нетология

Сеть и сетевые протоколы: **Firewall**

Артур
Сагутдинов

15+ лет в сфере ИТ

Разрабатываю и внедряю линуксовую инфраструктуру

Сисадминский блог

Артур Сагутдинов

Начальник IT отдела

ООО «Клинический институт репродуктивной медицины»

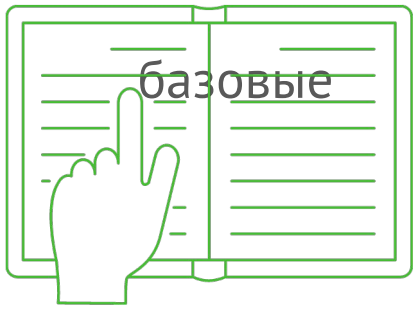
Артур Сагутдинов

Предисловие

На этом занятии мы поговорим о:

- фаерволле Netfilter;
- утилите iptables;
- виртуальных сетях VirtualBox;
- создании своего собственного маленького роутера!

По итогу занятия вы получите представление о том как управлять трафиком с помощью собственноручно созданного роутера и узнаете, как создавать базовые правила для iptables.



План занятия

1. Предисловие
2. Firewall
3. Историческая справка
4. Netfilter
5. iptables
6. netstat
7. Практика
8. Итоги
9. Домашнее задание

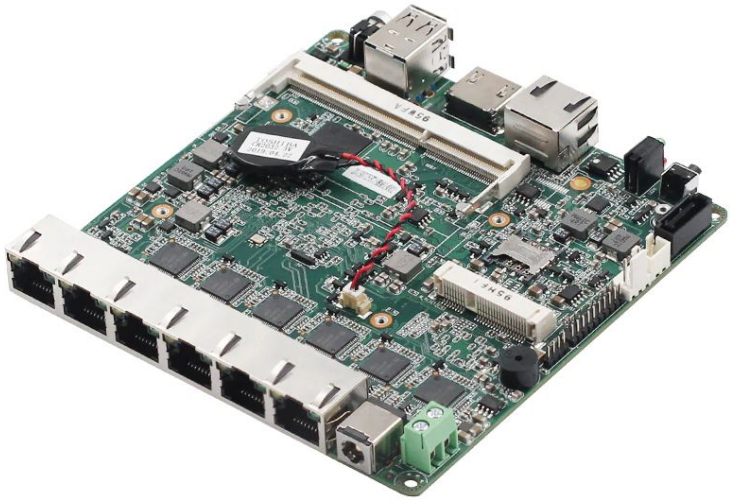
Firewall

Firewall

Firewall – это дополнительный слой защиты между вами и проблемами.

Существуют реализации как в виде программного решения, так и в виде программно аппаратных реализаций.

Задача фаерволла – фильтрация проходящего через него трафика на основе определённых ранее правил.



Основная цель

Основная цель фаерволла – защита сетей или отдельных хостов от атак направленных извне внутрь защищаемой сети.

Фаерволл бесполезен в борьбе с атаками проводимыми внутри периметра, чей трафик не проходит сквозь его интерфейсы.

Что будет без него?

Отсутствие фаерволла ➡️ проблемы!

Многие атаки направлены не на видимое повреждение оборудования, вызов всевозможных неполадок или работу по принципу крипто-вымогателей.

Многие атаки направлены на кражу информации. Поэтому несмотря на увещевания фаерволл-диссидентов, утверждающих что они живут без фаерволлов годами, это не значит что они не подвергаются атакам. Просто целью этих атаки являются их данные.

Историческая справка

Историческая справка

Brandmauer – в переводе с немецкого «противопожарная стена».

Firewall – противопожарная стена в переводе с английского.

В разрезе компьютерных сетей термин появился ближе к концу 1980-х годов. Предшественниками фаерволлов на поприще сетевой безопасности в конце 1980-х использовались роутеры. В то время они уже разделяли сети на сегменты и, как следствие, могли фильтровать проходящий через них трафик.

Впервые термин встречается в 1983 году в фильме про хакеров «Военные игры». Возможно это и привело к последующей популяризации термина в реальной жизни.

Историческая справка

Все мы сегодня здесь собрались из-за этого человека. Rusty Russel – основатель проекта ipchains, в последующем в 1998г. основавший проект **Netfilter/iptables**.

- в Linux 2.0 как фаерволл работал ipfwadm;
- с версии Linux 2.2 фаерволлом был ipchains;
- начиная с версии Linux 2.4 функцию фаерволла в ядре Linux выполняет Netfilter.

Rusty Russel

Источник изображения: wikipedia.org

Netfilter

Netfilter

Netfilter – межсетевой экран встроенный в ядро Linux начиная с версии 2.4.

Имеет свой «фронтенд»: утилиту iptables.

С помощью этой утилиты сисадмин может создавать и изменять правила фильтрующие трафик.

