Автор: lissyara.
Оригинал: http://www.lissyara.su/archive/squid_old/

Прокси-сервер - это программа которая выполняет роль "прокладки" между браузером пользователя и WWW сервером. Через него проходят все запросы пользователя по протоколу http и ответы серверов пользователю. Он может фильтровать проходящий траффик по тем или иным признакам, а так же разграничивать доступ к интернету по протоколу http (в случае если используется непрозрачный прокси-сервер).
SQUID - пожалуй, самый лучший прокси под UNIX платформу. Есть сборки и для win32, но, на мой взгляд, это уже совсем не то :) Очень богатая функциональность:
— Поддержка протоколов HTTP, FTP, SSL, HTCP, CAPR
— Каскадирование серверов
— возможность прозраного проксирования
— поддержка протокола SNMP
— кэширование DNS-запросов
Собираем, операционка - FreeBSD4.11. Обновляем дерево портов и приступаем:/root/>cd /usr/ports/www/squid
/usr/ports/www/squid/>make && make install && make clean

Появляется синенькое окошко с кучей опций. Я выбрал:
SQUID_UNDERSCORES - разрешил запрещённый символ подчёркивания(_) в именах - мало ли идиотов в интернете...
SQUID_CHECK_HOSTNAMES - пусть проверяет имена.
SQUID_RCNG - стартовый скрипт squid
Он качает много-много патчей и собирается (впрочем, если выпустят новую версию - то патчей первое время не будет :)). После чего топаем в /usr/local/etc/squid и редактируем squid.conf до такого состояния (все настройки даны для прозрачного прокси-сервера, у "непрозрачного" будут отсутствовать пункты httpd_accel_*):http_port 3128
icp_port 0
hierarchy_stoplist cgi-bin ?
acl QUERY urlpath_regex cgi-bin ?
no_cache deny QUERY
cache_mem 128 MB
maximum_object_size 8092 KB
maximum_object_size_in_memory 512 KB
cache_dir ufs /usr/local/squid/cache 2048 64 256
cache_access_log /var/log/squid/access.log
cache_log /var/log/squid/cache.log
cache_store_log /var/log/squid/store.log
cache_mgr admin@my_domain.ru
visible_hostname mail.my_domain.ru

```
tcp_outgoing_address 222.222.222.222
refresh_pattern ^ftp:           1440    20%     10080
refresh_pattern ^gopher:        1440    0%      1440
refresh_pattern .               0       20%     4320
redirect_program /usr/local/etc/squid/redirector.pl
redirect_children 10

acl     all             src             0.0.0.0/0.0.0.0
acl     allowed_sites   dstdomain
"/usr/local/my_doc_smb/squid/allowed_sites.conf"
acl     limited_IP      src
"/usr/local/my_doc_smb/squid/limited_IP.conf"
acl     localhost       src             127.0.0.0/8
acl     our_networks    src             192.168.0.0/24
#acl    denied_sites    dstdomain
#"/usr/local/my_doc_smb/squid/denied_ext.conf"
#http_access    deny    denied_sites
http_access     allow   allowed_sites
http_access     deny    limited_IP
http_access     allow   our_networks
http_access     allow   localhost
http_access     deny    all

httpd_accel_host virtual
httpd_accel_port 80
httpd_accel_uses_host_header on

coredump_dir /usr/local/squid/cache
pid_filename /usr/local/squid/logs/squid.pid
```

Это - самый минимум конфигурации, для прозрачного проксирования, и списков людей, которым можно посещать лишь малое количество сайтов (сайты описаны в файле allowed_sites.conf, а ip компов в файле limited_IP.conf), и списка сайтов на которые ходить нельзя никому (denied_ext.conf). Файлы выглядят примерно так:/usr/local/my_doc_smb/squid/allowed_sites.confwww.yandex.ru
mail.yandex.ru
www.ya.ru
www.mail.ru

/usr/local/my_doc_smb/squid/limited_IP.conf192.168.0.8
192.168.0.56
192.168.0.89

/usr/local/my_doc_smb/squid/denied_ext.confwww.sex.com
www.tetki.ru
www.soska.ru
www.porewo.com

