Министерство образования и науки Украины
Севастопольский национальный технический университет

# АДМИНИСТРИРОВАНИЕ ИНФОРМАЦИОННЫХ СИСТЕМ

Методические указания
к выполнению лабораторных работ №5 — № 6
по дисциплине
“Администрирование информационных систем”
для студентов специальности
7.080401 – "Информационные управляющие системы и технологии"
всех форм обучения

Севастополь
2008

Create PDF files without this message by purchasing novaPDF printer (http://www.novapdf.com)

УДК 004.92

Методические указания к выполнению лабораторных работ №5 — № 6 по дисциплине “Администрирование информационных систем”/ Сост. Ю.В. Доронина, А.В. Исаева. — Севастополь: Изд-во СевНТУ, 2008. — 28 с.

Целью методических указаний по дисциплине “Администрирование информационных систем” является выработка у студентов практических навыков администрирования информационных систем и закрепление их на примере реальных информационных систем. Рассматриваются вопросы управления пользователями на сервере, ограничения доступа к базам данных, нарушения целостности файла базы данных, средства сервера и действия администратора ИС по обнаружению повреждений файлов баз данных, а также защите от несанкционированного доступа из внешней сети.

Методические указания составлены в соответствии с требованиями программы дисциплины “Администрирование информационных систем” для студентов пятого курса дневной и заочной форм обучения специальности 7.080401 и утверждены на заседании кафедры информационных систем (протокол №7 от 6 февраля 2008 года).

Допущено учебно-методическим центром СевНТУ в качестве методических указаний.

Рецензент: Г. Г. Сергеев, канд. техн. наук, доцент кафедры КиВТ ;

Create PDF files without this message by purchasing novaPDF printer (http://www.novapdf.com)

# СОДЕРЖАНИЕ



Create PDF files without this message by purchasing novaPDF printer (http://www.novapdf.com)

# ОБЩИЕ ТРЕБОВАНИЯ

## 1. Цель и задачи лабораторных работ

Цель настоящих лабораторных работ состоит в исследовании вопросов защиты локальной сети или компьютера в локальной сети от несанкционированного доступа из внешней сети.

Задачами лабораторных работ являются:

– углубленное изучение основных теоретических положений дисциплины «Администрирование информационных систем»;

– получение практических навыков по осуществлению защиты от несанкционированного проникновения из внешней сети на примере работы с программным комплексом, обеспечивающим межсетевой экран (брандмауэром).

## 2. Описание лабораторной установки

Лабораторные работы выполняются на ЭВМ с частотой работы процессора не менее 200МГц при наличии оперативной памяти объемом не менее 128 Мб. В качестве операционной системы используется ОС LINUX. Для выполнения лабораторных работ необходим брандмауэр в виде программного комплекса Iptables.

## 3. Порядок выполнения лабораторных работ

Варианты заданий студенты получают от преподавателя или выбирают самостоятельно по номеру зачетной книжки. Номер варианта определяется как остаток от деления двух последних цифр номера зачетной книжки на количество вариантов задания. Например, две последние цифры 23 и всего 7 вариантов. Остаток от деления 23 на 7 равен 2. Следовательно, номер варианта задания 2.

При выполнении каждой лабораторной работы необходимо:

- изучить соответствующие теоретические положения, методические указания;
- выполнить задание на лабораторную работу на ЭВМ в соответствии с вариантом;
- ответить на контрольные вопросы;
- подготовить и защитить отчет о выполненной работе.

## 4. Содержание отчета

Отчеты по лабораторной работе оформляются каждым студентом индивидуально. Отчет должен включать: название и номер лабораторной работы; цель работы; краткие теоретические сведения; постановку задачи; тексты и описания программ; результаты обработки данных по разработанным программам; выводы по работе.

Create PDF files without this message by purchasing novaPDF printer (http://www.novapdf.com)

# ЛАБОРАТОРНАЯ РАБОТА №5

## “Основные сведения о брандмауэрах. Изучение порядка движения пакетов”

## 1. Цель работы

Изучить основные сведения о брандмауэрах, их типы и организацию. Рассмотреть простейшие возможности брандмауэра на примере программы Iptables. Изучить порядок движения транзитных пакетов, пакетов для локального приложения и от локальных процессов.

## 2. Основные положения

### 2.1. Основные сведения о брандмауэрах

Для защиты локальной сети используется комплекс программного обеспечения, известный как Firewall (брандмауэр), или межсетевой экран. Брандмауэр позволяет "отгородить" систему или сеть от внешней сети. Он используется для предотвращения получения посторонними данных или ресурсов защищаемой сети, а также для контроля внешних ресурсов, к которым имеют доступ пользователи данной сети.

Чаще всего брандмауэр – это набор программ маршрутизации и фильтрации сетевых пакетов. Такие программы позволяют определить, можно ли пропустить данный пакет и если можно, то отправить его точно по назначению. Для того чтобы брандмауэр мог это сделать, ему необходимо определить набор правил фильтрации. Главная цель брандмауэра – контроль удаленного доступа извне или изнутри защищаемой сети или компьютера.

Брандмауэр позволяет лишь частично решить проблемы, связанные с обеспечением безопасного функционирования сети. Как бы хорошо он ни был настроен, если вовремя не обновить программный пакет, в котором была найдена уязвимость, или кто-то узнал логин и пароль пользователя сети или компьютера – высока вероятность несанкционированного доступа. Основная задача брандмауэра – разрешать функционирование только тем службам, которым было разрешено работать в данной сети или защищаемом компьютере.

Брандмауэры можно разделить по типу построения защиты:

- пороговый брандмауэр и его разновидность бастионного типа;
- брандмауэр, организующий так называемую демилитаризованную зону.

Брандмауэр порогового типа призван защитить локальную сеть от атак извне, а при соответствующей настройке и от атак изнутри. Такого типа брандмауэры обычно используются для защиты небольшой сети или одного компьютера. Как правило, сетевые службы, предоставляющие услуги вне локальной сети (HTTP, FTP и т. п.), размещаются на том же компьютере, что и брандмауэр.

Организация демилитаризованной зоны оправдана тогда, когда в сети выделено несколько специальных компьютеров для интернет-сервисов, предоставляемых внешней сети, а также при отсутствии уверенности в благонадежности сотрудников. Для организации демилитаризованной зоны используются, по меньшей мере, два брандмауэра: один для защиты

Create PDF files without this message by purchasing novaPDF printer (http://www.novapdf.com)

демилитаризованной зоны от проникновения извне, а второй – от проникновения из вашей собственной локальной сети. Организация демилитаризованной зоны сложнее, чем организация брандмауэра бастионного типа, но при этом обеспечивается большая защита данных.

