Настройка фильтрации трафика на MikroTik.

Часть 1. ОСОБЕННОСТИ РАБОТЫ ФАЙРВОЛА

https://spw.ru/educate/articles/nastrojka-filtracii-trafika-na-mikrotik/

Для базового понимания работы файрвола, необходимо ознакомиться с понятиями цепочки (chain), состояния соединения (connection state), условия и действия (action).

Цепочки (chain)

При фильтрации трафик, в зависимости от своего предназначения попадает в одну из цепочек (chain) обработки трафика. В фильтре предопределены три основные цепочки:

- input входящий трафик предназначенный для маршрутизатора. Например, когда вы подключаетесь к маршрутизатору при помощи приложения winbox, трафик как раз попадает в эту цепочку.
- output Исходящий трафик. Трафик, создаваемый самим маршрутизатором. Например, если вы выполните команду ping непосредственно с самого маршрутизатора, трафик попадет в эту цепочку.
- forward Трафик, идущий через маршрутизатор. Например, если компьютер из локальной сети, установил соединение с внешним сайтом, данный трафик попадает в цепочку forward.

Таким образом мы видим, что для защиты самого маршрутизатора необходимо использовать цепочку input, а для защиты и фильтрации трафика между сетями необходимо использовать цепочку forward.
Кроме того, администратор имеет возможность создавать свои собственные цепочки обработки трафика, к которым можно обращаться из основных цепочек. Данная возможность будет рассмотрена в дальнейшем.
Состояние соединения (connection state)

Каждое из сетевых соединений MikroTik относит к одному из 4 состояний:

- New – Новое соединение. Пакет, открывающий новое соединение, никак не связанное с уже имеющимися сетевыми соединениями, обрабатываемыми в данный момент маршрутизатором.
- Established – Существующее соединение. Пакет относится у уже установленному соединению, обрабатываемому в данный момент маршрутизатором.
- Related – Связанное соединение. Пакет, который связан с существующим соединением, но не является его частью. Например, пакет, который начинает соединение передачи данных в FTP-сессии (он будет связан с управляющим соединением FTP), или пакет ICMP, содержащий ошибку, отправляемый в ответ на другое соединение.
- Invalid – Маршрутизатор не может соотнести пакет ни с одним из вышеперечисленных состояний соединения.

Исходя из вышеизложенного, мы видим, что хорошим вариантом настройки фильтрации пакетов будет следующий набор условий:

1. Обрабатывать новые соединения (connection state = new), принимая решение об пропуске или блокировке трафика.
2. Всегда пропускать соединения в состоянии established и related, так как решение о пропуске этого трафика было принято на этапе обработки нового соединения.
3. Всегда блокировать трафик, для которого состояние соединения равно invalid, потому что этот трафик не относится ни к одному из соединений и фактически является паразитным.

Условие

При прохождении пакета через фильтр, маршрутизатор последовательно проверяет соответствие пакета заданным условиям, начиная от правила, расположенного первым. и последовательно проверяя пакет на соответствие правилам номер два, три и так далее, пока не произойдет одно из двух событий:

4. Пакет будет соответствовать заданному условию. При этом сработает соответствующее правило, в котором это условие было задано, после чего обработка пакета будет завершена.

5. Закончатся все условия и пакет не будет признан соответствующим ни одному из них. При этом, по умолчанию он будет пропущен дальше.

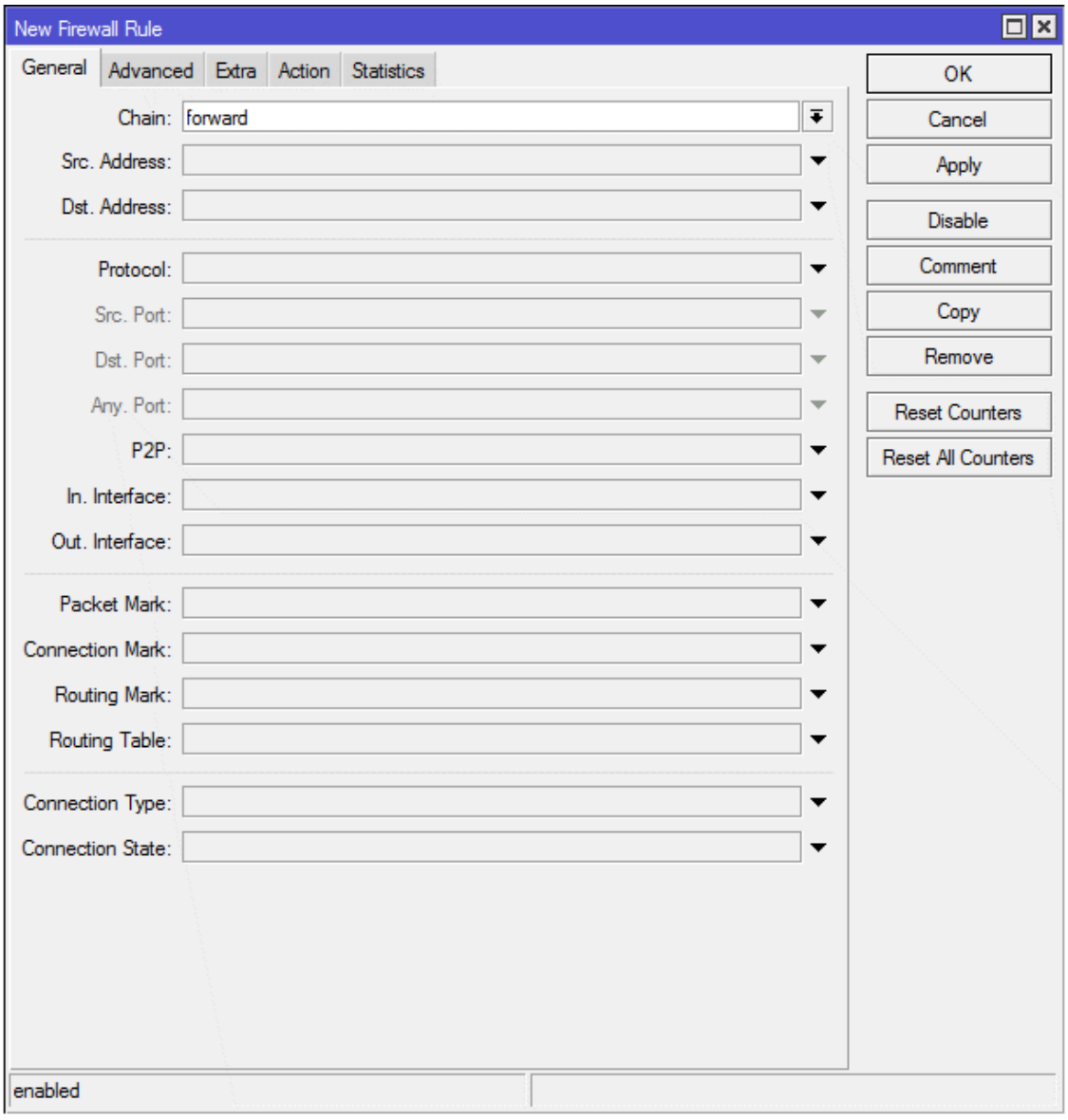
Исходя из п.2, нельзя не отметить, что есть две стратегии построения фильтра пакетов:

6. Нормально открытый файрвол. Данный тип настройки можно определить как «Все разрешено, что не запрещено». При этом мы запрещаем прохождение только некоторых типов трафика. Если пакет не соответствует этим типам – он будет пропущен. Обычно данный тип файрвола характерен для мест, где не предъявляется высоких требований к безопасности пользователей, а трафик может быть самым разнообразным и не поддающимся жесткой квалификации. Такая настройка характерна для операторов связи (Интернет-провайдеров), открытых точек доступа, домашних маршрутизаторов.
7. Нормально закрытый файрвол. Данный тип настройки можно определить как «Все запрещено, что не разрешено». При этом разрешается прохождение только определенных типов трафика, а последним правилом в файрволе стоит правило, запрещающее прохождение любого типа трафика. Такой тип настройки файрвола характерен для корпоративного использования, где существуют жесткие требования к безопасности.

Не могу сказать, что какая-то из стратегий является правильной, а какая-то неправильной. Обе стратегии имеют право на жизнь, но каждая — в определенных условиях.

Теперь подробнее распишем все варианты условий, на основании которых мы можем принимать решение о действии.

Закладка general



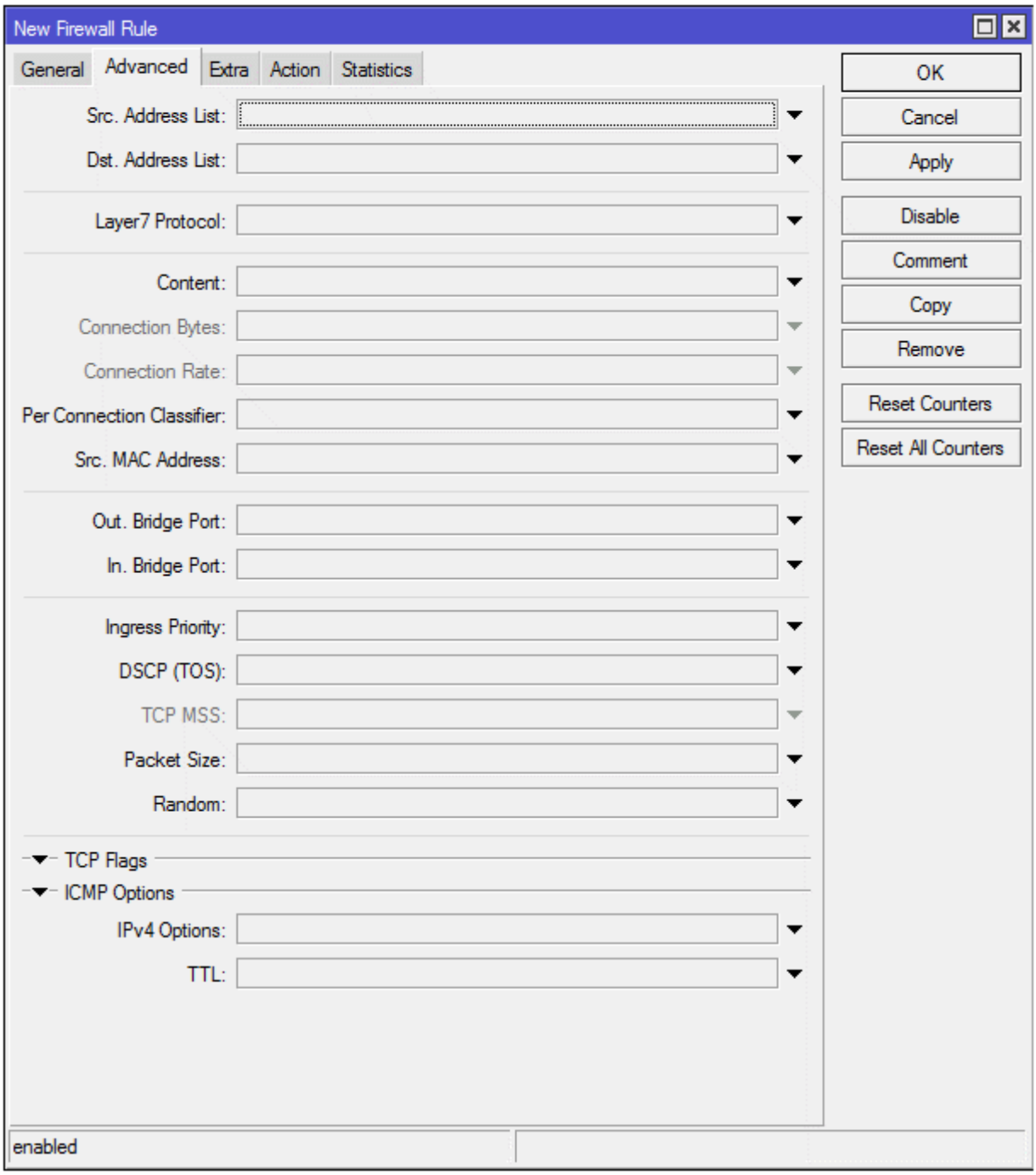
Наименование	Описание
Chain	Цепочка (см. выше). Варианты в списке: input, output, forward. Если ввести свое название – получим свою цепочку.
Src. Address	Адрес источника пакета. Варианты заполнения поля: Один адрес. Например, 192.168.0.5 Подсеть. Например, 192.168.0.0/24 Диапазон адресов. Например, 192.168.0.5-192.168.0.15 Обратите внимание: если вам надо задать несвязанный диапазон адресов, то это нельзя сделать в этом поле, но можно сделать через Src. Address List на закладке Advanced
Dst. Address	Адрес назначения пакета. Варианты заполнения поля см. выше
Protocol	Протокол соединения. TCP, UDP, ICMP и т.п.
Src. Port	Порт, с которого пришел пакет. Поле можно заполнить только если протокол соответствует TCP или UDP. Варианты заполнения поля: Один порт. Например, 22. Диапазон портов. Например, 10000-20000. Несколько портов. Например ,22,23,25. Несколько диапазонов портов. Например, 5060-5070,10000-20000 . Диапазон и несколько портов. Например, 22,23,10000-20000.
Dst. Port	Порт, на который пришел пакет. Поле можно заполнить, только если протокол соответствует TCP или UDP. Варианты заполнения поля см. выше.
Any Port	Любой порт. Например, или Src. Port, или Dst. Port Варианты заполнения поля см. выше.
P2P	Peer-to-Peer протокол. Пакет относится к одному из P2P протоколов. Например, edonkey или BitTorrent. Обратите внимание, что шифрованные сессии не идентифицируются посредством данного поля.
In Interface	Интерфейс, с которого пришел проверяемый пакет. (Не работает, если chain=output, т.к. источник трафика - сам маршрутизатор)
Out Interface	Интерфейс, куда будет передан пакет. (Не работает, если chain=input, так как трафик предназначен для маршрутизатора и дальше передан быть не может).
Packet Mark	Пакет имеет определенную маркировку, полученную ранее через Mangle.
Connection Mark	Пакет имеет определенную маркировку, полученную ранее через Mangle.
Routing Mark	Пакет имеет определенную маркировку, полученную ранее через Mangle.
Connection Type	Пакет относится к определенному типу соединения, включенному на закладке Firewall/Service Ports
Connection State	Состояние соединения. Описывалось выше.

