Курс «Администратор Linux»

Фильтрация пакетов

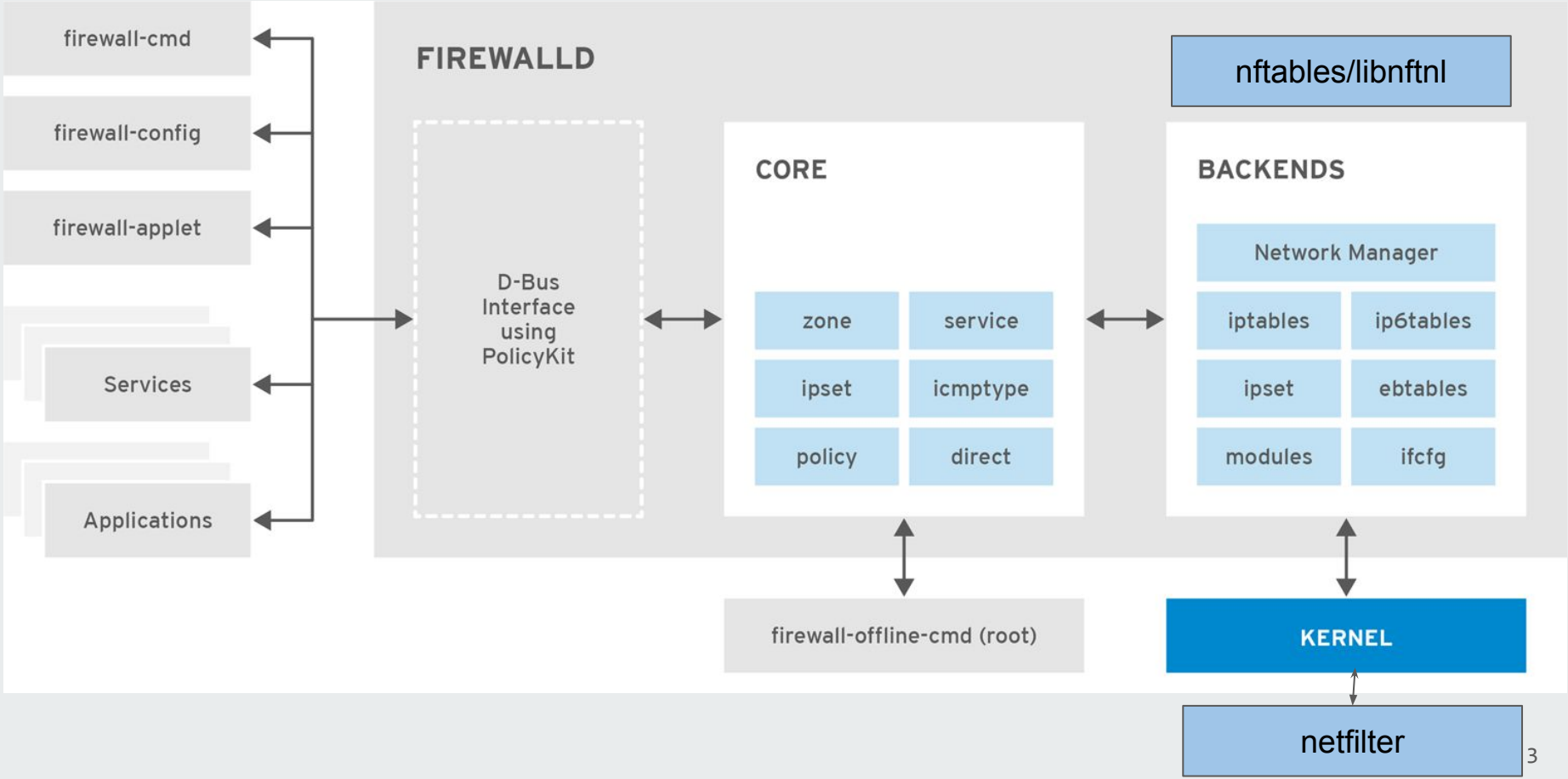
Занятие # 19

Дмитрий Молчанов
Григорий Ожегов
Алексей Цыкунов

Компоненты/утилиты управления

- Компоненты
  - xtables
    - ip_tables
    - ip6_tables
    - arp_tables
  - nf_conntrack
  - ebtables
  - **nftables**
- Утилиты управления
  - iptables/ip6tables
  - ipset
  - ebtables
  - arptables
  - firewalld
  - nft