**Брандмауэр с фильтрацией пакетов**

Брандмауэр с фильтрацией пакетов представляет собой "сито" для проходящих через него входящих и исходящих пакетов. В операционной системе Linux реализован брандмауэр, позволяющий контролировать ICMP-, UDP-и TCP-пакеты. Брандмауэр с фильтрацией пакетов организован как механизм, реализующий набор разрешающих и запрещающих правил для входящих и исходящих пакетов. Этот набор правил определяет, какие пакеты могут проходить через конкретный сетевой интерфейс.

Брандмауэр с фильтрацией пакетов может производить с проходящим пакетом всего три действия:

1. переслать пакет в узел назначения;
2. удалить пакет без уведомления посылающей пакет стороны;
3. вернуть передающему компьютеру сообщение об ошибке.

Несмотря на простоту таких действий, в большинстве случаев их достаточно для организации эффективной защиты.

Как правило, брандмауэр устанавливается для того, чтобы контролировать данные, которыми компьютеры обмениваются с Интернетом. В результате работы фильтрующего брандмауэра отсеиваются недопустимые обращения к узлам внутренней сети, и запрещается передача из внутренней сети в Интернет для пакетов, определенных правилами фильтрации.

В целях получения более гибкой системы правила фильтрации пакетов составляются для каждого сетевого интерфейса, в них учитываются IP-адреса источника и получателя, номера портов TCP и UDP, флаги TCP-соединений и ICMP-сообщений. Причем правила для входящих и исходящих пакетов различаются. Это значит, что при настройке фильтрующего брандмауэра правила для конкретного сетевого интерфейса представляются как отдельные правила для входящей и исходящей информации, поскольку входящие и исходящие пакеты обрабатываются брандмауэром независимо друг от друга. Списки правил, которые управляют фильтрацией сетевых пакетов, поступающих извне в локальную сеть и отправляемых из локальной сети в Интернет, принято называть цепочками (chains). Термин "цепочка" используется потому, что при проверке пакета правила применяются последовательно одно за другим, пока не обнаружится подходящее правило для сетевого пакета или список правил не будет исчерпан.

Описанный механизм фильтрующего брандмауэра достаточно эффективен, однако он не обеспечивает полной безопасности локальной сети. Брандмауэр всего лишь один из элементов общей схемы защиты. Анализ заголовков сетевых пакетов – операция слишком низкого уровня, для того чтобы реально выполнять аутентификацию и контролировать доступ. В процессе фильтрации пакетов практически невозможно распознать отправителя сообщения и проанализировать смысл передаваемой информации. Из всего набора данных, пригодных для аутентификации, на рассматриваемом уровне доступен только IP-адрес отправителя,

Create PDF files without this message by purchasing novaPDF printer (http://www.novapdf.com)

однако этот адрес очень легко подделать, на чем и базируется множество способов сетевых атак. Несмотря на то, что средства фильтрации пакетов позволяют эффективно контролировать обращение к портам, использование протоколов обмена и содержимое пакетов, проверку данных необходимо продолжить на более высоком уровне.

**Политика организации брандмауэра**

При построении брандмауэров используются два основных подхода:

1. запрещается прохождение всех пакетов, пропускаются лишь те, которые удовлетворяют явно определенным правилам;
2. разрешается прохождение всех пакетов, за исключением пакетов, удовлетворяющих определенным правилам.

Другими словами, запрещено все, что не разрешено, и разрешено все, что не запрещено.

С практической точки зрения лучше использовать подход, при котором поступающий пакет по умолчанию отвергается (запрещено все, что не разрешено). В этом случае организация безопасности сети достигается достаточно просто, но с другой стороны, приходится предусматривать возможность обращения к каждой сетевой службе и использование каждого конкретного протокола. Это означает, что администратор сети, занимающийся настройкой брандмауэра, должен точно знать, какие протоколы применяются в его локальной сети. При использовании подхода, предусматривающего запрет по умолчанию, приходится предпринимать специальные меры всякий раз, когда необходимо разрешить доступ к какому-то ресурсу, однако эта модель с нашей точки зрения более надежна, чем противоположный вариант.

Политика разрешения по умолчанию позволяет добиться функционирования системы малыми усилиями, но при этом необходимо предусмотреть каждый конкретный случай, при котором требуется запретить доступ. Может случиться так, что необходимость внесения запретов станет ясна лишь тогда, когда в результате несанкционированного доступа сети будет нанесен значительный ущерб.

В обоих случаях для конфигурации брандмауэра используются цепочки правил. Каждая цепочка представляет собой набор правил, заданных явным образом, и политику по умолчанию. Пакет проверяется на соответствие каждому из правил, а правила выбираются из списка последовательно до тех пор, пока не будет обнаружено соответствие сетевого пакета одному из них. Если пакет не удовлетворяет ни одному из заданных правил, с сетевым пакетом производятся действия, определенные политикой по умолчанию.

В процессе работы брандмауэр может пропустить сетевой пакет (ACCEPT), запретить прохождение сетевого пакета (DENY) либо отказать сетевому пакету в прохождении, т. е. отклонить его (REJECT).

При отклонении сетевого пакета (REJECT) сам пакет удаляется, а его отправителю возвращается ICMP-сообщение об ошибке.

При запрете прохождения сетевого пакета (DENY) сам пакет удаляется, но отправитель не оповещается об удалении сетевого пакета.

В большинстве случаев запрет сетевого пакета считается лучшим решением, чем отказ в прохождении сетевого пакета. Во-первых, отправка сообщения об

Create PDF files without this message by purchasing novaPDF printer (http://www.novapdf.com)

ошибке увеличивает сетевой трафик, а во-вторых, сообщения об ошибке могут быть использованы для организации атаки с целью вывода из строя сервера. Помимо этого, любое ответное действие на "неправильные" пакеты предоставляет взломщику дополнительную информацию о конфигурации системы.

## 2.2. Работа с программой IpTables

Дальнейшее рассмотрение брандмауэра продолжим на примере программы Iptables. При описании понадобятся следующие термины: цепочка – это набор правил, которые управляют фильтрацией пакетов; таблица – совокупность цепочек.

Когда пакет приходит на брандмауэр, он сначала попадает на сетевое устройство, перехватывается соответствующим драйвером и далее передается в ядро. Далее пакет проходит ряд таблиц и затем передается либо локальному приложению, либо переправляется на другую машину.

**Таблица Mangle**

Эта таблица предназначена для внесения изменений в заголовки пакетов (mangle - искажать, изменять). В ней устанавливаются биты ToS (Type Of Service) и прочие другие. В этой таблице не следует производить любого рода фильтрацию, маскировку или преобразование адресов (DNAT, SNAT, MASQUERADE).

В таблице Mangle можно выполнять только следующие действия:

- действие ToS выполняет установку битов поля Type of Service в пакете;
- действие TTL используется для установки значения поля TTL (Time To Live) пакета;
- действие MARK устанавливает специальную метку на пакет, которая затем может быть проверена другими правилами в iptables или другими программами, например iproute2. С помощью "меток" можно управлять маршрутизацией пакетов, ограничивать трафик и т.п.

