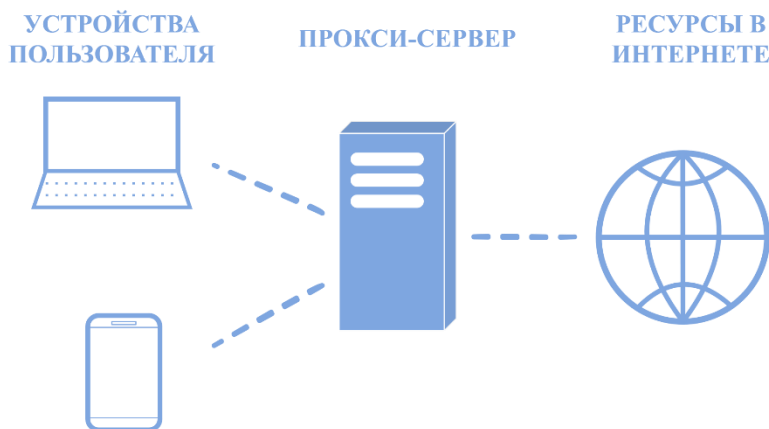


Практическая работа № 13. Настройка прокси-сервера SQUID

0. Теоретическая информация по прокси-серверу SQUID.

Прокси-сервер – промежуточный сервер в компьютерных сетях, выполняющий роль посредника между пользователем и целевым сервером, позволяющий клиентам как выполнять косвенные запросы к другим сетевым службам, так и получать ответы.



Сначала клиент подключается к прокси-серверу и запрашивает какой-либо ресурс (например e-mail), расположенный на другом сервере. Затем прокси-сервер либо подключается к указанному серверу и получает ресурс у него, либо возвращает ресурс из собственного кэша (в случаях, если прокси имеет свой кэш). В некоторых случаях запрос клиента или ответ сервера может быть изменён прокси-сервером в определённых целях.

Прокси-сервер позволяет защищать компьютер клиента от некоторых сетевых атак и помогает сохранять анонимность клиента, но также может использоваться мошенниками для скрытия адреса сайта, уличённого в мошенничестве, изменения содержимого целевого сайта (подмена), а также перехвата запросов самого пользователя.

Чаще всего прокси-серверы применяются для следующих целей:

- обеспечение доступа компьютеров локальной сети к сети Интернет;
- кэширование данных: если часто происходят обращения к одним и тем же внешним ресурсам для снижения нагрузки на канал во внешнюю сеть и ускорения получения клиентом запрошенной информации;
- сжатие данных: прокси-сервер загружает информацию из Интернета и передаёт информацию конечному пользователю в сжатом виде для экономии внешнего сетевого трафика клиента или внутреннего – организации, в которой установлен прокси-сервер;
- защита локальной сети от внешнего доступа: например, можно настроить прокси-сервер так, что локальные компьютеры будут обращаться к внешним ресурсам только через него, а внешние компьютеры не смогут обращаться к локальным вообще (они «видят» только прокси-сервер);
- ограничение доступа из локальной сети к внешней: например, можно запрещать доступ к определённым веб-сайтам, ограничивать использование интернета каким-то локальным пользователям, устанавливать квоты на трафик или полосу пропускания, фильтровать рекламу и вирусы;

– анонимизация доступа к различным ресурсам: прокси-сервер может скрывать сведения об источнике запроса или пользователе. В таком случае целевой сервер видит лишь информацию о прокси-сервере, например IP-адрес, но не имеет возможности определить истинный источник запроса; существуют также искажающие прокси-серверы, которые передают целевому серверу ложную информацию об истинном пользователе;

– обход ограничений доступа: используется, например, пользователями стран, где доступ к некоторым ресурсам ограничен законодательно и фильтруется.

Прокси-сервер, к которому может получить доступ любой пользователь сети интернет, называется открытым.

SQUID – кэширующий HTTP/FTP прокси-сервер

1. Установка, первоначальная настройка и диагностика SQUID

1.1. Установите SQUID на server1.

```
sudo apt install squid
```

1.2. Сохранить исходный вариант конфигурационного файла SQUID.

```
cd /etc/squid  
sudo cp squid.conf squid.conf.orig
```

1.3. На основе исходного конфигурационного файла SQUID создайте новый конфигурационный файл, исключив комментарии и пустые строки, содержащиеся в исходном файле.

```
sudo -i  
cd /etc/squid  
cat squid.conf.orig | grep -v '^#' | grep -v '^$' > squid.conf  
exit
```

1.4. Создайте в конфигурационном файле /etc/squid/squid.conf (в самом верху) список управления доступом типа src с именем localnet, добавив в этот список управления доступом все хосты из сети 192.168.1.0/24.

```
acl localnet src 192.168.1.0/24
```

1.5. Включите в /etc/squid/squid.conf правило, разрешающее хостам, определенным в списке управление доступом localnet, обращаться к прокси-серверу.