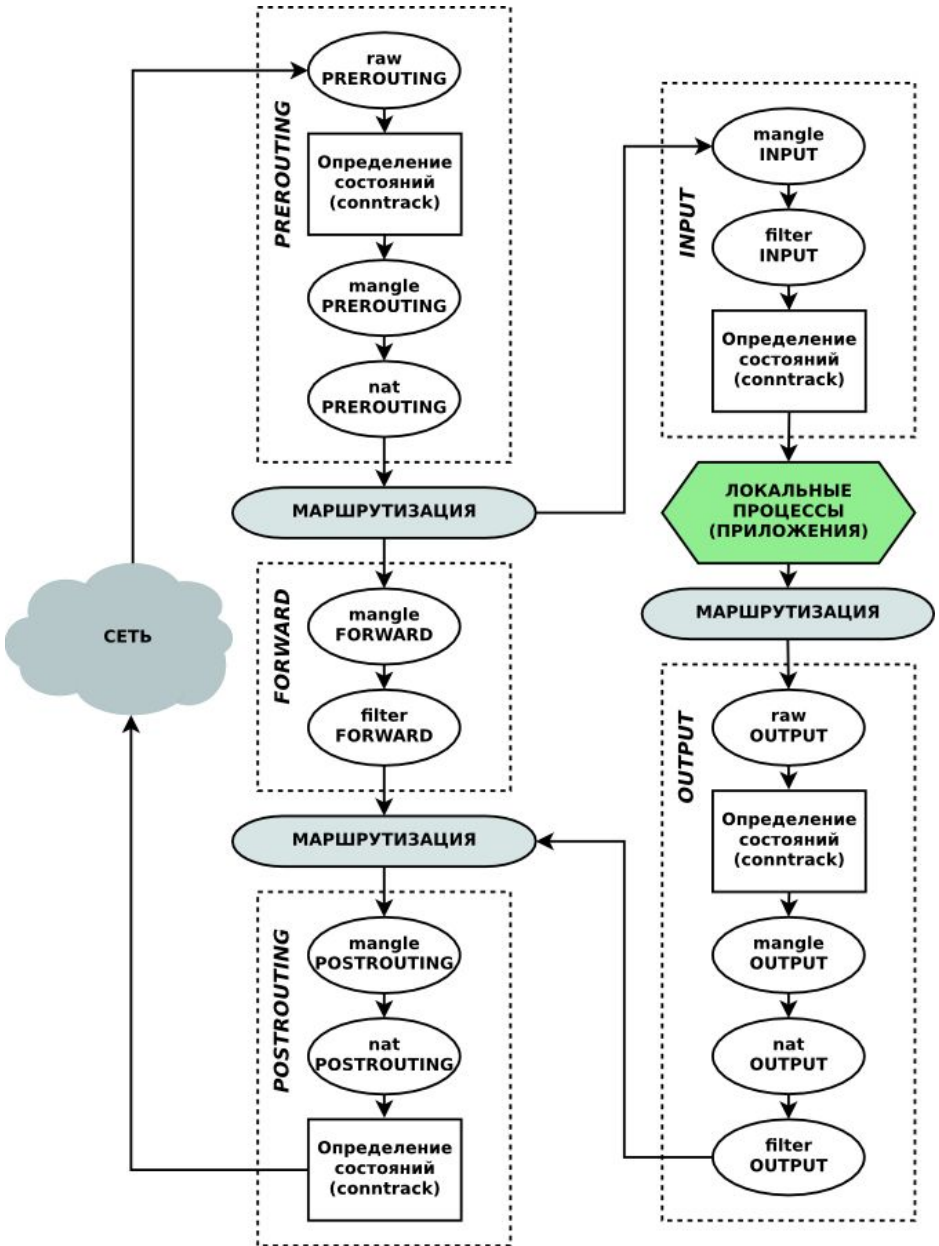
Аналогом iptables в таких современных версиях OS как CentOS, Fedora, OpenSUSE, Red Hat Enterprise Linux, SUSE Linux Enterprise служит утилита firewalld.

Архитектура Netfilter

Архитектура netfilter подразумевает прохождение пакетов через цепочки правил.

Каждое правило в упорядоченном списке может содержать различные критерии и действие или переход, выполняющиеся в случае полного соответствия пакета критериям.

Отсутствие критериев применяет правило ко всем проходящим через него пакетам.



iptables

iptables

iptables – интерфейс управления netfilter.

➡️ Оперирует правилами, цепочками и таблицами.

Является полноценным инструментом позволяющим настроить фаерволл.

Когда вы настраиваете фаерволл в своём домашнем роутере, вы просто дёргаете веб-интерфейс, являющийся не более чем утилитой конфигурирования iptables.

Правило iptables

В состав правила входят следующие сущности:

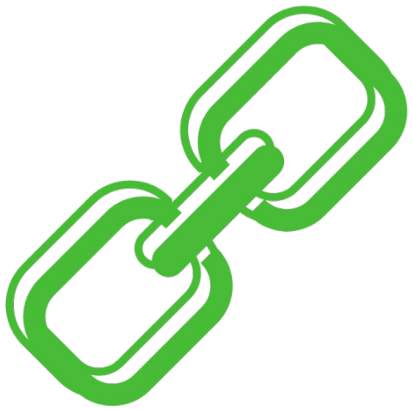
- условие – логическое выражение на основании которого происходит анализ свойств пакета / соединения и которое определяет попадание пакета / соединения под текущее правило;
- действие – выполняется в случае соответствия пакета / соединения текущему правилу;
- счётчик – учитывает количество пакетов попавших под условие текущего правила.

Цепочки iptables

Цепочка – упорядоченная последовательность правил.

Бывают:

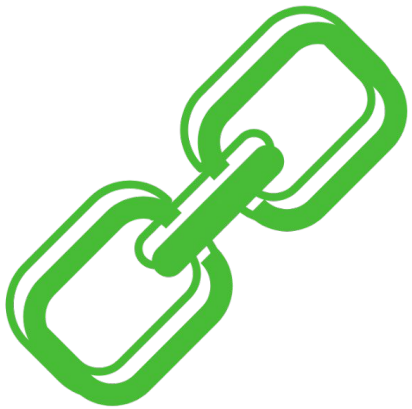
- пользовательская цепочка – создаётся пользователем и используется только в пределах своей таблицы;
- базовая цепочка – создаётся по умолчанию при создании таблицы и в отличии от пользовательской обладает действием по умолчанию.



Цепочки iptables

В список базовых цепочек входят:

- PREROUTING;
- INPUT;
- FORWARD;
- OUTPUT;
- POSTROUTING.



