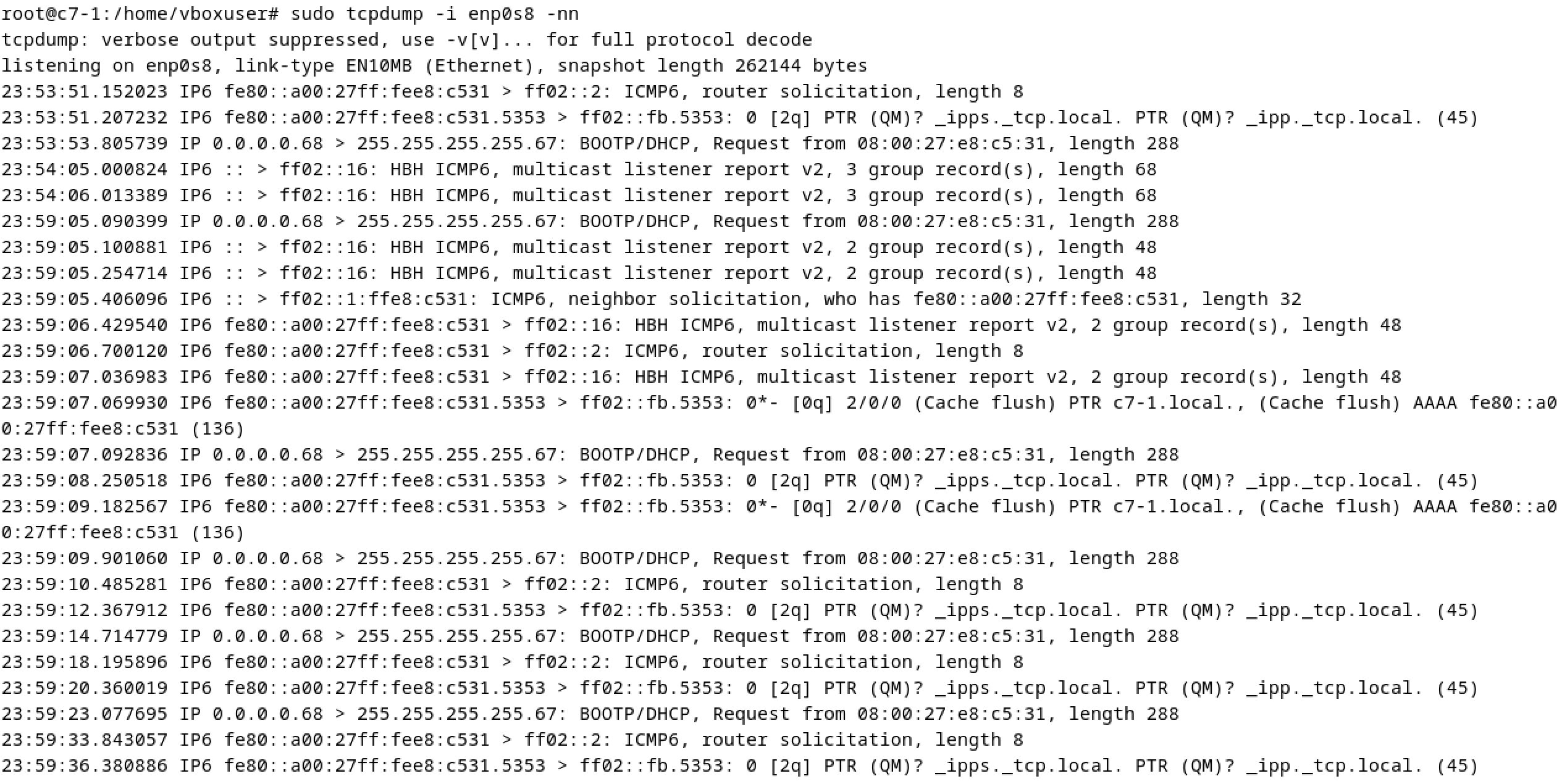
sudo tcpdump -i enp0s8 -nn

Изображение выглядит как текст, Шрифт, снимок экрана, черно-белый

Автоматически созданное описание

Отправка TCP пакетов с c7-2 на ya.ru:

mtr -c 5 ya.ru

В выводах tcpdump наблюдаем, как изменяется IP-адрес отправителя при пересылке через NAT-интерфейс на c7-1: 