Обратите внимание, что перед частью полей можно поставить флаг восклицательного знака. Этот флаг будет обозначать отрицание. Например:

обозначает что адрес источника любой, кроме 192.168.0.0/24 . Также обратите внимание, что если поле не заполнено, оно должно быть серым. Если вы передумали заполнять поле, чтобы его исключить и сделать серым – нажмите стрелку «вверх», справа от поля.

Закладка Advanced

На этой закладке собраны расширенные опции выбора пакета.

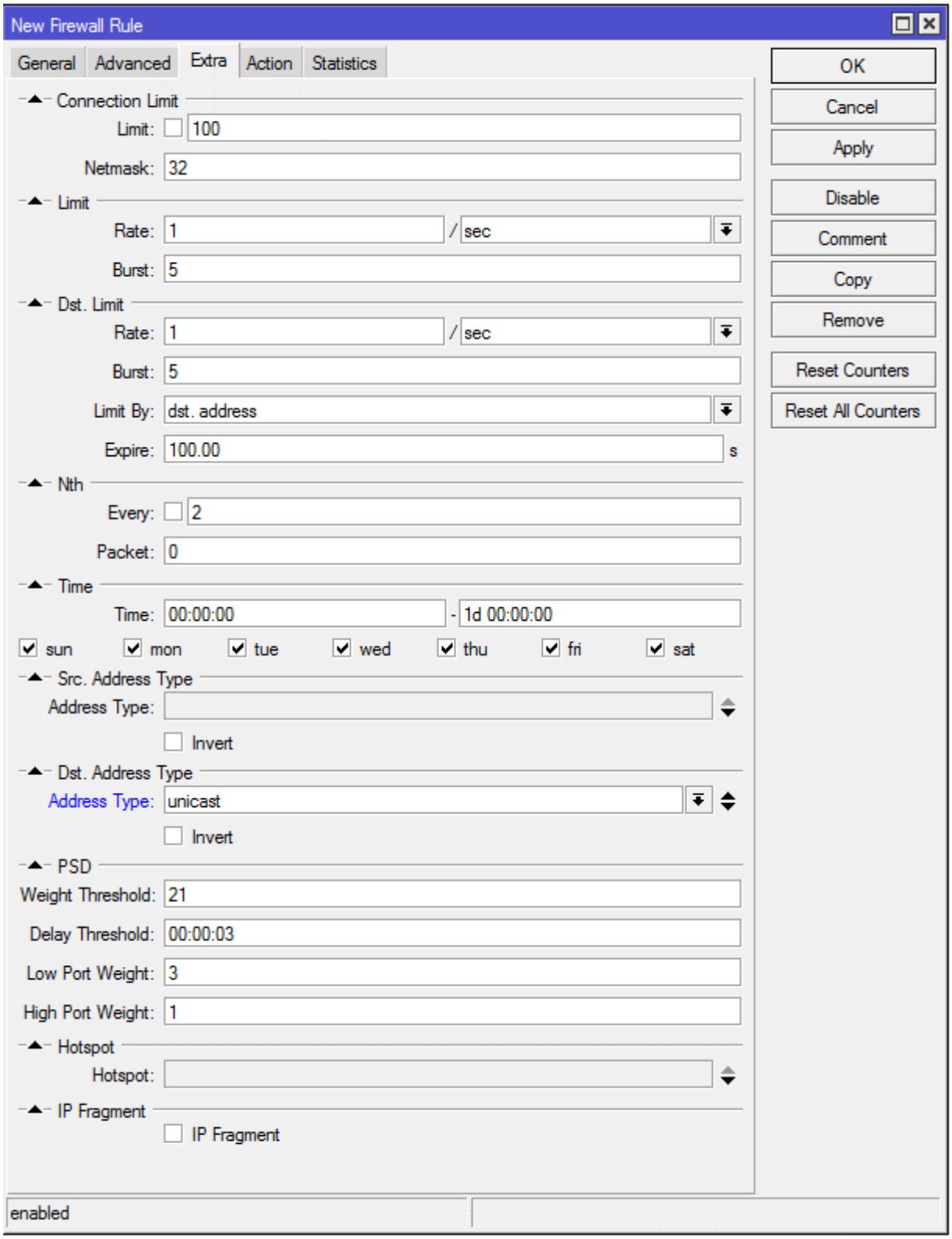


Наименование	Описание
Src. Address List	Адрес источника пакета совпадает с одним из адресов в именованном списке адресов, заданном на закладке Firewall/Address Lists.
Dst. Address List	Адрес назначения пакета совпадает с одним из адресов в именованном списке адресов, заданном на закладке Firewall/Address Lists.
Layer 7 Protocol	При проверке пакета L7-фильтром, заданным на закладке Firewall/Layer 7 Protocols, он был отнесен к одному из определенных на этой закладке протоколов.
Content	Внутри пакета содержится определенная строка символов.

Connection Bytes	Количество байт, прошедших через соединение. При этом 0 обозначает бесконечность. Например, 1000000-0 = более 1МБ.
Connection Rates	Скорость соединения. Например, 0-128000. Это правило сработает, если скорость подключения менее 128 килобит в секунду. (Поставив флаг [!] перед таким правилом, мы заставим срабатывать правило на соединение более 128kbps)
Per Connection Classifier	Используется при необходимости разделения трафика на несколько потоков. Позволяет держать пакеты с определенным набором опций в одном потоке. Подробнее: http://wiki.mikrotik.com/wiki/Manual:PCC
Src. MAC Address	MAC-адрес сетевой карты источника. Сработает, только если источник пакета находится в одном Ethernet-сегменте с маршрутизатором.
Out Bridge Port	Порт назначения интерфейса типа bridge, при активированной в Bridge опции Use IP Firewall.
In Bridge Port	Порт источника интерфейса типа bridge, при активированной в Bridge опции Use IP Firewall.
Ingress Priority	Приоритет пакета. Может быть получен из VLAN, WMM или MPLS ext. bit
DSCP (TOS)	Определяет DSCP, заданный в заголовке пакета.
TCP MSS	Размер MSS (Maximum segment size) TCP пакета.
Packet Size	Размер пакета.
Random	Случайное срабатывание правила. Число задается в диапазоне 1-99, что соответствует вероятности срабатывания правила от 1 до 99 процентов. Обычно используется при тестировании сервисов, когда надо изобразить случайную потерю пакетов на нестабильном канале.
TCP Flags	Флаги TCP соединения.
ICMP Options	Опции (типы сообщения) ICMP.
IPv4 Options	В заголовке пакета имеется заданная опция протокола Ipv4.
TTL	Time To Live – Время жизни пакета соответствует …

Закладка Extra

Эта закладка продолжает список расширенных опций, не поместившихся на закладку Advanced.



Итак:

Как мы видим, в маршрутизаторе существует достаточно большое количество правил выбора пакетов, которые позволяют очень гибко и тонко настраивать работы с трафиком.

Теперь, когда мы поняли, на основании каких правил мы можем найти интересующий нас пакет, давайте посмотрим, что можно сделать после срабатывания правила.

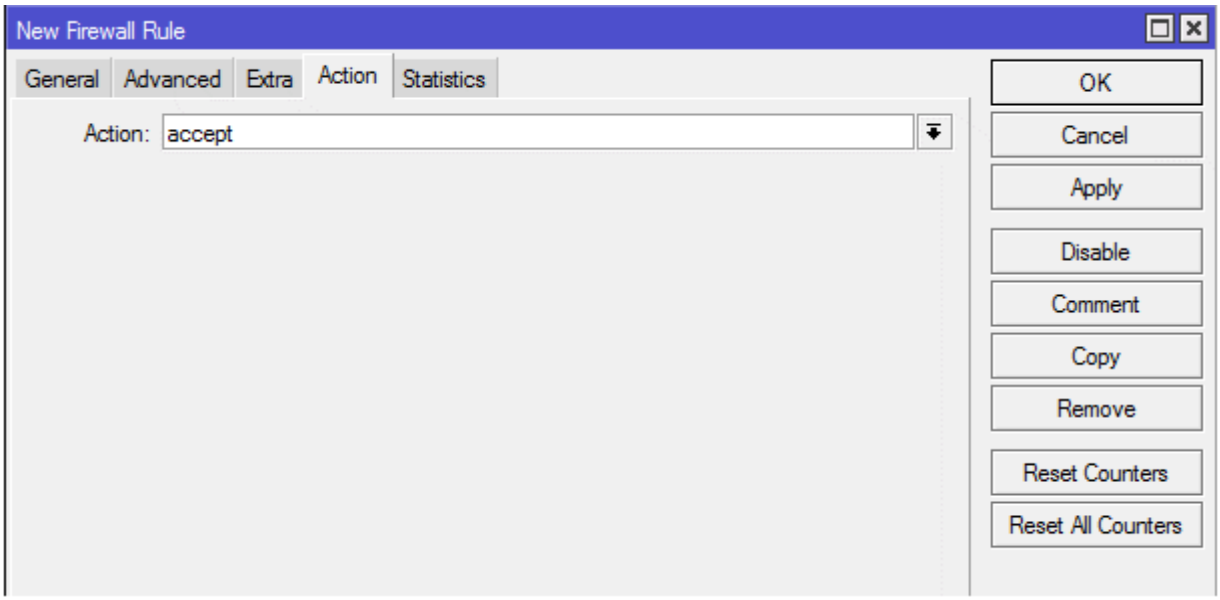
Наименование	Описание
Connection Limit	Предел количества соединений для адреса или подсети. Адрес или подсеть задается полем netmask (для 1 адреса 32).
Limit	Предназначено для ограничения количества передаваемых пакетов: Поля: Rate – количество пакетов в секунду (минуту/час). Burst – Количество неучитываемых пакетов (пакетов не входящих в packet rate).
Dst. Limit	Ограничение количества передаваемых пакетов по адресу источника/назначения. В отличии от limit, учитываются пакеты для каждого адреса или адреса/порта в зависимости от выбранных опций. Поля rate и burst соответствуют таковым в опции Limit. Дополнительный поля: Limit By – по какому критерию (src|dst address | address/port) учитывать пакеты. Expire - через какой промежуток времени запомненный адрес/порт будут удалены.
Nth	Каждый из: Every – из какого числа пакетов. Packet – сколько. Например Every=3, packet=2 Обозначает «Каждые 2 из 3 пакетов или проще 2/3 пакетов). Опцию часто используют при балансировке нагрузки между каналами.
Time	Время действия правила. Позволяет ограничить действие правила во времени и по дням недели. Так как у маршрутизатора нет аппаратно-независимых часов, для корректной работы опции требуется настроенный SNTP-клиент (System/SNTP-Client) и часовой пояс (System/Clock)
Src. Address Type	Тип IP-адреса источника (Local, Unicast, Broadcast, Multicast)
Dst. Address Type	Тип IP-адреса назначения (Local, Unicast, Broadcast, Multicast)
PSD	Port Scan Detect. Опция позволяющая настроить определение события сканирования портов. Поля: Weight Threshold = При каком значении сработает. Delay Threshold = Максимальная задержка между пакетами с разными портами назначения, пришедшими с одного адреса. Low Port Weight = сколько при подсчете стоит каждый порт в диапазоне 0-1023. High Port Weight = сколько при подсчете стоит каждый порт в диапазоне 1024-65535. Например, на скриншоте правило сработает, если будет просканировано 7 и более портов в привилегированном диапазоне; Или 21 и более портов в непривилегированном диапазоне. При этом пауза между поступающими пакетами с одного источника, направленного на разные порты будет не более 3 секунд.
Hotspot	Опции, связанные с работой хотспот, если он настроен на маршрутизаторе.
IP Fragment	Пакет является фрагментом другого пакета.

