3proxy

Настройка брандмауэра

По умолчанию, в Ubuntu брандмауэр разрешает все подключения. Однако, если у нас настроен фаервол для запрета лишних соединений, необходимо открыть порт для прокси.

Если используется Iptables:

```
iptables -I INPUT 1 -p tcp --dport 3128 -j ACCEPT


netfilter-persistent save
```

- если система вернет ошибку при вводе команды для сохранения правил, устанавливаем пакет командой **apt-get install iptables-persistent**.*
- **3128** — порт по умолчанию, по которому работает 3proxy в режиме прокси;*

Установка и запуск 3proxy

3proxy отсутствует в репозиториях Ubuntu, поэтому есть два варианта установки. Полегче, и для тех, кто любит хардкор.

Готовые пакеты

У 3proxy на официальном сайте есть два вида готовых программных пакетов для возможной установки. Стабильные и те, что в разработке.

Сначала скачаем стабильный пакет для Ubuntu системы:

```
wget https://github.com/z3APA3A/3proxy/releases/download/0.9.3/3proxy-0.9.3.x86_64.deb
```

После чего, мы можем установить его с помощью команды:

```
sudo dpkg -i 3proxy-0.9.3.x86_64.deb
```

Исходники

Для начала устанавливаем необходимые в системе пакеты:

```
apt-get install build-essential git
```

Клонируем официальный репозиторий

```
git clone https://github.com/3proxy/3proxy
```

Переходим в каталог программы:

```
cd 3proxy
```

Запускаем компиляцию 3proxy:

```
make -f Makefile.Linux
```

Создаем системную учетную запись:

```
adduser --system --disabled-login --no-create-home --group 3proxy --force-badname
```

- *Поскольку в linux не позволяет по умолчанию создать пользоватля с именем, начинающемсяя с цифры, мы добавляем команду **--force-badname**, что позволит нам создать этого пользователя*

Создаем каталоги и копируем файл 3proxy в /usr/bin:

```
mkdir -p /var/log/3proxy
mkdir /etc/3proxy
cp bin/3proxy /usr/bin/
```

Задаем права на созданные каталоги:

```
chown 3proxy:3proxy -R /etc/3proxy
chown 3proxy:3proxy /usr/bin/3proxy
chown 3proxy:3proxy /var/log/3proxy
```

Создаем конфигурационный файл:

```
vi /etc/3proxy/3proxy.cfg
```

Со следующим содержанием:

```
nserver 77.88.8.8
nserver 8.8.8.8

nscache 65536
timeouts 1 5 30 60 180 1800 15 60

external 0.0.0.0
internal 0.0.0.0

daemon

auth none

allow * * * 80-88,8080-8088 HTTP
allow * * * 443,8443 HTTPS

proxy -n
```

- Необходимо обратить внимание на настройки **external** и **internal** — внешний и внутренний интерфейсы (если наш прокси работает на одном адресе, то IP-адреса должны совпадать).*
- Параметр proxy -n **-p3128** определяет порт запуска

Запускаем 3proxy:

Если вы производили установку из готовых пакетов, вы можете запустить программу следующей командой:

```
systemctrl start 3proxy
```

Если вы собирали программу из исходников, проверяем работоспособность командой:

```
/usr/bin/3proxy /etc/3proxy/3proxy.cfg
```

Далее. вам необходимо создать сервис для автозапуска

Настройка автозапуска

Для автозагрузки 3proxy настроим его как сервис. Создаем файл в systemd:

```
vi /etc/systemd/system/3proxy.service
```

С содержанием:

```
[Unit]
Description=3proxy Proxy Server

[Service]
Type=simple
ExecStart=/usr/bin/3proxy /etc/3proxy/3proxy.cfg
ExecStop=/bin/kill `/usr/bin/pgrep -u proxy3`
RemainAfterExit=yes
Restart=on-failure

[Install]
WantedBy=multi-user.target
```

Обновляем конфигурацию systemd:

```
systemctl daemon-reload
```

Разрешаем запуск сервиса и стартуем его:

```
systemctl enable 3proxy
systemctl start 3proxy
```

