



Министерство науки и высшего образования Российской Федерации
Калужский филиал
федерального государственного бюджетного
образовательного учреждения высшего образования
«Московский государственный технический университет имени Н.Э. Баумана
(национальный исследовательский университет)»
(КФ МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ ИУК «Информатика и управление»

КАФЕДРА ИУК4 «Программное обеспечение ЭВМ, информационные технологии»

ЛАБОРАТОРНАЯ РАБОТА №5

«Использование системы в качестве прокси-сервера»

ДИСЦИПЛИНА: «Операционные системы»

Выполнил: студент гр. ИУК4-62Б

_____ (Калашников А.С.)
(Подпись) (Ф.И.О.)

Проверил:

_____ (Красавин Е.В.)
(Подпись) (Ф.И.О.)

Дата сдачи (защиты):

Результаты сдачи (защиты):

- Балльная оценка:

- Оценка:

Цель: получение практических навыки по настройке прокси-сервера Squid под ОС FreeBSD.

Задачи:

1. Научиться получать и устанавливать прокси-сервер Squid под ОС FreeBSD.
2. Научиться настраивать и управлять прокси-сервером Squid под ОС FreeBSD.

Задание:

Под руководством преподавателя самостоятельно настроить прокси-сервер Squid и проверить его работоспособность.

1. Скачать дистрибутив из интернета и скинуть его на флешку.
2. Смонтировать флешку.
3. Создать директорию, в которую будем копировать дистрибутив.
4. Скопировать архив в созданную директорию и распаковать его.
5. Скомпилировать и установить дистрибутив.
6. Настроить файл конфигурации squid.conf согласно примеру.
7. Запустить прокси-сервер.
8. Изменить права у файла squid.sh.
9. Проверить работоспособность прокси-сервера.
10. Завершить работу FreeBSD.

Ответить на контрольные вопросы и подготовить отчет

Описание процесса выполнения лабораторной работы:

```
checking for SSL_CTX_get0_certificate in -lssl... yes
checking whether SSL_CTX_new and similar openssl API functions require 'const SSL_METHOD *'... yes
checking whether SSL_get_new_ex_index() dup callback accepts 'const CRYPTO_EX_DATA *'... yes
checking whether SSL_CTX_sess_set_get_cb() callback accepts a const ID argument'... yes
checking "whether X509_get0_signature() accepts const parameters"... yes
checking whether the TXT_DB use OPENSSL_PSTRING data member... yes
checking whether the squid workaround for buggy versions of sk_OPENSSL_PSTRING_value should used... no
checking whether the workaround for OpenSSL IMPLEMENT_LHASH_ macros should used... yes
checking whether hello message can be overwritten in SSL struct... no
configure: OpenSSL library support: yes -L/usr/lib -lcrypto -lssl
checking for krb5-config... /usr/bin/krb5-config
checking for krb5-config... (cached) /usr/bin/krb5-config
checking for LIB_KRB5... no
configure: Use krb5-config to get CXXFLAGS and LIBS
configure: Try to find Kerberos headers in given path
checking for gssapi.h... yes
checking for gssapi/gssapi.h... yes
checking for gssapi/gssapi_krb5.h... ^Z
Suspended
root@root:/usr/ports/www/squid # make install clean
```