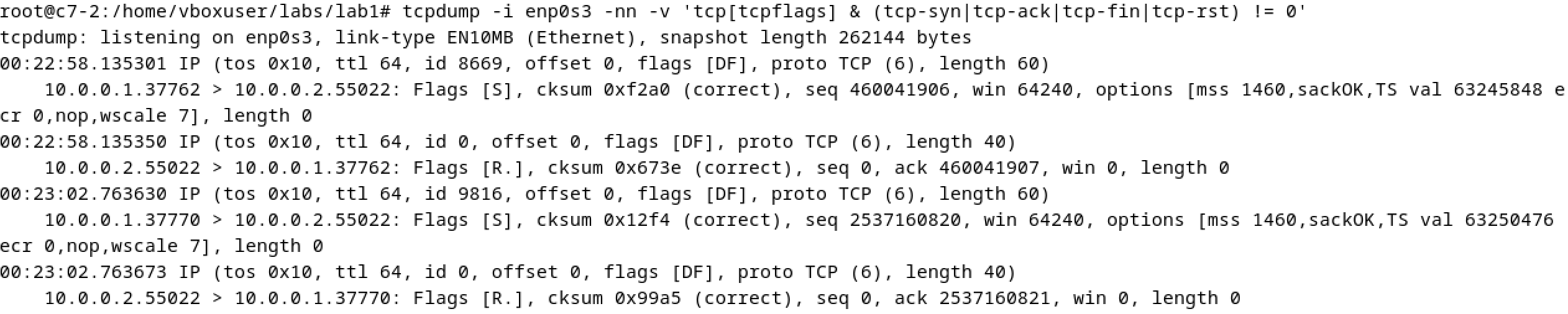
Закрытие сессий SSH и перехват трафика:

Закрываем все SSH-сессии с c7-2:

sudo pkill sshd

На c7-2 запускаем tcpdump для перехвата всех TCP-пакетов и их флагов:

sudo tcpdump -i enp0s8 -nn -v 'tcp[tcpflags] & (tcp-syn|tcp-ack|tcp-fin|tcp-rst) != 0'



Во время подключения на c7-2 в терминале с tcpdump можно увидеть последовательность установления TCP-соединения:

[S] (SYN) — начало подключения.

[S.] (SYN-ACK) — ответ на SYN.

[.] (ACK) подтверждение установления соединения.

Анализ изменения флагов ACK и SYN:

При передаче данных, вы заметили, что:

ACK - флаг подтверждает успешную передачу пакетов,

SYN и SYN-ACK происходят только при установлении соединения,

FIN и RST будут видны при завершении или сбросе соединения.

5. Текст итоговых правил iptables с c7-1.

Часть №6

На c7-1 установим политику запрета для цепочек INPUT и FORWARD:

sudo iptables -P INPUT DROP

sudo iptables -P FORWARD DROP

Добавление правил:

Разрешить подключение к SSH на c7-2 (порт 55022) из сети хоста:

sudo iptables -A INPUT -p tcp --dport 55022 -s <real\_host\_network> -j ACCEPT

Разрешить подключение к DNS (8.8.8.8 и 77.88.8.1) из внутренней сети:

sudo iptables -A FORWARD -s 10.0.0.0/24 -d 8.8.8.8 -p udp --dport 53 -j ACCEPT

sudo iptables -A FORWARD -s 10.0.0.0/24 -d 77.88.8.1 -p udp --dport 53 -j ACCEPT

Разрешить доступ из внутренней сети к веб-протоколам и POP3, SSH:

sudo iptables -A FORWARD -s 10.0.0.0/24 -p tcp -m multiport --dports 22,80,443,8080,110 -j ACCEPT

Разрешить доступ к SMTP из основной сети:

sudo iptables -A FORWARD -p tcp --dport 25 -s <real\_host\_network> -j ACCEPT

Запретить весь трафик с 192.56.0.11 и сети 14.12.44.0/18:

sudo iptables -A INPUT -s 192.56.0.11 -j DROP

sudo iptables -A INPUT -s 14.12.44.0/18 -j DROP

sudo iptables -A FORWARD -s 192.56.0.11 -j DROP

sudo iptables -A FORWARD -s 14.12.44.0/18 -j DROP

Запретить доступ к SSH на c7-1 извне:

sudo iptables -A INPUT -p tcp --dport 22 -i enp0s3 -j DROP

Разрешить доступ к SSH на c7-1 из внутренней сети:

sudo iptables -A INPUT -p tcp --dport 22 -i enp0s8 -j ACCEPT

Разрешить ICMP-запросы из внутренней сети наружу на 8.8.8.8:

sudo iptables -A FORWARD -s 10.0.0.0/24 -d 8.8.8.8 -p icmp --icmp-type echo-request -j ACCEPT

Запретить c7-1 давать ICMP-ответы, но оставить возможность отправлять запросы:

sudo iptables -A INPUT -p icmp --icmp-type echo-request -j DROP

6. Команду подключения из Части 7, п.1.

