Санкт-Петербург, ул. Седова, 80, телефон (812) 449-0770, факс (812) 449-0771, E-mail: info@bilim.com

Николай Малых

nmalykh@bilim.com

Создание динамических правил фильтрации пакетов

IPSET[1]

ipset[2] представляет собой программу для работы с наборами адресов IP (IP set) в ядре Linux. Программа позволяет создавать, изменять и проверять наборы адресов IP, с которыми работает ядро Linux (включая netfilter/iptables). IP set может включать в себя адреса IP, номера портов TCP и UDP, а также дополнительную информацию.

Любой элемент набора может быть связан (bind) с любым другим набором. Набор может иметь принятую по умолчанию связку, которая будет применима ко всем элементам данного набора, для которых явно не определена другая связка. Связываемую с другим набором запись включать в этот набор не требуется.

Связки указывающие на наборы, а также на соответствия (match) и операции (target) iptables, создают ссылки, которые защищают данный набор в ядре. Защищенный набор не может быть удален, пока на него имеется хотя бы одна ссылка.

Синтаксис

```
ipset -N set type [options]
ipset -[XFLSHh] [set] [options]
ipset -[EW] from-set to-set
ipset -[ADU] set entry
ipset -B set entry -b binding
ipset -T set entry [-b binding]
ipset -R
```

Опции

Опции **ipset** можно разделить на две группы – команды и параметры (прочие опции).

Команды

Командные опции или просто команды задают выполняемое программой действие. В командной строке может использоваться только одна опция этого типа (т. е., нельзя задать одновременное выполнение нескольких операций), если в описании команды явно не указано иное. Длинные варианты командных опций можно вводить в сокращенной форме – главное, чтобы программа ipset могла идентифицировать нужную команду (отличить ее от прочих команд и параметров).

`-N, --create setname type options`

Создает набор заданного типа (**type**) с именем, указанным параметром **setname**. В зависимости от типа создаваемого набора команда должна включать те или иные опции.

`-X, --destroy [setname]`

Эта команда удаляет набор, указанный параметром **setname**, или все наборы (если имя набора не указано). Перед удалением набора удаляются все связки элементов этого набора и связка, принятая для набора по умолчанию.

Если на удаляемый набор имеются ссылки, команда просто ничего не делает и набор сохраняется.

`-F, --flush [setname]`

Удаляется все элементы указанного набора или всех наборов (если параметр **setname** не задан или в качестве имени набора указано ключевое слово **:all:**). При удалении элементов связки не изменяются.

`-E, --rename from-setname to-setname`

Эта команда служит для изменения имени указанного параметром **from-setname** набора. Набора с новым именем (**to-setname**) до выполнения команды не должно существовать.

`-W, --swap from-setname to-setname`

Меняет ссылки в ядре Linux, правилах iptables и связках ipset (ссылки на набор **from-setname** заменяются ссылками на набор **to-setname** и наоборот). Оба набора должны существовать до выполнения команды.

`-L, --list [setname]`

Выводит список элементов и связок для указанного набора или всех наборов (если набор не указан или использовано ключевое слово **:all:**). При использовании вместе с опцией **-n**, **--numeric** отключается преобразование адресов и номеров портов в символьные имена. Использование с опцией **-s**, **--sorted** обеспечивает вывод отсортированного списка, если данный набор поддерживает сортировку. Ниже показан пример вывода по команде **ipset -l -n SrcTree** для набора **SrcTree** типа **iptree**, не имеющего связок и созданного со значением тайм-аута 5 секунд. Этот набор включает адреса отправителей пакетов, полученных хостом в течение истекших 5 секунд.

```
Name: SrcTree
Type: iptree
References: 1
Default binding:
Header: timeout: 5
Members:
```

[1] Этот параграф является переводом man ipset с некоторыми дополнениями.

[2] Программу можно загрузить с сайта http://ipset.netfilter.org.

```
172.16.33.35%3
193.111.91.1%3
205.188.5.240%2
212.48.200.33%0
Bindings:
```

-S, --save [setname]
Сохраняет указанный набор или все наборы (если набор не задан или используется ключевое слово **:all:**) на stdout в формате, понятном для опции **–restore**. Вывод может быть направлен в файл стандартными средствами перенаправления.

-R, --restore
Восстанавливает сессию, сохраненную с помощью опции **--save**. Восстанавливаемая сессия может считываться со стандартного устройства ввода stdin.