Таблицы iptables

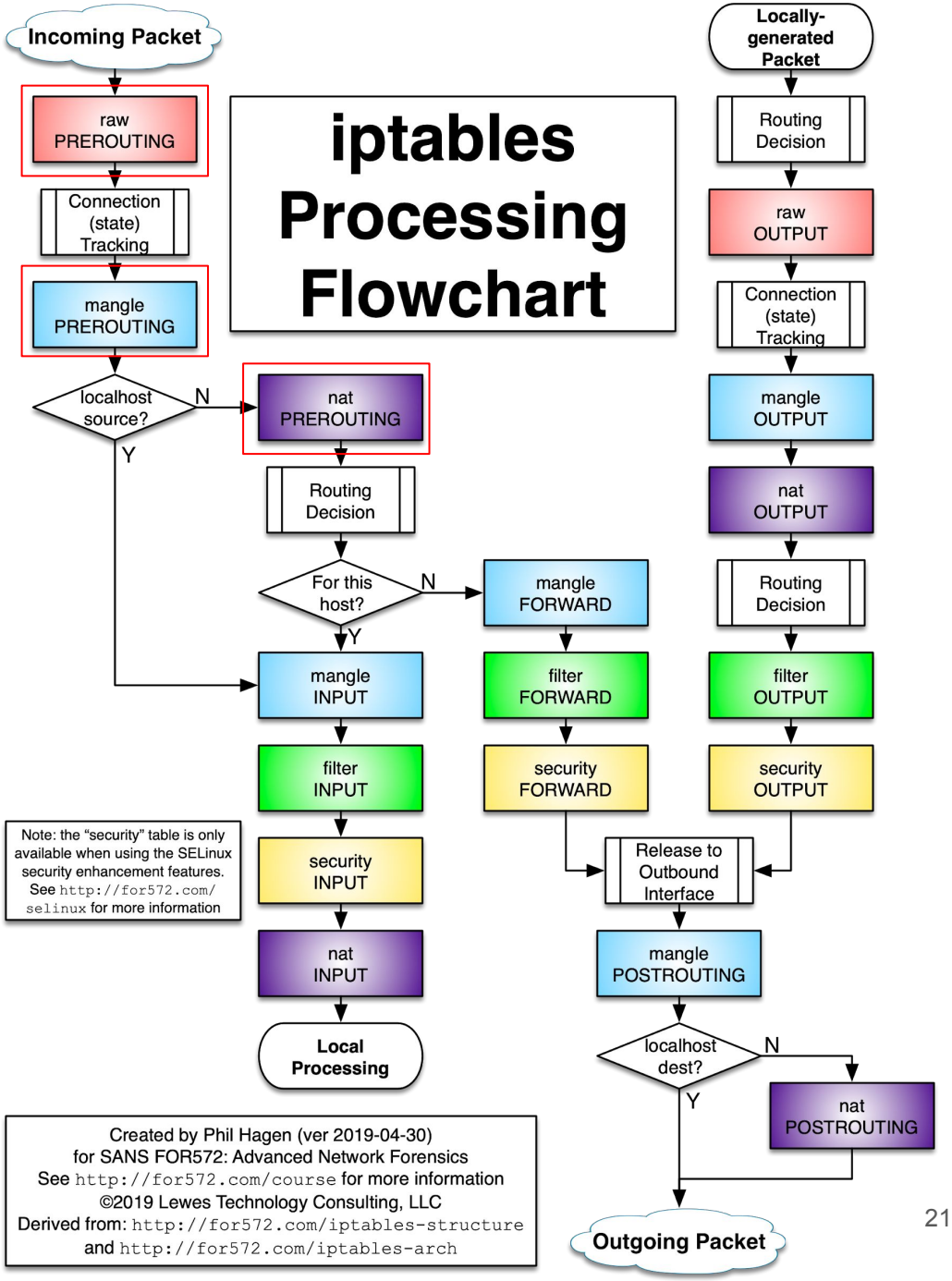
Таблица – совокупность базовых и пользовательских цепочек, имеющих общее назначение.

iptables имеет 4 типа таблиц:

- Filter;
- NAT;
- Mangle;
- Raw.

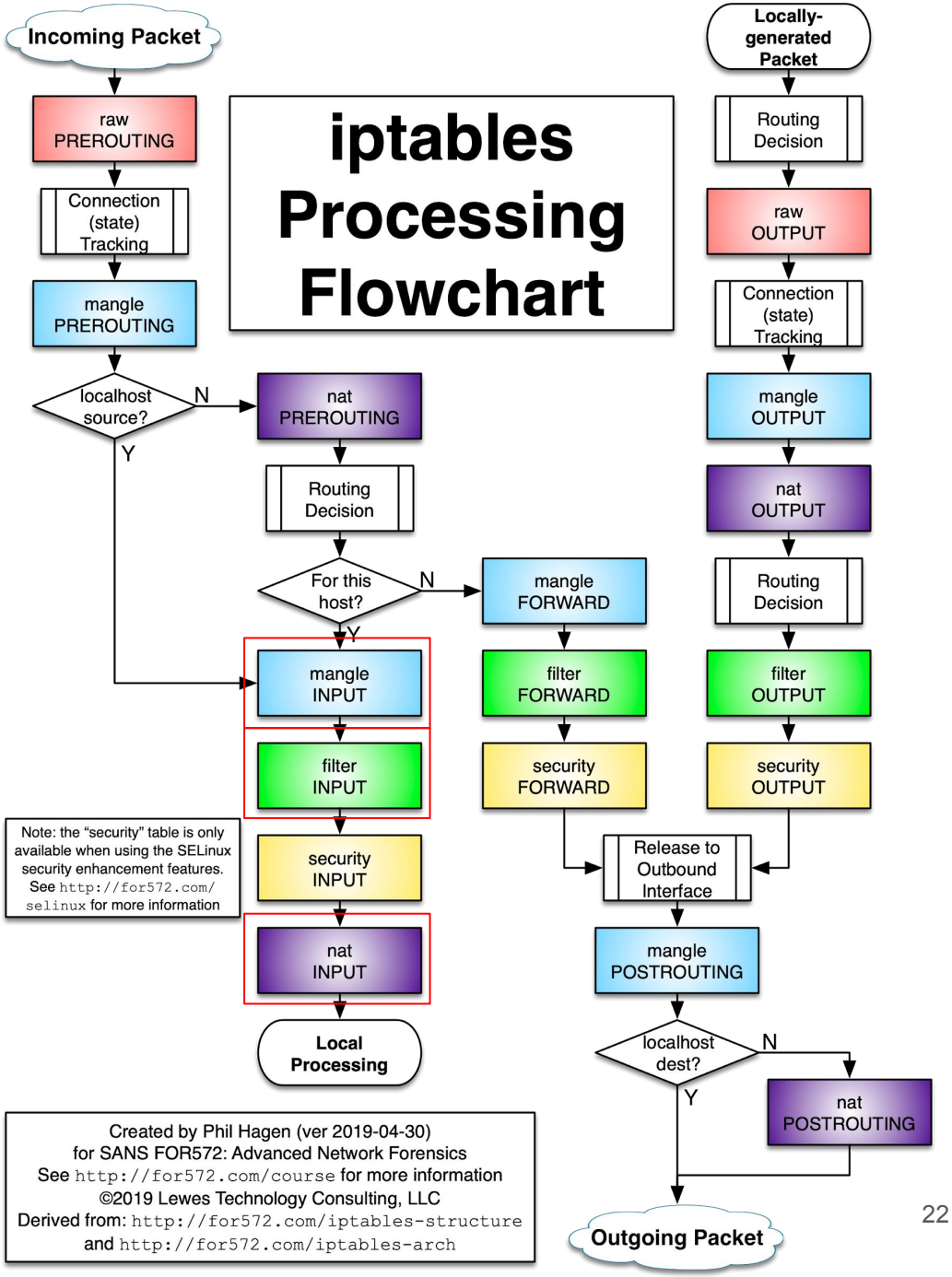
Цепочка PREROUTING

В данной цепочке обрабатываются все входящие пакеты.