Как мы видим, в маршрутизаторе существует достаточно большое количество правил выбора пакетов, которые позволяют очень гибко и тонко настраивать работы с трафиком.

Теперь, когда мы поняли, на основании каких правил мы можем найти интересующий нас пакет, давайте посмотрим, что можно сделать после срабатывания правила.

ДЕЙСТВИЯ ПРИ ФИЛЬТРАЦИИ ПАКЕТОВ

Действия задаются на закладке Action сформированного правила.



Рассмотрим их:

Accept

Разрешить прохождение пакета. Дальнейшие действия по фильтрации прекращаются, пакет передается на следующий этап обработки.

add-dst-to-address-list

Добавить адрес назначения пакета в именованный список адресов (address list).

Опции:

- Address-List – Имя списка адресов. Выбирается из списка или задается новое.
- Timeout – Время, которое данный адрес будет присутствовать. По истечении заданного времени адрес будет удален из списка.

add-src-to-address-list

Добавить адрес источника пакета в именованный список адресов (address list).

Опции:

- Address-List – Имя списка адресов. Выбирается из списка или задается новое.
- Timeout – Время, которое данный адрес будет присутствовать. По истечении заданного времени адрес будет удален из списка.

Обратите внимание, что динамические списки адресов являются очень мощным инструментом. Так как мы можем учитывать списки адресов в правилах выбора пакета на закладке Advanced, фактически, таким образом мы можем динамически менять правила фильтрации трафика.

Drop

Удалить пакет. Пакет уничтожается и никуда дальше не передается.

Jump

Перейти на собственную цепочку (chain) обработки пакетов.
Опция – наименование цепочки.

Log

Занести информацию о пакете в Log-файл маршрутизатора. При этом пакет будет передан на следующее правило. Данная опция часто используется при отладке.

Passthrough

Ничего не делать. Передать пакет на следующее правило. Однако при этом счетчики работают, показывая сколько пакетов соответствовало этому правилу. Обычно используется для статистики.

Reject

Запретить прохождение пакета и отправить отправляющему узлу ICMP-сообщение об ошибке.
Опция – вид сообщения.

Return

Досрочно прервать обработку собственной цепочки (chain) и вернуться на следующее правило за правилом с Action=jump, которое передало пакет в эту цепочку.

Tarpit

Очень интересная опция. Может использоваться только с протоколом TCP. Суть в том, что маршрутизатор дает разрешение на создание соединения, при этом выставляя нулевое окно передачи (т.е. скорость соединения = 0). Позволяет «завесить» атакующий хост на этом соединении.

ЗАКЛЮЧЕНИЕ

В этой части мы разобрались с основными опциями файрвола. Обратите внимание, что условия выбора пакетов одинаковы для всего файрвола и будут нам требоваться в дальнейшем при изучении NAT и расширенной обработки трафика (Mangle). А вот названия цепочек и действия там будут совершенно другими.

Часть 2. ПЕРЕД НАСТРОЙКОЙ ФАЙРВОЛА

https://spw.ru/educate/articles/nastrojka-filtracii-trafika-na-mikrotik-chast-2/

В этой и следующей части статьи используется следующая топология сети:

- WAN – 172.30.10.26/24, Default Gateway 172.30.10.1
- DMZ – 10.10.10.1./24
- LAN – 192.168.88.1/24

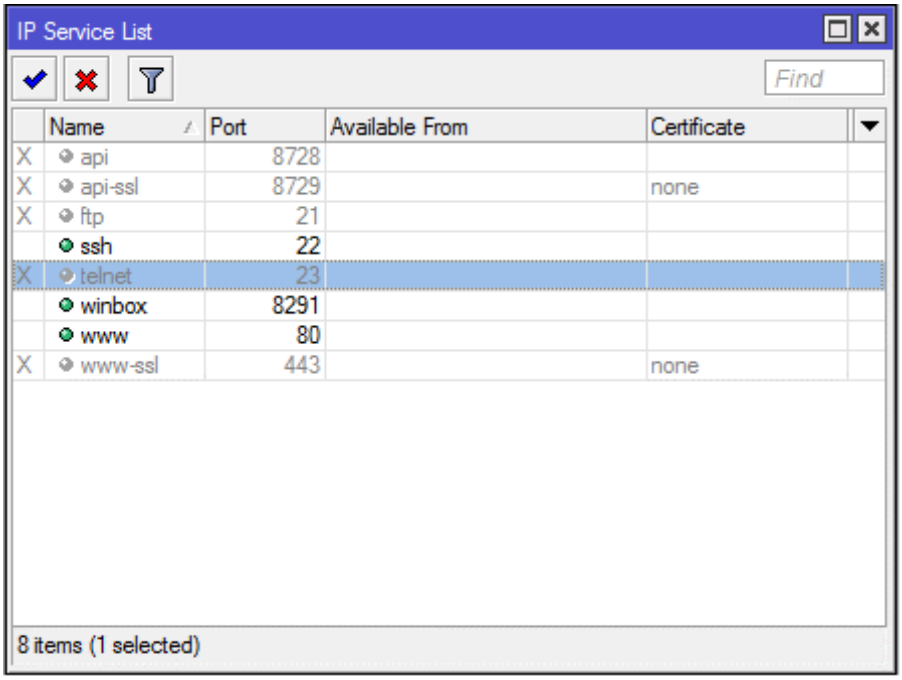
Как это настроить — смотрите статью "Разумная настройка RB2011"

Немного повторяя статью по настройке RB 2011, я позволю себе напомнить, что для начала необходимо отключить неиспользуемые на маршрутизаторе сервисы. Фактически для настройки и работы с маршрутизатором нам достаточно сервисов:

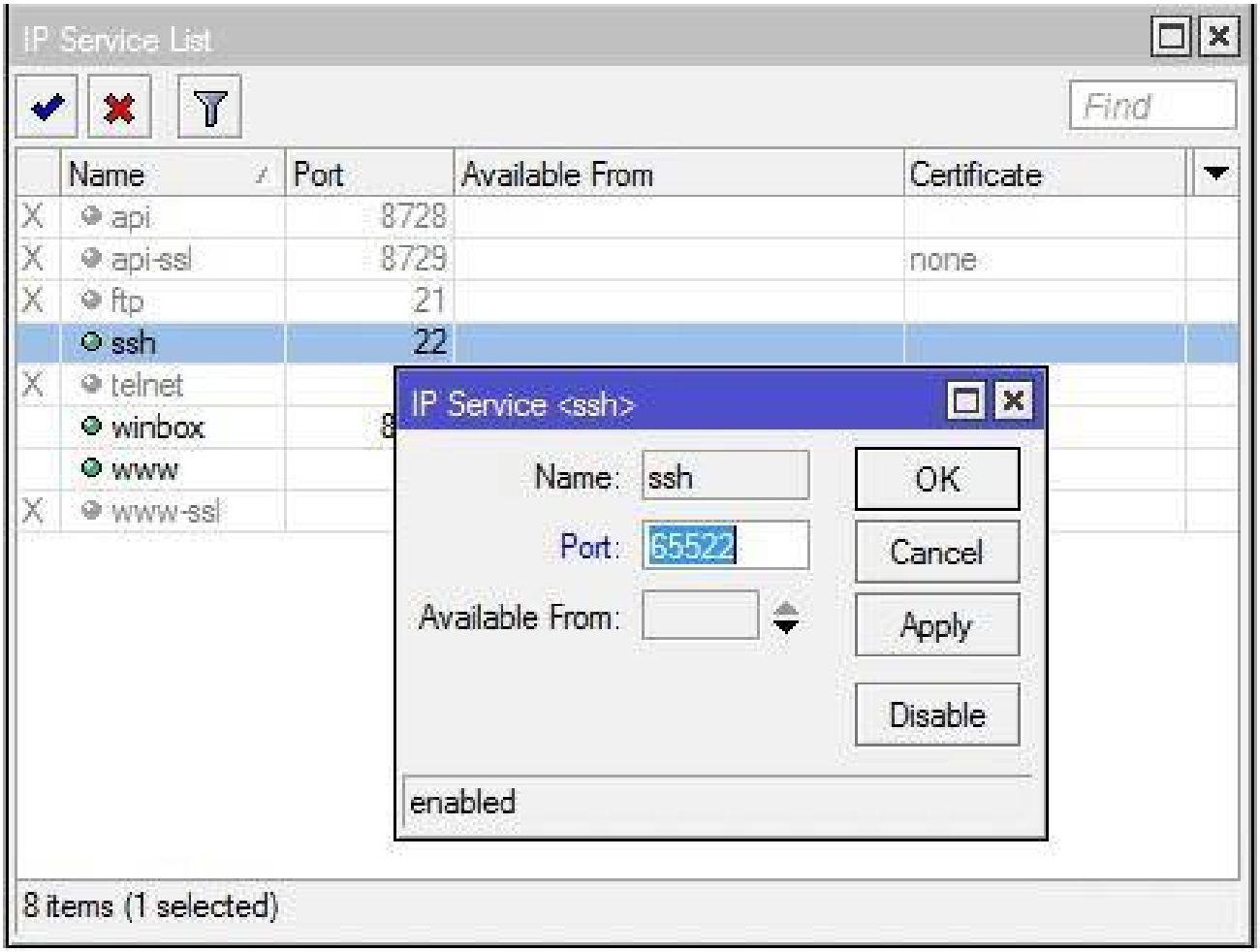
- Winbox – порт 8291 TCP;
- Ssh – порт 22 TCP;
- www – порт 80 TCP;

ЭТАП 1. ОТКЛЮЧЕНИЕ НЕНУЖНЫХ СЕРВИСОВ

Открываем меню IP / Services и воспользовавшись кнопкой Disable отключаем ненужные нам сервисы.



