

Основы построения защищенных компьютерных сетей

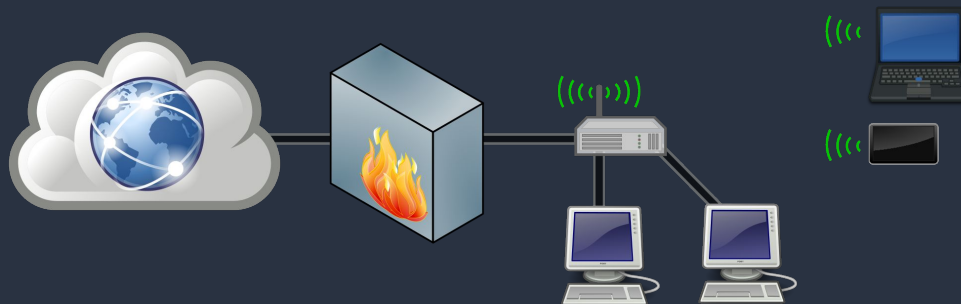
Лекция 7 Межсетевые экраны

Семён Новосёлов

2022



Межсетевой экран (файервол, брандмауэр) — комплекс аппаратных или программных средств, осуществляющий контроль и фильтрацию проходящих через него сетевых пакетов в соответствии с заданными правилами



Классификация МЭ

По уровню модели OSI:

- канальный уровень — управляемые коммутаторы
 - фильтрация по MAC-адресу
- сетевой уровень — сетевые фильтры
 - фильтрация по полям протокола IP: адресу, порту назначения, ...
- сеансовый уровень — шлюзы
 - трансляция адресов (NAT, PAT) или сетевых протоколов
- прикладной уровень — прокси-серверы, шлюзы уровня приложений
- смешанные, работающие на нескольких уровнях

Основные возможности МЭ

- фильтрация доступа к защищенным службам
- препятствование получению закрытой информации
- препятствование внедрению ложных данных с помощью уязвимых служб
- регистрация попыток доступа
- регламентирование порядка доступа к сети
- уведомление о подозрительной активности
- контроль доступа к узлам сети

Недостатки МЭ

- может снижать пропускную способность сети и время отклика, так как фильтрация происходит не мгновенно
- не защищает сети от проникновения через уязвимости в ПО
- не защищает от вирусов

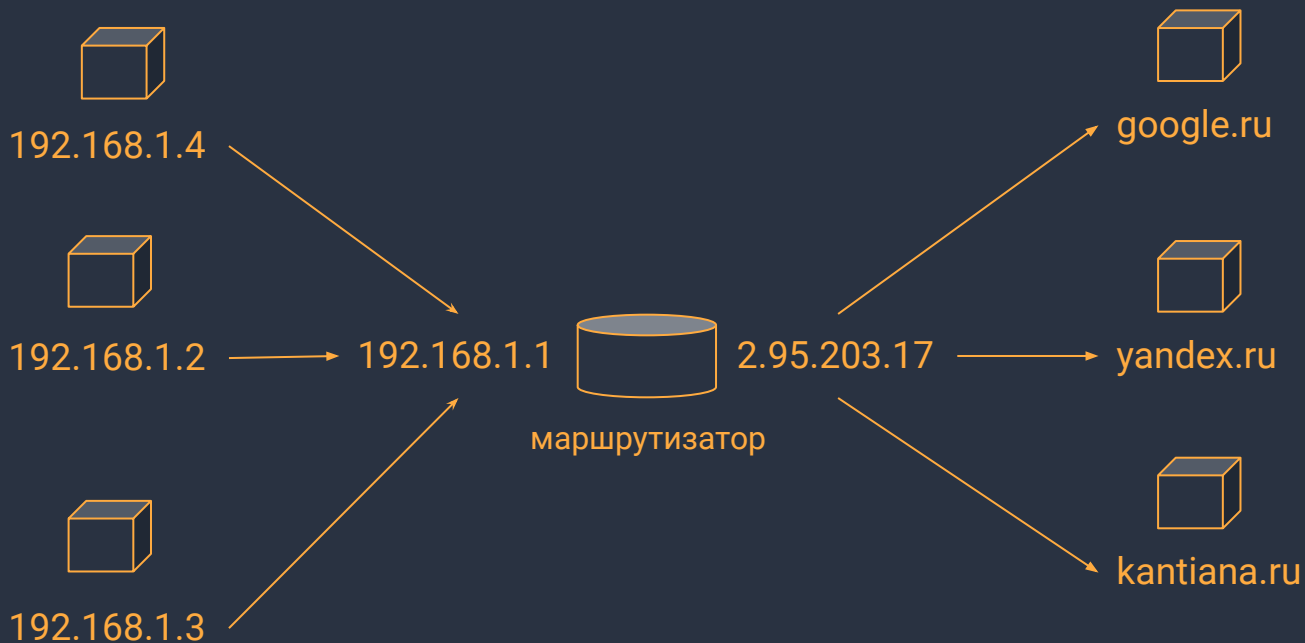
Технология SPI (Stateful Packet Inspection)