**Таблица Nat**

Эта таблица используется для преобразований сетевых адресов NAT (Network Address Translation). Только первый пакет из потока проходит через цепочки этой таблицы, трансляция адресов или маскировка применяются ко всем последующим пакетам в потоке автоматически.

Для таблицы Nat характерны следующие действия.

- Действие DNAT (Destination Network Address Translation) производит преобразование адресов назначения в заголовках пакетов. То есть производится перенаправление пакетов на другие адреса, отличные от указанных в заголовках пакетов.
- SNAT (Source Network Address Translation) используется для изменения исходных адресов пакетов. С помощью этого действия можно скрыть структуру локальной сети, а заодно и разделить единственный внешний IP-адрес между компьютерами локальной сети для выхода в Интернет. В этом случае брандмауэр с помощью SNAT автоматически производит прямое и обратное преобразование адресов, давая возможность выполнять подключение к серверам в Интернете с компьютеров в локальной сети.
- Маскировка (MASQUERADE) применяется в тех же целях, что и SNAT, но в отличие от последнего MASQUERADE сильнее загружает систему.

Create PDF files without this message by purchasing novaPDF printer (http://www.novapdf.com)

Происходит это потому, что каждый раз, когда требуется выполнение этого действия, производится запрос IP-адреса для указанного в действии сетевого интерфейса, в то время как для SNAT IP-адрес указывается непосредственно. Однако благодаря такому отличию, MASQUERADE может работать в случаях с динамическим IP-адресом, т.е. при подключении к Интернет, например, через PPP, SLIP или DHCP.

**Таблица Filter**

В этой таблице содержатся наборы правил для выполнения фильтрации пакетов. Пакеты могут пропускаться далее либо отвергаться (действия ACCEPT и DROP соответственно) в зависимости от их содержимого. Можно отфильтровывать пакеты и в других таблицах, но эта таблица существует именно для нужд фильтрации. В ней допускается использование большинства из существующих действий, однако ряд действий, должны выполняться только в присущих им таблицах. Порядок следования транзитных пакетов демонстрирует таблица 1.

Таблица 1 – Порядок движения транзитных пакетов

| Шаг | Таблица | Цепочка | Примечание |
|---|---|---|---|
| 1 | | | Кабель (т.е. Интернет) |
| 2 | | | Сетевой интерфейс (например, eth0) |
| 3 | mangle | PREROUTING | Обычно эта цепочка используется для внесения изменений в заголовок пакета, например для изменения битов ToS (Type of Servise – тип обслуживания) и т.д. |
| 4 | nat | PREROUTING | Эта цепочка используется для трансляции (преобразования) сетевых адресов (Destination Network Address Translation). Source Network Address Translation выполняется позднее, в другой цепочке. Любого рода фильтрация в этой цепочке может производиться только в исключительных случаях |
| 5 | | | Принятие решения о дальнейшей маршрутизации, т.е. в этой точке решается, куда пойдет пакет: локальному приложению или на другой узел сети. |
| 6 | mangle | FORWARD | Далее пакет попадает в цепочку FORWARD таблицы mangle, которая должна использоваться только в исключительных случаях, когда необходимо внести некоторые изменения в заголовок пакета между двумя точками (в случае принятия решения о дальнейшей маршрутизации). |

Create PDF files without this message by purchasing novaPDF printer (http://www.novapdf.com)

Продолжение таблицы 1

| | | | |
|---|---|---|---|
| 7 | Filter | FORWARD | В цепочку FORWARD попадают только те пакеты, которые идут на другой хост. Вся фильтрация транзитного трафика должна выполняться здесь. Через эту цепочку проходит трафик в обоих направлениях, это обстоятельство следует учитывать при написании правил фильтрации. |
| 8 | mangle | POSTROUTING | Эта цепочка предназначена для внесения изменений в заголовок пакета, после того как принято последнее решение о маршрутизации. |
| 9 | nat | POSTROUTING | Эта цепочка предназначена в первую очередь для Source Network Address Translation. Не следует использовать ее для фильтрации без особой необходимости. Здесь же выполняется маскарадинг (Masquerading). |
| 10 | | | Выходной сетевой интерфейс (например, eth1). |
| 11 | | | Кабель (например, LAN). |

Пакет проходит несколько этапов, прежде чем он будет передан далее. На каждом из них пакет может быть остановлен, будь то цепочка iptables или что-либо другое. Заметим, что нет каких-либо цепочек, специфичных для отдельных интерфейсов. Цепочку FORWARD проходят все пакеты, которые движутся через брандмауэр/роутер. При этом цепочка INPUT для фильтрации транзитных пакетов не используется, они в нее не попадают. Через нее движутся только те пакеты, которые предназначены данному хосту.

Теперь рассмотрим порядок движения пакета, предназначенного локальному процессу/приложению. Он представлен в таблице 2.

Таблица 2 – Порядок движения пакетов для локального приложения

| Шаг | Таблица | Цепочка | Примечание |
|---|---|---|---|
| 1 | | | Кабель (т.е. Интернет) |
| 2 | | | Входной сетевой интерфейс (например, eth0) |
| 3 | mangle | PREROUTING | Обычно используется для внесения изменений в заголовок пакета: для установки битов ToS и пр. |
| 4 | nat | PREROUTING | Преобразование адресов (Destination Network Address Translation). Фильтрация пакетов здесь допускается только в исключительных случаях. |
| 5 | | | Принятие решения о маршрутизации. |
| 6 | mangle | INPUT | Пакет попадает в цепочку INPUT таблицы mangle. Здесь вносятся изменения в заголовок пакета перед тем, как он будет передан локальному приложению. |

Create PDF files without this message by purchasing novaPDF printer (http://www.novapdf.com)

Продолжение таблицы 2

| | | | |
|---|---|---|---|
| 7 | | INPUT | Здесь производится фильтрация входящего трафика. Все входящие пакеты, адресованные нам, проходят через эту цепочку, независимо от того, с какого интерфейса они поступили. |
| 8 | | | Локальный процесс/приложение (программа-сервер или программа-клиент) |

В этом случае пакеты идут через цепочку INPUT, а не через FORWARD.

В заключение рассмотрим порядок движения пакетов, созданных локальными процессами. Он представлен в таблице 3 и в виде схемы на рисунке 1.