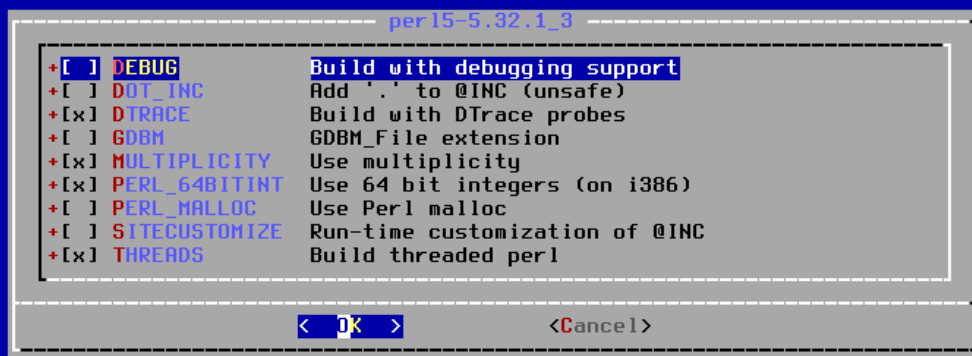


Рис.1. Установка дистрибутива SQUID из портов

```
o ascii code      ^x search        ^l undelete line ^n next line     ^v next page
^u end of file    ^a begin of line ^w delete word   ^b back 1 char  ^z next word
^t top of text    ^e end of line   ^r restore word  ^f forward char
^c command        ^d delete
ESC-Enter: exit
=====line 1 col 21 lines fr=====
#
# Recommended minimum configuration:
#
# Example rule allowing access from your local networks.
# Adapt to list your (internal) IP networks from where browsing
# should be allowed
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 (LAN)
acl localnet src 100.64.0.0/10        # RFC 6598 shared address space (CGN)
acl localnet src 169.254.0.0/16       # RFC 3927 link-local (directly plugged)
acl localnet src 172.16.0.0/12        # RFC 1918 local private network (LAN)
acl localnet src 192.168.0.0/16       # RFC 1918 local private network (LAN)
acl localnet src fc00::/7             # RFC 4193 local private network range
acl localnet src fe80::/10            # RFC 4291 link-local (directly plugged)

acl SSL_ports port 443
```

Рис.2. Настройка squid.conf согласно примеру

```

root@root:/usr/local/etc/squid # mkdir -p /usr/local/etc/squid/var/logs
root@root:/usr/local/etc/squid # ls
cachemgr.conf      icons              squid.conf.documented
cachemgr.conf.sample  mib.txt          squid.conf.sample
errorpage.css      mime.conf         var
errorpage.css.sample mime.conf.sample
errors             squid.conf
root@root:/usr/local/etc/squid #
root@root:/usr/local/etc/squid # chmod 777 /usr/local/etc/squid/var/logs
root@root:/usr/local/etc/squid # mkdir -p /usr/local/etc/squid/var/cache
root@root:/usr/local/etc/squid # chmod 777 /usr/local/etc/squid/var/cache

```

Рис.3. Создание разделов для хранения кэша и логов

```

root@root:/var/squid/cache # squid -z
root@root:/var/squid/cache # 2023/04/24 21:51:03 kid1l Set Current Directory to
/var/squid/cache
2023/04/24 21:51:03 kid1l Creating missing swap directories
2023/04/24 21:51:03 kid1l No cache_dir stores are configured.
2023/04/24 21:51:03l Removing PID file (/var/run/squid/squid.pid)

```

Рис.4. Создание структуры разделов для хранения кэша перед первым запуском

```

root@root:/usr/local/etc/rc.d # chmod 755 squid
root@root:/usr/local/etc/rc.d # cd ..
root@root:/usr/local/etc # cd squid/
root@root:/usr/local/etc/squid # ls
cachemgr.conf      icons              squid.conf.documented
cachemgr.conf.sample  mib.txt          squid.conf.sample
errorpage.css      mime.conf         var
errorpage.css.sample mime.conf.sample
errors             squid.conf
root@root:/usr/local/etc/squid # cd var/
root@root:/usr/local/etc/squid/var # ls
cache  logs
root@root:/usr/local/etc/squid/var # cd logs
root@root:/usr/local/etc/squid/var/logs # ls
root@root:/usr/local/etc/squid/var/logs # grep squid
^Z
Suspended
root@root:/usr/local/etc/squid/var/logs # ps aux | grep squid
root  937   0.0   0.0 12840   2320 v0  T   21:59   0:00.00 grep squid
root  950   0.0   0.0 12840   2320 v0  S+  22:01   0:00.00 grep squid

```

Рис.5. Изменение прав файла squid.sh – разрешение выполнения файла

```

root@root:/usr/local/etc/squid/var/logs # ps aux | grep squid
root  937   0.0   0.0 12840   2320 v0  T   21:59   0:00.00 grep squid
root  950   0.0   0.0 12840   2320 v0  S+  22:01   0:00.00 grep squid
root@root:/usr/local/etc/squid/var/logs # squid
root@root:/usr/local/etc/squid/var/logs # ps aux | grep squid
root  952   0.0   0.3 115200 15016  - Ss   22:02   0:00.00 squid
squid 954   0.0   0.8 216552 39016  - S    22:02   0:00.14 (squid-1) --kid squ
squid 955   0.0   0.1 19856   7244  - S    22:02   0:00.01 (logfile-daemon) /v
squid 956   0.0   0.2 20492   8000  - S    22:02   0:00.01 (pinger) (pinger)
root  937   0.0   0.0 12840   2320 v0  T   21:59   0:00.00 grep squid
root  958   0.0   0.0 12892   2316 v0  R+   22:02   0:00.00 grep squid
root@root:/usr/local/etc/squid/var/logs #
root@root:/usr/local/etc/squid/var/logs # service squid stop
Performing sanity check on squid configuration.
Configuration for squid passes.
Stopping squid.
Waiting for PIDS: 952.
root@root:/usr/local/etc/squid/var/logs # ps aux | grep squid
root  937   0.0   0.0 12840   2320 v0  T   21:59   0:00.00 grep squid
root  988   0.0   0.0 12840   2320 v0  S+   22:03   0:00.00 grep squid

```

Рис.8. Демонстрация завершения работы SQUID

Вывод: в ходе выполнения данной лабораторной работы были приобретены практические навыки по настройке прокси-сервера Squid под ОС FreeBSD.

