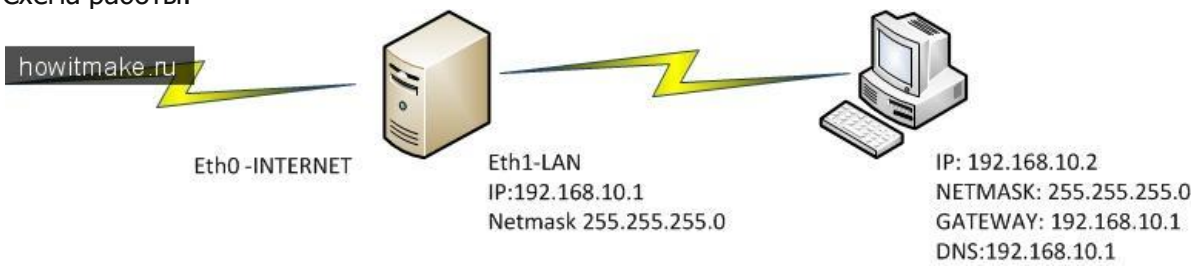
Настройка шлюза локальной сети, на базе Ubuntu 12.04

Если у вас есть локальная сеть, то для ее клиентов необходимо предоставить доступ в интернет. Для обеспечения данной возможности необходимо настроить шлюз, который будет принимать запросы клиентов и пересылать их во внешний мир, а поступающие ответы, передавать обратно.

В этой статье, я расскажу как настроить шлюз для локальной сети на Ubuntu, схема его работы очень простая, но и защиты он не обеспечивает. Существует распространенное заблуждение что **NAT** (Network Address Translation -транслятор сетевых адресов) обеспечивает защиту от внешних угроз, но это далеко не так, у него совершенно другое назначение, фактически это дверь, которая открывается в обе стороны, со всеми вытекающими последствиями…
Если есть необходимость в повышении уровня безопасности, а такая необходимость есть всегда, то тут нужно настраивать «фаерволл», я намеренно не привожу никаких правил на этот случай, т.к. они настраиваются исходя из конкретных задач.
Для работы нам понадобится
1) Система с двумя сетевыми картами, которая будет выполнять функцию шлюза.
2) Клиентский ПК, с которого мы будем тестировать работу шлюза.

Схема работы:



Предполагается, что операционная система у вас установлена, на сервере который имеет 2 сетевых интерфейса.
eth0 — подключение к интернет. (может получать IP адрес динамически, может иметь статический, я опишу оба варианта)
eth1 — подключение к локальной сети, будет иметь статический IP **192.168.10.1** и маску **255.255.255.0**
Также, для тестирования нам понадобится клиентская машина, которая будет находиться в локальной сети (операционная система значения не имеет).

Первым делом, настраиваем сетевые интерфейсы сервера:
Поднимаем права до **root**

```
sudo su
```

Вводим пароль.
Редактируем настройки сетевых интерфейсов:

```
nano /etc/network/interfaces
```

Навастриваем **eth0** (по которому осуществляется подключение к интернет)
Вариант №1- Получение IP по DHCP от провайдера:

```
auto eth0
iface eth0 inet dhcp
```

Вариант №2-Статический IP

```
auto eth0
iface eth0 inet static
address XXX.XXX.XXX.XXX
netmask 255.YYY.YYY.YYY
gateway XXX.XXX.XXX.XXX
dns-nameservers ZZZ.ZZZ.ZZZ.ZZZ
```

Где:
Вместо XXX.XXX.XXX.XXX вписываем IP адрес, который мы получили от провайдера
Вместо 255.YYY.YYY.YYY -выписываем маску подсети.
Ну и вместо ZZZ.ZZZ.ZZZ.ZZZ вписываем IP адрес DNS сервера.

Настраиваем **eth1** (по которому подключается локальная сеть)

```
auto eth1
iface eth1 inet static
address 192.168.10.1
netmask 255.255.255.0
```

В результате действий у нас должен получиться файл interfaces, примерно, следующего содержания:

```
auto lo
iface lo inet loopback

auto eth0
iface eth0 inet static
address XXX.XXX.XXX.XXX
netmask 255.YYY.YYY.YYY
gateway XXX.XXX.XXX.XXX

auto eth1
iface eth1 inet static
address 192.168.10.1
netmask 255.255.255.0
```

Сохраняем изменения, выходим.
Перезапустим сеть:

```
/etc/init.d/networking restart
```

Таким образом в локальной сети адрес шлюза и dns сервера, будет 192.168.10.1.

Переходим к нашему тестовому клиенту, т.к. у нас в сети нет DHCP сервера, то IP адрес мы будем назначать в ручную.
Присваиваем клиенту:
IP 192.168.10.2
маску подсети **255.255.255.0**
шлюз **192.168.10.1**
DNS **192.168.10.1**

пробуем с клиента пинговать 192.168.10.1 — запросы должны бегать нормально.