Таблица 3 – Порядок движения пакета от локальных процессов

| Шаг | Таблица | Цепочка | Примечание |
|---|---|---|---|
| 1 | | | Локальный процесс (т.е. программа-сервер или программа-клиент). |
| 2 | | | Принятие решения о маршрутизации. Здесь решается, куда пойдет пакет дальше: на какой адрес, через какой сетевой интерфейс и пр. |
| 3 | mangle | OUTPUT | Здесь производится внесение изменений в заголовок пакета. Выполнение фильтрации в этой цепочке может иметь негативные последствия. |
| 4 | nat | OUTPUT | Эта цепочка используется для трансляции сетевых адресов (NAT) в пакетах, исходящих от локальных процессов брандмауэра. |
| 5 | Filter | OUTPUT | Здесь фильтруется исходящий трафик. |
| 6 | mangle | POSTROUTING | Цепочка POSTROUTING таблицы mangle, в основном, используется для правил, которые должны вносить изменения в заголовок пакета перед тем, как он покинет брандмауэр, но уже после принятия решения о маршрутизации. В эту цепочку попадают все пакеты, как транзитные, так и созданные локальными процессами брандмауэра. |
| 7 | nat | POSTROUTING | Здесь выполняется Source Network Address Translation. В этой цепочке не рекомендуется производить фильтрацию пакетов во избежание нежелательных побочных эффектов. Однако и здесь можно останавливать пакеты, применяя политику по умолчанию DROP. |
| 8 | | | Сетевой интерфейс (например, eth0) |
| 9 | | | Кабель (Internet) |

Create PDF files without this message by purchasing novaPDF printer (http://www.novapdf.com)

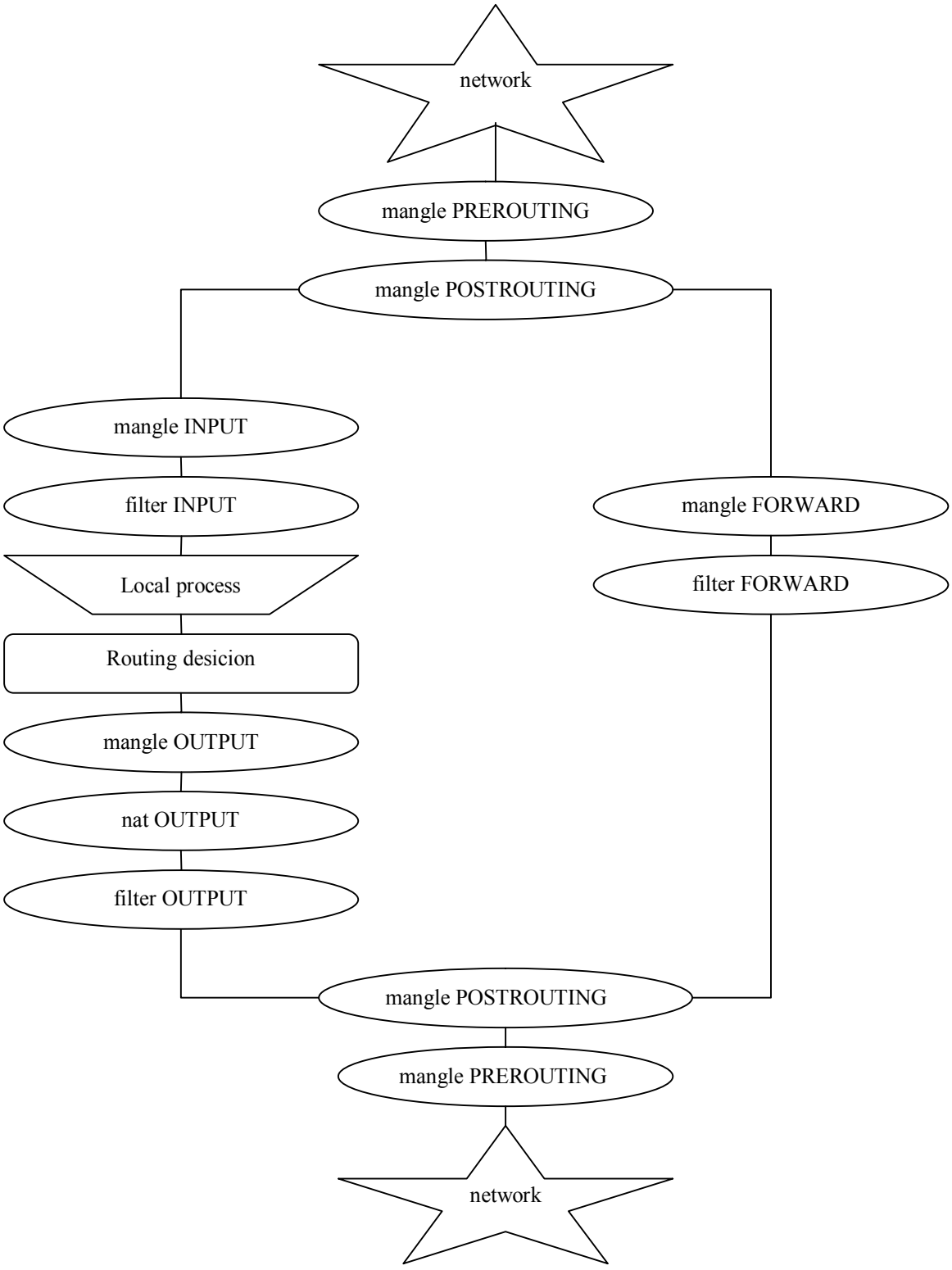


Рисунок 1 – Порядок прохождения пакетов.

Create PDF files without this message by purchasing novaPDF printer (http://www.novapdf.com)

**Механизм определения состояния**

Механизм определения состояния (state machine) является отдельной частью iptables, фактически он является механизмом трассировки соединений. Трассировщик соединений создан для того, чтобы netfilter мог постоянно иметь информацию о состоянии каждого конкретного соединения. Наличие трассировщика позволяет создавать более надежные наборы правил по сравнению с брандмауэрами, которые не имеют поддержки такого механизма.

В пределах iptables соединение может иметь одно из 4-х базовых состояний: NEW, ESTABLISHED, RELATED и INVALID.

Таблица 4 – Состояния соединений и их описание

| Состояние | Описание |
|---|---|
| NEW | Признак NEW сообщает о том, что пакет является первым для данного соединения. Это означает, что это первый пакет в данном соединении, который «увидел» модуль трассировщика. Например, если получен SYN пакет, являющийся первым пакетом для данного соединения, то он получит статус NEW. Однако пакет может и не быть SYN пакетом и, тем не менее, получить статус NEW. Это может иногда породить определенные проблемы, но может оказаться и весьма полезным, например, когда желательно "подхватить" соединения, "потерянные" другими брандмауэрами или в случаях, когда таймаут соединения уже истек, но само соединение не было закрыто. |
| RELATED | Состояние RELATED одно из самых сложных. Соединение получает статус RELATED, если оно связано с другим соединением, имеющим признак ESTABLISHED. Это означает, что соединение получает признак RELATED тогда, когда оно инициировано из уже установленного соединения, имеющего признак ESTABLISHED. Хорошим примером соединения, которое может рассматриваться как RELATED, является соединение FTP-data, связанное с портом FTP control, а также DCC соединение, запущенное из IRC. Следует помнить, что большинство протоколов TCP и некоторые из протоколов UDP весьма сложны и передают информацию о соединении через область данных TCP или UDP пакетов, поэтому требуют наличия специальных вспомогательных модулей для корректной работы. |
| ESTABLISHED | Состояние ESTABLISHED говорит о том, что это не первый пакет в соединении. Единственное требование, предъявляемое к соединению, заключается в том, что для перехода в состояние ESTABLISHED необходимо, чтобы узел сети передал пакет и получил на него ответ от другого узла (хоста). После получения ответа состояние соединения NEW или RELATED будет изменено на ESTABLISHED. |