- МЭ с поддержкой SPI позволяют отслеживать соединения (TCP, UDP и др.) и отбрасывать пакеты, не принадлежащие соединению

Технология DPI (Deep Packet Inspection)

- проверка данных пакета для поиска вирусов, спама, вторжений, недопустимых файлов, попытки открытия заблокированного сайта и т. п.
- может также использоваться для сбора статистики и модификации пакетов
- недостатки DPI: требуется более мощное оборудование, декодеры протоколов могут содержать уязвимости

NAT (Network Address Translation)



Межсетевой экран Linux

- в ядро Linux встроен межсетевой экран **netfilter**
- для управления его работой в большинстве систем используется утилита **iptables**

Замечание: файерволы для **Windows** и других систем имеют схожие возможности.

Архитектура netfilter

- в **netfilter** пакеты пропускаются через цепочки, которые являются упорядоченными списками правил
- каждое правило состоит из критерия, действия и счетчика
- цепочки объединяются в таблицы, в зависимости от функционального назначения

```
iptables -L
Chain INPUT (policy ACCEPT)
target      prot opt src      dst
ACCEPT      all  --  any      any      state RELATED,ESTABLISHED
LOG          tcp  --  any      any      tcp flags:SYN,RST,ACK/SYN limit: avg 6/hour
burst 5 ...
DROP        all  --  anywhere anywhere
```

Цепочки netfilter

Существует 5 типов стандартных цепочек:

- **PREROUTING** — для начальной обработки входящих пакетов
- **INPUT** — для входящих пакетов адресованных непосредственно локальному процессу (клиенту или серверу)
- **FORWARD** — для входящих пакетов перенаправленных на выход
- **OUTPUT** — для пакетов генерируемых локальными процессами
- **POSTROUTING** — для окончательной обработки исходящих пакетов

Таблицы netfilter

Цепочки организованы в 4 таблицы:

- **raw** — просматривается до передачи пакета системе определения состояний
- **mangle** — содержит правила модификации (обычно заголовка) IP-пакетов
- **nat** — просматривает только пакеты, создающие новое соединение
- **filter** — основная таблица, используется по умолчанию, если название таблицы не указано

Механизм определения состояний netfilter

Каждый пакет, проходящий через систему определения состояний может иметь 4 состояния:

- **NEW** — пакет открывает новый сеанс. Классический пример -- пакет TCP с флагом SYN
- **ESTABLISHED** — пакет является частью уже существующего сеанса
- **RELATED** — пакет открывает новый сеанс, связанный с уже открытым сеансом
- **INVALID** — все прочие пакеты

Блокировка адреса/подсети

```
iptables -I INPUT -s address/mask -j {DROP/REJECT} [--reject-with reject_type]
```

- блокировка по маске может быть полезна, если атака идёт из определенной сети
- также может использоваться модуль **iprange**

Блокировка доступа к сайтам

```
iptables -I OUTPUT -s address1,address2,... -j {DROP/REJECT} [--reject-with reject_type]
```

Можно блокировать по:

- IP-адресу (флаг **-s**: входящий адрес, **-d**: исходящий адрес)
- по подстроке (используя модуль string)
 - "Host: kantiana.ru"
 - работает только для незашифрованных соединений и при MiTM.

Разблокировка и удаление правил

Есть два способа:

- Просто повторение правила с ключом -D
- Удаление по номеру:
 - `iptables -L --line-numbers`
 - `iptables -D OUTPUT 1`

1. Блокировка яндекса:

```
iptables -A OUTPUT -d yandex.ru -j DROP
```

2. Разблокировка яндекса:

```
iptables -D OUTPUT -d yandex.ru -j DROP
```


Фильтрация TCP

- Используется модуль tcp
- По порту:
 - `--dport` для исходящего порта
 - `--sport` для порта отправителя

```
iptables -A OUTPUT -d kantiana.ru -p tcp -m tcp --dport 80 -j  
REJECT --reject-with tcp-reset
```