Переходим на наш шлюз.
Установим пакет **dnsmasq**, он необходим для перенаправления DNS запросов, вышестоящим серверам.

```
apt-get install dnsmasq
```

Возвращаемся к клиенту, выполняем на нем

```
nslookup mail.ru
```

В ответ подучаем:

```
Сервер: UnKnown
Address: 192.168.10.1

Имя: mail.ru
Addresses: 94.100.191.201
94.100.191.204
94.100.191.203
94.100.191.202
```

Отлично, разрешение имен работает, но если мы попробуем открыть страницу **mail.ru**, то у нас ничего не получится, потому что не настроена маршрутизация пакетов.
Настроим ее: Первым делом, разрешаем перенаправление пакетов:

```
nano /etc/sysctl.conf
```

необходимо найти строку и снять с нее комментарий:

```
net.ipv4.ip_forward=1
```

Сохраняем изменения и выходим

Теперь, нам необходимо добавить правила для маршрутизации пакетов:

```
nano /etc/rc.local
```

Добавим перед строкой **exit 0**

```
iptables -F
iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
iptables -A FORWARD -i eth0 -o eth0 -j REJECT
iptables -I FORWARD -p tcp --tcp-flags SYN,RST SYN -j TCPMSS --clamp-mss-to-
pmtu
```

Значение можно разобрать самостоятельно (будет в качестве домашнего задания), на форуме ubuntu был отличный FAQ по настройке iptables.
Сохраняем изменения, и перезагружаем сервер

```
reboot
```

Дождемся загрузки сервера и перейдя к клиентской системе, пробуем открыть сайт Mail.ru-все должно открываться! Вот так, настраивается шлюз для локальной сети. Это, одно из самых простых решений.

Настройка DHCP сервера под управлением Ubuntu 12.04

Почти 2 года прошло с публикации предыдущей статьи по настройке DHCP сервера на Ubuntu, я решил ее обновить, благо новая ОС вышла и в ней есть не большие изменения. Недавно, я рассказал о настройке шлюза для локальной сети, а эта статья будет, дополнением к предыдущей, т.е. у нас есть шлюз, теперь неплохо настроить локальную сеть. Настраивать клиентские ПК в ручную-скучно, по этому, процесс настройки сети, на клиентских устройствах, мы автоматизируем.

Предполагается что у нас:
1) Операционная система установлена.
2) Шлюз настроен по статье: Настройка шлюза локальной сети, на базе Ubuntu 12.04
3) В локальной сети у нас есть клиентский ПК, операционная система значения не имеет. (для тестирования работоспособности)

Переходим к настройке.
поднимаем права до root:

```
sudo su
```

Установим пакет DHCP сервера:

```
aptitude install isc-dhcp-server
```

Зависимости подтянутся автоматически.
Адресное пространство, в нашей локальной сети, будет находиться в диапазоне 192.168.10.0/24 т.е. в нашей подсети может находиться максимум 254 сетевых устройства.

Для начала, укажем на каком интерфейсе будет работать наш DHCP сервер

```
nano /etc/default/isc-dhcp-server
```

Нас интересует строка **INTERFACES** т.к. к локальной сети у нас подключается **eth1** вот его и укажем:

```
INTERFACES="eth1"
```

Теперь нам необходимо настроить конфигурационный файл DHCP сервера:

```
nano /etc/dhcp/dhcpd.conf
```

Сервер планируется единственным в сети, по этому будет работать в авторитарном режиме, для этого снимаем комментарий со строки:

```
authoritative;
```

Закомментируем некоторые строки, они нам не понадобятся, все параметры будут храниться в

одном месте, но об этом дальше.

```
default-lease-time 600;
max-lease-time 7200;
```

Теперь создадим нашу подсеть, диапазон IP у нам будет начиная со 192.168.10.10 и заканчивая 192.168.10.254, маска подсети 255.255.255.0 (или 24 bit), в качестве шлюза, DNS сервера у нас выступает сам сервер, указываем IP интерфейса eth1-192.168.10.1
Время аренды адреса, указывается в секундах, я указал 7 дней.

```
subnet 192.168.10.0 netmask 255.255.255.0 {
range 192.168.10.10 192.168.10.254;
option domain-name-servers 192.168.10.1;
option domain-name "example.org";
option routers 192.168.10.1;
option broadcast-address 192.168.10.255;
default-lease-time 604800;
max-lease-time 604800;
}
```

Сохраняем изменения выходим.
перезапустим DHCP

```
/etc/init.d/isc-dhcp-server restart
```

Переходим к нашему тестовому клиентскому ПК, устанавливаем в настройках сетевого соединения- получение IP адреса от DHCP сервера. Получаем настройки сети, лезем в интернет и если вы настраивали шлюз по моей статье, то с доступом в интернет у вас проблем возникнуть не должно.

В случае возникновения проблем, то их причины нужно узнавать из логов. По умолчанию **isc-dhcp-server** кидает записи в syslog, который находится в **/var/log/syslog** в случае если информации в нем, не достаточно, то можно изменить уровень логирования событий, его настройки хранятся в:

```
nano /etc/dhcp/dhcpd.conf
```

нас интересует строка:

```
log-facility local7;
```

Если потребовалось изменить уровень логирования, то можно выставить 1, тогда, чтобы выглядело:

```
log-facility local1;
```

Тогда в лог будут заноситься все события DHCP сервера, просто в огромных объемах, но это необходимо исключительно в целях выявления причины сбоя, после устранения необходимо

перевести в обычный режим.

Если есть необходимость в резервировании IP адреса за определенной машиной, то бегать к клиентскому ПК, чтобы забить там статический IP, нет необходимости, да и это совершенно не правильно. Гораздо удобнее выполнить резервацию этого IP адреса на DHCP сервере. После выполнения резервации данный IP адрес будет выдаваться только тому MAC адресу, за которым он зарегистрирован.

Делается это очень просто:
В **dhcpd.conf** добавляется следующее:

```
host testhost {
  hardware ethernet 00:01:8a:e3:s8:92;
  fixed-address 192.168.10.11;
}
```

Где:
hardware ethernet -Указываем MAC адрес сетевой карты
Остальное можно писать «от фанаря».

Если понадобилось посмотреть, какие адреса были выданы, а также узнать их статус (свободен/занят), то идем в:

```
/var/lib/dhcp/dhcpd.leases
```

На этом, пожалуй и все.