Create PDF files without this message by purchasing novaPDF printer (http://www.novapdf.com)

Продолжение таблицы 4

| | |
|---|---|
| INVALID | Признак INVALID говорит о том, что пакет не может быть идентифицирован и поэтому не имеет определенного статуса. Это может происходить по разным причинам, например при нехватке памяти или при получении ICMP-сообщения об ошибке, которое не соответствует какому-либо известному соединению. Для таких пакетов рекомендуется действие DROP. |

## 3. Порядок выполнения работы

1) Ознакомиться с теоретическим материалом.
2) Имеется сервер локальной сети со службой http. Необходимо настроить доступ к серверу. Для этого выполнить следующие действия:

1. Сбросить все цепочки командой

```
iptables –F
```

2. Установить политики по умолчанию командами

```
iptables –P INPUT DROP
iptables –P OUTPUT DROP
iptables –P FORWARD DROP
```

3. Разрешить прохождение данных существующих и «связанных» соединений командами

```
iptables –A INPUT –m state -state RELATED,ESTABLISHED  -j ACCEPT
iptables –A OUTPUT –m state –state RELATED, ESTABLISHED -j ACCEPT
```

4. Разрешить входящий запрос по HTTP протоколу командой

```
iptables –A INPUT –p tcp –dport 80 –j ACCEPT
```

Примечание: прикладные программы могут работать некорректно из-за отсутствия настройки всех протоколов и интерфейса «обратной петли».

## 4. Контрольные вопросы

1. Что такое брандмауэр? Типы брандмауэров.
2. Политика разрешения и политики запрещения.
3. Поясните понятия: пропускание, запрещение и отклонение сетевого пакета
4. Таблицы mangle, nat, filter.
5. Порядок движения транзитных пакетов.
6. Порядок движения пакета от локальных процессов.
7. Порядок движения пакета к локальному приложению.
8. Механизм определения состояний. Назовите основные состояния и определите их свойства.

Create PDF files without this message by purchasing novaPDF printer (http://www.novapdf.com)

# ЛАБОРАТОРНАЯ РАБОТА №6

## “Построение правил для функционирования брандмауэра под управлением ОС LINUX”

## 1. Цель работы

Научиться составлять правила для функционирования брандмауэров под управлением ОС LINUX.

## 2. Основные положения

Каждое правило – это строка, содержащая в себе критерии, определяющие, подпадает ли пакет под заданное правило, и действие, которое необходимо выполнить в случае выполнения критерия. В общем виде правила записываются так: iptables [-t table] command [match] [target/jump]

Описание действия (target/jump) не обязательно должно стоять последним в строке, однако такая запись удобнее в прочтении

Если в правило не включается спецификатор [-t table], то по умолчанию предполагается использование таблицы filter.

Непосредственно за именем таблицы стоит команда. Если спецификатора таблицы нет, то команда всегда должна стоять первой. Команда определяет действие iptables: вставить правило, добавить правило в конец цепочки, удалить правило и т.п. Подробно команды рассмотрены в таблице 5.

Раздел match задает критерии проверки, по которым определяется, подпадает ли пакет под действие этого правила. Здесь могут быть самые разные критерии: IP-адрес источника пакета или сети, IP-адрес места назначения, порт, протокол, сетевой интерфейс и т.д.

Раздел target указывает, какое действие должно быть выполнено при условии выполнения критериев в правиле. Здесь можно заставить ядро передать пакет в другую цепочку правил, "сбросить" пакет, выдать на источник сообщение об ошибке и т.д. Основные виды критериев рассмотрены в таблице 6.

Таблица 5 – Команды и их описание

| | |
|---|---|
| Команда | -A, –append |
| Пример | iptables -A INPUT ... |
| Описание | Добавляет новое правило в конец заданной цепочки. |
| Команда | -D, –delete |
| Пример | iptables -D INPUT –dport 80 -j DROP, iptables -D INPUT 1 |
| Описание | Удаление правила из цепочки. Команда имеет 2 формата записи: первый – когда задается критерий сравнения с опцией -D (см. первый пример), второй – порядковый номер правила. Если задается критерий сравнения, то удаляется правило, которое имеет в себе этот критерий, если задается номер правила, то будет удалено правило с заданным номером. Счет правил в цепочках начинается с 1. |

Create PDF files without this message by purchasing novaPDF printer (http://www.novapdf.com)

Продолжение таблицы 5

| | |
|---|---|
| Команда | -R, –replace |
| Пример | iptables -R INPUT 1 -s 192.168.0.1 -j DROP |
| Описание | Эта команда заменяет одно правило другим. В основном, она используется во время отладки новых правил. |
| Команда | -I, –insert |
| Пример | iptables -I INPUT 1 –dport 80 -j ACCEPT |
| Описание | Вставляет новое правило в цепочку. Число, следующее за именем цепочки, указывает номер правила, перед которым нужно вставить новое правило. Другими словами, число задает номер для вставляемого правила. В примере выше, указывается, что данное правило должно быть 1-м в цепочке INPUT. |
| Команда | -L, –list |
| Пример | iptables -L INPUT |
| Описание | Вывод списка правил в заданной цепочке в данном примере предполагается вывод правил из цепочки INPUT. Если имя цепочки не указывается, то выводится список правил для всех цепочек. Формат вывода зависит от наличия дополнительных ключей в команде, например -n, -v, и пр. |
| Команда | -F, –flush |
| Пример | iptables -F INPUT |
| Описание | Сброс (удаление) всех правил из заданной цепочки (таблицы). Если имя цепочки и таблицы не указывается, то удаляются все правила, во всех цепочках. Если таблица не указана ключом -t (–table), то очистка цепочек производится только в таблице filter. |
| Команда | -N, –new-chain |
| Пример | iptables -N allowed |
| Описание | Создается новая цепочка с заданным именем в заданной таблице. В вышеприведенном примере создается новая цепочка с именем allowed. Имя цепочки должно быть уникальным и не должно совпадать с зарезервированными именами цепочек и действий (такими как DROP, REJECT и т.п.) |
| Команда | -X, –delete-chain |
| Пример | iptables -X allowed |
| Описание | Удаление заданной цепочки из заданной таблицы. Удаляемая цепочка не должна иметь правил и не должно быть ссылок из других цепочек на удаляемую цепочку. Если имя цепочки не указано, то будут удалены все цепочки заданной таблицы кроме встроенных. |
| Команда | -P, –policy |
| Пример | iptables -P INPUT DROP |
| Описание | Задает политику по умолчанию для заданной цепочки. Политика по умолчанию определяет действие, применяемое к пакетам, не попавшим под действие ни одного из правил в цепочке. В качестве политики по умолчанию используют DROP и ACCEPT. |