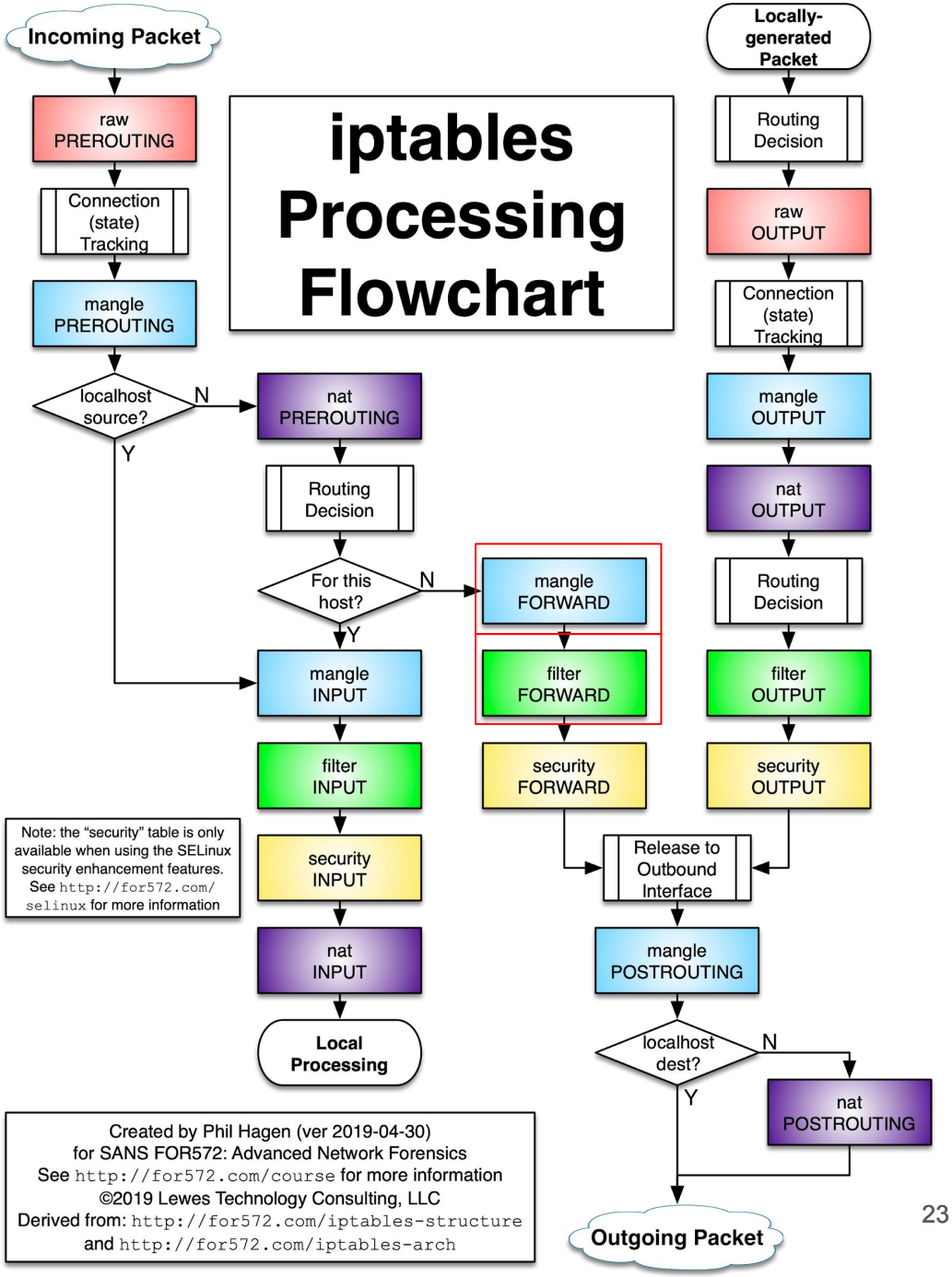
Цепочка INPUT

Данная цепочка предназначена для обработки входящих пакетов, адресованных локальному процессу