Фильтрация пакетов



Firewalld: Зоны

- Изменяемые/неизменяемые зоны
- Зону можно присвоить
  - Интерфейсу (ifcfg-*, ZONE=)
  - Соединениям созданным в NetworkManager
  - Источникам трафика
- Компоненты зоны
  - Службы
  - Порты
  - Блоки ICMP
  - Маскарадинг
  - Проброс портов

Firewalld: Zones

```
# nmcli device
# firewall-cmd --get-zone-of-interface=eth0

# firewall-cmd --list-interfaces
# firewall-cmd --zone=trusted --list-interfaces

# firewall-cmd --zone=trusted --add-interface=eth0
# firewall-cmd --zone=work --change-interface=eth0
# firewall-cmd --zone work --remove-interface=eth0

# firewall-cmd --zone=trusted --add-source=192.168.1.0/24
# firewall-cmd --zone=work --change-source=192.168.1.0/24
# firewall-cmd --zone work --remove-source=192.168.1.0/24
```

Firewalld: Зоны

- Drop - только исходящие сетевые соединения. Все входящие дропаются.
- Block - соединения только внутри системы. Все входящие режектятся с icmp ответом
- Public (по умолчанию) - не доверять, разрешены отдельные входящие соединения
- External - (для роутеров) включен маскарадинг, разрешены отдельные входящие соединения
- Dmz - доверять, разрешены отдельные входящие соединения
- Home - доверять, разрешены отдельные входящие соединения
- Work - доверять, разрешены отдельные входящие соединения
- Internal - разрешены отдельные входящие соединения
- Trusted - разрешено все

Firewalld: примеры

- firewall-cmd --state
- firewall-cmd --reload
- firewall-cmd --get-(zones/services/icmptypes/active-zones)
- firewall-cmd [--zone=<zone>] --list-all
- firewall-cmd [--zone=<zone>] --add-interface=<interface>
- firewall-cmd [--zone=<zone>] --change-interface=<interface>
- firewall-cmd [--zone=<zone>] --[add/remove/query]-service=<service> [--timeout=<seconds>]
- firewall-cmd [--zone=<zone>] --[add/remove/query]-port=<port>[-<port>]/<protocol> [--timeout=<seconds>]
- firewall-cmd [--zone=<zone>] --[add/remove/query]-masquerade
- firewall-cmd [--zone=<zone>] --[add/remove/query]-icmp-block=<icmptype>
- firewall-cmd [--zone=<zone>] --[add/remove/query]-forward-port=port=<port>[-<port>]:proto=<protocol> { :toport=<port>[-<port>] | :toaddr=<address> | :toport=<port>[-<port>]:toaddr=<address> }

Firewalld: примеры

- firewall-cmd --permanent …
- firewall-cmd --direct --passthrough { ipv4 | ipv6 | eb } <args>
- firewall-cmd --direct --add-chain { ipv4 | ipv6 | eb } <table> <chain>
- firewall-cmd --direct --remove-chain { ipv4 | ipv6 | eb } <table> <chain>
- firewall-cmd --direct --query-chain { ipv4 | ipv6 | eb } <table> <chain>
- firewall-cmd --direct --get-chains { ipv4 | ipv6 | eb } <table>
- firewall-cmd --direct --add-rule { ipv4 | ipv6 | eb } <table> <chain> <priority> <args>
- firewall-cmd --direct --remove-rule { ipv4 | ipv6 | eb } <table> <chain> <args>
- firewall-cmd --direct --query-rule { ipv4 | ipv6 | eb } <table> <chain> <args>
- firewall-cmd --direct --get-rules { ipv4 | ipv6 | eb } <table> <chain>

Netfilter: Принцип работы

Пакетный фильтр это набор правил. В iptables правила организованы по таблицам и цепочкам.
Каждая таблица состоит из цепочек.
Каждая цепочка - упорядоченный набор правил, которые просматриваются последовательно начиная с первого.
Каждое правило состоит из:

- Критериев срабатывания
- Действия

Каждое правило имеет счетчики срабатываний.