Create PDF files without this message by purchasing novaPDF printer (http://www.novapdf.com)

Таблица 6 – Критерии и их описание

| | |
|---|---|
| Критерий | -p, –protocol |
| Пример | iptables –A INPUT -p tcp |
| Описание | Для указания типа протокола. Примеры протоколов: TCP, UDP и ICMP. Список протоколов – в файле /etc/protocols. В качестве имени протокола в данный критерий передается один из трех вышеупомянутых протоколов или ключевое слово ALL. В качестве протокола можно передавать число – номер протокола: например, протоколу ICMP соответствует число 1, TCP – 6 и UDP – 17. Соответствия между номерами протоколов и их именами находятся в файле /etc/protocols. Если данному критерию передается числовое значение 0, это эквивалентно использованию спецификатора ALL, который подразумевается по умолчанию, когда критерий – protocol не используется. Для логической инверсии критерия перед именем протокола используется символ «!»: например, – protocol ! tcp подразумевает пакеты протоколов UDP и ICMP. |
| Критерий | -s, –src, –source |
| Пример | iptables -A INPUT -s 192.168.1.1 |
| Описание | IP-адрес(а) источника пакета. Адрес источника может указываться так, как показано в примере, тогда подразумевается единственный IP-адрес. Можно указать адрес в виде address/mask, например как 192.168.0.0/255.255.255.0 или более современным способом 192.168.0.0/24, т.е. фактически определяя диапазон адресов. Символ !, установленный перед адресом, означает логическое отрицание, т.е. –source ! 192.168.0.0/24 означает любой адрес кроме адресов 192.168.0.x. |
| Критерий | -d, –dst, –destination |
| Пример | iptables -A INPUT -d 192.168.1.1 |
| Описание | IP-адрес(а) получателя. Имеет синтаксис схожий с критерием –source, за исключением того, что подразумевает адрес места назначения. Точно так же может определять как единственный IP-адрес, так и диапазон адресов. Символ ! используется для логической инверсии критерия. |
| Критерий | -i, –in-interface |
| Пример | iptables -A INPUT -i eth0 |
| Описание | Интерфейс, с которого был получен пакет. Использование этого критерия возможно в цепочках INPUT, FORWARD и PREROUTING, в других случаях вызывается сообщение об ошибке. При отсутствии этого критерия предполагается любой интерфейс, что равносильно использованию критерия -i +. Символ ! инвертирует результат совпадения. Если имя интерфейса завершается символом +, то критерий задает все интерфейсы, начинающиеся с заданной строки: например, -i PPP+ обозначает любой PPP интерфейс, а запись -i ! eth+ – любой интерфейс, кроме любого eth. |

Create PDF files without this message by purchasing novaPDF printer (http://www.novapdf.com)

Продолжение таблицы 6

| | |
|---|---|
| Критерий | -o, –out-interface |
| Пример | iptables -A FORWARD -o eth0 |
| Описание | Задает имя выходного интерфейса. Этот критерий допускается использовать только в цепочках OUTPUT, FORWARD и POSTROUTING, в противном случае будет генерироваться сообщение об ошибке. При отсутствии этого критерия предполагается любой интерфейс, что равносильно использованию критерия -o +. Символ ! инвертирует результат совпадения. Если имя интерфейса завершается символом +, то критерий задает все интерфейсы, начинающиеся с заданной строки, например -o eth+ обозначает любой eth интерфейс, а запись -o ! eth+ - любой интерфейс, кроме любого eth. |
| Критерий | -f, –fragment |
| Пример | iptables -A INPUT -f |
| Описание | Правило распространяется на все фрагменты фрагментированного пакета, кроме первого. Сделано это потому, что нет возможности определить исходящий/входящий порт для фрагмента пакета, а для ICMP-пакетов определить их тип. С помощью фрагментированных пакетов могут производиться атаки на брандмауэр, так как фрагменты пакетов могут не отлавливаться другими правилами. Как и раньше, допускается использования символа ! для инверсии результата сравнения, только в данном случае символ ! должен предшествовать критерию -f, например ! -f. Инверсия критерия трактуется как "все первые фрагменты фрагментированных пакетов и/или нефрагментированные пакеты, но не вторые и последующие фрагменты фрагментированных пакетов". |

Для использования следующих критериев необходимо указать тип протокола tcp или udp. Данные критерии применимы только для этих протоколов. Эти критерии описаны в таблице 7.

Create PDF files without this message by purchasing novaPDF printer (http://www.novapdf.com)

Таблица 7 – TCP, UDP – критерии и их описание.

| Критерий | –sport, –source-port |
|---|---|
| Пример | iptables -A INPUT -p tcp –sport 22 |
| Описание | Исходный порт, с которого был отправлен пакет. В качестве параметра может указываться номер порта или название сетевой службы. Соответствие имен сервисов и номеров портов находятся в файле /etc/services. При указании номеров портов правила отрабатывают несколько быстрее, однако это менее удобно при разборе листингов скриптов. Если же нужно создать значительные по объему наборы правил, например, порядка нескольких сотен и более, то предпочтительнее использовать номера портов. Номера портов могут задаваться в виде интервала из минимального и максимального номеров, например, –source-port 22:80. Если опускается минимальный порт, т.е. когда критерий записывается как –source-port :80, то в качестве начала диапазона принимается число 0. Если опускается максимальный порт, т.е. когда критерий записывается как –source-port 22:, то в качестве конца диапазона принимается число 65535. Допускается такая запись:–source-port 80:22, в этом случае iptables поменяет числа 22 и 80 местами, т.е. подобного рода запись будет преобразована в –source-port 22:80. Как и раньше, символ ! используется для инверсии. Например, критерий –source-port ! 22 подразумевает любой порт, кроме 22. Инверсия может применяться и к диапазону портов, например, –source-port ! 22:80. Дополнительная информация находится в описании критерия multiport. |
| Критерий | –dport, –destination-port |
| Пример | iptables -A INPUT -p tcp –dport 22 |
| Описание | Порт или диапазон портов, на который адресован пакет. Аргументы задаются в том же формате, что и для –source-port. |

ICMP – критерии описаны в таблице 8, явные критерии – в таблице 9.