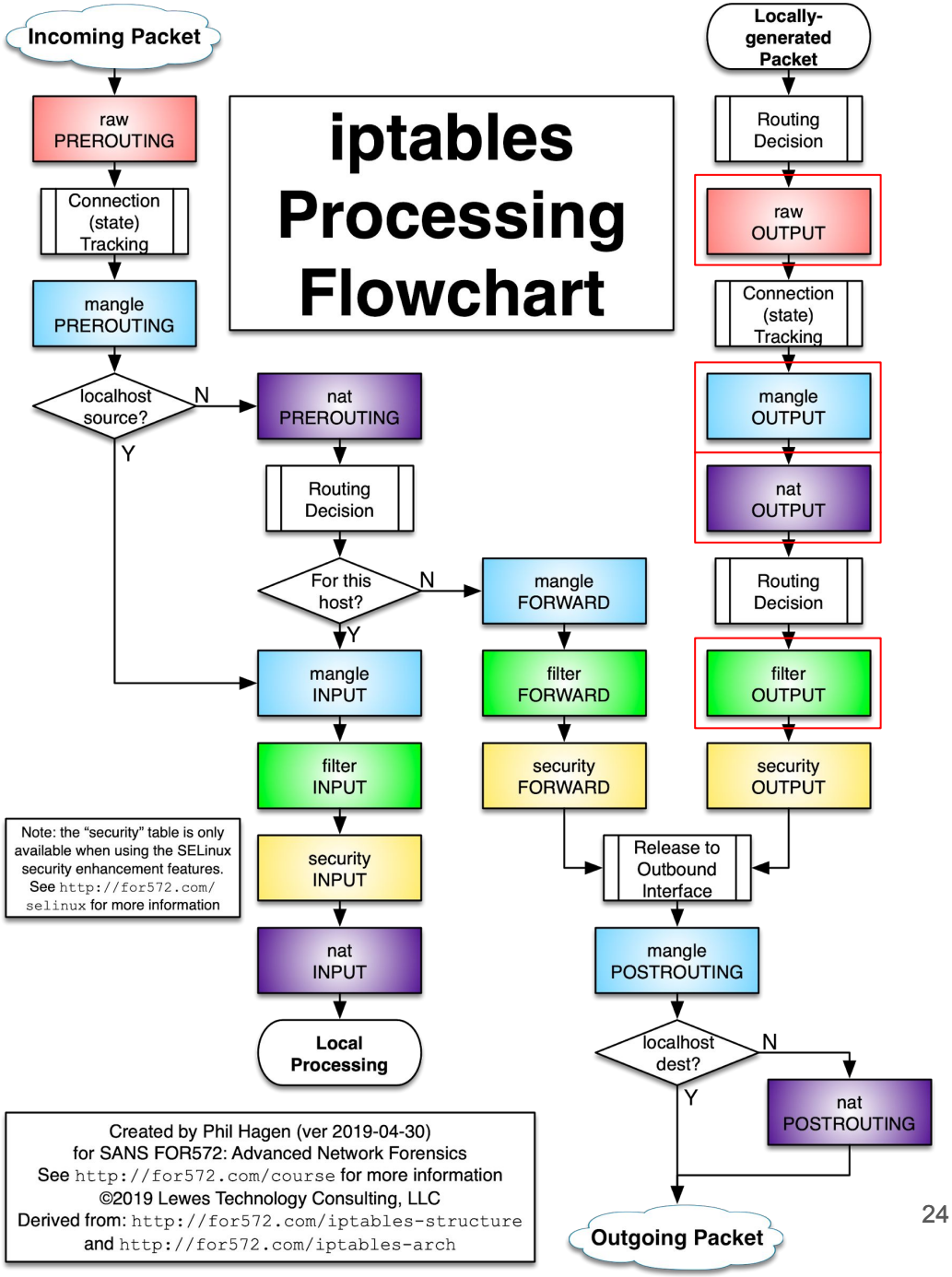
Цепочка FORWARD

Данная цепочка предназначена для обработки входящих пакетов, которые пересылаются дальше.



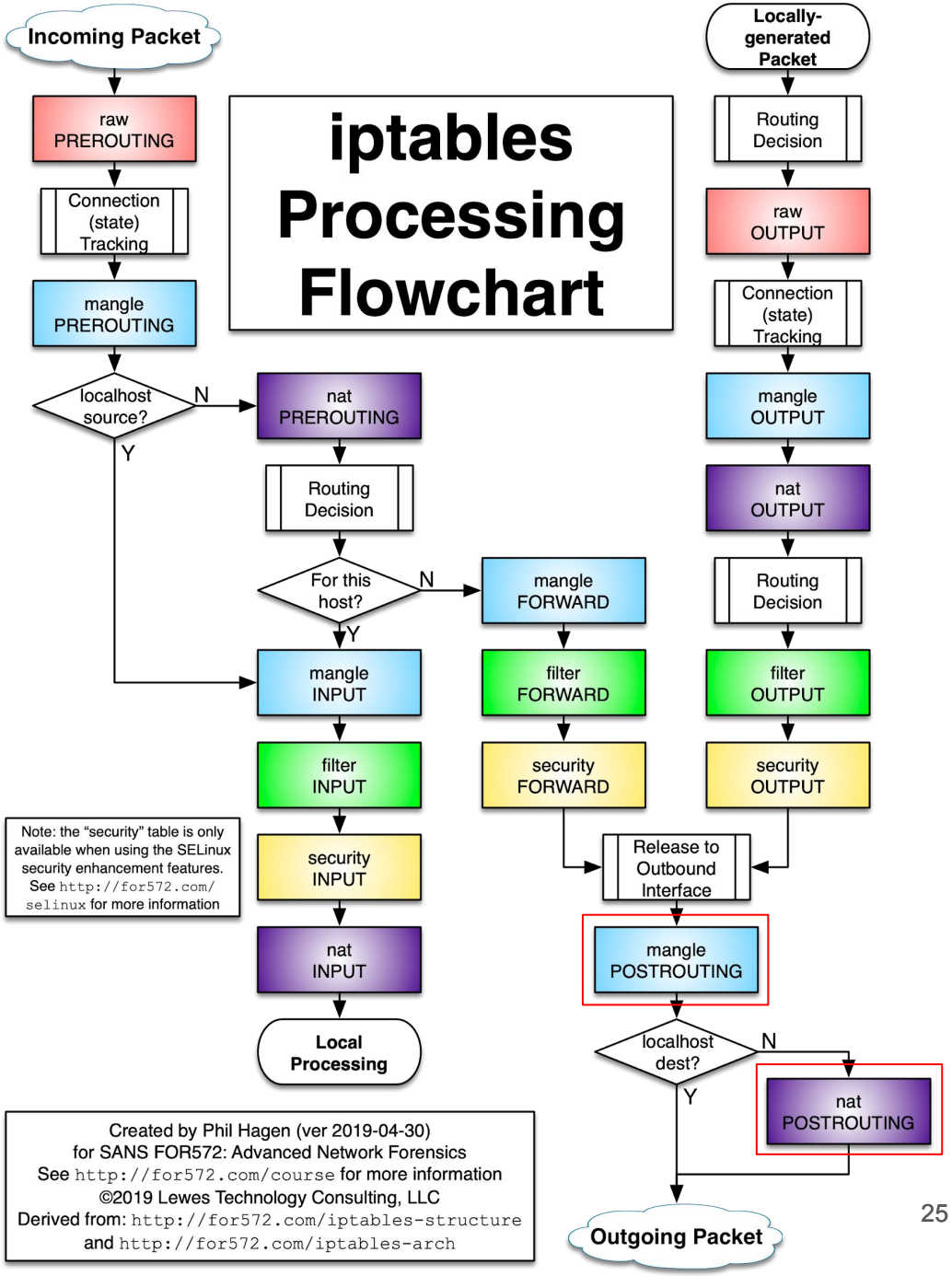
Цепочка OUTPUT

Данная цепочка предназначена для пакетов, исходящих от внутренних процессов.