Настройка аутентификации

Для доступа к web прокси мы можем настроить аутентификацию, для этого нам необходимо отредактировать конфигурационный файл вновь:

```
vi /etc/3proxy/3proxy.cfg
```

Сначала необходимо изменить строчку:

```
auth none
```

На:

```
auth strong
```

После которой вы можете добавить секцию пользователей:

```
users admin:CL:password
users "3err0:CR:$1$UsbY5l$ufEATFfFVL3xZieuMtmqC0"
```

В данном примере мы добавили 2-х пользователей:

- **admin** с паролем **password**
- **3err0** с паролем **password** в виде md5 и солью **mysalt**

Для получения хэша пароля можно воспользоваться командой

```
openssl passwd -1 -salt mysalt
```

- Где в качестве соли можно использовать любую комбинацию.

Возможные типы паролей:

- *CL — текстовый пароль*
- *CR — зашифрованный пароль (md5)*
- *NT — пароль в формате NT.*

Возможные варианты для **auth**:

- *none — без авторизации.*
- *iponly — авторизация по IP-адресу клиента.*
- *nbname — по Netbios имени.*
- *strong — по логину и паролю.*

Также можно использовать двойную авторизацию, например:

```
auth nbname strong
```

После внесения изменений, перезапускаем службу:

```
systemctl restart 3proxy
```

SOCKS

Для прозрачного прохождения пакетов через прокси можно настроить SOCKS5. Необходимо изменить конфигурационный файл:

```
vi /etc/3proxy/3proxy.cfg
```

Добавив в него строчку:

```
socks
```

- ***socks*** *Запустится на стандартном орту* ***1080****.*

Если необходимо определить порт запуска, необходимо указать аргумент **-p**:

```
socks -p8083 -i192.168.1.23 -e111.111.111.111
```

** запускаем* ***socks*** *на порту* ***8083;*** *внутренний интерфейс —* ***192.168.1.23****, внешний —* ***111.111.111.111****.*

После перезапускаем сервис:

```
systemctl restart 3proxy
```

Настройка анонимности

Необходимо, чтобы время на сервере совпадало с временем на компьютере.

На стороне сервера необходимо задать часовой пояс, например, если наш прокси находится в Германии, вводим:

```
timedatectl set-timezone Asia/Irkutsk
```

На стороне клиента либо меняем часовой пояс в системе, либо устанавливаем плагин для браузера, например, для Mozilla Firefox и меняем часовой пояс уже в нем.

Отключение icmp.

По времени ответа на ping можно определить отдаленноесть клиента от прокси. Чтобы проверку нельзя было выполнить, отключаем на сервере icmp. Для этого создаем правило в брандмауэре:

```
iptables -I INPUT -p icmp --icmp-type 8 -j REJECT
```

И применяем настройку:

```
netfilter-persistent save
```

Дополнительные настройки

Настройки, которые могут не понадобиться. Но они позволят настроить дополнительные возможности прокси-сервера.

Логирование

Для включения лог файла необходимо добавить следкующие строчки в конфиг файл программы:

```
log filename LOGTYPE
rotate 30
```

LOGTYPE задает тип ротации и может принимать значения:

- M, ежемесячная ротация
- W, еженедельная ротация
- D, ежедневная ротация
- H, ежечасная ротация
- C, ежеминутная ротация

Для ведения журнала в syslog необходимо изменить строчку к такому виду:

```
  log @ident
```

- Где **@ident** соответствуют ведению журнала через syslog с соотвествующим идентификатором.