Таблица 8 – ICMP критерии

| Критерий | –icmp-type |
|---|---|
| Пример | iptables -A INPUT -p icmp –icmp-type 8 |
| Описание | Тип сообщения ICMP определяется номером или именем. Числовые значения определяются в RFC 792. Чтобы получить список имен ICMP значений, надо выполнить команду iptables –protocol icmp –help, или посмотреть приложение «Типы ICMP». Символ ! инвертирует критерий, например –icmp-type ! 8. |

Create PDF files without this message by purchasing novaPDF printer (http://www.novapdf.com)

Таблица 9 – Явные критерии

| | |
|---|---|
| Критерий state используется совместно с кодом трассировки соединений и позволяет получить информацию о признаке состояния соединения, что позволяет судить о состоянии соединения, причем даже для таких протоколов как ICMP и UDP. Данное расширение необходимо загружать явно с помощью ключа -m state. | |
| Ключ | –state |
| Пример | iptables -A INPUT -m state –state RELATED,ESTABLISHED |
| Описание | Проверяется признак состояния соединения (state). Можно указать 4 состояния: INVALID, ESTABLISHED, NEW и RELATED. INVALID подразумевает, что пакет связан с неизвестным потоком или соединением и, возможно, содержит ошибку в данных или в заголовке. Состояние ESTABLISHED указывает на то, что пакет принадлежит уже установленному соединению, через которое пакеты идут в обоих направлениях. Признак NEW подразумевает, что пакет открывает новое соединение или пакет принадлежит однонаправленному потоку. Признак RELATED указывает на то, что пакет принадлежит уже существующему соединению, но при этом он открывает новое соединение. Примером тому может служить передача данных по FTP, или выдача сообщения ICMP об ошибке, которое связано с существующим TCP или UDP соединением. Следует заметить, что признак NEW это не то же самое, что установленный бит SYN в пакетах TCP, посредством которых открывается новое соединение. Подобного рода пакеты могут быть потенциально опасны в случае, когда для защиты сети используется один сетевой экран. |
| Расширение multiport позволяет указывать в тексте правила несколько портов и диапазонов портов.<br>Использовать стандартную проверку портов и расширение -m multiport (например –sport 1024:63353 -m multiport –dport 21,23,80) одновременно нельзя. Подобные правила будут просто отвергаться iptables. | |
| Ключ | –source-port |
| Пример | iptables -A INPUT -p tcp -m multiport –source-port 22,53,80,110 |
| Описание | Указывает список исходящих портов. С помощью данного критерия можно указать до 15 различных портов. Названия портов в списке должны отделяться друг от друга запятыми, пробелы в списке не допустимы. Данное расширение может использоваться только вместе с критериями -p tcp или -p udp. Часто используется как расширенная версия обычного критерия –source-port. |
| Ключ | –destination-port |
| Пример | iptables -A INPUT -p tcp -m multiport –destination-port 22,53,80,110 |
| Описание | Служит для указания списка входных портов. Формат задания аргументов полностью аналогичен -m multiport –source-port. |
| Ключ | –port |

Create PDF files without this message by purchasing novaPDF printer (http://www.novapdf.com)

Продолжение таблицы 9

| | |
|---|---|
| Пример | iptables -A INPUT -p tcp -m multiport –port 22,53,80,110 |
| Описание | Данный критерий проверяет как исходящий, так и входящий порт пакета. Формат аргументов аналогичен критерию –source-port и –destination-port. Следует обратить внимание на то, что данный критерий проверяет порты обоих направлений, т.е. под критерий –m multiport –port 80 попадают пакеты, идущие с порта 80 в порт 80. |

**Действия и переходы**

Действия ACCEPT, REJECT и DROP были рассмотрены ранее.

Действие **DNAT** может выполняться только в цепочках PREROUTING и OUTPUT таблицы nat, и во вложенных подцепочках. Важно запомнить, что вложенные подцепочки, реализующие DNAT, не должны вызываться из других цепочек кроме PREROUTING и OUTPUT. Ключи для действий и переходов указаны в таблице 10.

Таблица 10 – Ключи для действий и переходов и их описание

| | |
|---|---|
| Ключ | –to-destination |
| Пример | iptables -t nat -A PREROUTING -p tcp -d 15.45.23.67 –dport 80 -j DNAT –to-destination 192.168.1.1-192.168.1.10 |
| Описание | Ключ –to-destination указывает, какой IP-адрес должен быть подставлен в качестве адреса места назначения. В выше приведенном примере во всех пакетах, пришедших на адрес 15.45.23.67, адрес назначения будет изменен на один из диапазона от 192.168.1.1 до 192.168.1.10. Как указывалось выше, все пакеты из одного потока будут направляться на один и тот же адрес, а для каждого нового потока будет выбираться один из адресов в указанном диапазоне случайным образом. Можно также определить единственный IP-адрес. Можно дополнительно указать порт или диапазон портов, на который (которые) будет перенаправлен трафик. Для этого после IP-адреса через двоеточие надо указать порт, например, –to-destination 192.168.1.1:80, а указание диапазона портов выглядит так: –to-destination 192.168.1.1:80-100. Синтаксис действий DNAT и SNAT во многом схож. Следует помнить, что указание портов допускается только при работе с протоколом TCP или UDP при наличии опции –protocol в критерии. |

Маскарадинг (**MASQUERADE**) представляет собой то же самое, что и SNAT, но не имеет ключа –to-source. Причиной тому то, что маскарадинг может работать, например, с dialup подключением или DHCP, т.е. в тех случаях, когда IP-адрес присваивается устройству динамически. Если подключение динамическое, то нужно использовать маскарадинг, если имеется статическое IP подключение, то лучше использовать действия SNAT.

Create PDF files without this message by purchasing novaPDF printer (http://www.novapdf.com)

Действие MASQUERADE указывается только в цепочке POSTROUTING таблицы nat, так же как и действие SNAT. Действие MASQUERADE имеет ключ, описываемый ниже в таблице 11, использование которого необязательно.

Таблица 11 – Ключи для маскарадинга и их описание

| Ключ | –to-ports |
|---|---|
| Пример | iptables -t nat -A POSTROUTING -p TCP -j MASQUERADE –to-ports 1024-31000 |
| Описание | Ключ –to-ports используется для указания порта источника или диапазона портов исходящего пакета. Можно указать один порт, например, –to-ports 1025 или диапазон портов: –to-ports 1024-3000. Этот ключ можно использовать только в правилах, где критерий содержит явное указание на протокол TCP или UDP с помощью ключа –protocol. |

**REDIRECT** выполняет перенаправление пакетов и потоков на другой порт той же самой машины. Например, можно пакеты, поступающие на HTTP порт перенаправить на порт HTTP proxy. Действие REDIRECT очень удобно для выполнения "прозрачного" проксирования (transparent proxying), когда машины в локальной сети даже «не подозревают» о существовании прокси.

REDIRECT может использоваться только в цепочках PREROUTING и OUTPUT таблицы nat. Это действие также можно выполнять в подцепочках, вызываемых и вышеуказанных. Для действия REDIRECT предусмотрен только один ключ, описанный в таблице 12.

