Защита RDP от перебора паролей при помощи оборудования Mikrotik

Записки IT специалиста

Технический блог специалистов ООО"Интерфейс"

- Главная
- Защита RDP от перебора паролей при помощи оборудования Mikrotik

Атака с полным перебором паролей (брутфорс) - одна из наиболее часто встречающихся угроз в современном интернете. Она базируется не на уязвимостях ПО, а на нарушении политики паролей, что иногда оказывается гораздо более продуктивным. Пользователи не любят сложных паролей и даже когда есть явные требования по сложности стремятся использовать более простые комбинации, либо словарные слова. Многие также используют одну пару логин - пароль для всех учетных записей и при компрометации одной из них эти данные могут попасть в руки злоумышленников. Одним из наиболее часто атакуемых сервисов является RDP и сегодня мы посмотрим, как можно его защитить.

В среде администраторов бытует мнение, что выставлять RDP без VPN "наружу" нельзя. На сегодняшний день это не так, современные реализации протокола используют даже по умолчанию SSL-защиту и проверку подлинности на уровне сети (NLA), если использовать поддерживаемые версии ОС и своевременно устанавливать обновления, то RDP будет достаточно надежным для использования без дополнительной защиты.

Но у него, как и у многих иных сервисов, есть слабое место - пароли, а усугубляется ситуация отсутствием штатных средств защиты от их перебора. Существуют решения, которые устанавливаются прямо на RDP-сервер и добавляют необходимую функциональность, но мы пойдем другим путем и будем использовать более подходящее для этого средство - межсетевой экран.

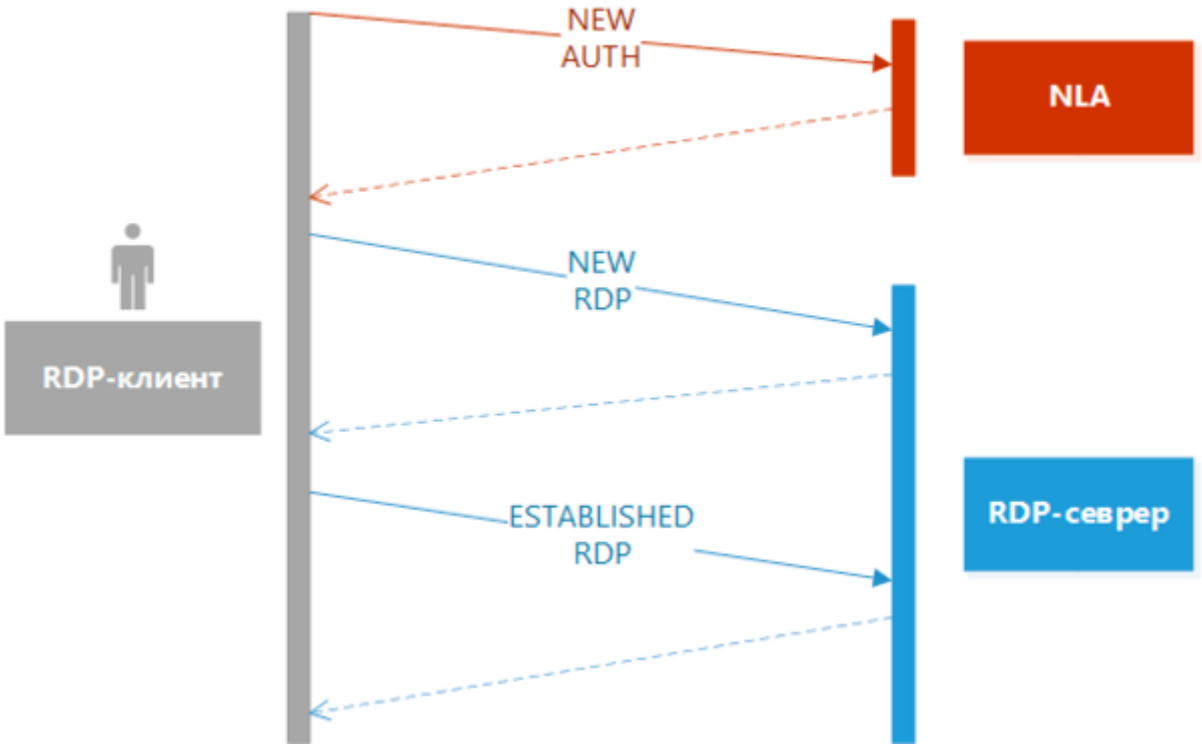
На первый взгляд задача труднорешаемая. Каким образом роутер сможет определить ведется перебор паролей, либо просто подключаются и работают пользователи, ведь через него проходит только транзитный трафик к RDP-серверу, при этом сама сессия шифруется. Но не будем делать скоропалительных выводов. Существует специальный механизм **Connection Tracking**, который определяет состояние соединения и принадлежащих ему пакетов, всего таких состояний четыре:

- **NEW** - новый пакет, не принадлежащий ни одному соединению
- **ESTABLISHED** - пакет, принадлежащий уже установленному соединению
- **RELATED** - пакет нового соединения, порождённого уже существующим соединением, многие протоколы, например, FTP открывают дополнительные соединения для передачи данных
- **INVALID** - все что не относится к первым трем состояниям, это пакет, не принадлежащий ни одному соединению и не являющийся первым пакетом нового соединения.

Исходя из вышесказанного можно прийти к следующему выводу: у работающего RDP-клиента состояние пакетов будет ESTABLISHED, а NEW будет появляться только в начале соединения, если же мы постоянно получаем NEW от клиента - то с большой долей вероятности он занимается перебором паролей. Давайте рассмотрим, как происходит RDP-соединение, схема предельно упрощенная, на уровне минимально необходимом для понимания происходящих процессов.

Современные решения предусматривают **проверку подлинности на уровне сети** (NLA) и клиент сначала должен пройти процесс аутентификации, прежде чем он сможет подключиться к компьютеру, RDP-сессия при этом **не создается**. Пакет с запросом аутентификации будет иметь состояние **NEW**, при неудачной аутентификации каждый новый запрос будет создавать еще один пакет с состоянием **NEW**.

При успешном вводе учетных данных клиент создаст новое RDP-соединение, первый пакет этого соединения также будет иметь состояние **NEW**, а все последующие **ESTABLISHED**.



Таким образом успешное RDP соединение предусматривает получение двух пакетов с состоянием NEW, каждая попытка аутентификации еще одного. Это дает возможность достаточно легко отличать легальных пользователей от ботов и злоумышленников. Фактически если в течении короткого промежутка времени мы получили более 2 пакетов в состоянии NEW - то это однозначно свидетельствует, что на той стороне ошиблись при вводе учетных данных.