По остальным цифирькам:
cache_mem - сколько памяти под кэш потратит (реально в 2.5 раза больше зажрёт. Любит он оперативку. У меня в точно такой конфигурации занимает 297 мегов памяти, примерно через день - когда наберёт объектов в память)
maximum_object_size - максимальный размер объекта сохраняемый на диск (частенько неслушается и сохраняет обекты раза в 2-3 большие)
maximum_object_size_in_memory - максимальный размер объекта хранимого в оперативке
cache_dir - директория для кэша. Должна существовать и юзер от которого работает сквид должен иметь право писать в неё. Там же - ufs - тип файловой системы на которой расположена папка кэша, 2048 - максимальный размер кэша, 64 - число директорий первого уровня
256 - число директорий второго уровня (на директориях экономить не советую, сам столкнулся - кончились папки, но лимит по размеру кэша ещё не был достигнут, инет в итоге работает, но жутко тормозит. Как на модеме хреновеньком.... На 10-ти мегабитной-то линии....)
cache_access_log - местоположение файла логов доступа пользователей к инету - кто, куда, сколько.
cache_log - лог собственно сквида - результаты запусков-остановок, результаты работы с кэшем.
cache_store_log - лог что сохранено в кэше на диске
cache_mgr - е-майл администратора, выводится при ошибках или если доступ к странице запрещён.
visible_hostname - видимое снаружи имя хоста
tcp_outgoing_address - внешний адрес сервера
redirect_program - программа редиректор (занимается анализом запрашиваемых URL и может производить с ними какие-то действия, у меня раньше, таким макаром был прикручен антивирус на проверку входящего http-траффика, а щас висит скрипт срезающий порнуху, от антивиря пришлось отказаться - примерно 30-40% лишнего траффика было, т.к. сайты нынче в основном динамические...)
redirect_children - число процессов программы-редиректора
Затем идут ACL-ы, разрешающие или запрещающие пользование http и поддержка прозрачного проксирования. ACL denied_sites закомментирован, можно пользоваться им самим, но я предпочитаю натравить на него внешнюю программу-редиректор, тогда можно будет вносить в него не сайты целиком, а ключевые слова по которым будет резаться URL - типа sex, deffki, porewo.... Если в запросе будет такое слово (неважно, в середине пути, в имени сервера, или названии файла) то не такой адрес пользователя не пустят. Можно таким макаром зарезать всю графику, например, написать jpg, jpeg, png, gif и всё - графики больше нет :) Можно резать флэши, файлы с нежелательными расширениями, да что угодно....

Учтите, строки типа acl allowed_sites dstdomain "/usr/local/my_doc_smb/squid/allowed_sites.conf" - это одна строка, просто у меня в листинге конфига не влезло и я её так перенёс.
coredump_dir - директория куда будет писаться дамп программы в случае критической ошибки и последующего "выпадания в корку"
pid_filename - имя файла где хранится идентификатор запущенного squid`a
Файлы с запретами и разрешениями (/usr/local/my_doc_smb/squid/allowed_sites.conf, /usr/local/my_doc_smb/squid/limited_IP.conf, /usr/local/my_doc_smb/squid/denied_ext.conf) так странно лежат по причине, что из локалки у меня к ним открыт доступ по самбе, просто мне их так удобней редактировать :)
Создаём файлы, папки и запускаем squid:/usr/local/etc/squid/>mkdir -p my_doc_smb/squid
/usr/local/etc/squid/>mkdir -p mkdir /var/log/squid
/usr/local/etc/squid/>touch /usr/local/my_doc_smb/squid/allowed_sites.conf
/usr/local/etc/squid/>touch /usr/local/my_doc_smb/squid/limited_IP.conf
/usr/local/etc/squid/>touch /usr/local/my_doc_smb/squid/denied_ext.conf
/usr/local/etc/squid/>touch /usr/local/etc/squid/redirector.pl
/usr/local/etc/squid/>chmod +x redirector.pl
/usr/local/etc/squid/>chown -R squid:wheel /var/log/squid/
/usr/local/etc/squid/>echo 'squid_enable="YES"' >> /etc/rc.conf
/usr/local/etc/squid/>squid -z
2005/09/20 14:51:04| aclParseAclLine: WARNING: empty ACL: acl
2005/09/20 14:51:04| aclParseAclLine: WARNING: empty ACL: acl
2005/09/20 14:51:04| aclParseAclLine: WARNING: empty ACL: acl
2005/09/20 14:51:04| Creating Swap Directories