Цепочка POSTROUTING

В данной цепочке происходит окончательная обработка исходящих пакетов.



netstat

Утилита netstat

Позволяет смотреть состояния соединений, таблиц маршрутизации, чисто сетевых интерфейсов и статистику по протоколам, а именно:

- посмотреть слушает ли сервер порт 22;

  ```
  netstat -an | grep ":22"
  ```

- посмотреть все сокеты с состоянием LISTEN;

  ```
  netstat -l
  ```

- узнать статистику для каждого протокола;

  ```
  netstat -s
  ```

- посмотреть руководство по netstat;

  ```
  man netstat
  ```

Просмотр таблиц iptables

При работе с iptables всегда приходится обращаться к просмотру содержимого таблиц. Иначе мы не узнаем текущие настройки.

Чтобы сделать это нам понадобится следующая команда:

sudo iptables -nvL -t raw

sudo iptables -nvL -t mangle

sudo iptables -nvL -t nat

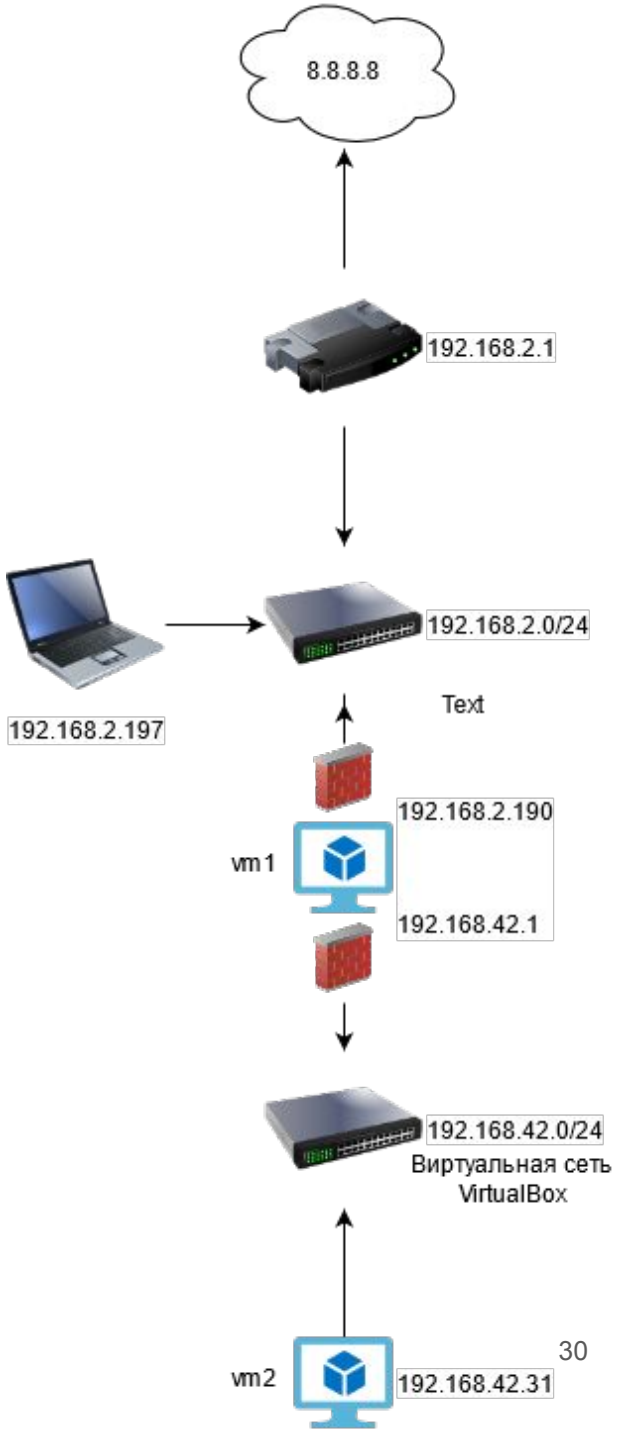
sudo iptables -nvL -t filter

➡️ Если не указать имя таблицы, команда выдаст содержимое таблицы filter.

Практика

Исходные данные

- ноутбук подключён к сети 192.168.2.0 и на нём установлен VirtualBox;
- vm1 подключена к сети 192.168.2.0 (enp0s3) и к виртуальной сети 192.168.42.0 (enp0s8);
- vm2 подключена только к сети 192.168.42.0;
- сеть 192.168.2.0 – выход в интернет;
- сеть 192.168.42.0 – виртуальная сеть Virtual Box;
- vm1 в сети 42.0 имеет IP 192.168.42.1;
- vm2 в сети 42.0 имеет IP 192.168.42.31.



Блокируем порт извне

К примеру, мы хотим закрыть доступ из вне к какому-то порту.

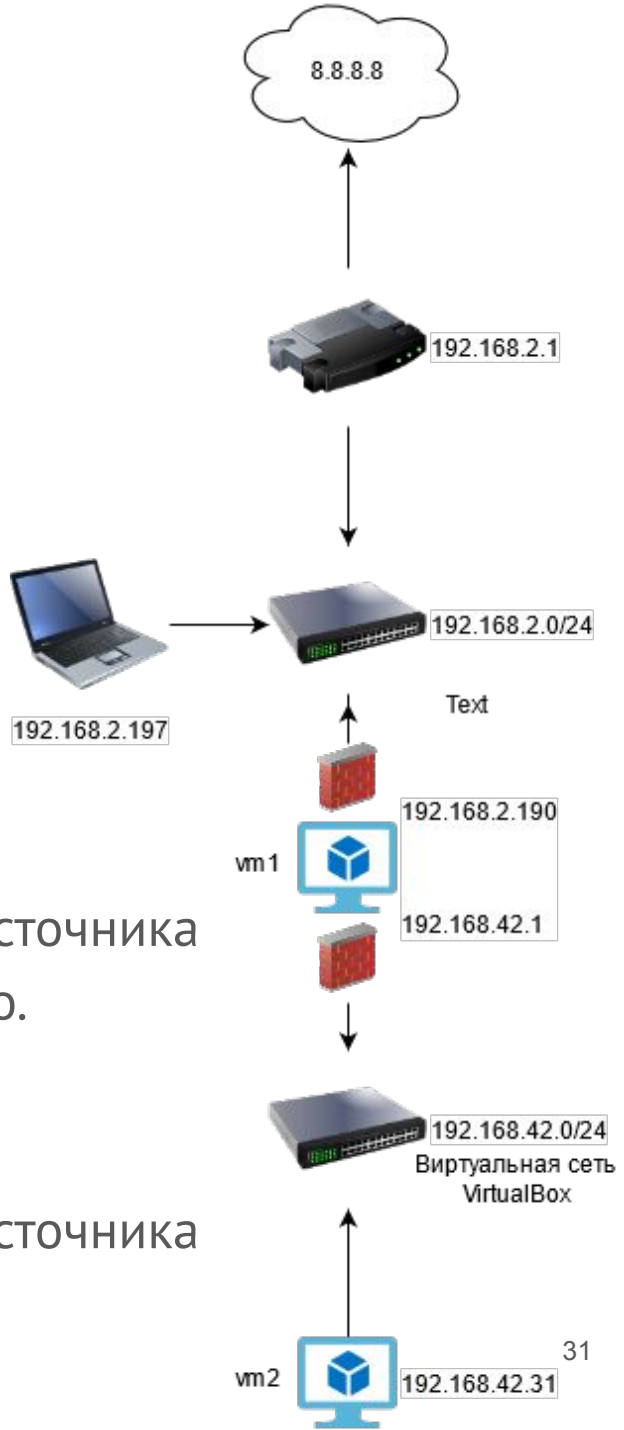
Для этого в таблицу INPUT нам необходимо добавить условие, и соответствующее для него действие:

```
sudo iptables -A INPUT -p tcp --dport 22 -j DROP
```