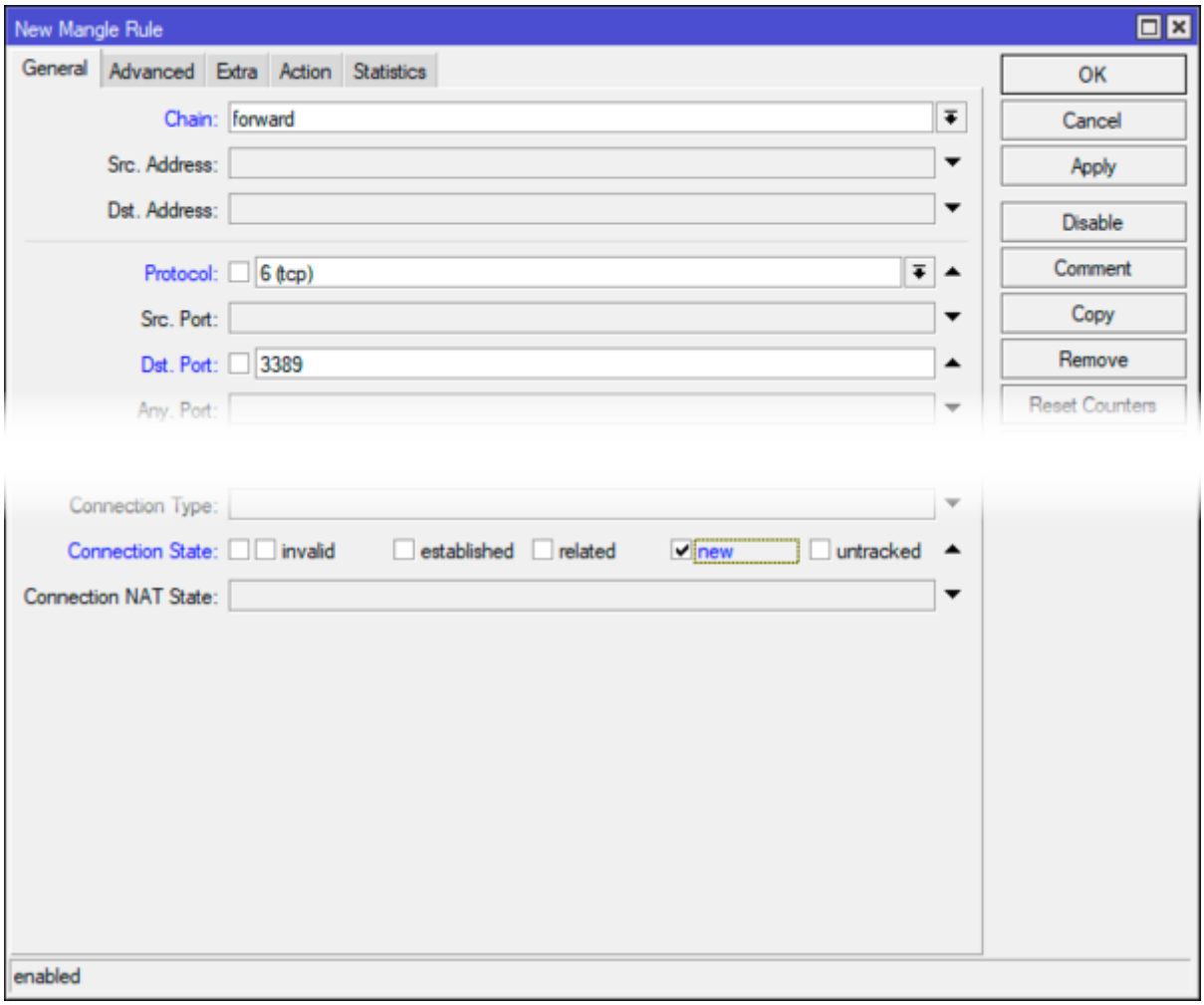
Так как человеку свойственно ошибаться, то оставим легальным пользователям право на ошибку: три неверных попытки и одна правильная - итого пять пакетов NEW в течении короткого времени, злоумышленник за это время успеет пять раз попробовать ввести учетные данные. Какой промежуток времени взять? Одной-двух минут в целом будет достаточно, даже при ручных попытках ввода. Но не следует устанавливать слишком большой интервал, так как пользователь может случайно отключиться - подключиться снова и попасть под блокировку. Брутфорсом же занимаются преимущественно боты и даже минуты им с лихвой хватит чтобы выйти за пределы лимитов.

Теперь, когда необходимый теоретический минимум получен, можно переходить к практической реализации и это будет не слепое копирование мануалов из интернета, а вполне осмысленный набор действий. Прежде всего нам надо выявить и занести в отдельный список тех клиентов, которые пытаются прислать нам более пяти новых пакетов на порт RDP-сервера за определенный интервал времени.