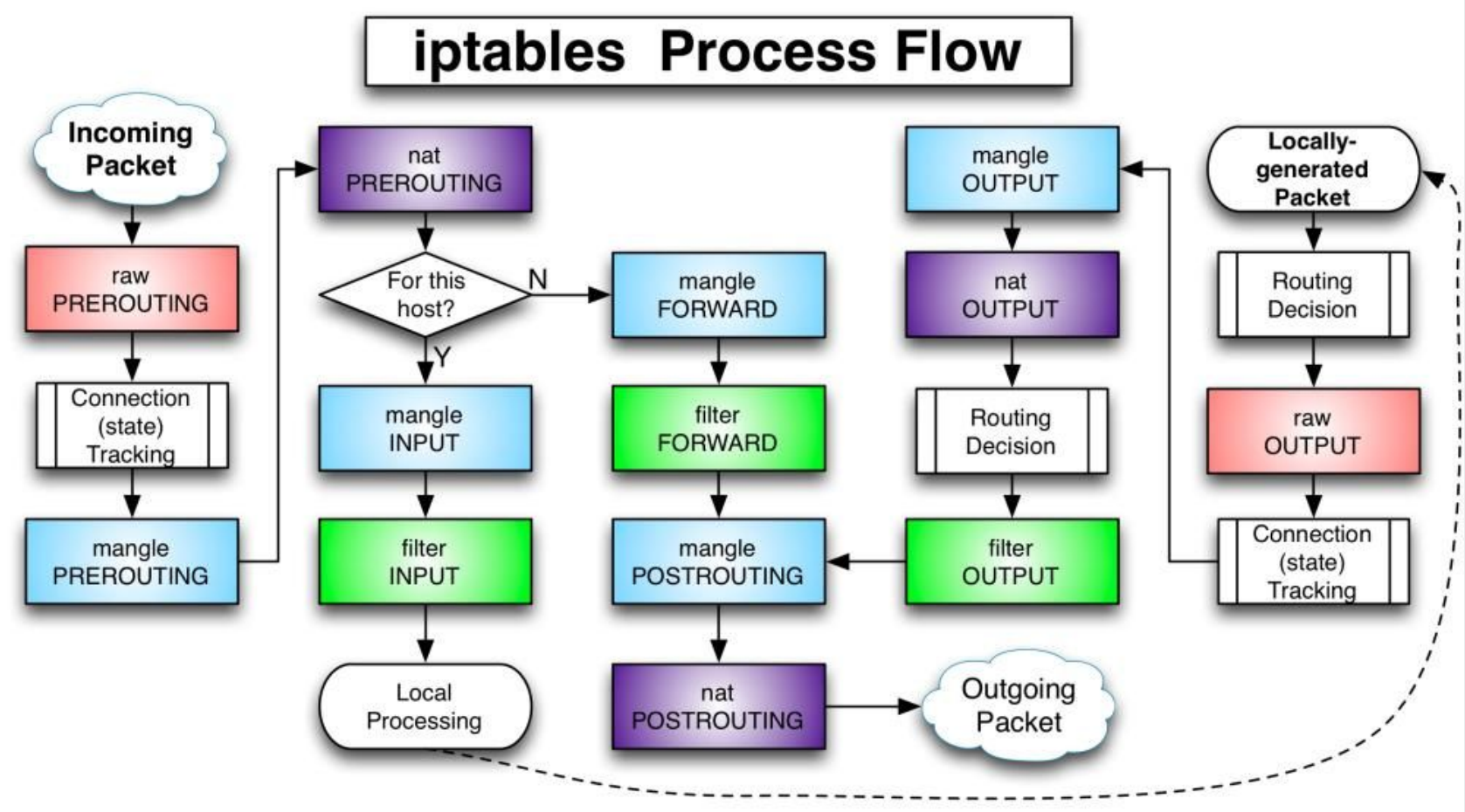
Таблицы

- **raw** - изначальная обработка, до conntrack
- **mangle** - модификация заголовков пакетов и маркировка пакетов
- **nat** - трансляция адресов
- **filter** - фильтры
- **security** - работа с SELinux