Настройка портов и прокси-интерфейсов

При необходимости, можно настроить 3proxy на использование разных интерфейсов на разных портах:

```
vi /etc/3proxy/3proxy.cfg
```

```
proxy -n -a -p3128 -i192.168.0.23 -e222.222.222.222
proxy -n -a -p8080 -i192.168.1.23 -e111.111.111.111
```

- 3proxy будет слушать на порту **3128** с внутреннего интерфейса **192.168.0.23** и направлять пакеты в сеть Интернет через внешний интерфейс **222.222.222.222**
- Также, на порту **8080** для внутреннего и внешнего интерфейсов **192.168.1.23** и **111.111.111.111** соответственно.
- Не забываем также настраивать брандмауэр (вначале инструкции мы открывали только 3128 порт).*

Ограничение пропускной способности

При необходимости, можно ограничить скорость.

```
vi /etc/3proxy/3proxy.cfg
```

```
bandlimin 1000000 user1,user3
bandlimin 5000000 user2,user4
```

- В данном примере пользователям **user1** и **user3** установлено ограничение в **1000000** бит/сек (1 мбит) для **user2** и **user4** — 5 мбит/сек.*

```
bandlimin 240000 * 192.168.0.2,192.168.0.3
```

- Разрешаем качать компьютерам с IP-адресами 192.168.0.2 и 192.168.0.3 со скоростью 24 кбит в секунду, причем это 24 кбит приходятся не на каждый из этих компьютеров, а на оба в совокупности, т.е. если оба будут к примеру, качать одновременно файлы с достаточно быстрых сайтов, то каждому придется только по 12 кбит в секунду (заметьте, килобит, а не килобайт, если надо пересчитать в килобайты, разделите числа на 8)

```
bandlimin 48000 * 192.168.0.4
```

- А этому счастливчику единолично скорость 48 кбит в секунду

```
bandlimout 24000 *
```

- исходящую скорость тоже ограничим всем до 24 килобит в секунду

```
nobandlimin * * * 110
```

- Ну и наконец, если вы хотите, чтобы эти жесткие ограничения не касались, к примеру, скачивания почты, то снимите ограничения на порт 110
- И не забудьте поставить эту команду **ПЕРЕД** прочими командами ограничения скорости - конфиг обрабатывается последовательно до первого удовлетворяющего условия.

Ограничения доступа

Можно ограничить доступ и разрешить только для определенных портов, сетей и пользователей.

```
vi /etc/3proxy/3proxy.cfg
```

Синтаксис:

```
allow <userlist> <sourcelist> <targetlist> <targetportlist> <commandlist> <weekdays>
<timeperiodslist>
deny <userlist> <sourcelist> <targetlist> <targetportlist> <commandlist> <weekdays>
<timeperiodslist>
```

Где:

- ***userlist*** *— список пользователей через запятую.*
- ***sourcelist*** *— сети клиентов через запятую.*
- ***targetlist*** *— сети назначения через запятую.*
- ***targetportlist*** *— порты назначения через запятую.*
- ***commandlist*** *— команды, к которым применяется правило.*
- ***weekdays*** *— в какие дни недели работает правило. 0 - 6 — Пн - Вс, 7 — тоже Вс.*
- ***timeperiodslist*** *— время, когда работает правило. Указываются диапазоны.*

Примеры:

```
allow * * * 80 HTTP
allow * * * 443,8443 HTTPS
allow * * * 21 FTP
```

Также, ограничить доступ можно по количеству одновременных соединений для каждой службы:

```
maxconn 700
proxy -n -a -p3128 -i192.168.0.23 -e222.222.222.222
proxy -n -a -p8080 -i192.168.1.23 -e111.111.111.111
```

- таким образом, мы установим **700** максимальных соединений для прокси на порту **3128** и **700** — для proxy на порту **8080**.*

Другие ограничения

```
deny * 192.168.200.4 * 110 * 1-5 18:00:00-23:59:59,00:00:00-08:00:00
```

- Запрещаем получение почты пользователю с IP .4 в нерабочее время по будням

```
allow * 192.168.200.4 * * * 1-5 09:00:00-18:00:00
```

- Разрешаем доступ в интернет пользователю с IP .4 исключительно с понедельника по пятницу с 9 часов утра до 6 вечера, но зато разрешены запросы на любые порты, то есть можно и с FTP-серверов качать, и HTTPS, к примеру.

```
allow * 192.168.0.2,192.168.0.3 * 80,443 * 1-7 00:00:00-23:59:59
```