Таблица 12 – Ключи для REDIRECT и их описание

| Ключ | –to-ports |
|---|---|
| Пример | iptables -t nat -A PREROUTING -p tcp –dport 80 -j REDIRECT –to-ports 8080 |
| Описание | Ключ –to-ports определяет порт или диапазон портов назначения. Без указания ключа –to-ports, перенаправления не происходит, т.е. пакет идет на тот порт, куда и был назначен. В примере, приведенном выше, –to-ports 8080 указан один порт назначения. Если нужно указать диапазон портов надо написать –to-ports 8080-8090. Этот ключ можно использовать только в тех правилах, где критерий содержит явное указание на протокол TCP или UDP с помощью ключа –protocol. |

**SNAT** преобразует сетевые адреса (Source Network Address Translation), т.е. изменяет исходящий IP-адрес в IP заголовке пакета. Это действие можно использовать для предоставления выхода в Интернет другим компьютерам из локальной сети, имея лишь один уникальный IP-адрес. Для этого необходимо включить пересылку пакетов (forwarding) в ядре, затем создать правила, которые будут транслировать исходящие IP-адреса данной локальной сети в реальный внешний адрес. В результате для внешней сети будет ничего не известно о данной локальной сети, она будет считать, что запросы пришли с брандмауэра.

Create PDF files without this message by purchasing novaPDF printer (http://www.novapdf.com)

SNAT допускается выполнять только в таблице nat, в цепочке POSTROUTING. Ключи для SNAT и их описание находятся в таблице 13.

Таблица 13 – Ключи для SNAT и их описание

| Ключ | –to-source |
|---|---|
| Пример | iptables –t nat -A POSTROUTING -p tcp -o eth0 -j SNAT –to-source 194.236.50.155-194.236.50.160:1024-32000 |
| Описание | Ключ –to-source используется для указания адреса, присваиваемого пакету. Надо указать IP-адрес, который будет подставлен в заголовок пакета как исходящий. Если надо перераспределить нагрузку между несколькими брандмауэрами, можно указать диапазон адресов, где начальный и конечный адреса диапазона разделяются дефисом, например: 194.236.50.155-194.236.50.160. Тогда конкретный IP-адрес будет выбираться из диапазона случайным образом для каждого нового потока. Дополнительно можно указать диапазон портов, которые будут использоваться только для нужд SNAT. Все исходящие порты будут после этого перекартированы в заданный диапазон. Iptables старается избегать перекартирования портов, однако не всегда это возможно. Если диапазон портов не задан, то исходные порты ниже 512 перекартируются в диапазоне 0-511, порты в диапазоне 512-1023 – в диапазоне 512-1023, порты из диапазона 1024-65535 – в диапазоне 1024-65535. Порты назначения не подвергаются перекартированию. |

Действие **TTL** используется для изменения содержимого поля Time To Live в IP заголовке. Действие TTL можно указывать только в таблице mangle и нигде больше. Для данного действия предусмотрено 3 ключа, описанных в таблице 14.

Таблица 14 – Ключи для TTL и их описание

| Ключ | –ttl-set |
|---|---|
| Пример | iptables -t mangle -A PREROUTING -i eth0 -j TTL –ttl-set 64 |
| Описание | Устанавливает поле TTL в заданное значение. Оптимальным считается значение около 64. Слишком большое значение может иметь неприятные последствия для локальной сети: если пакет "зацикливается" между двумя неправильно сконфигурированными роутерами, тогда при больших значениях TTL, есть риск "потерять" значительную долю пропускной способности канала. |

На что следует обратить особое внимание:

- При прохождении правила по цепочке правила применяются к пакету последовательно. Т.е. если к пакету было применено действие DROP, ACCEPT или REJECT, то пакет прекращает продвижение в данной цепочке.
- Так как правила применяются последовательно, необходимо самые часто используемые правила располагать в начале цепочки (правила для протоколов

Create PDF files without this message by purchasing novaPDF printer (http://www.novapdf.com)

обмена файлами и т.п.), а самые редко используемые – в конце цепочки (правила взаимодействия по интерфейсу «обратной петли» и т.п.).

## 3. Варианты заданий

1. Имеется маршрутизатор с включенным брандмауэром. Разрешить доступ из локальной сети (ЛС) к сервисам Internet (HTTP, HTTPS, FTP, DNS). Для самого сервера разрешить доступ только к DNS (определенный сервер с адресом IP_DNS).
2. Имеется ЛС, имеющая выход в Интернет через маршрутизатор (с фиксированными адресами портов). Обеспечить взаимодействие ЛС с Интернет «от одного лица». На сервере включен протокол SSH для обеспечения управления сервером.
3. Доступ из ЛС в Internet обеспечивает маршрутизатор с фильтрацией пакетов. На маршрутизаторе работает HTTP сервер Apache (порт 9684). Сделать данный сервер доступным через его стандартный порт (80). Разрешить доступ к серверам E-Mail в глобальной сети, функционирующим на основе протоколов SMTP, POP3, IMAP.
4. Доступ из ЛС в Internet обеспечивает маршрутизатор с фильтрацией пакетов. Обеспечить обмен информацией локальной сети с DNS сервером за пределами ЛС. Включить SNAT. Обеспечить работу в Internet сервера HTTP, который представлен PC, находящейся в сети (рассмотреть 3 случая: обращение из Internet, из локальной сети и от маршрутизатора).
5. Дана локальная машина с доступом в Интернет. Разрешить взаимодействие по протоколам HTTP, FTP, telnet, SSH, DNS.
6. ЛС взаимодействует с Интернет при помощи выделенной машины (маршрутизатор). Маршрутизатор подключен к глобальной сети Интернет через модем (IP-адрес назначается динамически). Обеспечить работу ЛС в Интернет с использованием SNAT, установкой TTL в определенное значение (для скрытия сети). Запретить приложениям маршрутизатора работать в Интернет, при «безграничном» доступе к/из локальной сети.
7. Имеется маршрутизатор с включенным брандмауэром. Запретить вхождение в сеть запросов типа Echo-request. Для самого сервера разрешить доступ только к DNS (определенный сервер с адресом IP_DNS). Разрешить любое взаимодействие ЛС с Интернет. Разрешить взаимодействие ЛС с установленным на маршрутизаторе SSH. Запретить остальным приложениям маршрутизатора работу в сети.

## 4. Контрольные вопросы

1. Понятие правила. Общий формат записи.
2. Назовите и опишите основные команды.
3. Критерии и их описание. Критерии различных протоколов.
4. Основные действия и переходы. Объясните их функции и назначение.

Create PDF files without this message by purchasing novaPDF printer (http://www.novapdf.com)

# БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Oskar Andreasson Iptables Tutorial 1.1.19=Руководство по Iptables 1.1.19 / Перевод Киселева А.– Copyright © 2001—2002 by Oskar Andreasson.– 430с.

2. Ботте Т. Руководство администратора сети/ Т. Ботте, Т. Доусон, Г. Перди. – Кудиц-Образ, 2004.– 386с.

Create PDF files without this message by purchasing novaPDF printer (http://www.novapdf.com)

Create PDF files without this message by purchasing novaPDF printer (http://www.novapdf.com)

Create PDF files without this message by purchasing novaPDF printer (http://www.novapdf.com)

Заказ№……  от «…..»  Тираж……экз.
Изд-во СевНТУ

Create PDF files without this message by purchasing novaPDF printer (http://www.novapdf.com)