Примечание: правило должно быть размещено до запрещающего правила `http_access deny all`. Можно указать данное правило второй строчкой.

```
http_access allow localnet
```

1.6. Закомментируйте правило `acl localnet src 192.168.0.0/16`. Результат должен выглядеть следующим образом:

```
acl localnet src 192.168.1.0/24
http_access allow localnet
acl localnet src 0.0.0.1-0.255.255.255 # RFC 1122 "this" network (LAN)
acl localnet src 10.0.0.0/8 # RFC 1918 local private network $
acl localnet src 100.64.0.0/10 # RFC 6598 shared address space ($
acl localnet src 169.254.0.0/16 # RFC 3927 link-local (directly p$
acl localnet src 172.16.0.0/12 # RFC 1918 local private network $
#acl localnet src 192.168.0.0/16 # RFC 1918 local private $
acl localnet src fc00::/7 # RFC 4193 local private network $
acl localnet src fe80::/10 # RFC 4291 link-local (directly p$
acl SSL_ports port 443
acl Safe_ports port 80 # http
acl Safe_ports port 21 # ftp
acl Safe_ports port 443 # https
```

1.7. Выполните команду, которая позволяет пересчитать измененный конфигурационный файл.

```
sudo squid -k reconfigure
```

1.8. Проверьте доступность сервера SQUID и возможность доступа к Интернет-ресурсам на VM `server1` (используйте браузер Firefox). В браузере нажмите на меню и перейдите в Настройки. В поиске введите слово Прокси. Выполните следующую конфигурацию:

Параметры соединения

✕

Настройка прокси для доступа в Интернет

☐ Без прокси

☐ Автоматически определять настройки прокси для этой сети

☐ Использовать системные настройки прокси

☒ Ручная настройка прокси

HTTP прокси

server1.astra.test

Порт

3128

☒ Также использовать этот прокси для HTTPS

HTTPS прокси

server1.astra.test

Порт

3128

Справка

Отмена

OK

1.9. Для проверки работоспособности прокси-сервера можно, например, попробовать перейти на сайт `astralinux.ru`.

1.10. Просмотрите журнал для SQUID.

```
sudo journalctl -u squid.service
```

1.11. На client1 в файл /etc/hosts внесите следующую конфигурацию:

```
192.168.1.10 server1.astra.test server1
```

1.12. Проверьте доступность сервера SQUID и возможность доступа к Интернет-ресурсам на VM client1 (задайте аналогичные настройки для браузера).

2. Настройка доступа в веб-ресурсам

2.1. Создадим список управления доступом типа dstdomain, который включает домен rutube.ru. Внесите дальнейшую конфигурацию в самый верх файла /etc/squid/squid.conf.

```
acl rutube_site dstdomain .rutube.ru
```

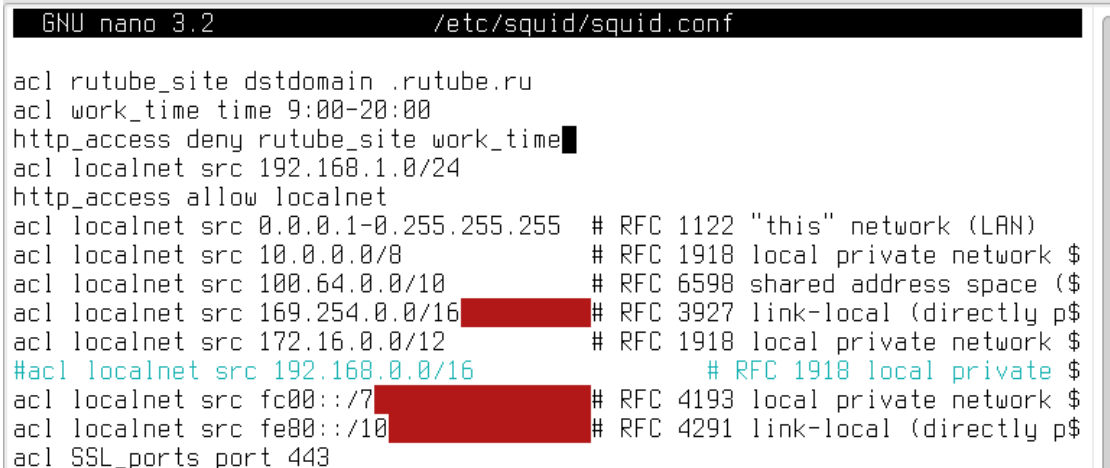
2.2. Создадим список управления доступом типа time, который определяет рабочее время.

```
acl work_time time 9:00-20:00
```