Если вы используете для работы с MikroTik протокол ssh, хорошей идеей будет изменить его стандартный порт на какой-нибудь другой, например 65522. Для этого дважды щелкаем по строке с ssh и в открывшемся окне меняем порт:



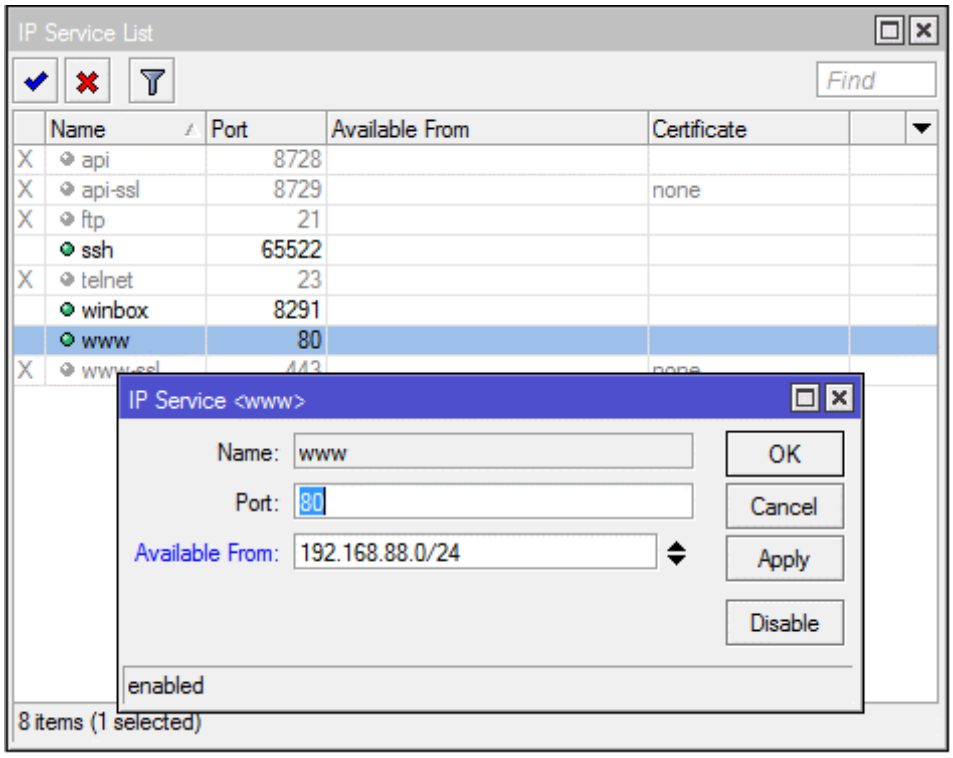
Нажав OK, подтверждаем выбор.

Также можно поменять порты winbox и www. Однако мы не наблюдали, чтобы порт 8291 (winbox) подвергался атаке по подбору пароля.

А вот в случае атаки порта ssh к маршрутизатору, находящемуся на внешнем IP адресе, производится до нескольких сотен попыток несанкционированного подключения в сутки.

Web-интерфейс, конечно, тоже лучше отключить, но если вы по каким-то причинам не можете использовать для настройки winbox, то будет хорошим решением разрешить доступ к web-интерфейсу маршрутизатора только из локальной сети.

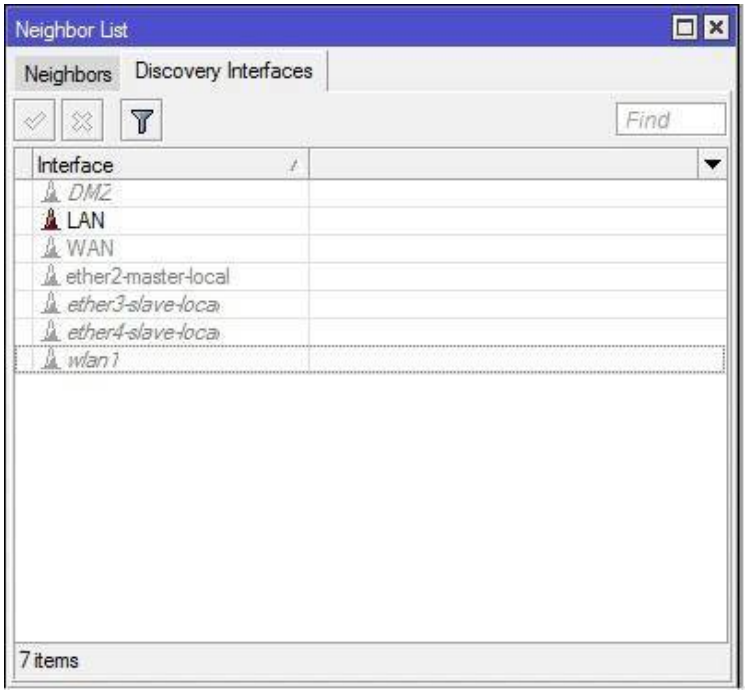
Для этого дважды щелкаем по строке с www и заполняем поле available from :



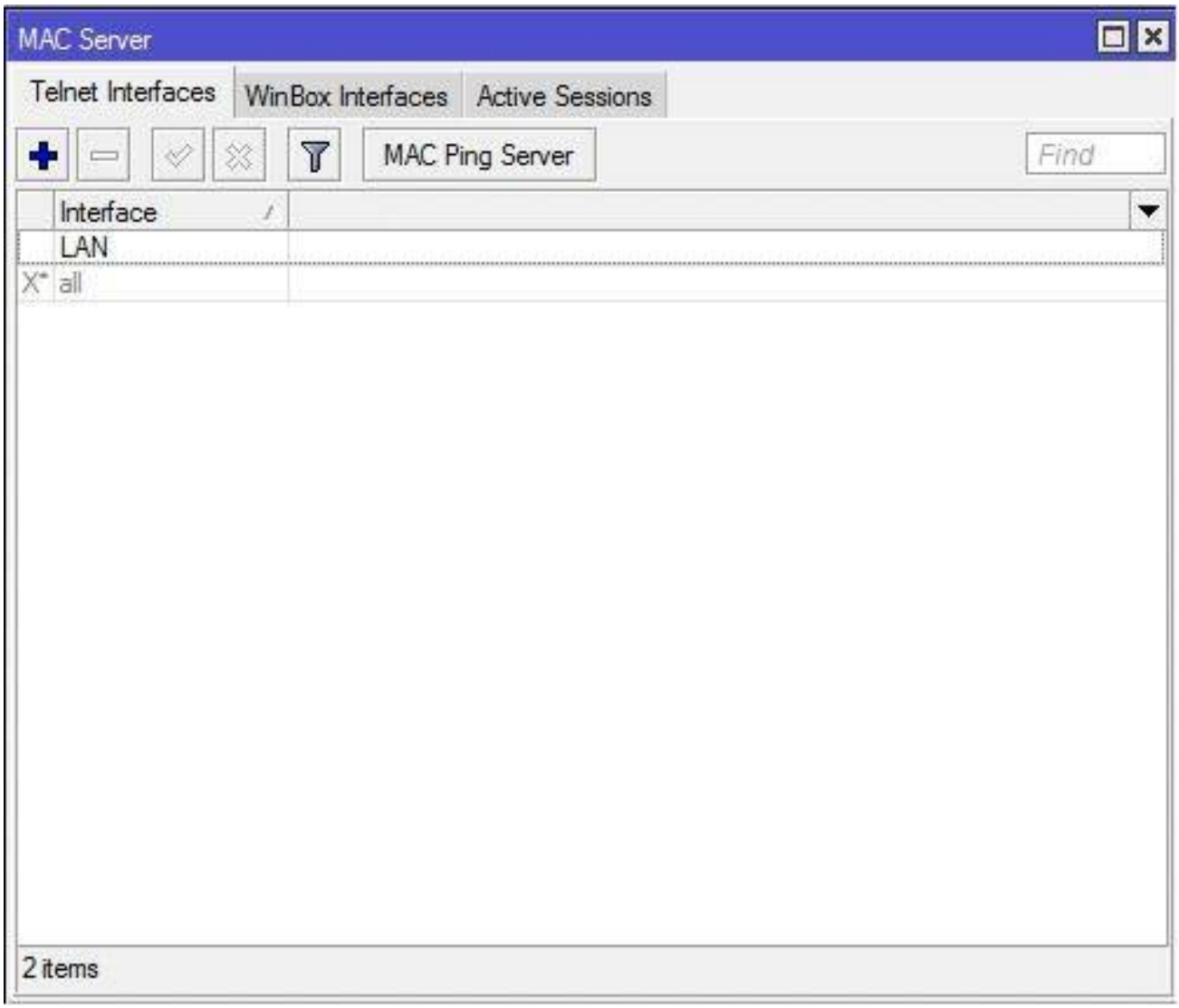
Нажав кнопку ОК, подтверждаем выбор.

ЭТАП 2. ОТКЛЮЧЕНИЕ ПОИСКА СОСЕДЕЙ И MAC-СЕРВЕРА НА ВНЕШНИХ ИНТЕРФЕЙСАХ

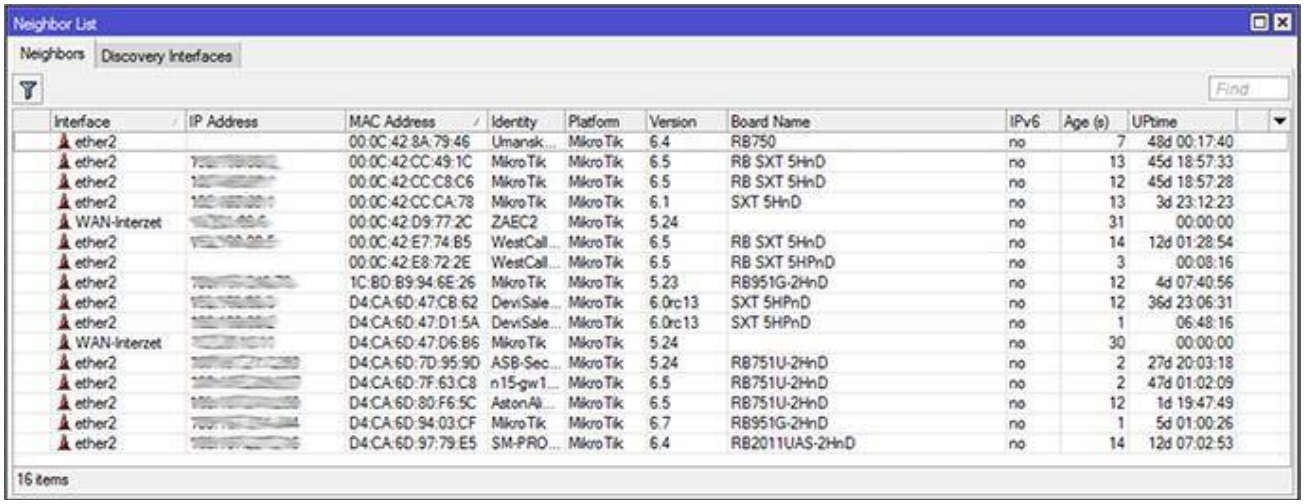
Заходим в меню ip/neighbors, переходим на закладку Discovery Interfaces и отключаем все кроме интерфейса LAN, нажатием на кнопку Disable.