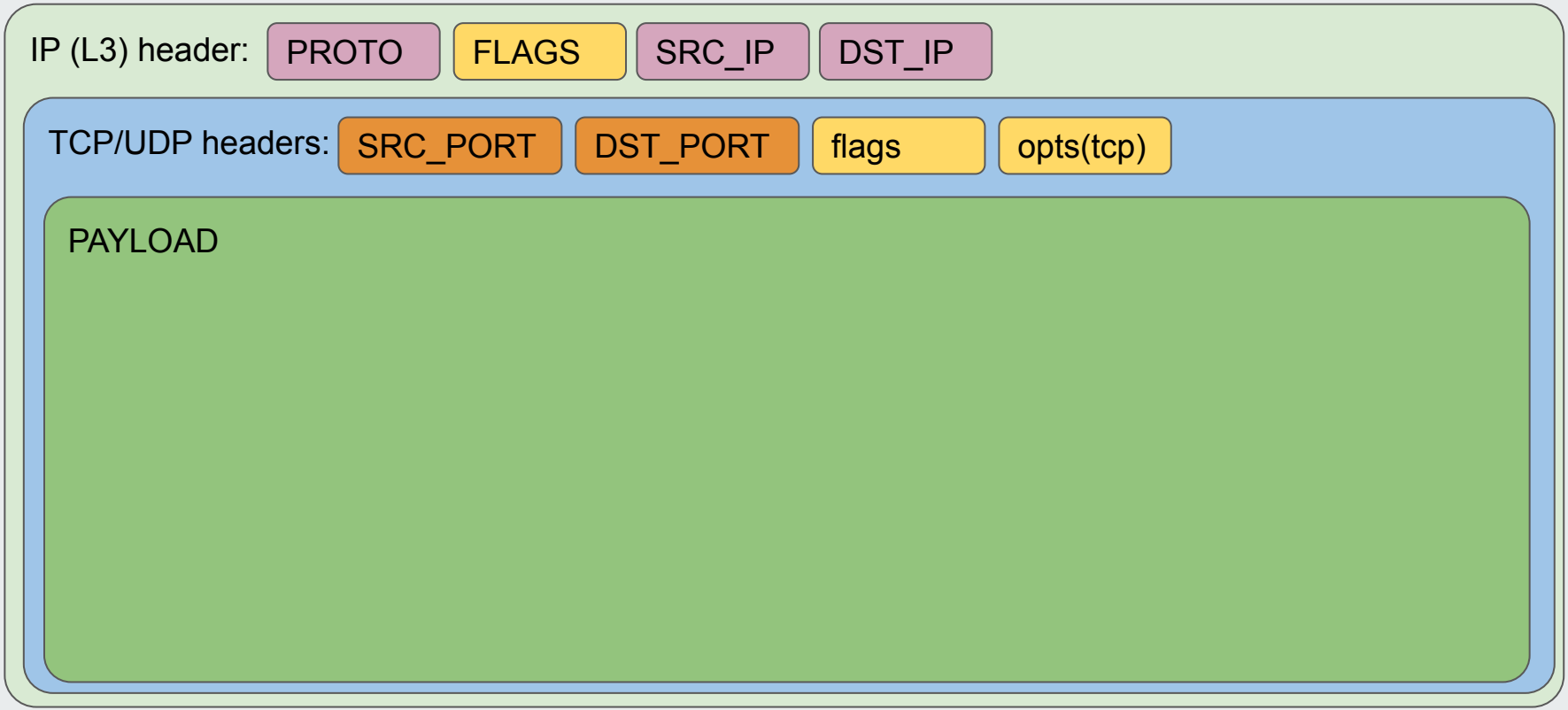
Цепочки

- PREROUTING - до принятия решения о маршрутизации
- POSTROUTING - после принятия решения о маршрутизации
- OUTPUT - пакеты сгенерированные локальными приложениями
- INPUT - пакеты предназначающиеся локальной системе
- FORWARD - пакеты проходящие через систему

Схема работы



IP-пакет


IP (L3) header:
PROTO
FLAGS
SRC_IP
DST_IP
TCP/UDP headers:
SRC_PORT
DST_PORT
flags
opts(tcp)
PAYLOAD

Критерии срабатывания(общие, L3)