2.3. Добавьте правило, которое запрещает доступ к сайтам в домене rutube.ru в рабочее время, указав одновременно оба списка управления доступом.

```
http_access deny rutube_site work_time
```

Результат должен выглядеть следующим образом:



```
GNU nano 3.2 /etc/squid/squid.conf
acl rutube_site dstdomain .rutube.ru
acl work_time time 9:00-20:00
http_access deny rutube_site work_time
acl localnet src 192.168.1.0/24
http_access allow localnet
acl localnet src 0.0.0.1-0.255.255.255 # RFC 1122 "this" network (LAN)
acl localnet src 10.0.0.0/8 # RFC 1918 local private network $
acl localnet src 100.64.0.0/10 # RFC 6598 shared address space ($
acl localnet src 169.254.0.0/16 # RFC 3927 link-local (directly p$
acl localnet src 172.16.0.0/12 # RFC 1918 local private network $
#acl localnet src 192.168.0.0/16 # RFC 1918 local private $
acl localnet src fc00::/7 # RFC 4193 local private network $
acl localnet src fe80::/10 # RFC 4291 link-local (directly p$
acl SSL_ports port 443
```

1.4. Выполните пересчитывание конфигурационного файла сервером SQUID.

```
sudo squid -k reconfigure
```

1.5. Попробуйте открыть сайт rutube.ru. Получилось ли открыть данный сайт?

3.3. Базовая аутентификация NCSA

3.1. Установите программный пакет `apache2-utils` на `server1`, если он еще не был установлен.

```
sudo apt install apache2-utils
```

3.2. Командой `htpasswd` создайте пользователя `iuser`.

```
sudo htpasswd -c /etc/squid/passwd iuser
```

Примечание: используйте параметр `-c` имя_файла_с_паролями, когда первый раз выполняете команду `htpasswd`. Пароль может быть любым на ваше усмотрение.

3.3. Установите владельцев `proxy:proxy` и права `640` на файл с паролями.

```
sudo chown proxy:proxy /etc/squid/passwd
```

```
sudo chmod 640 /etc/squid/passwd
```

3.4. Внесите изменения в конфигурационный файл SQUID `/etc/squid/squid.conf` (сразу после конфигурации из пункта 2):

1) Настройте схему аутентификации NCSA с помощью директивы `auth_param`.
Записывается в одну строку:

```
auth_param basic program /usr/lib/squid/basic_ncsa_auth  
/etc/squid/passwd
```

2) Определите список управления доступом типа `proxy_auth` с именем `authusers`, который содержит всех валидных пользователей.

```
acl authusers proxy_auth REQUIRED
```

3) Добавьте правило, запрещающее доступ пользователям, не указанным в списке управления доступом `authusers`, и правило, разрешающее доступ валидным пользователям.

Примечание: правила должны быть размещены до последнего правила, запрещающего доступ всем, и до правила `localnet`.

```
http_access deny !authusers
```

```
http_access allow authusers
```

Результат должен выглядеть следующим образом:

```
GNU nano 3.2 /etc/squid/squid.conf Изменён
acl rutube_site dstdomain .rutube.ru
acl work_time time 9:00-20:00
http_access deny rutube_site work_time
auth_param basic program /usr/lib/squid/basic_ncsa_auth /etc/squid/passwd
acl authusers proxy_auth REQUIRED
http_access deny !authusers
http_access allow authusers
acl localnet src 192.168.1.0/24
http_access allow localnet
acl localnet src 0.0.0.1-0.255.255.255 # RFC 1122 "this" network (LAN)
acl localnet src 10.0.0.0/8 # RFC 1918 local private network $
acl localnet src 100.64.0.0/10 # RFC 6598 shared address space ($
acl localnet src 169.254.0.0/16 # RFC 3927 link-local (directly p$
acl localnet src 172.16.0.0/12 # RFC 1918 local private network $
```

3.5. Выполните перечитывание конфигурационного файла сервером SQUID.

```
sudo squid -k reconfigure
```

3.6. Проверьте, работает ли аутентификация для SQUID с VM server1 и client1.
Для этого заново откройте браузер и зайдите на любой сайт.