Стара статья но разжевано все детально

Взято: http://linuxportal.ru/entry.php/95_0_3_0_C/

Автор статьи: **Alexey Dmitriev**
Дата: 30.07.2003

Маскарадинг (masquerading) и трансляции ip-адресов (NAT)

Проблема маскарада (masquerading) и трансляции ip-адресов в Линуксе. Что это такое, что там можно и что нельзя.

Автор оригинала Иван Н. Песин

Но так как статья морально устарела, то она будет взята лишь за основу. (претензии автора принимаются). Каждому Linux admin у рано или поздно приходится сталкиваться с проблемой маскарада (masquerading) и трансляции ip-адресов на Линуксе (кроме тех у кого есть куча реальных адресов и они раздают их всем компам в локальной сети :-)). Если у Вас есть локальная сеть, подключенная к сети Интернет через linux сервер, то без маскарада не обойтись.
Начнем, пожалуй, с определений.

1. Маскарад (masquerade) и трансляция адресов (NAT) в мире Linux не являются синонимами.
2. Маскарад - замена адреса на адрес машины, выполняющей маскарад.
3. Трансляция адресов - замена адреса на любой указанный.

Если вы используете ОС Linux с ядром 2.0 или 2.2, Вам нужно будет опустить глаза вниз - в конце этой статьи находится авторское описание настроек ipfwadm и ipchains.
Я же опишу настройку NAT, masquerading для ОС Linux с ядром 2.4- 2.6, де используются iptables.
Для чего это вообще нужно? Вы имеете сеть с адресами:
10.0.0.0-10.255.255.255 - сеть класса A
172.16.0.0 - 172.31.255.255 - сеть класса B
192.168.0.0-192.168.255.255 - сеть класса C
Эти сети зарезервированы для использования в локальных сетях (intranet) и в сети Интернет не используются.

Допустим ваш компьютер в локальной сети с ip 192.168.0.2 получает доступ к Интернет через сервер с внутренним ip адресом 192.168.0.1 (eth1) и внешним адресом 111.111.111.111 (eth0).

Если пакет с вашего компьютера будет иметь ваш адрес источника (192.168.0.2) то он просто не придет обратно, так как удаленный хост не будет знать по какому маршруту отослать его обратно. Для того, чтобы этого не случилось и придуманы NAT и masquerading.

Итак как работает маскарад.

Ваш пакет (например на www.ibm.com) проходит через сервер и в нем адрес источника меняется на адрес сервере (111.111.111.111). Пакет приходит на www.ibm.com и хост отвечает по адресу в пакете (111.111.111.111). Так как Ваш сервер запомнил, что пакет для www.ibm.com посылали вы, то он принимает пакет и отдает его вашему компьютеру.

Вот и все пакет ушел и вернулся.

Включается маскарад в iptables очень просто:

iptables -A FORWARD -s 192.168.0.0/24 -j ACCEPT

(этой командой вы разрешили прохождение пакетов между сетевыми интерфейсами из локальной сети 192.168.0.0/24)

iptables -A FORWARD -d 192.168.0.0/24 -j ACCEPT

(этой командой вы разрешили прохождение пакетов между сетевыми интерфейсами в локальную сеть 192.168.0.0/24)

iptables -t nat -A POSTROUTING -o eth0 -s 192.168.0.0/24 -j MASQUERADE

(и последняя команда - ей вы включили маскарад для сети 192.168.0.0/24).

Еще вам нужно проверить чтобы был включен forward ip в вашем ядре.

Сделать это можно командой

cat /proc/sys/net/ipv4/ip_forward

если Вы получили 1 на выходе значит все в порядке, если 0, тогда вам нужно включить ip forward командой

echo 1 >/proc/sys/net/ipv4/ip_forward

Как работает NAT ?

Ваш пакет (например на www.ibm.com) проходит через сервер и в нем адрес источника меняется на указанный адрес (мы укажем адрес сервера (111.111.111.111). Пакет приходит на www.ibm.com и хост отвечает по адресу в пакете (111.111.111.111). Пакет приходит на Ваш сервер и происходит обратная замена.

Включается NAT в iptables так:

iptables -A FORWARD -s 192.168.0.0/24 -j ACCEPT

(этой командой вы разрешили прохождение пакетов между сетевыми интерфейсами из локальной сети 192.168.0.0/24)

iptables -A FORWARD -d 192.168.0.0/24 -j ACCEPT

(этой командой вы разрешили прохождение пакетов между сетевыми интерфейсами в локальную сеть 192.168.0.0/24)

iptables -t nat -A POSTROUTING -o eth0 -s 192.168.0.0/24 -j SNAT --to-source 111.111.111.111

(и последняя команда - ей вы включили трансляцию адресов сети 192.168.0.0/24 на адрес 111.111.111.111).
Еще вам нужно проверить чтобы был включен forward ip в вашем ядре.
Сделать это можно командой
cat /proc/sys/net/ipv4/ip_forward
если Вы получили 1 на выходе значит все в порядке, если 0, тогда вам нужно включить ip forward командой
echo 1 >/proc/sys/net/ipv4/ip_forward
Вот собственно и все. Дальше идет статья Ивана Песина
Решение проблемы маскарада, на ядрах разных версий отличается, и потому мы разделим нашу повесть на две части, ядра версии от, если не ошибаюсь, 2.0.29 до 2.2.9ac, и от 2.2.10 до 2.4.0. Ядра версии 2.4.x я обсуждать не буду, потому как там имеется вполне сносное описание. Итак,