-A, --add setname IP
Добавляет IP-адрес в указанный набор.

-D, --del setname IP
Удаляет IP-адрес из указанного набора.

-T, --test setname IP
Проверяет наличие адреса IP в указанном наборе. При обнаружении адреса возвращается нулевое значение, а при отсутствии – отличное от нуля.

-T, --test setname IP --binding to-setname
Проверяет относится ли адрес IP к набору, указываемому заданной связкой. При обнаружении адреса в наборе возвращается 0, при отсутствии – ненулевое значение. С помощью ключевого слова **:default:** можно проверить наличие адреса в наборе, указываемом принятой по умолчанию связкой.

-B, --bind setname IP --binding to-setname
Связывает IP-адрес из набора **setname** с набором **to-setname**.

-U, --unbind setname IP
Удаляет связку, относящуюся к IP-адресу из набора **setname**.

-H, --help [settype]
Выводит на экран краткую справку о работе с программой или типе набора (**settype**).

В командах -B, -U и -T может использоваться ключевое слово **:default:** для связывания, проверки или удаления связки, используемой по умолчанию, вместо связки для адреса IP. В команде -U можно использовать ключевое слово **:all:** для удаления связок всех элементов набора.

Параметры
Описанные ниже опции используются в качестве дополнительных параметров команд

-b, --binding setname
Задает связку для команды **-B** и является обязательным параметром данной команды. Вы можете также использовать эту опцию в команде **-T** для тестирования связки.

-s, --sorted
Задает сортировку выводимых данных.

-n, --numeric
Отключает преобразование адресов IP и номеров портов в символьные имена при выводе информации. По умолчанию программа пытается преобразовать адреса и номера портов в имена хостов и служб, что может существенно замедлять работу и повышать нагрузку на серверы DNS.

-q, --quiet
Подавляет вывод сообщений на stdout и stderr. После завершения работы ipset будет возвращать значение кода ошибки.

Типы наборов
Программа ipset поддерживает несколько типов наборов, которые описаны ниже.

ipmap
Тип **ipmap** служит для задания непрерывных блоков адресов IP и использует диапазон памяти, в котором каждый бит представляет один IP-адрес. Этот тип позволяет создавать наборы, содержащие до 65535 адресов (сеть класса B). Тип **ipmap** обеспечивает высокую скорость работы и экономный расход памяти, что весьма удобно для проверки принадлежности адресов к некому диапазону. Использование опции **--netmask** со значением размера маски CIDR[3] (0-32) при создании набора позволяет сохранять наборы адресов и проверять принадлежность IP-адреса к данному набору – если значение, получаемое наложением указанной маски на адрес, включено в набор, значит и данный адрес относится к этому набору.

Опции, используемые при создании наборов типа **ipmap**, включают:

--from from-IP
--to to-IP
задают начальный и конечный адреса диапазона.

--network IP/mask
создают набор **ipmap** для указанного префикса и маски сети.

--netmask CIDR-netmask

[3] См. RFC 1517 – 1519, переводы которых имеются на сайте http://www.protocols.ru.

при использовании необязательного параметра **--netmask** в наборе будут сохраняться префиксы сетей вместо адресов IP; параметр **from-IP** в таких случаях должен указывать префикс сети.

macipmap

Тип **macipmap** служит для задания непрерывных блоков адресов IP вместе со связанными с каждым адресом MAC-адресами и использует диапазон памяти, в котором каждые 8 байтов представляют одну пару адресов IP - MAC. Этот тип позволяет сохранять наборы с числом элементов до 65535 (сеть класса B). При добавлении записи в набор типа **macipmap** нужно указать параметры как IP%MAC. При удалении или проверке записи вторая часть (%MAC) является необязательной.

При создании наборов типа **macipmap** используются следующие опции:

`--from from-IP`
`--to to-IP`
создает набор **macipmap** для заданного диапазона адресов IP.

`--network IP/mask`
создает набор **macipmap** для заданной сети.

`--matchunset`
При использовании необязательного параметра **--matchunset** IP-адреса, которые могут быть включены в набор, но еще не включены в него, будут считаться включенными в набор.

Отметим, что модули ядра set и операция SET (iptables) всегда используют MAC-адрес из пакета для проверки, добавления или удаления записи из наборов типа **macipmap**.

portmap

Тип **portmap** служит для задания непрерывного диапазона номеров портов и использует диапазон памяти, каждый бит которого представляет один порт. Тип **portmap** позволяет создавать наборы, содержащие до 65535 портов. Этот тип наборов работает очень быстро и экономно расходует память.