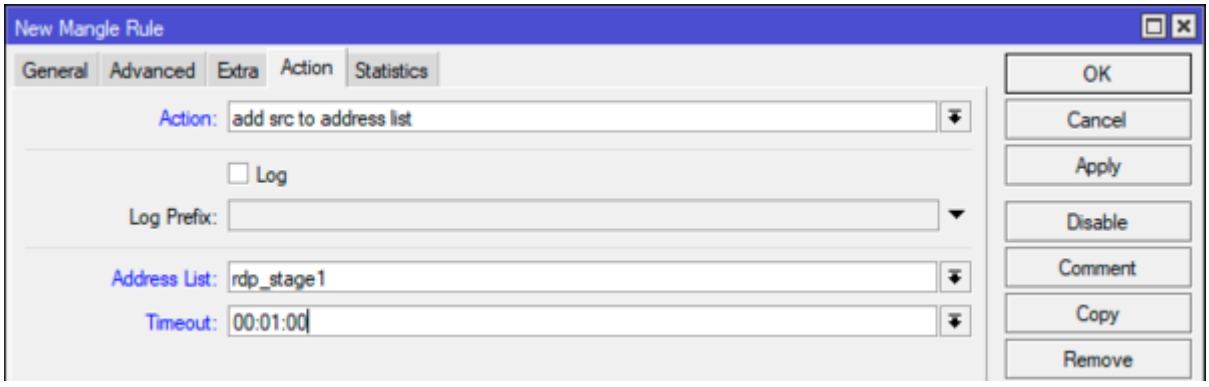
Для этой цели мы будем использовать добавление адреса-источника в **Address List** и таблицу **Mangle**. Почему именно Mangle, ведь действие **add src to address list** доступно и в Filter? Один из основных принципов построения брандмауэра - это читабельность правил конфигурации. В Filter мы ожидаем увидеть действия, производимые с пакетами на основании тех или иных условий. А вот Mangle

предназначена для маркировки пакетов и соединений, и, хотя, добавление в списки не совсем маркировка, логически более верным будет разместить эти правила здесь.

Итак, переходим в **IP - Firewall - Mangle** и создаем следующее новое правило: **Chain - forward, Protocol - tcp, Dst. Port - 3389, Connection State - new**.



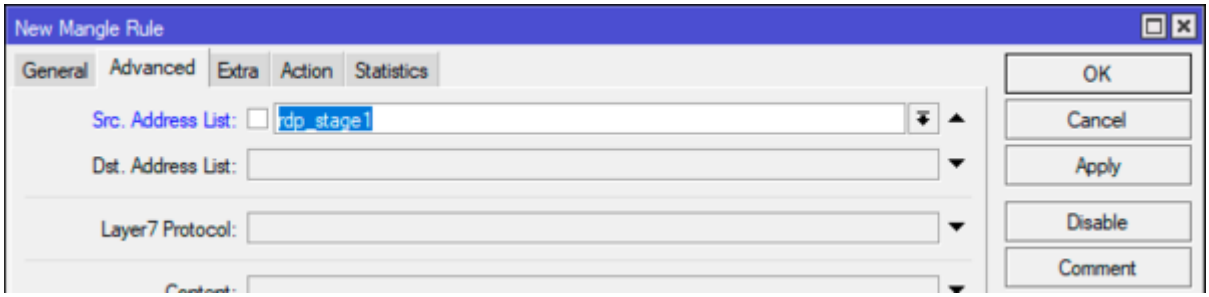
На закладке **Action** добавляем действие **add src to address list** и укажем имя списка **rdp_stage1** и время действия записи - **1 минута**.



В терминале можете выполнить:

```
/ip firewall mangle
add action=add-src-to-address-list address-list=rdp_stage1 address-list-timeout=1m
chain=forward connection-state=new dst-port=3389 protocol=tcp
```

Что мы сейчас сделали? Для каждого транзитного пакета на порт 3389 с состоянием NEW мы помещаем его адрес-источник в список **rdp_stage1** на 1 минуту. Получив еще один такой пакет, мы должны проверить, нет ли его уже в указанном списке, а если есть, то поместить в список **rdp_stage2**, для этого создаем еще одно аналогичное правило, но с дополнительным условием, на закладке **Advanced** указываем **Src. Address List - rdp_stage1**, а на закладке Action в качестве списка указываем **rdp_stage2**.