- А здесь разрешаем только WEB-серфинг, зато круглосуточно и любой день недели

Настройка браузера

Проверяем работоспособность нашего 3proxy. Для этого настраиваем браузер для работы через прокси-сервер, например, Mozilla Firefox:

Пример настройки Mozilla Firefox для работы через прокси

Image not found or type unknown

Настройка почты

3proxy позволяет настроить его не только на WEB-серфинг. Электронная почта - это то, на что можно так-же испольовать это ПО. Для начала настроим получение почты. Для этого в составе 3proxy имеется свой pop3-прокси:

Вновь отредактируем наш конфиг файл:

```
vi /etc/3proxy/3proxy.cfg
```

Добавив строчку:

```
pop3p
```

В этом случае надо будет настроить ваши мэйл-клиенты. Если, к примеру, для ящика vasya.pupkin@mail.ru в настройках TheBat! в настройках доставки почты (закладка Транспорт) раньше стояли такие параметры:

```
pop3-сервер: pop.mail.ru
пользователь: vasya.pupkin
```

То чтобы получать почту через прокси, надо будет их поменять на следующие:

```
pop3-сервер: 192.168.0.1
пользователь: vasya.pupkin@pop.mail.ru
```

- **Внимание**, именно последовательность "имя пользователя"@"POP3-сервер вашего ящика", а не ваш адрес.

Для большей наглядности рассмотрим еще вариант настройки клиента:

```
pop3-сервер: mail.example.com
пользователь: vasya.pupkin@example.com
```

Тогда для получения почты через прокси пришлось бы поменять настройки клиента на следующие:

```
pop3-сервер: 192.168.0.1
пользователь: vasya.pupkin@example.com@mail.example.com
```

Что же касается отправки почты, то разработчик прокси-сервера рекомендует воспользоваться для этого портмаппингом:

```
tcppm -i192.168.0.1 25 smtp.provider.ru 25
```

То есть, мы просто все запросы по порту 25, по которому, собственно, и отправляются почтовые сообщения, перекидываем на 25-ый же порт почтового сервера провайдера.

В настройках вашего почтового клиента в этом случае для упомянутого выше ящика в общем случае поменяется запись только для SMTP-сервера:

```
SMTP-сервер: 192.168.0.1
```

Должен заметить, что ловкий прием с портмаппингом можно было бы использовать и для получения почты. В этом случае используется команда:

```
tcppm -i192.168.0.1 2110 pop.mail.ru 110
```

Есть в этом варианте одна неприятная сторона - если пользователи будут пользоваться ящиками на разных почтовых серверах, благо что их много развелось (yandex, rambler,newmail...), то для каждого почтового сервера придется завести новую запись портмаппинга в нашем конфиге, к примеру:

```
tcppm -i192.168.0.1 3110 pop.yandex.ru 110
```

И дополнительно надо будет поменять порт на 2110 и не забыть дать пользователям разрешение на этот нестандартный порт:

```
allow * 192.168.0.2,192.168.0.3 * 2110,25,110 * 1-5 00:00:00-20:00:00
```

Админка

И наконец, рассмотрим важную составляющую прокси-сервера - WEB-интерфейс администрирования. Для доступа к нему надо запустить команду

```
admin
```

- Службу admin можно так-же как и любую другую запускать с параметром **-p80** с указанием порта на котором необходимо запустить ваш сервис

Для доступа к интерфейсу пропишите в адресной строке браузера следующий адрес: http://192.168.0.1:80, не забыв, конечно, дать разрешение пользователю на использование стандартного для службы admin порта 80. В открывшемся окне можно будет получить доступ к информации о максимально допустимом трафике и его текущем значении, посмотреть конфиг целиком и даже перезапустить прокси-сервер. Поэтому разрешения на эту службу раздавайте крайне внимательно! Впрочем, у службы admin есть ключ **-s**, который не дает делать пользователю ничего недозволенного. Считаю своим долгом упомянуть о команде writable, которая, будучи размещена в самом начале конфига, дает возможность не только читать конфиг через WEB-интерфейс, но и менять его! Автор программы предлагает дважды подумать, прежде чем включать эту опцию.