Далее идем в меню Tools/MAC Server и на закладках Telnet Interfaces и WinBox Interfaces добавляем интерфейс LAN, удаляем если есть любые другие интерфейсы и отключаем интерфейс "*all"



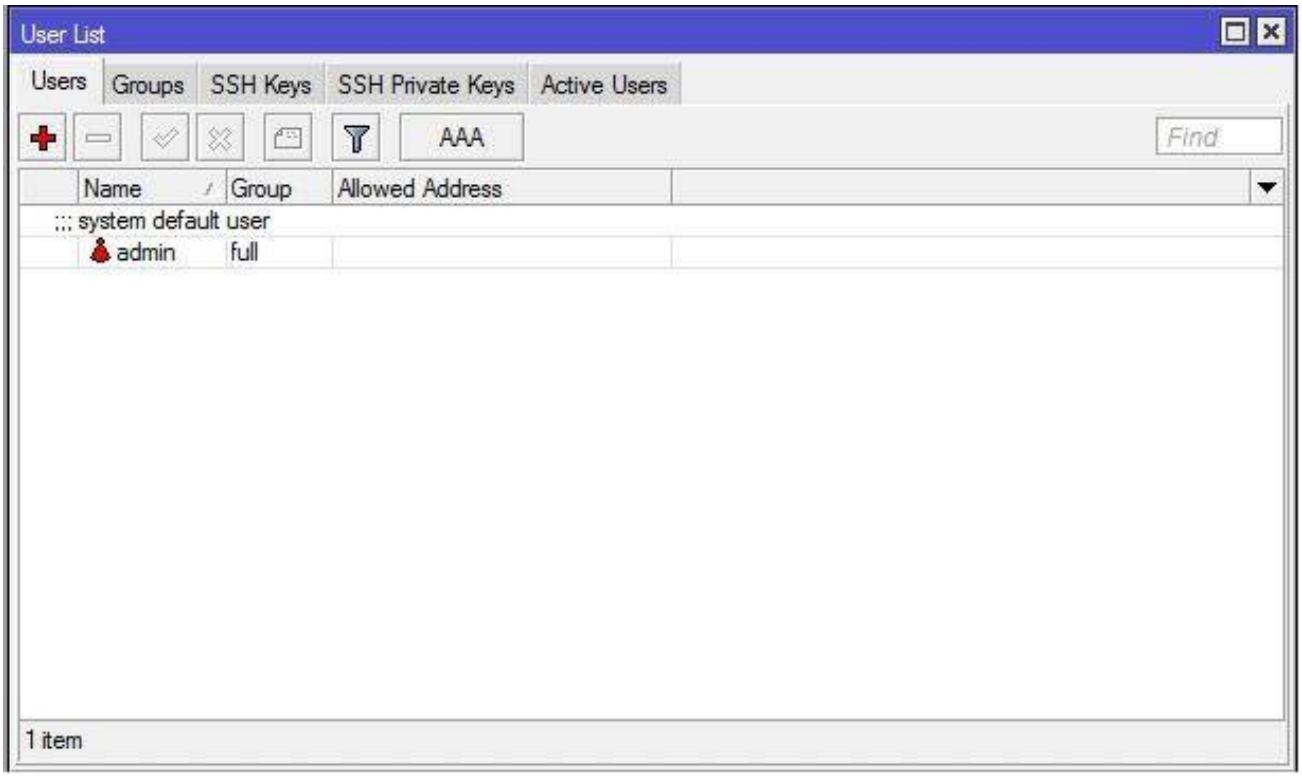
Это не даст возможности подключиться к маршрутизатору снаружи, при помощи MAC - Telnet
Если кто-то считает что это излишние меры безопасности, вот что видит на WAN -портах один из установленных в работу маршрутизаторов.



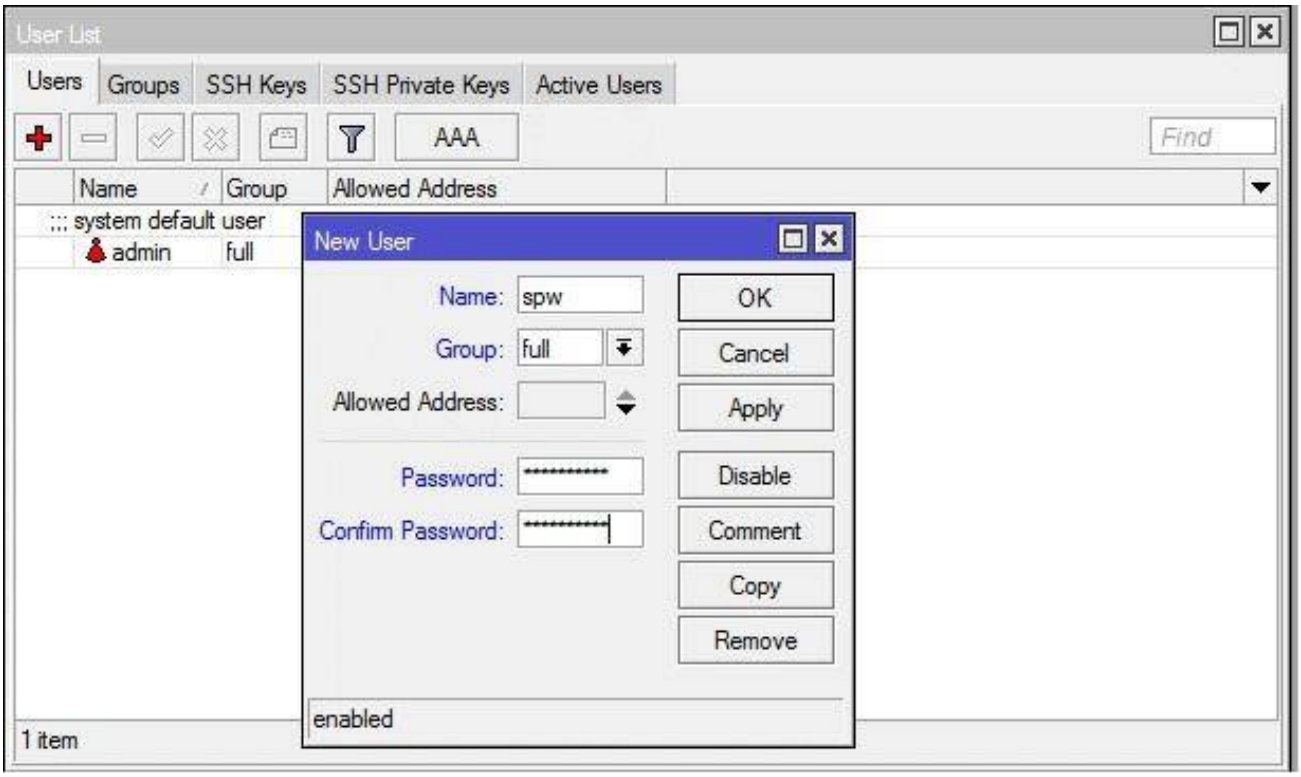
При этом на два из них удалось зайти mac-telnet и получить доступ к управлению.

ЭТАП 3. ПОЛЬЗОВАТЕЛЬ И ПАРОЛЬ

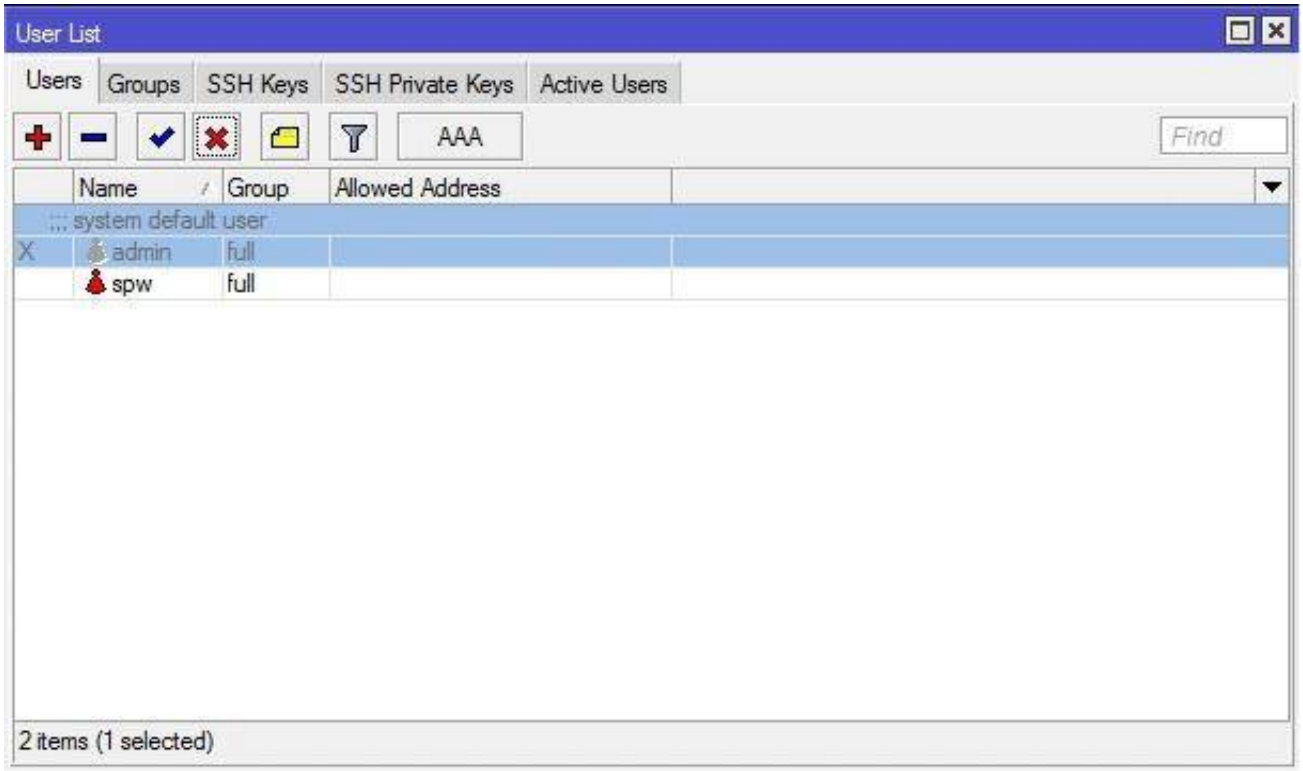
Теперь надо заменить пользователя по умолчанию и установить ему пароль.
Переходим в меню System/Users.



Создаем нового пользователя с правами администратора.



После чего закрываем программу Winbox, запускаем его заново и заходим под новым пользователем, открываем меню System/Users и отключаем учетную запись администратора.



На этом подготовительные операции можно считать законченными. Переходим к настройке firewall.

НАСТРОЙКА ФАЙРВОЛА

В предыдущей части статьи мы с вами узнали, что:

1. Трафик, идущий на маршрутизатор, попадает в цепочку файрвола input;
2. Трафик, создаваемый маршрутизатором, попадает в цепочку файрвола output;
3. Трафик, идущий через маршрутизатор, попадает в цепочку forward;
4. Существуют четыре состояния соединения: new, established, related, invalid.

То есть, исходя из состояний соединения и цепочек, общие правила защиты маршрутизатора можно сформулировать как:

1. Мы работаем только с цепочкой input;
2. Мы пропускаем соединения с состоянием established и related, как уже установленные;
3. Мы пропускаем протокол ICMP;
4. Мы считаем как WAN, так и DMZ недоверенными сетями;
5. Мы разрешаем прохождение некоторого трафика на маршрутизатор. Остальной трафик блокируем.

Теперь давайте определим разрешенный трафик с недоверенных интерфейсов. Итак, мы разрешаем:

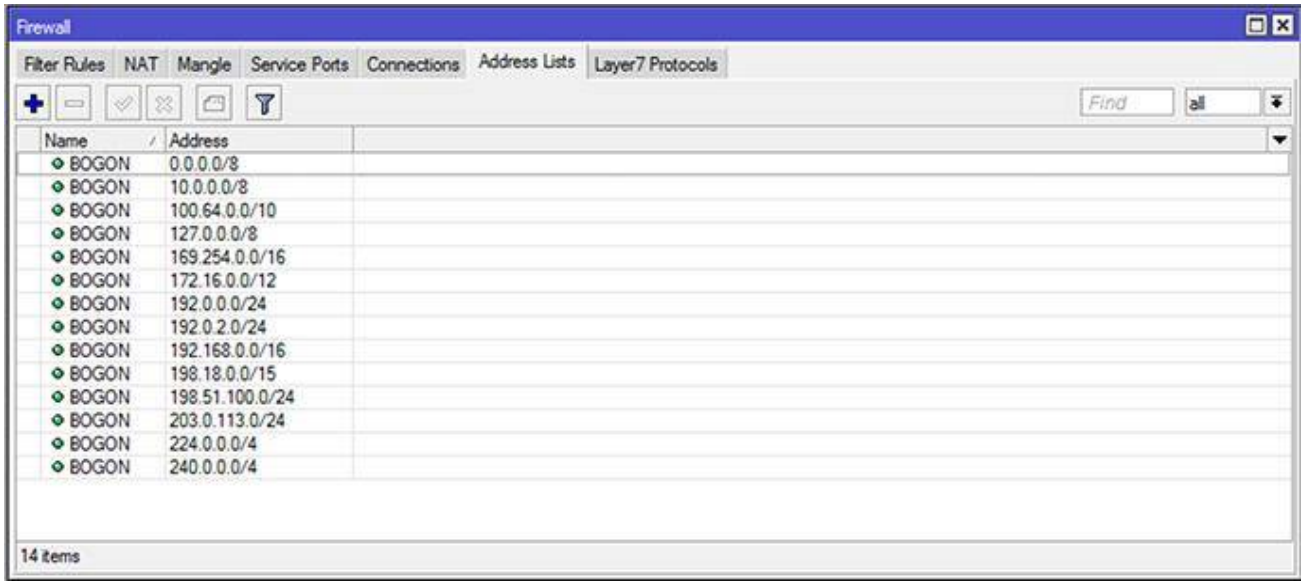
1. TCP порт 8291 – winbox, удаленное управление снаружи;
2. 65522 ssh на измененном порту;
3. Предположим, что у нас в дальнейшем будет настраиваться VPN-сервер по протоколу PPTP и мы разрешим порт 1723 по протоколу TCP.