Для создания наборов **portmap** используются опции:

`--from from-port`
`--to to-port`
создает набор типа **portmap** для заданного диапазона портов.

iphash

Тип **iphash** служит для создания произвольных наборов адресов IP и использует для хранения IP-адресов хэш-таблицу. Для предотвращения совпадающих хэш-значений используется метод двойного хэширования (double-hashing) и (как крайняя мера) динамическое увеличение размера хэша. Тип **iphash** является достаточно быстрым и удобен для хранения произвольных наборов адресов. С помощью опции **--netmask** с маской CIDR (0-32) можно сохранять в наборе блоки адресов взамен единичных адресов IP. Адрес считается относящимся к набору, если соответствующий ему префикс (блок адресов), относится к данному набору.

При создании наборов **iphash** используются следующие опции:

`--hashsize hashsize`
задает начальный размер хэша (по умолчанию 1024)

`--probes probes`
задает число попыток проверки наличия совпадений при добавлении IP-адреса в хэш с использованием двойного хэширования (по умолчанию 8).

`--resize percent`
задает увеличение размера хэша в процентах (по умолчанию 50), когда не удается добавить адрес после заданного числа попыток двойного хэширования.

`--netmask CIDR-netmask`
необязательный параметр **--netmask** позволяет сохранять в наборе префиксы сетей вместо IP-адресов.

nethash

Тип **nethash** использует хэширование для включения в набор префиксов сетей различных размеров. Адреса, используемые в наборах типа **nethash**, должны задаваться в формате префикс IP/размер маски CIDR (1-31). Для того, чтобы избавиться от совпадения хэш-значений используется метод двойного хэширования и (в качестве крайней меры) динамическое увеличение размера хэша.

При создании наборов **nethash** используются опции:

`--hashsize hashsize`
задает начальный размер хэша (по умолчанию 1024)

`--probes probes`
задает число попыток проверки наличия совпадений при добавлении префикса в хэш с использованием двойного хэширования (по умолчанию 2).

`--resize percent`
задает увеличение размера хэша в процентах (по умолчанию 50) когда не удается добавить адрес после заданного числа попыток двойного хэширования.

Адрес IP будет считаться входящим в набор, если он относится к любому из блоков, включенных в этот набор. Проверка производится начиная с блоков минимального размера и заканчивается самым крупным блоком. При добавлении и удалении адресов IP в хэш-блок с помощью модуля ядра **SET** будет добавляться/удаляться блок минимального размера, который может быть найден в наборе.

iptree

Тип **iptree** использует дерево для хранения адресов IP и может поддерживать тайм-ауты (время жизни элемента набора).

При создании набора типа **iptree** может использоваться опция:

`--timeout value`

значение тайм-аута для записи в секундах (по умолчанию 0). По истечении заданного тайм-аутом времени запись удаляется из набора. Если задано нулевое значение тайм-аута, записи из набора не удаляются автоматически.

Тайм-аут можно также задавать при добавлении адреса или номера порта в набор, указывая его как %timeout после адреса или порта (например, src%10).

Ограничения

Имена наборов (**setname**) не могут начинаться с двоеточия (**:**). Пустые элементы набора не могут использоваться.

Рекомендации

Если вы хотите создать набор из сетей одинакового размера, относящихся к одному префиксу (например, блоки /24 из сети /8), используйте тип **ipmap**. Для создания набора блоков одинакового размера (например, /24) из произвольных сетей используйте тип **iphash**. Для создания набора префиксов произвольных размеров лучше всего подходит тип **nethash**. Тип **iptree** позволяет создавать наборы элементов с заранее заданным сроком жизни. Такая возможность полезна, например, для ограничения частоты подключений с одного адреса IP.

Диагностика

Программа выводит различные сообщения об ошибках. Код 0 говорит об отсутствии ошибок, код 2 – о некорректном использовании параметров, а во всех остальных случаях возвращается код 1.

Автор

Программу ipset создал Jozsef Kadlecsik на основе программы ippool, авторами которой являются Joakim Axelsson, Patrick Schaaf и Martin Josefsson.