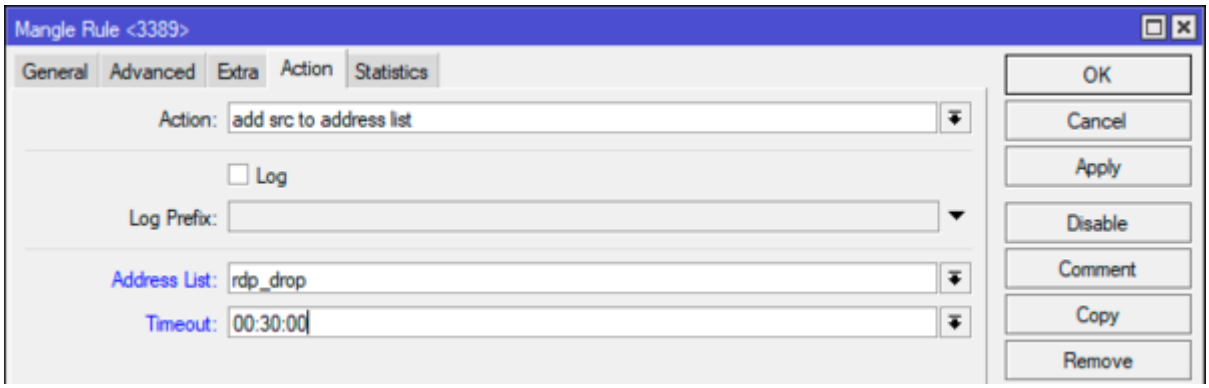
Это же действие в терминале:

```
/ip firewall mangle
add action=add-src-to-address-list address-list=rdp_stage2 address-list-timeout=1m
chain=forward connection-state=new dst-port=3389 protocol=tcp src-address-
list=rdp_stage1
```

Затем создаем еще одно правило, где указываем **Src. Address List - rdp_stage2**, а список на закладке **Action - rdp_stage3**, и еще два, пока не заполним список **rdp_stage5**.

```
/ip firewall mangle
add action=add-src-to-address-list address-list=rdp_stage3 address-list-timeout=1m
chain=forward connection-state=new dst-port=3389 protocol=tcp src-address-
list=rdp_stage2
add action=add-src-to-address-list address-list=rdp_stage4 address-list-timeout=1m
chain=forward connection-state=new dst-port=3389 protocol=tcp src-address-
list=rdp_stage3
add action=add-src-to-address-list address-list=rdp_stage5 address-list-timeout=1m
chain=forward connection-state=new dst-port=3389 protocol=tcp src-address-
list=rdp_stage4
```

И, наконец, последнее правило, с условием **Src. Address List - rdp_stage5** и действием **Action - add src to address list, Address List - rdp_drop** и таймаутом, скажем, **30 минут**.

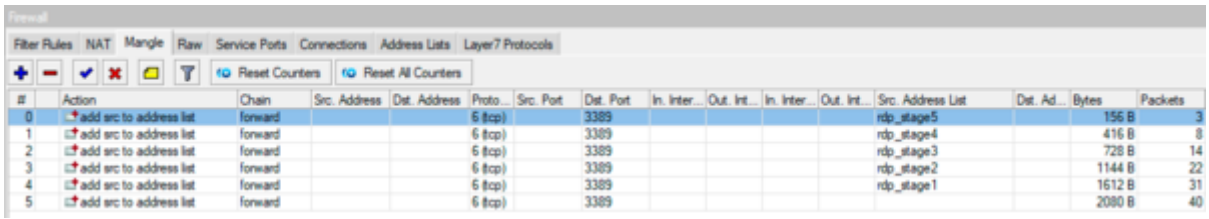


Или в терминале:

```
/ip firewall mangle
add action=add-src-to-address-list address-list=rdp_drop address-list-timeout=30m
chain=forward connection-state=new dst-port=3389 protocol=tcp src-address-
list=rdp_stage5
```