Также с этого момента мы начинаем работать с командной строкой маршрутизатора. Все команды вставляются в терминал маршрутизатора. Если необходимо – вы можете посмотреть в графическом интерфейсе, что конкретно было сделано. Очень скоро вы научитесь читать команды и соотносить их с графическим интерфейсом. Определяем так называемые bogon-сети (сети приватных или не распределенных IP-адресов).

```
/ip firewall address-list
add address=0.0.0.0/8 disabled=no list=BOGON
add address=10.0.0.0/8 disabled=no list=BOGON
add address=100.64.0.0/10 disabled=no list=BOGON
```

```
add address=127.0.0.0/8 disabled=no list=BOGON
add address=169.254.0.0/16 disabled=no list=BOGON
add address=172.16.0.0/12 disabled=no list=BOGON
add address=192.0.0.0/24 disabled=no list=BOGON
add address=192.0.2.0/24 disabled=no list=BOGON
add address=192.168.0.0/16 disabled=no list=BOGON
add address=198.18.0.0/15 disabled=no list=BOGON
add address=198.51.100.0/24 disabled=no list=BOGON
add address=203.0.113.0/24 disabled=no list=BOGON
add address=224.0.0.0/4 disabled=no list=BOGON
add address=240.0.0.0/4 disabled=no list=BOGON
```

Должно получиться вот так:



И запрещаем с этих подсетей соединения на WAN-порт маршрутизатора:

```
/ip firewall filter
add action=drop chain=input in-interface=WAN src-address-list=BOGON
```

Разрешаем все уже установленные подключения (connection state=established):

```
/ip firewall filter
add chain=input connection-state=established
```

Разрешаем все зависимые подключения (connection state=related):

```
/ip firewall filter
add chain=input connection-state=related
```

Разрешаем ICMP:

```
/ip firewall filter
add chain=input protocol=icmp
```

Разрешаем новые соединения по портам 65522 и 8291 с любого интерфейса:

```
/ip firewall filter
add chain=input connection-state=new dst-port=8291,65522 protocol=tcp
```

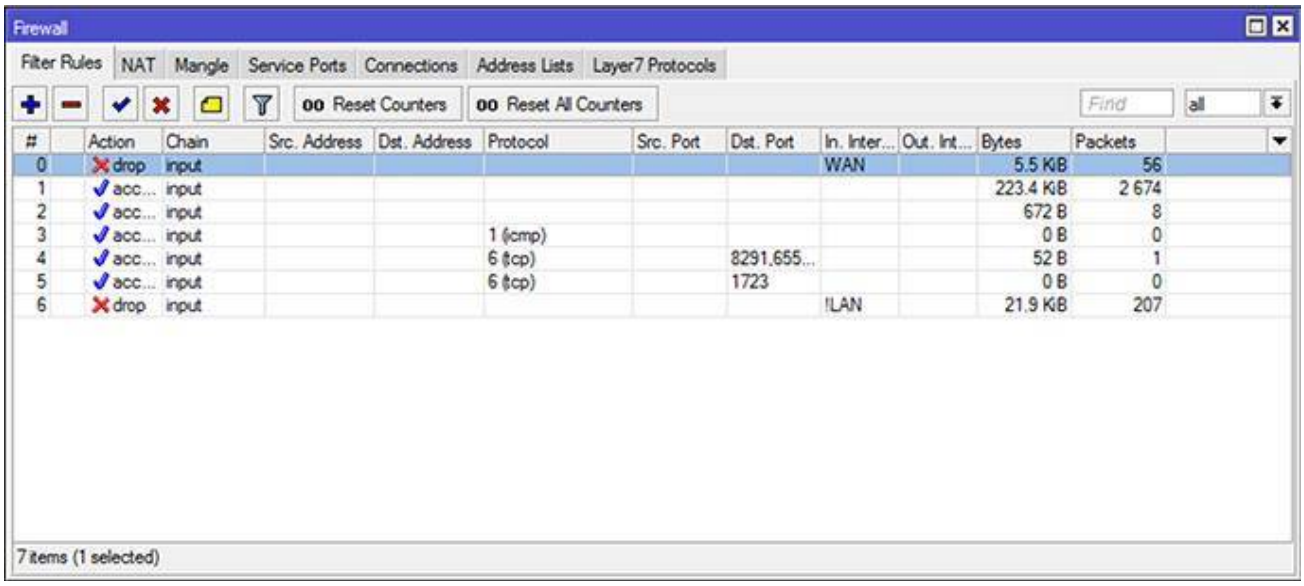
Разрешаем новые соединения по порту 1723 (PPTP) любого интерфейса:

```
/ip firewall filter
add chain=input dst-port=1723 protocol=tcp
```

И блокируем все новые соединения со всех интерфейсов, кроме LAN:

```
/ip firewall filter
add action=drop chain=input connection-state=new in-interface=!LAN
```

Должно получиться следующее:

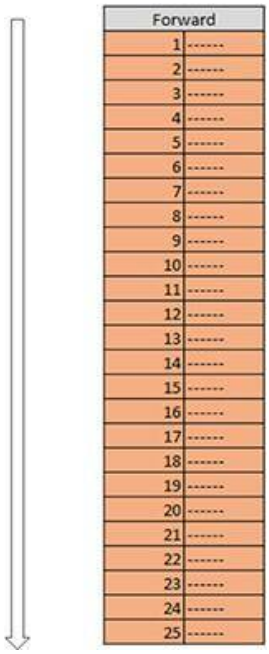


На этом базовая настройка безопасности маршрутизатора завершена. В следующей части мы рассмотрим защиту локальной сети, демилитаризованной зоны и создание собственных цепочек фильтрации трафика.

Часть 3. ИСПОЛЬЗОВАНИЕ СОБСТВЕННЫХ ЦЕПОЧЕК ОБРАБОТКИ ТРАФИКА (CUSTOM CHAIN)

https://spw.ru/educate/articles/nastrojka-filtracii-trafika-na-mikrotik-chast-3/

В прошлых частях статьи мы ознакомились с базовыми настройками файрволла, из которых мы, в том числе, узнали, что межсетевой экран маршрутизатора последовательно проверяет пакет на соответствие правилам фильтрации сверху-вниз. Проверка прекращается в тот момент, когда пакет будет либо пропущен на следующий этап обработки трафика (Action=Accept), либо прохождение пакета будет запрещено (Action=reject,drop,tarpit). Таким образом, если мы представим себе файрволл, состоящий из 25 правил, то выглядеть это будет следующим образом:

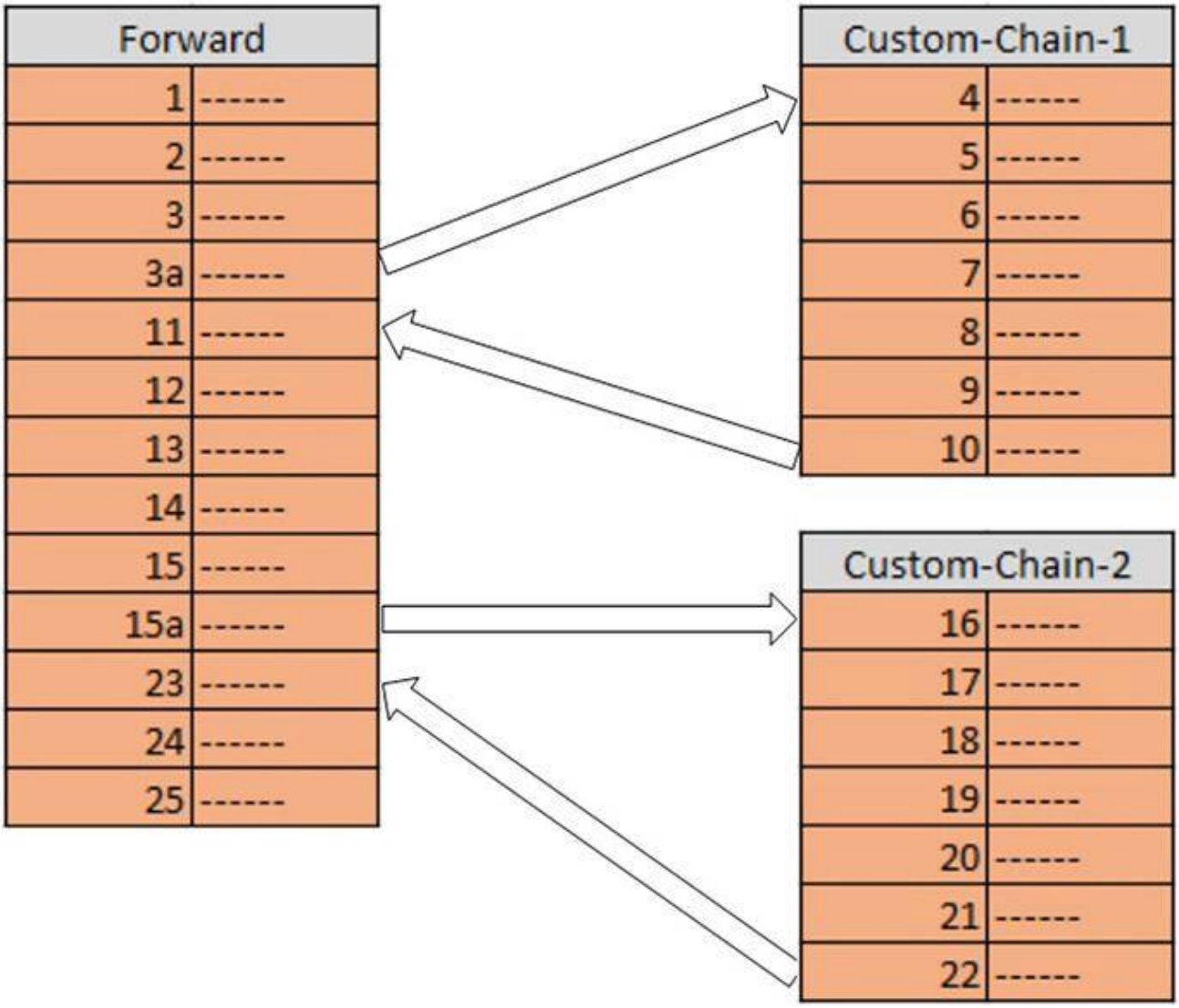


При этом, если пакет соответствует только правилу номер 25, он все равно будет проверен на соответствие правилам с номера 1 по номер 24, на что будут потрачены ресурсы маршрутизатора.

Ситуация осложняется тем, что есть «очень дорогие», с точки зрения процессорного времени, правила фильтрации. В этих случаях приходит на выручку возможность писать собственные цепочки (Chain) обработки правила.

Если вы когда-либо занимались программированием, то собственная цепочка очень похожа на процедуру, вызов которой осуществляется указанием в поле Action команды Jump с именем цепочки, возврат же происходит на следующее правило за вызовом цепочки, по окончании обработки трафика в цепочке, либо если в каком-либо правиле собственной цепочки было использовано Action=Return, что прервало дальнейшую обработку цепочки.

ГРАФИЧЕСКИ ЭТО МОЖНО ОТОБРАЗИТЬ ТАК



При этом надо обратить внимание, что пакет попадет в цепочку Custom-Chain-1, только если он будет соответствовать условиям, обозначенным в правиле 3a, а в цепочку Custom-Chain-2, только если он соответствует правилу 15a.