Условие set и операция SET в iptables

Наборы IP могут использоваться в правилах iptables вместе с условием **set** или операцией **SET**. Можно проверять пакеты на принадлежность адресов и или портов к указанным наборам (условие **set**), добавлять или удалять адреса и номера портов в существующие наборы (операция **SET**). Оба расширения поддерживают до 6 уровней “вложенности” связок. Аргументы **src** и **dst** (указываются через запятую) определяют чей (отправителя или получателя) адрес или номер порта can будет использоваться для проверки условия.

Операция SET

Операция SET позволяет добавлять адрес или номер порта из заголовка соответствующего правилу пакета в указанный набор или удалять адрес (номер порта) из указанного набора. Набор, используемый в качестве параметра операции должен быть создан заранее с помощью **ipset** (см. выше). Если указанного в правиле набора не существует, iptables выдает сообщение об ошибке.

`-j SET --add-set SetName flag[,flag[,flag...]]]`
`-j SET --del-set SetName flag[,flag[,flag...]]]`

Параметр **--add-set** задает добавление в набор, указанный параметром **SetName**, а параметр **--del-set** – удаление из указанного набора. Адрес или номер порта для добавления и удаления берется из заголовка пакета, соответствующего остальной части правила. Поле **flag** представляет собой одно или множество разделенных запятыми значений **src** и **dst**, определяющих откуда (отправитель или получатель) брать значение параметра для включения в набор. Каждый элемент списка относится к соответствующему элементу или набору связки (по порядку связывания).

При добавлении в набор уже имеющегося в нем адреса/порта или при попытке удаления отсутствующего в наборе адреса/порта не выполняется никаких реальных действий и сообщения об ошибке не выдается.

Созданные с помощью этой операции наборы можно использовать в других правилах iptables вместе с условием **set**.

Условие set

Условие **set** позволяет проверять наличие адреса IP, пары IP-MAC, номера порта в том или ином наборе или связке наборов. На основании проверки может приниматься решение о дальнейшей судьбе пакета.

`-m set [!] --set SetName flag[,flag[,flag...]]] -j ACCEPT`

Опция **-m** указывает на необходимость использования модуля соответствия **set**, **--set SetName** указывает имя набора, в котором проверяется наличие (или отсутствие при использовании знака инверсии **!**) адреса или номера порта. Параметр **flag** представляет собой одно или множество разделенных запятыми значений **src** и **dst**, каждое из которых относится к соответствующему (по порядку) элементу или набору связки и определяет из какого поля заголовка (отправитель или получатель) берется значение для проверки.

Указанный в правиле набор должен быть создан заранее с помощью **ipset** (см. выше). Если указанного в правиле набора не существует, iptables выдает сообщение об ошибке.

Использование связок

Наборы IP позволяют связать (bind) элемент одного набора с другим набором. Например, вы можете связать те или иные элементы набора адресов IP с набором номеров портов. Наборы могут иметь используемую по умолчанию связку, которая относится к каждому элементу данного набора, для которого связка не определена явно.

Сами по себе связки не представляют интереса. Однако при использовании наборов в правилах iptables связки обеспечивают целый ряд новых возможностей. Пакет будет соответствовать правилу iptables, содержащему условие –set тогда и только тогда, когда этот пакет будет соответствовать всем наборам относящихся к данному правилу связок. Отметим, что связки могут быть

“вложенными”, т. е., элемент набора, связанного с данным элементом, в свою очередь может быть связан с другим набором и т. д. Уровень “вложенности” связок может достигать 6.

Рассмотрим простой пример:

```
# создадим набор servers типа ipmap, содержащий IP-адреса двух машин
ipset -N servers ipmap --network 192.168.0.0/16
ipset -A servers 192.168.0.1
ipset -A servers 192.168.0.2
# создадим набор ports типа portmap, содержащий номера разрешенных портов для хоста 192.168.0.2
ipset -N ports portmap --from 1 --to 1024
ipset -A ports 21
ipset -A ports 22
ipset -A ports 25
# свяжем набор портов с адресом 192.168.0.2
ipset -B servers 192.168.0.2 -b ports
```

Далее создадим правила iptables, использующие эту связку

```
iptables -A FORWARD -m set --set servers dst,dst -j ACCEPT
iptables -A FORWARD -j DROP
```

В результате пакетный фильтр, использующий наборы и связки, будет пропускать пакеты, адресованные в любой порт хоста 192.168.0.1, а для 192.168.0.2 будут пропускаться только пакеты, адресованные в порты 21, 22 и 25.