➡️ После этого любое подключение из любого источника к текущему хосту на порт 22 будет заблокировано.

```
sudo iptables -A INPUT -p tcp --dport 22 -j ACCEPT
```

➡️ После этого любое подключение из любого источника к текущему хосту на порт 22 будет разрешено.

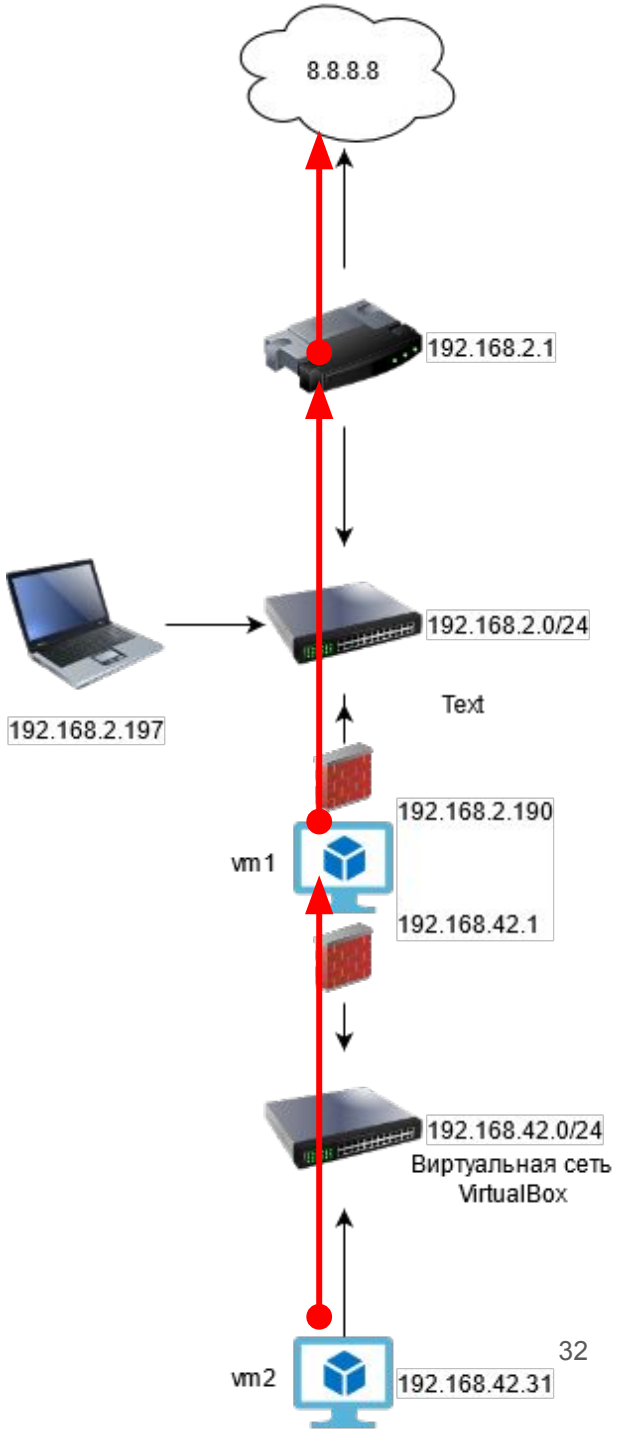


Настраиваем NAT маскарадинг

Сделаем так, чтобы трафик с vm2 выходил интернет.

```
# На vm1
# Включаем IP форвардинг в ядре Linux
cat /proc/sys/net/ipv4/ip_forward # Проверяем включён ли ip форвардинг
sudo nano /proc/sys/net/ipv4/ip_forward # Изменяем на 1 если было 0
# Разрешаем форвардинг уже установленных соединений
sudo iptables -A FORWARD -j ACCEPT -m conntrack --ctstate \
ESTABLISHED,RELATED -m comment --comment "established traffic"
# Разрешаем форвардинг новых соединений с интерфейса enp0s8 на enp0s3
sudo iptables -A FORWARD -j ACCEPT -i enp0s8 -o enp0s3 \
-m comment --comment "forward"
# Включаем маскарадинг всех соединений идущих через enp0s3
sudo iptables -t nat -A POSTROUTING -o enp0s3 -j MASQUERADE \
-m comment --comment "masquerade"
# Чтобы ip_forward сохранился после перезагрузки, пригодится команда:
sudo sysctl -w net.ipv4.ip_forward=1
```

```
# На vm2
ping 8.8.8.8
```