Часть №7

Настройка проброса порта для доступа к веб-серверу c7-2:

создания туннеля:

ssh -L 127.0.0.80:8888:10.0.0.2:80 FIOuser@<c7-1\_external\_ip> -p 55022

После этого можно будет обратиться к адресу 127.0.0.80:8888 в браузере, чтобы открыть веб-страницу на c7-2.

***Вопросы и задания:***

1. В чем разница между действиями SNAT или MASQUERADE? Когда уместно использовать одно, а когда другое?

*Ответ*:

* SNAT (Source NAT) — это метод изменения исходного IP-адреса пакетов, отправляемых с локальной сети на внешнюю. SNAT используется, когда известен IP-адрес, который будет использоваться для подмены (например, статический IP). Он позволяет сохранить состояние соединений, что может быть полезно для определенных приложений.
* MASQUERADE — это более динамичный вариант SNAT, используемый обычно для подключения к интернету через динамический IP-адрес. При использовании MASQUERADE Linux автоматически подставляет текущий внешний IP-адрес, что делает его удобным для случаев, когда IP может изменяться (например, при подключении через DSL).

Когда использовать:

* Используйте SNAT, когда у вас есть фиксированный IP-адрес и вы хотите управлять правилами NAT более точно.
* Используйте MASQUERADE, когда у вас динамический IP, и вам не нужно беспокоиться о его изменениях.

2. Какие цепочки и какие таблицы существуют в iptables по умолчанию?

*Ответ*: В iptables по умолчанию существуют следующие таблицы:

* filter — основная таблица для управления доступом к трафику (принимает, отклоняет, пересылает).
* nat — используется для NAT (изменение адресов в заголовках пакетов).
* mangle — для изменения заголовков пакетов (например, для QoS).
* raw — для управления необработанными пакетами (например, для исключения от состояния соединений).

По умолчанию в таблице filter существуют следующие цепочки:

* INPUT — для входящих пакетов.
* FORWARD — для пакетов, которые пересылаются через маршрутизатор.
* OUTPUT — для исходящих пакетов.

3. Как добавить новую цепочку? Как перенаправить в нее трафик?

*Ответ*:

1. Чтобы добавить новую цепочку в iptables:

iptables -N <имя\_цепочки>

1. Чтобы перенаправить трафик в новую цепочку, используйте команду для добавления правила в существующую цепочку (например, INPUT, FORWARD или OUTPUT):

iptables -A <существующая\_цепочка> -j <имя\_цепочки>

4. Имеет ли смысл порядок правил?

*Ответ*: да, порядок правил имеет большое значение в iptables. Правила обрабатываются последовательно, и как только пакет соответствует одному из правил, дальнейшая проверка прекращается. Это означает, что более специфичные правила должны быть выше в списке, чтобы их могли применять до того, как пакет дойдет до более общих правил.

5. Как с помощью iptables можно реализовать настройки, при которых брандмауэр пропускает пакеты тех соединений, которые были инициированы изнутри. Учтите, что правило позволяло установить соединение, т. е. передать пакеты наружу, так и получать ответы, то есть принять ответные пакеты.

*Ответ*: для реализации этой настройки вы можете использовать состояние соединения в iptables. Вот пример правил, которые позволяют пропускать пакеты для установленных соединений и вновь создаваемые изнутри:

*# Разрешить входящие пакеты для установленных соединений*

*iptables -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT*

*# Разрешить исходящие соединения*

*iptables -A OUTPUT -m state --state NEW,ESTABLISHED -j ACCEPT*

* Первое правило позволяет принимать входящие пакеты, которые являются частью уже установленных соединений или связаны с ними.
* Второе правило разрешает новым соединениям исходить из внутренней сети и также позволяет принимать ответные пакеты для уже установленных соединений.

Эти правила обеспечат необходимую функциональность для брандмауэра.