Ядра версий 2.0.29 - 2.2.9

Для функции маскарада в этих версиях необходим пакет ipfwadm либо ipchains, зависит от версии. Установив их можно приступать к осмысленным действиям. Очень тривиальная задача решается следующим образом:

ipchains -A forward -i ethX -s 192.168.1.0/24 -j MASQ

результатом этой команды будет замена исходных адресов пакетов из сети 192.168.1.0 класса C, при маршрутизации их через интерфейс ethX, на адрес машины на которой выполнены команды.

Перейдем теперь к более сложным случаям. Допустим, вам необходимо, в зависимости от адреса источника/назначения маскарадить под разными адресами. Как же это выполнить? Тут нам придется обратиться к нестандартным средствам.

Для начала посмотрим урл http://www.suse.de/~mha/HyperNews/get/linux-ip-nat.html. Хотя там написано не очень много, зато содержательно. После прочтения остается лишь вытащить соответствующий архив с программой и патчем ядра. Пропатчив ядро, его надо перекомпилировать (прочитать почитайки обязательно!) с соответствующими изменениями в конфигурации. Устанавливаем скомпилированное ядро, при этом не забываем про "ядро на всякий случай" (рабочее ядро, которое вы будете загружать, если новое ядро не загрузится). После перезагрузки станет возможным использование команды ipnatadm. Название говорит само за себя ;). Теперь легко можно изобразить команды вроде:

ipnatadm -O -i -b -S 192.168.1.1/24 -M 207.46.230.219
ipnatadm -O -i -b -S 192.168.1.2/24 -M 207.46.230.229

ну и так далее, думаю, разобраться можно. Есть еще полезная опция -W, позволяющая задавать интерфейс. Да, не забудьте, что псевдонимы интерфейсов не понимаются ни ipchains, ни ipfwadm, ни ipnatadm. Перейдем теперь к теме

Ядра версий 2.2.10 - 2.4.0

На этих версиях, к сожаленью, программа ipnatadm предательски перестает функционировать. Почему предательски? Потому что удается пропатчить ядро, скомпилировать, загрузиться и даже приконнектиться с помощью telnet, например. Но при попытке выполнить, допустим, даже вышеописанные команды, интерфейс подвисает. Ну что ж, значит нужно искать другие ходы. Если нужно, значит надо.

Внимательно прочитав информацию, размещенную на сайте с ipnatadm, можно обнаружить любопытный факт. А именно то, что один наш земляк Алексей Кузнецов, пишет часть ядра, которая, в том числе, ведает маршрутизацией и маскарадом. Из этого прямо следует, что функциональность этого кода огромна, а документация - м-да, хромает, мягко выражаясь. Сам Кузнецов советует разбирать исходники ядра - для тех, кто сможет, захочет, имеет время, и т.д. и т.п.

Ну а я попробую, насколько это возможно, сжато рассмотреть стандартный и нестандартные методы маскарада, и NAT´а. Стандартный случай: необходимо маскарадить исходящие пакеты хоста 192.168.1.1 в 207.46.230.219. Подход к решению этой задачи полностью аналогичен случаю с более ранними ядрами. Необходимо лишь учесть, что в системах с этими ядрами утилиты ipfwadm нет, вместо нее используется ipchains. Сразу перейдем к рассмотрению более сложных случаев. Допустим, как и в предыдущем примере, пакеты с разных машин, необходимо транслировать в разные адреса.

Тут придется прибегнуть к утилите ip из пакета iproute. Изобразим следующие команды:

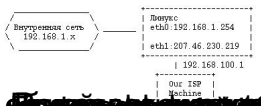
```
ip rule add from 192.168.1.1 nat 207.46.230.219
ip rule add from 192.168.1.2 nat 207.46.230.229
```

кроме того:

```
ip route add nat 207.46.230.219 via 192.168.1.1
ip route add nat 207.46.230.229 via 192.168.1.2
```

Результатом этих действий будет маскарад и демаскарад адресов 192.168.1.1 и 192.168.1.2, соответственно в 207.46.230.219 и 229.

Вот такие пирожки. Это все теория и стандартные случаи, давайте рассмотрим более интересный и совсем реальный пример. Имеется сеть на ~250 машин, линукс красная шапочка версии 6.2, в роли маршрутизатора, и выполняющий маскарад внутренних машин под адрес, допустим, 207.46.230.219. Линукс включен в машину провайдера витой парой, из сетевой в сетевую. Кроме того, провайдер, естественно, не желает тратить отдельный ip-адрес на сетевую в которую включен наш линукс. Другими словами (рисунком) ситуация следующая


Внутренняя сеть
192.168.1.x
Люпукс
eth0:192.168.1.254
eth1:207.46.230.219
192.168.100.1
Our ISP
Machine