Примечание

Хочу прояснить на примерах еще один момент, который я сам понял не сразу и понимание которого, возможно, может вызвать проблемы и у вас. Это использование авторизации auth strong. Предположим, что нам хочется, что доступ к WEB-администрированию был доступен только с компьютера 192.168.0.4, причем даже в этом случае запрашивался пароль доступа. В этом случае соответствующая секция конфига будет выглядеть так:

```
flush
auth strong
allow Administrator 192.168.0.4 * 80 * 1-5 00:00:00-20:00:00
admin
```

И разумеется, прежде чем пользоваться авторизацией strong, надо завести пользователя, в нашем случае это Administrator:

```
users Administrator:CL:password
```

Теперь при попытке доступа на страницу WEB-администрирования с компьютера Сидорова будет запрашиваться пароль, а на других компьютера доступ будет полностью закрыт. Если же третья строчка в приведенной выше секции конфига будет выглядеть так:

```
allow Administrator * * 80 * 1-5 00:00:00-20:00:00
```

То доступ к администрированию можно будет получить с любого компьютера при предъявлении пароля для пользователя Administrator.

Вы, наверное, заметили новую команду **flush**. Мы ее применили для сброса заданного выше списка доступа, чтобы можно было изменить разрешения доступа для пользователей. Замечу, что сброс не действует на ограничение скорости, которое действует на все службы. И не забываем после его примения задавать способ авторизации.

Конфиг файл

```
nserver 8.8.8.8
nserver 8.8.4.4

nscache 65536
timeouts 1 5 30 60 180 1800 15 60
```

```
external 192.168.0.0
internal 192.168.0.0

daemon

auth strong
users 3err0:CL:password



nobandlimin * * * 110
bandlimin 24000 * 192.168.0.2,192.168.0.3
bandlimin 48000 * 192.168.0.4
bandlimout 24000 *

###################################################################
# РАЗДАЕМ WEB
###################################################################

auth iponly
allow * 192.168.0.4 * * * 1-5 09:00:00-18:00:00
allow * 192.168.0.2,192.168.0.3 * 80,443 * 1-7 00:00:00-23:59:59
proxy -n -p3128

###################################################################
# РАЗДАЕМ ПОЧТУ
###################################################################

flush
auth iponly
deny * 192.168.200.4 * 110 * 1-5 18:00:00-23:59:59,00:00:00-08:00:00
allow * 192.168.0.2,192.168.0.3,192.168.0.4 * 25,110 * 1-5 00:00:00-20:00:00
tcppm -i192.168.0.1 25 smtp.provider.ru 25
pop3p

###################################################################
# РАЗДАЕМ FTP
###################################################################

flush
auth iponly
```

```
allow * 192.168.0.2,192.168.0.3 * 2110,25,110 * 1-5 00:00:00-20:00:00
ftppr
socks

#####################################################################
# АДМИНИСТРАТИВНЫЙ ДОСТУП
#####################################################################

flush
auth iponly
allow 3err0 192.168.0.4 * 3129 * 1-5 00:00:00-20:00:00
admin -s -p3129

end
```

В случае получения почты через портмаппинг секция конфига, идущая после коментария # ПОЧТА выглядела бы немного по другому:

```
#####################################################################
# РАЗДАЕМ ПОЧТУ
#####################################################################

flush
auth iponly
deny * 192.168.200.4 * 110 * 1-5 18:00:00-23:59:59,00:00:00-08:00:00
allow * 192.168.0.2,192.168.0.3,192.168.0.4 * 25,110 * 1-5 00:00:00-20:00:00
tcppm -i192.168.0.1 25 smtp.provider.ru 25
tcppm -i192.168.0.1 2110 pop.mail.ru 110
tcppm -i192.168.0.1 3110 pop.yandex.ru 110
```

Revision #1
Created 9 April 2023 03:11:09 by 3err0
Updated 9 April 2023 06:14:53 by 3err0