Всё нормально, не считая ругани на пустые файлы ACL. На это можно не обращать внимания, или забить туда какие-ньть адреса. Тогда он ругаться перестанет.
Вот содержимое файла /usr/local/etc/squid/redirector.pl#!/usr/bin/perl

```
$0 = 'redirect' ;
$| = 1 ;

open (IN_FILE, "/usr/local/my_doc_smb/squid/denied_ext.conf") || die $!;
my @tmp_data = <IN_FILE>;
chomp @tmp_data;
push @banners, map { qr /Q$_E/ }  grep { ! /^s*$/ } @tmp_data;
close IN_FILE;

while (<>) {
 ($url, $who, $ident, $method) = /^(S+) (S+) (S+) (S+)$/ ;
 $url = 'http://mail.my_domain.ru/zaglushka.jpg'
 if grep ($url=~/$_/i, @banners) ;
 print "$url $who $ident $methodn" ;
}
```

Это простенький перловый скрипт, перебирающий переданный ему URL на соответствие шаблонам лежащим в файле denied_ext.conf, и если они подходят, то вместо этого УРЛа он отдаёт другой - http://mail.my_domain.ru/zaglushka.jpg по которому лежит мелкий рисунок серого цвета (белый неудобно - у меня до кучи он баннеры режет и белые дыры на страницах не смотрятся вообще...)
Ну, а теперь пристегните ремни, сейчас мы попробуем взлететь со всем этим хозяйством (копирайт из старого-старого анекдота):/usr/local/etc/squid/>../rc.d/squid.sh start

```
Starting squid.
2005/09/20 15:28:35| aclParseAclLine: WARNING: empty ACL: acl allowed_sites
2005/09/20 15:28:35| aclParseAclLine: WARNING: empty ACL: acl limited_IP
2005/09/20 15:28:35| aclParseAclLine: WARNING: empty ACL: acl denied_sites
/usr/local/etc/squid/>ps -ax | grep squid
73072  ??  Ss     0:00.00 /usr/local/sbin/squid -D
73074  ??  D      0:04.75 (squid) -D (squid)
73087  p0  D+     0:00.00 grep squid
/usr/local/etc/squid/>ps -ax | grep perl
73075  ??  Is     0:00.03 redirect (perl)
73076  ??  Is     0:00.03 redirect (perl)
73077  ??  Is     0:00.03 redirect (perl)
73078  ??  Is     0:00.03 redirect (perl)
/usr/local/etc/squid/>sockstat | grep perl
squid    squid    73074   14 tcp4   127.0.0.1:3248       127.0.0.1:4480
squid    squid    73074   15 tcp4   127.0.0.1:3922       127.0.0.1:2536
squid    squid    73074   16 tcp4   127.0.0.1:2393       127.0.0.1:3906
squid    squid    73074   21 tcp4   *:3128               *:*
squid    squid    73074   22 udp4   *:3401               *:*
squid    squid    73072    4 dgram  syslogd[83]:3
squid    perl     73076    0 tcp4   127.0.0.1:4371       127.0.0.1:1506
squid    perl     73076    1 tcp4   127.0.0.1:4371       127.0.0.1:1506
squid    perl     73075    0 tcp4   127.0.0.1:1596       127.0.0.1:2215
squid    perl     73075    1 tcp4   127.0.0.1:1596       127.0.0.1:2215
```

Всё нормально. Добавляем правило в файрволл (ipfw), до divert natd:fwd 127.0.0.1,3128 tcp from 192.168.0.0/24 to any 80 via fxp0

Где fxp0 - внешний интерфейс, и всё, можно работать. Если на ходу возникает необходимость переконфигурить squid то перезапускать необязательно, можно дать команду/usr/local/etc/squid/>squid -k reconfigure

или/usr/local/etc/squid/>killall -1 squid

этого вполне достаточно.
 Заполняйте файлы с разрешениями-запретами, и пользуйтесь. Клиентов настраивать не нужно - достаточно указать шлюзом по-умолчанию машину со squid - всё остальное сделает ipfw - перекинет пакеты на squid.