Как это все работает? Адрес-источник первого NEW пакета мы помещаем в список **rdp_stage1** на одну минуту, если с этого адреса снова придет новый пакет, то адрес источник попадет в список **rdp_stage2**, тоже на минуту, если в течении этой минуты снова придет новый пакет - то он будет помещен в **rdp_stage3** и т.д. Шестой пакет с состоянием NEW попадет в список **rdp_drop** на полчаса. Время блокировки вы можете выбирать самостоятельно, в зависимости от текущих условий, но следует избегать больших значений, если пользователь таки попадет под блокировку, а администратор случайно не на связи, то полчаса гораздо лучше, чем сутки.

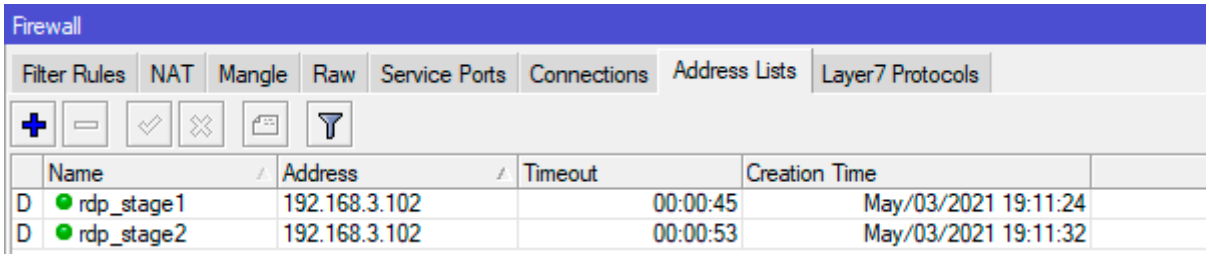
Следующий момент, после того как вы добавили все эти правила, их следует разместить в обратном порядке, от последнего к первому. Почему? Потому что действие **add src to address list** не является терминальным и пакет продолжает свое движение по цепочке. И если правила оставить в том порядке, в котором они был добавлены то получится следующее: первое правило добавит новый пакет в список **rdp_stage1**, второе проверит, нет ли его в этом списке, а оно там есть, и добавит в **rdp_stage2**. Далее вы поняли, адрес источник с первого же пакета добавится последовательно во все списки включая **rdp_drop**. Поэтому проверять списки надо в обратном порядке, от **rdp_stage5** к **rdp_stage1**.



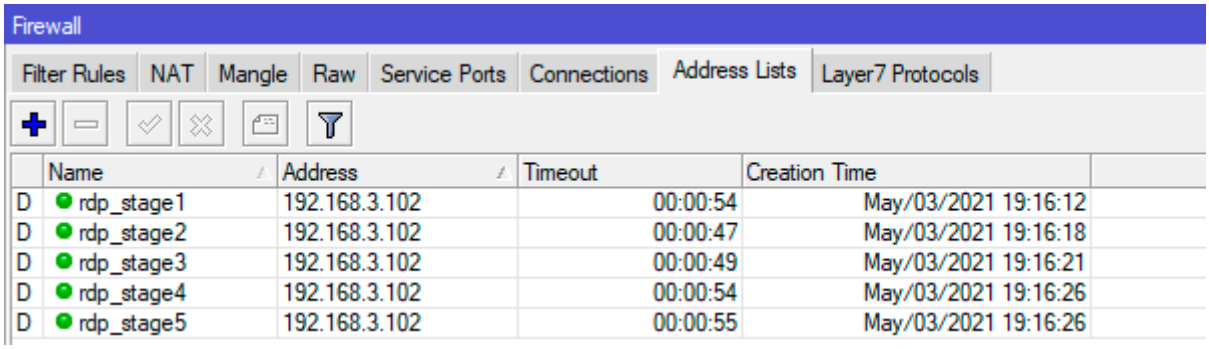
Если вы работаете в терминале, то правильная последовательность действий будет выглядеть так:

```
/ip firewall mangle
add action=add-src-to-address-list address-list=rdp_drop address-list-timeout=30m
chain=forward connection-state=new dst-port=3389 protocol=tcp src-address-
list=rdp_stage5
add action=add-src-to-address-list address-list=rdp_stage5 address-list-timeout=1m
chain=forward connection-state=new dst-port=3389 protocol=tcp src-address-
list=rdp_stage4
add action=add-src-to-address-list address-list=rdp_stage4 address-list-timeout=1m
chain=forward connection-state=new dst-port=3389 protocol=tcp src-address-
list=rdp_stage3
add action=add-src-to-address-list address-list=rdp_stage3 address-list-timeout=1m
chain=forward connection-state=new dst-port=3389 protocol=tcp src-address-
list=rdp_stage2
add action=add-src-to-address-list address-list=rdp_stage2 address-list-timeout=1m
chain=forward connection-state=new dst-port=3389 protocol=tcp src-address-
list=rdp_stage1
add action=add-src-to-address-list address-list=rdp_stage1 address-list-timeout=1m
chain=forward connection-state=new dst-port=3389 protocol=tcp
```