-i/--in-interface	Входящий интерфейс
-o/--out-interface	Исходящий инерфейс
-s/--source	Адрес источника
-d/--destination	Адрес назначения
-p/--protocol	IP-Протокол (tcp,udp,icmp…)
-f/--fragment	Является ли фрагментом (2+ в серии)

Критерии срабатывания (L4)

TCP/UDP

--sport port[:port]	порт или диапазон портов источника
--dport port[:port]	порт или диапазон портов назначения

TCP

--tcp-flags mask flags (SYN,ACK,RST,FIN SYN)	флаги TCP
--syn	взведен SYN

ICMP

--icmp-type	тип icmp-пакета

Дополнительные модули срабатывания

- conntrack/state - критерии срабатывания основанные на состоянии соединения.
- multiport - критерий срабатывания позволяющий указывать список портов, а не диапазон.
- iprange - критерий срабатывания, который позволяет указать ip-range, вместо cidr-префикса.
- mark/connmark - критерий срабатывания основанный на маркировке пакета/соединения.
- set - критерий срабатывания основанный на ipset
- u32 - гибкий критерий срабатывания который позволяет работать напрямую с заголовками пакетов и отдельными битами.
- mac - критерий на основе MAC адреса
- limit - критерий на основе кол-ва пакетов в ед. времени (ex. 10/hour)
- addrtype - критерий на основе типа адресации (UNICAST/MULTICAST/BROADCAST)

ограничения с помощью iptables

Ограничения делаются с помощью критериев матчинга.

- connlimit - ограничение количества соединений с одного адреса

```
# allow 2 ssh connections per client host
        iptables -A INPUT -p tcp --syn --dport 22 -m connlimit --connlimit-above 2 -j REJECT

# you can also match the other way around:
        iptables -A INPUT -p tcp --syn --dport 22 -m connlimit --connlimit-upto 2 -j ACCEPT
```

- ratelimit - ограничение частоты срабатывания правила. Чаще всего применяется в комбинации с действием LOG
  - --limit *limit*/unit(second,minute,hour,day)
  - --limit-burst - максимальный всплеск пакетов

iptables: балансировка трафика:

```
iptables -A PREROUTING -p tcp --dport 80 -m state -m nth --every 3 --packet 0 -j
DNAT --to-destination 10.0.0.4
iptables -A PREROUTING -p tcp --dport 80 -m state -m nth --every 3 --packet 1 -j
DNAT --to-destination 10.0.0.5
iptables -A PREROUTING -p tcp --dport 80 -m state -m nth --every 3 --packet 2 -j
DNAT --to-destination 10.0.0.6
```

iptables: модуль limit

```
# Limit the number of incoming tcp connections
# Interface 0 incoming syn-flood protection
iptables -N syn_flood
iptables -A INPUT -p tcp --syn -j syn_flood
iptables -A syn_flood -m limit --limit 1/s --limit-burst 3 -j RETURN
iptables -A syn_flood -j DROP

#Limiting the incoming icmp ping request:
iptables -A INPUT -p icmp -m limit --limit  1/s --limit-burst 1 -j ACCEPT

iptables -A INPUT -p icmp -m limit --limit 1/s --limit-burst 1 -j LOG --log-prefix PING-DROP:
iptables -A INPUT -p icmp -j DROP

iptables -A OUTPUT -p icmp -j ACCEPT
```

iptables: модуль limit

```
# limit incoming connection to ssh server (port 22) no more than 10 connections in a 10 minute:

SERVER_IP = 192.168.10.20
iptables -I INPUT -p tcp -s 0/0 -d $SERVER_IP --sport 513:65535 --dport 22 -m state --state
NEW,ESTABLISHED -m recent --set -j ACCEPT

iptables -I INPUT -p tcp --dport 22 -m state --state NEW -m recent --update --seconds 600
--hitcount 11 -j DROP

iptables -A OUTPUT -p tcp -s $SERVER_IP -d 0/0 --sport 22 --dport 513:65535 -m state
--state ESTABLISHED -j ACCEPT
```