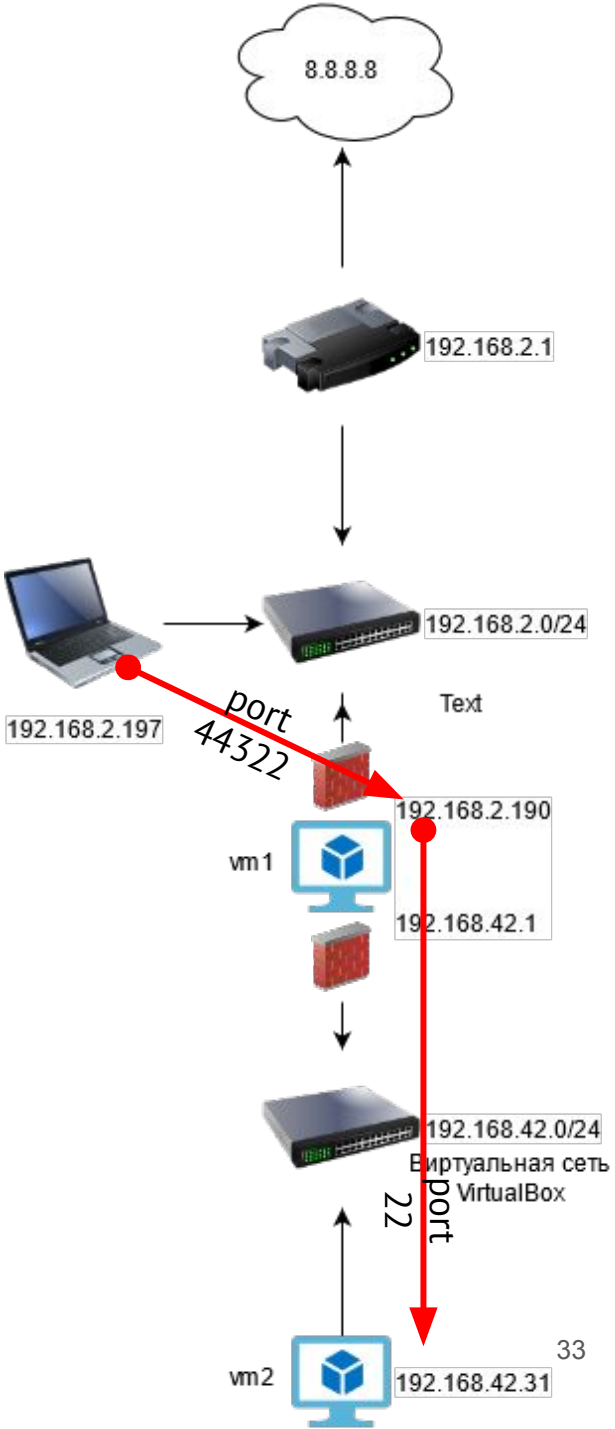


Пробрасываем порт

Сделаем так, чтобы порт 22 vm2 был доступен ноутбуку по адресу 192.168.2.190:44322

```
#Пробрасываем трафик с “публичного” IP шлюза порт 44322 на IP адрес 192.168.42.31 порт 22
sudo iptables -t nat -A PREROUTING -d 192.168.2.190 -p tcp \
--dport 44322 -j DNAT --to-destination 192.168.42.31:22
# Разрешаем пропускать трафик с enp0s3 через enp0s8 на 192.168.42.31 порт 22
sudo iptables -I FORWARD 1 -i enp0s3 -o enp0s8 -d 192.168.42.31 \
-p tcp -m tcp --dport 22 -j ACCEPT
```

На ноутбуке остаётся через Putty подключиться по SSH на IP 192.168.2.190 на порт 44322

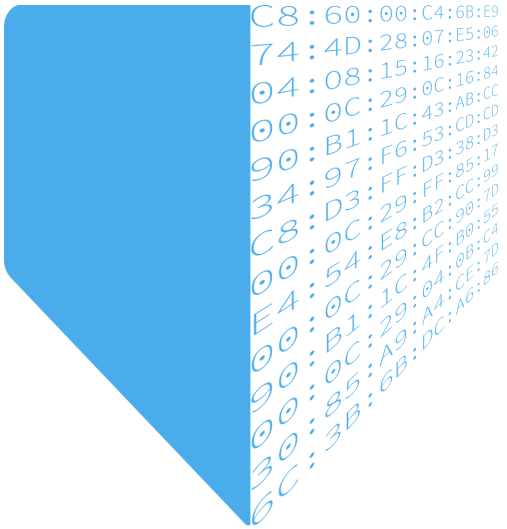


iptables блокировка по MAC

Обозначим критерием блокировки трафика – MAC адрес. В случае если аппаратный адрес сетевой карты подключающегося устройства будет соответствовать указанному в правиле, оно будет отбрасывать трафик.

```
#Отбрасываем трафик если он исходит от MAC адреса 08:00:27:47:88:ce
sudo iptables -A INPUT -m mac --mac-source 08:00:27:47:88:CE -j DROP
#Отбрасываем трафик если он исходит НЕ от MAC адреса
08:00:27:47:88:ce
sudo iptables -A INPUT -m mac ! --mac-source 08:00:27:47:88:CE -j DROP
```

Теперь попытки пинга vm1 с vm2 или какие-либо подключения непосредственно к шлюзу обречены на неудачу.



ebtables

Похожая на iptables утилита. В отличии от iptables работает с трафиком живущим на втором уровне модели OSI

В связи со своим уровнем в модели OSI, обработка трафика с помощью ebtables происходит раньше чем до трафика добирается iptables.

➡️ Предназначена для фильтрации трафика бриджей.

Например чтобы отбросить трафик от конкретного MAC адреса необходима следующая команда:

ebtables -A INPUT -s 08:00:27:47:88:CE -j DROP

Вариант который мы использовали в iptables:

sudo iptables -A INPUT -m mac --mac-source 08:00:27:47:88:CE -j DROP

Итоги

Итоги

Сегодня мы рассмотрели настройку фаерволла iptables для Linux и узнали как:

- превратить машину с двумя сетевыми интерфейсами в шлюз;
- перенаправлять порты в локальную сеть;
- блокировать трафик по MAC;
- фильтровать L2 трафик.

Домашнее задание

Домашнее задание

Давайте посмотрим ваше домашнее задание.

- Вопросы по домашней работе задавайте **в чате** мессенджера Slack.
- Задачу можно сдавать **по частям**.
- Зачёт по домашней работе проставляется после того, как **приняты задача полностью**.

нетология

Задавайте вопросы и пишите отзыв о лекции!

Артур Сагутдинов

Артур Сагутдинов