Теперь проверим работу наших правил, выполним подключение к RDP одновременно контролируя **IP - Firewall - Address Lists**. Если все прошло нормально, то мы увидим наш адрес-источник в списках **rdp_stage1** и **rdp_stage2**.

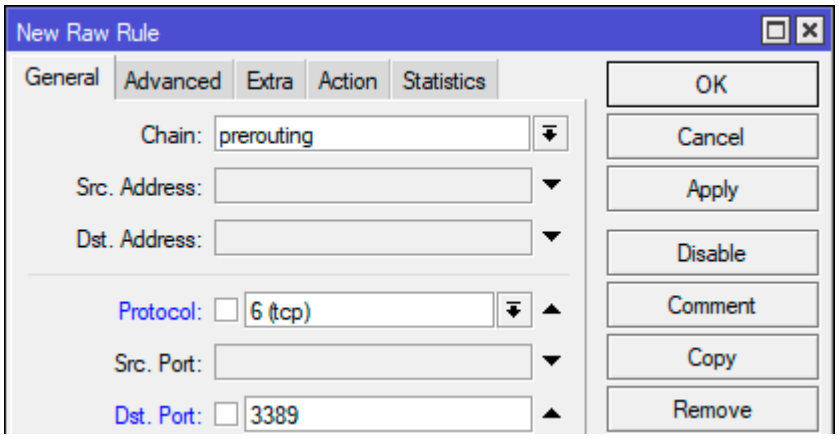


Теперь попробуем три раза ошибиться и на четвертый зайти:

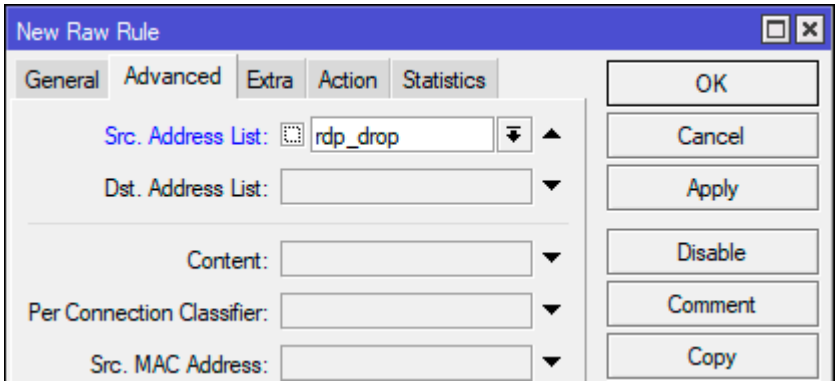


Как видим, практика полностью совпадает с теорией, следовательно, можно переходить к блокировкам, не опасаясь, что пострадают легальные пользователи. Блокировать мы будем в таблице **Raw**, почему? **Raw** содержит "сырые" данные о пакетах, до того, как они будут переданы **Connection Tracking**, а так как определение состояния пакета достаточно ресурсозатратная операция, то отсекая лишний трафик в **Raw** мы снижаем нагрузку на процессор роутера, что важно в случае достаточно большого списка.