В результате очевидно, что среднее количество правил, на которые проверяется пакет, уменьшается, что увеличивает производительность маршрутизатора.

Так же хочется обратить внимание что одну и ту же цепочку можно вызывать из разных цепочек (в том числе и своих) фильтрации трафика. Например, вы можете написать цепочку с защитой ssh-сервера от подбора пароля и обращаться к ней как из цепочки input (подбор пароля на маршрутизатор), так и из цепочки forward, защищая один или несколько внутренних серверов.

Для примера обсудим, как производится настройка роутера Микротик, имеющего несколько WAN-интерфейсов. В этом случает у вас возникает необходимость либо написания достаточно большого количества одинаковых правил для каждого WAN-интерфейса, либо создания собственной цепочки обработки трафика, и вызов ее при попадании пакета извне на любой из WAN-интерфейсов.

СОЗДАЕМ ДЕМИЛИТАРИЗОВАННУЮ ЗОНУ (DMZ)

Теперь, после небольшого теоретического отступления про собственные цепочки, которые мы с вами будем использовать, мы переходим к обещанному созданию DMZ. Исходя из второй части статьи, у нас есть 3 интерфейса:

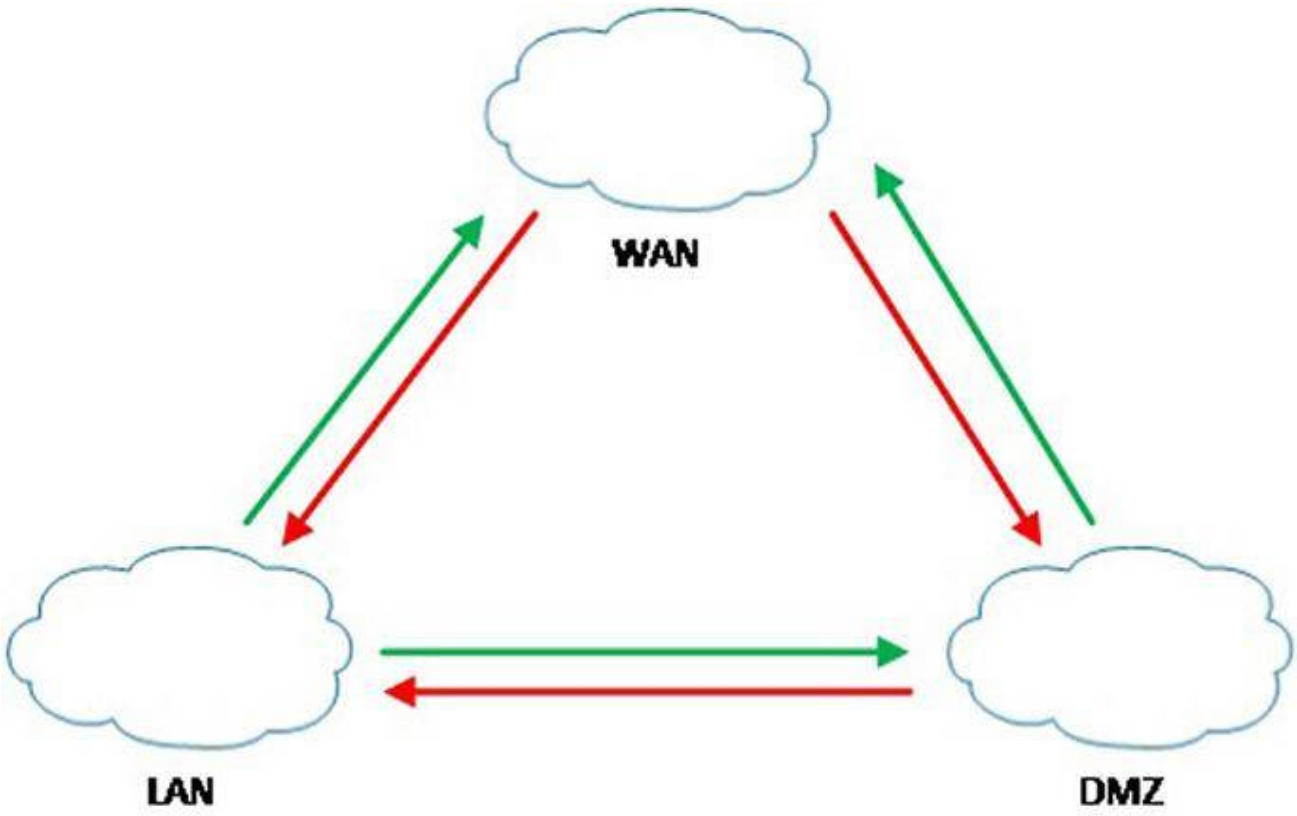
- WAN (Выход в интернет);
- DMZ (Демилитаризованная зона);
- LAN (Локальная сеть).

Р
азделим их по уровню доверия.

LAN – самая доверенная сеть. Из нее можно ходить как в WAN, так и в DMZ без ограничений. Чтобы из других сетей попасть в LAN, требуется отдельное правило на файрволле.

DMZ – сеть с промежуточным уровнем доверия. Из нее можно ходить в WAN, однако в LAN без специального разрешения доступ закрыт.

WAN – самая небезопасная сеть. С нее по умолчанию закрыт доступ как в LAN, так и в DMZ.
Графически это можно представить так:



Так же предположим, что в DMZ у нас есть www-сервер с адресом 10.10.10.100, который должен быть доступен из WAN (Вопросы настройки NAT в этой статье не рассматриваются). Так же из DMZ разрешено обращение на порт 22(ssh) хоста 192.168.88.200 расположенного в LAN.

НАСТРОЙКА МАРШРУТИЗАТОРА

Так как в этой части статьи мы работаем с трафиком, идущим через маршрутизатор, основной цепочкой (Chain) для такого трафика является forward.

Сначала разрешим все соединения с состояниями соединения равными established и related, и запретим с состоянием соединения равным invalid, независимо от интерфейсов. Так как мы уже знаем, что фильтровать трафик логично только на новых соединениях (connection-state=new).

Кроме того, наибольшее количество пакетов как раз относятся к уже установленным соединениям, и указание этого правила в начале списка несколько увеличит производительность.

```
ip firewall filter
add action=accept chain=forward connection-state=established disabled=no
add action=accept chain=forward connection-state=related disabled=no
add action=drop chain=forward connection-state=invalid disabled=no
```

Теперь создадим необходимые нам цепочки правил.

LAN-WAN

В этой цепочке мы разрешаем любой трафик от LAN интерфейса к WAN-интерфейсу. При необходимости можно добавить запрещающие правила. *(Обратите внимание, что мы не указываем интерфейсы. Они будут указаны в цепочке forward, при переходе на свои цепочки)*.

```
/ip firewall filter
add action=accept chain=LAN-WAN disabled=no
```

WAN-LAN

А здесь мы, наоборот, запрещаем весь трафик идущий из интерфейса WAN в локальную сеть. Ранее установленные соединения обрабатывается правилом, их разрешающим, заданным ранее.

```
/ip firewall filter
add action=drop chain=WAN-LAN disabled=no
```

LAN-DMZ

```
/ip firewall filter
add action=accept chain=LAN-DMZ disabled=no
```

DMZ-LAN

Здесь нужно пропустить 22 порт на хост 192.168.88.200

```
/ip firewall filter
add action=accept chain=DMZ-LAN disabled=no dst-address=192.168.88.200 dst-port=22 protocol=tcp
add action=drop chain=DMZ-LAN disabled=no
```

DMZ-WAN

```
/ip firewall filter
add action=accept chain=DMZ-WAN disabled=no
```

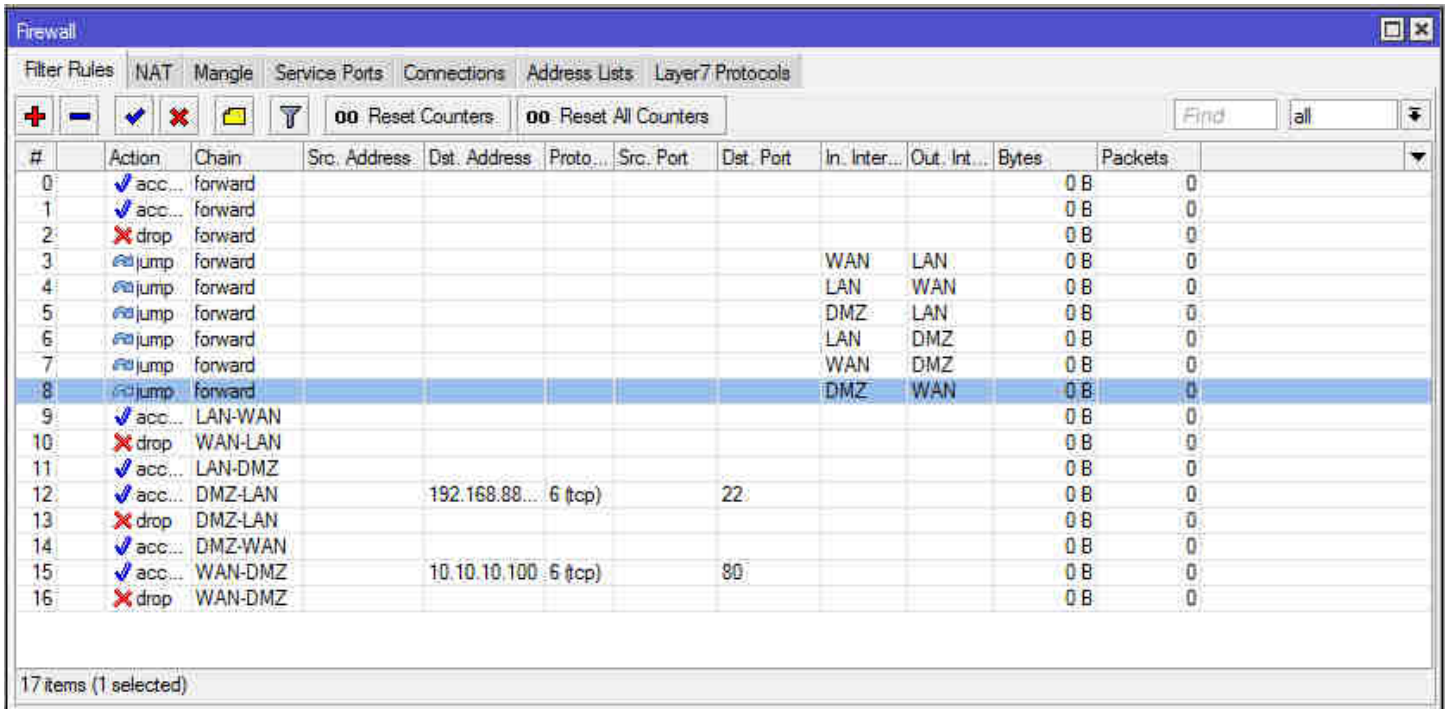
WAN-DMZ
Здесь нам нужно разрешить обращение на хост 10.10.10.100 на порт 80 (www).

```
/ip firewall filter
add action=accept chain=WAN-DMZ disabled=no dst-address=10.10.10.100 dst-port=80 protocol=tcp
add action=drop chain=WAN-DMZ disabled=no
```

На этом создание собственных цепочек закончено. Осталось к ним обратиться.
Создаем группу правил в основной цепочке forward. В ней мы, в зависимости от интерфейсов, будем отправлять пакет в подходящую цепочку:

```
/ip firewall filter
add action=jump chain=forward disabled=no in-interface=WAN jump-target=WAN-LAN out-interface=LAN
add action=jump chain=forward disabled=no in-interface=LAN jump-target=LAN-WAN out-interface=WAN
add action=jump chain=forward disabled=no in-interface=DMZ jump-target=DMZ-LAN out-interface=LAN
add action=jump chain=forward disabled=no in-interface=LAN jump-target=LAN-DMZ out-interface=DMZ
add action=jump chain=forward disabled=no in-interface=WAN jump-target=WAN-DMZ out-interface=DMZ
add action=jump chain=forward disabled=no in-interface=DMZ jump-target=DMZ-WAN out-interface=WAN
```

Для удобства, я поднял эти правила в файрволле повыше, сразу за первыми созданными правилами.
Должно получиться следующее:



#	Action	Chain	Src. Address	Dst. Address	Proto...	Src. Port	Dst. Port	In. Inter...	Out. Int...	Bytes	Packets
0	acc...	forward								0 B	0
1	acc...	forward								0 B	0
2	drop	forward								0 B	0
3	jump	forward						WAN	LAN	0 B	0
4	jump	forward						LAN	WAN	0 B	0
5	jump	forward						DMZ	LAN	0 B	0
6	jump	forward						LAN	DMZ	0 B	0
7	jump	forward						WAN	DMZ	0 B	0
8	jump	forward						DMZ	WAN	0 B	0
9	acc...	LAN-WAN								0 B	0
10	drop	WAN-LAN								0 B	0
11	acc...	LAN-DMZ								0 B	0
12	acc...	DMZ-LAN		192.168.88...	6 (tcp)		22			0 B	0
13	drop	DMZ-LAN								0 B	0
14	acc...	DMZ-WAN								0 B	0
15	acc...	WAN-DMZ		10.10.10.100	6 (tcp)		80			0 B	0
16	drop	WAN-DMZ								0 B	0

При дальнейшей оптимизации логично понаблюдать за счетчиками пакетов и более часто используемые правила поместить повыше.