Цели

ACCEPT	T	Принять пакет
DROP	T	Отбросить пакет
REJECT	T	Отбросить пакет и сообщить источнику icmp-сообщением
RETURN	T	вернуться в вышестоящую цепочку или применить правило по-умолчанию
LOG	N	
chain_name	N	Перейти в цепочку *chain_name*
DNAT	T	Destination NAT
SNAT	T	Source NAT
MASQUERADE	T	Source NAT для динамически-конфигурируемых интерфейсов
SET		Добавление/удаления адреса в ipset

conntrack

Подсистема отслеживания состояния соединений. Базово соединения имеют состояния:

- NEW - новое соединение. Отбираются пакеты устанавливающие соединения.
- ESTABLISHED - Установленное соединение. Отбираются пакеты не !syn/syn+ack, !rst/fin, которые относятся к уже отслеживаемым соединениям
- RELATED - Относящиеся к другому, уже установленному соединению (passive ftp, icmp-messages)
- INVALID - пакеты принадлежность которых к отслеживаемым соединениям установить не удалось.

conntrack

- Подсистема очень удобна при невысокой нагрузке, при высокой же требует настройки, иначе может приводить к потере пакетов или связности в целом из-за переполнения таблицы **conntrack** или слишком большого её размера. в **HL** рекомендуется отключать, т.к. просмотр таблиц добавляет время к обработке пакета повышая **latency** приложения.

ipset

Иногда возникает необходимость добавлять большое количество правил, которые можно сгруппировать по какому-либо признаку. Например это адреса с которых были запросы отвечающие определенному паттерну, коих могут быть тысячи или сети принадлежащие одной стране.

Пусть, допустим, сверка с одним правилом, занимает ~1us, правил у нас 2000, в лучшем случае проверка iptables завершится через 1us (первое правило), в худшем - через 2ms, а это уже существенно, когда речь идет об одном пакете в рамках соединения.

Для того, чтобы можно было сократить количество правил фильтрации, объединяя правила по определенным признакам, можно использовать инструмент ipset, который позволяет строить списки адресов, вместо списков правил, что существенно упрощает обработку больших списков адресов.

ipset

- типы сетов
  - hash - использует hash-таблицу для хранения (больше памяти)
    - hash:net - список сетей с VLSM (Variable Length Subnet Mask) (0-32)
    - hash:ip - список сетей/ip с одинаковой длиной маски
  - bitmap - использует bitmap для хранения (max 65536 записей)
    - bitmap:ip,mac (8 байт на запись)
    - bitmap:ip - 1 bit на адрес
  - list:set - списки списков

ipset: полезные расширения

- skbinfo - сохранение “дополнительной” информации о пакете
  - skbmark - fwmark, например 0x22 или 0x1111/0xff00ffff
  - skbprio - id класса (maj:min - hex без 0x), например 1:10
  - skbqueue - id очереди - число, например 10
- nomatch - для hash-таблиц хранящих сеть **(net)**. Позволяет добавлять в список исключения
- timeout - зание TTL записи в списке.
- counters - разрешение подсчета пакетов/байт по элементу в списке.

ipset hash:

Как понятно из названия для хранения используется hash-таблица. Потребляет больше памяти, чем bitmap.

hash:net - наиболее гибкий, т.к. позволяет хранить сети с разной длиной маски, что дает возможность хранить любой адрес(/32) или сеть(/0-31).

```
ipset create set_hn hash:net [timeout SECS] [counters]
```

hash:ip - предназначен для хранения элементов одинаковой длины (ip или сети с одинаковой длиной маски.

```
ipset create set_hip hash:ip [timeout SECS] [counters]
```

Это не все типы, больше в *`ipset help`* или *`man ipset`*

ipset bitmap:

bitmap - сеты потенциально более производительные, но менее гибкие и ограничены по количеству записей (65536 записей).

bitmap:ip - тип сета для хранения ip или сетей с одинаковой длиной маски

bitmap:ip,mac - тип сета позволяющий хранить пару ip-mac, что может быть использовано для отлова адресов у которых меняются mac-адреса. “прибивание гвоздями” ip к mac’у лучше делать через

```
ip neigh add ADDR lladdr MAC nud permanent
```