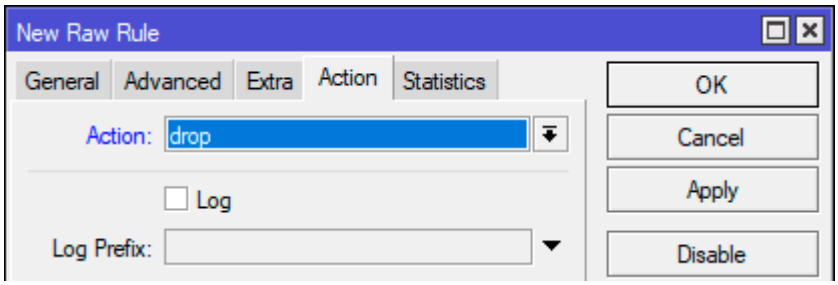
Переходим в **IP - Firewall - Raw** и создаем новое правило: **Chain - prerouting, Protocol -tcp, Port - 3389**.



На закладе **Advanced** указываем **Src. Address List - rdp_drop**:



И на закладке **Action** - действие **drop**.

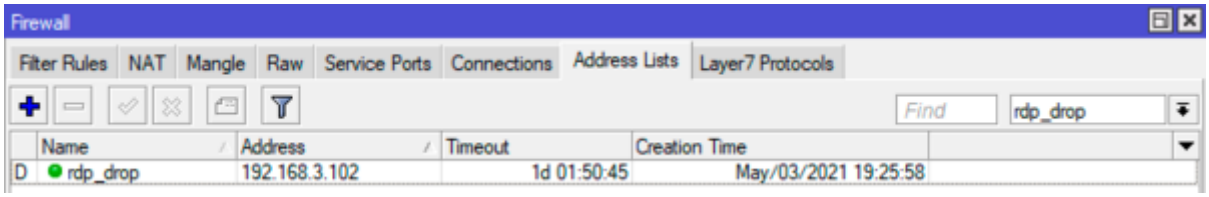


Все тоже самое быстро делается в терминале следующей командой:

```
/ip firewall raw
add action=drop chain=prerouting dst-port=3389 protocol=tcp src-address-
list=rdp_drop
```

Заметьте, мы заблокировали вероятному злоумышленнику только доступ к RDP, хотя, убрав условия о протоколе и порте, мы можем полностью блокировать этот адрес. Но мы не советуем этого делать, так как случаи бывают разные и даже сам администратор может несколько раз неправильно ввести пароль. Если мы блокировали только RDP, то он сможет подключиться к роутеру и разблокировать себя, если мы блокируем источник полностью, то связь с корпоративной сетью будет полностью потеряна.

На этом можно считать защиту от брутфорса готовой, на первых порах мы рекомендуем регулярно контролировать содержимое списка rdp_drop на предмет попадания туда легальных клиентов, после чего либо корректировать условия, либо разъяснять пользователям новые правила работы.



Кроме того, данный метод может быть применен для любого сервиса, не только RDP, все что вам потребуется - это установить, можно даже эмпирическим путем, схему соединения по нужному вам протоколу и определить, какое количество новых пакетов является допустимым, а какое говорит о попытке подбора паролей.

Онлайн-курс по MikroTik

Научиться настраивать MikroTik с нуля или систематизировать уже имеющиеся знания можно на углубленном курсе по администрированию MikroTik. Автор курса, сертифицированный тренер MikroTik Дмитрий Скоромнов, лично проверяет лабораторные работы и контролирует прогресс каждого своего студента. В три раза больше информации, чем в вендорской программе MTCNA, более 20 часов практики и доступ навсегда.