Обратите внимание, что фильтрацию трафика теперь логично проводить в соответствующей цепочке, что значительно упрощает конфигурирование, особенно в случае большого количества правил.

Общий код файрволла:

```
/ip firewall filter
add action=accept chain=forward connection-state=established disabled=no
add action=accept chain=forward connection-state=related disabled=no
add action=drop chain=forward connection-state=invalid disabled=no
add action=jump chain=forward disabled=no in-interface=WAN jump-target=WAN-LAN out-interface=LAN
add action=jump chain=forward disabled=no in-interface=LAN jump-target=LAN-WAN out-interface=WAN
add action=jump chain=forward disabled=no in-interface=DMZ jump-target=DMZ-LAN out-interface=LAN
add action=jump chain=forward disabled=no in-interface=LAN jump-target=LAN-DMZ out-interface=DMZ
add action=jump chain=forward disabled=no in-interface=WAN jump-target=WAN-DMZ out-interface=DMZ
add action=jump chain=forward disabled=no in-interface=DMZ jump-target=DMZ-WAN out-interface=WAN
add action=accept chain=LAN-WAN disabled=no
add action=drop chain=WAN-LAN disabled=no
add action=accept chain=LAN-DMZ disabled=no
```

```
add action=accept chain=DMZ-LAN disabled=no dst-address=192.168.88.200 dst-port=22 protocol=tcp
add action=drop chain=DMZ-LAN disabled=no
add action=accept chain=DMZ-WAN disabled=no
add action=accept chain=WAN-DMZ disabled=no dst-address=10.10.10.100 dst-port=80 protocol=tcp
add action=drop chain=WAN-DMZ disabled=no
```

Таким образом, мы выполнили базовую настройку Firewall MikroTik и научились работать с собственными цепочками фильтрации трафика.

ЧАСТЬ 4. ИСПОЛЬЗОВАНИЕ СПИСКА АДРЕСОВ (ADDRESS LIST)

https://spw.ru/educate/articles/nastrojka-filtracii-trafika-na-mikrotik-chast-4/

ВВЕДЕНИЕ

Использование списка адресов (address list)

В предыдущих частях статьи мы с Вами получили вполне работоспособную заготовку файервола, которая позволяет строить достаточно сложные правила фильтрации трафика. Однако все эти правила являются статическими. Что не всегда удобно.

Одним из вариантов создания динамических правил в файерволе является использование именованных списков адресов (address lists)

В именованный список, можно добавлять, как адреса, так и подсети. Вполне корректны следующие команды:

```
/ip firewall address-list
add address=192.168.0.10 list=test
add address=192.168.1.0/24 list=test
add address=192.168.2.1-192.168.2.15 list=test
```

Здесь мы добавили в Address-list с именем test сразу три разных записи

Одиночный хост 192.168.0.10 Подсеть 192.168.1.0/24 Диапазон адресов 192.168.2.1-192.168.2.15
Теперь мы все это одновременно можем обработать в файерволе в одном правиле. Например:

```
/ip firewall filter
add chain=forward src-address-list=test
```

разрешит прохождение трафика через маршрутизатор, если адрес источника находится в именованном списке test.

Однако у именованного списка адресов есть еще один замечательный параметр, который называется timeout. Он обозначает,что через какое время, после попадания в список, адрес будет оттуда автоматически удален. И если раньше, когда у Вас возникала необходимость временно разрешить или запретить какому-нибудь хосту прохождение трафика, Вы создавали правило в файерволе, а потом старались не забыть что его надо удалить, то теперь все проще.

Пример:

Нам иногда надо временно запретить прохождение трафика с какого либо из компьютеров сети.
Для начала создадим правило:

```
/ip firewall filter
add action=drop chain=forward src-address-list=deny-forward
```

А теперь мы хотим отключить хост 192.168.100.100 от Интернет на 1 час.
Набираем команду:

```
/ip firewall address-list
```

```
add list=deny-forward address=192.168.100.100 timeout=1h
```

И в общем-то и все. По истечении часа, эта запись автоматически удалится и хост получит доступ в Интернет. Удобно? Безусловно да. Но самая сильная функциональность именованных списков заключается в том, что их можно добавлять из других правил файервола, при помощи Action add dst to address list и add src to address list Это позволяет делать динамические правила фильтрации трафика.

Например мы хотим защитить встроенный ssh сервер микротик от атаки на подбор пароля (brute force)

Создадим следующие правила:

```
/ip firewall filter
add action=drop chain=input dst-port=22 protocol=tcp src-address-list=ssh_blacklist
add action=add-src-to-address-list address-list=ssh_blacklist \
address-list-timeout=1w3d chain=input connection-state=new dst-port=22 \
protocol=tcp src-address-list=ssh_stage3
add action=add-src-to-address-list address-list=ssh_stage3 \
address-list-timeout=1m chain=input connection-state=new dst-port=22 \
protocol=tcp src-address-list=ssh_stage2
add action=add-src-to-address-list address-list=ssh_stage2 \
address-list-timeout=1m chain=input connection-state=new dst-port=22 \
protocol=tcp src-address-list=ssh_stage1
add action=add-src-to-address-list address-list=ssh_stage1 \
address-list-timeout=1m chain=input connection-state=new dst-port=22 \
protocol=tcp
```

Разберемся, как это работает.

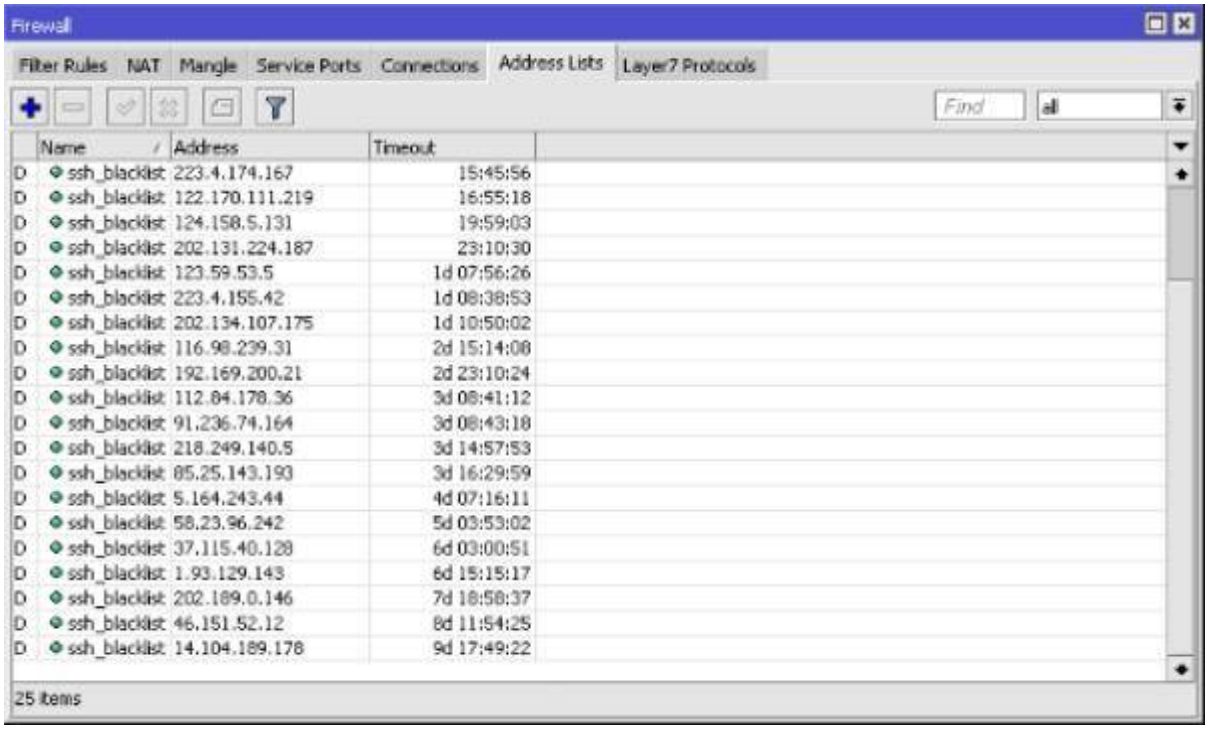
Итак, если происходит подключение к ssh-серверу, срабатывает последнее правило из списка, в результате чего адрес отправителя пакета попадает в список адресов с именем ssh_stage1 на одну минуту.Если подключение к серверу было осуществлено успешно, соединение перейдет в состояние established и эти правила, ориентированные на connection-state=new не будут срабатывать.

Если же, подключающийся не знает пароля, сервер его отключит. И при попытке подключения в течении одной минуты после первой попытки, сработает второе снизу правило. В результате чего адрес отправителя пакета попадет на одну минуту в список адресов с именем ssh_stage2.

Третья попытка подбора пароля в течении минуты приведет к попаданию атакующего на одну минуту в список с именем ssh_stage3. Если атакующий попытается подключиться в течении одной минуты после попадания его адреса в список ssh_stage3, он попадет в список ssh_blacklist на длительное время (в примере на 10 дней).

Первое правило этого примера блокирует прохождение пакетов на ssh-сервер, от адресов входящих в последний список.

Результат работы примера:



Понятно, что если Вы замените в примере цепочку input на forward, а dst-port замените на 3389, Вы защитите таким образом опубликованный терминальный сервер. Кстати, данный пример легко модифицируется для реализации такой технологии как Port Knocking.

Суть данной технологии заключается в том, что для того, чтобы адрес источника пакета получил доступ к защищаемому ресурсу, он должен сначала «постучать» в заранее определенной последовательности в заранее определенные порты по протоколам TCP и/или UDP.

Попробуем теперь модифицировать пример. Теперь нам надо добиться того, чтобы удаленный компьютер получил доступ к ssh серверу, «простучав» порты в последовательности tcp:2000, udp:1000, tcp:5000
Между «стуками» не должно пройти более 5 секунд. Доступ для попытки подключения после успешного простукивания портов предоставляется на 1 минуту.

Решение:

```
/ip firewall filter
add chain=input connection-state=new dst-port=22 protocol=tcp src-address-list=port_knock-OK
add action=add-src-to-address-list address-list=port_knock-OK address-list-timeout=1m \
chain=input dst-port=5000 protocol=tcp src-address-list=port_knock2
add action=add-src-to-address-list address-list=port_knock2 address-list-timeout=5s \
chain=input dst-port=1000 protocol=udp src-address-list=port_knock1
add action=add-src-to-address-list address-list=port_knock1 address-list-timeout=5s \
chain=input dst-port=2000 protocol=tcp add action=drop chain=input connection-state=new dst-
port=22 protocol=tcp
```

Таким образом, работа с именованными списками адресов значительно расширяет возможности настройки фильтрации пакетов. А возможность заполнять эти списки из правил файрвола, а так же возможность указания времени на которое адрес попадает в список, позволяет создавать сложные интеллектуальные правила фильтрации трафика.