Это не все типы, больше в ***ipset help*** или ***man ipset***

ipset пример:

```
ipset create otuset bitmap:ip,mac range 192.168.0.0/16
ipset add otuset 192.168.254.2,08:00:27:b2:b9:ab
ipset test otuset 192.168.254.2,08:00:27:b2:b9:ab
ipset list otuset
ipset flush otuset
ipset destroy otuset
```

Цепочки: raw

Предназначена для базовой обработки пакетов до conntrack, в частности для управления conntrack в отношении некоторых пакетов.
Цепочки:

- PREROUTING
- OUTPUT

Действия:

- NOTRACK - отключить conntrack для пакетов попадающих в правило
- CT - настроить работу с модулем conntrack
- DROP - отбросить пакет.

mangle

Предназначена для маркировки и классификации пакетов, модификации заголовков (tos, mss, ttl)

Цепочки:

- PREROUTING
- INPUT
- FORWARD
- OUTPUT
- POSTROUTING

mangle

Действия:

- TTL - установить ttl
- TOS - используется для установки разрядов в поле Type of Service заголовка IP
- MARK/CONNMARK - установить метку (fwmark) пакета/соединения
- CLASSIFY - классифицировать пакет для обработки в шейпере
- TPROXY - прозрачный прокси на локальный порт
- TCPMSS - позволяет изменять значение MSS в TCP SYN пакетах, для контроля максимального размера пакетов в этом соединении

    - iptables -t mangle -A FORWARD -p tcp --tcp-flags SYN,RST SYN -j TCPMSS --clamp-mss-to-pmtu
    - iptables -A FORWARD -p tcp --tcp-flags SYN,RST SYN -j TCPMSS --set-mss 128

Mangle: пример

```
iptables -A PREROUTING -i eth1 -t mangle -p tcp --dport 25 -j MARK --set-mark 1
echo 201 mail.out >> etc/iproute/rt_tables
ip rule add fwmark 1 table mail.out
ip rule ls
ip route add default via 195.96.98.253 dev ppp0 table mail.out
```

nat

Предназначена для манипуляций с адресами источника/назначения.
Цепочки:

- PREROUTING
- INPUT
- OUTPUT
- POSTROUTING

Действия:

- SNAT/MASQUERADE - Source NAT
- DNAT - Destination NAT
- REDIRECT - Подмена dst_ip:dst_port на свои собственные (частный случай DNAT)

Nat: пример

```
*nat
:PREROUTING ACCEPT [578:36692]
:POSTROUTING ACCEPT [0:0]
:OUTPUT ACCEPT [103:7355]
-A PREROUTING -p tcp -m tcp --dport 25271 -j DNAT --to-destination 5.5.5.5:25171
-A POSTROUTING -j MASQUERADE
COMMIT
```

Основная таблица, где происходит фильтрация пакетов.

Цепочки:

- INPUT
- FORWARD
- OUTPUT

Действия:

- ACCEPT
- REJECT
- DROP

Таблица предназначена для работы совместно с selinux.
Цепочки:

- INPUT
- FORWARD
- OUTPUT

Действия:

- SECMARK/CONNSECMARK - установить SELinux context для пакета/соединения

ICMP фильтрация

IPv4	
Action	Types
ALLOW	3, 0, 8, 11
DROP	4, 6, 13-18, 30-37
DROP when not needed	other types

ICMP фильтрация

Action	Types (Codes)
ALLOW	1, 2, 3 (0), 4 (1,2), 128, 129
Consider allowing	3 (1), 4 (0), 144-147
Policy dependent	15, 5-99, 102-126 154-199, 202-254
Consider dropping	100, 101, 127 138-140, 200, 201, 255
DROP addressed to example network	5-99, 102-126, 144-147 150, 154-199, 202-254 & consider dropping

ICMP фильтрация

IPv6 приходящий на файрвалл	
Action	Types (Codes)
ALLOW	1, 2, 3 (0), 4 (1,2), 128, 129 130-136, 141-143, 148, 149 151-153
Consider allowing	3 (1), 4 (0), 144-147, 150
Policy dependent	4-99, 102-126, 137, 139, 140
Consider dropping	100, 101, 127 154-199, 200-255
DROP in example	144-147, 150 policy dependent & consider dropping

Спасибо за внимание

Дмитрий Молчанов
Григорий Ожегов
Алексей Цыкунов