Ответы на контрольные вопросы:

1. Дайте определение прокси-сервера.

Прокси-сервер (от англ. proxy — «представитель, уполномоченный») — служба в компьютерных сетях, позволяющая клиентам выполнять косвенные запросы к другим сетевым службам.

2. Укажите цели применения прокси-сервера.

Чаще всего прокси-серверы применяются для следующих целей:

- Обеспечение доступа с компьютеров локальной сети в Интернет.
- Кэширование данных.
- Сжатие данных.
- Защита локальной сети от внешнего доступа .
- Ограничение доступа из локальной сети к внешней.
- Анонимность доступа к различным ресурсам.

3. Опишите преимущества в работе в сети Интернет с использованием прокси.

- Кэширование данных: если часто происходят обращения к одним и тем же внешним ресурсам, то можно держать их 7 копию на прокси-сервере и выдавать по запросу, снижая тем самым нагрузку на канал во внешнюю сеть и ускоряя получение клиентом запрошенной информации.
- Сжатие данных: прокси-сервер загружает информацию из Интернета и передаёт информацию конечному пользователю в сжатом виде. Такие прокси-серверы используются в основном с целью экономии внешнего трафика.
- 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 запретить доступ к определённым веб-сайтам, ограничить использование интернета каким-то локальным пользователям, устанавливать квоты на трафик или полосу пропускания, фильтровать рекламу и вирусы.
- Анонимность 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.

4. Перечислите виды прокси.

- HTTP-прокси.
- FTP-прокси.
- HTTPS-прокси.
- Mapping-прокси.
- Socks-прокси.

5. Дайте определение HTTP-прокси и перечислите его возможности.

HTTP-прокси – самый распространенный. Он предназначен для организации работы браузеров и других программ, использующих протокол HTTP. Браузер передает прокси- серверу URL ресурса, прокси-сервер получает его с запрашиваемого веб-сервера (или с другого прокси-сервера) и отдает браузеру. У HTTP-прокси широкие возможности при выполнении запросов:

- Можно сохранять полученные файлы на диске сервера.
- Можно ограничивать доступ к ресурсам.
- Можно выдавать не тот ресурс, который запрашивается браузером.
- Можно ограничивать скорость работы для отдельных пользователей, групп или ресурсов.
- Ведутся журналы работы прокси.
- Можно маршрутизировать веб-запросы.

6. Дайте определение FTP-прокси.

FTP проху – это узкоспециализированный тип прокси серверов и предназначен он исключительно для работы с FTP (File Transfer Protocol) серверами.

7. Дайте определение HTTPS-прокси и укажите его отличия от HTTP-прокси.

HTTPS-прокси – фактически часть HTTP-прокси. S в названии означает «secure», т.е. безопасный.

Прокси-серверу HTTPS дается только команда «соединится с таким-то сервером», и после соединения прокси передает в обе стороны зашифрованный трафик, не имея возможности узнать подробности, т.к. в процессе шифрации/дешифрации прокси тоже участия не принимает – это делают клиентская программа и целевой сервер.

8. Дайте определение Mapping-прокси.

Mapping-прокси – способ заставить работать через прокси те программы, которые умеют работать с интернетом только напрямую. При настройке такого прокси администратор создает как бы «копию» целевого сервера, но доступную через один из портов прокси-сервера для всех клиентов локальной сети – устанавливает локальное «отображение» заданного сервера.

9. Дайте определение Socks-прокси.

Socks-прокси – протокол для прокси-сервера, позволяющий пропускать через прокси почти любой прикладной TCP- или UDP-протокол.

10. Опишите принцип кэширования.

Web-контент перемещается в некий сетевой кэш поближе к пользователям, нуждающимся в нем, вследствие чего уменьшается число участков маршрутизации или коммутации, через которые он должен пройти.

11. Опишите принцип сетевых кэшей и их назначение.

Сетевые кэши – это специальные промежуточные узлы в компьютерных сетях, которые хранят наиболее часто запрашиваемую информацию. Они позволяют ускорить доступ к данным и сократить трафик в сети.

Работа сетевых кэшей происходит следующим образом. Когда запрос на доступ к определенному ресурсу приходит к серверу, кэш проверяет, не содержится ли запрашиваемая информация в его памяти. Если информация имеется, то кэш передает ее напрямую пользователю, без обращения к серверу. Если информации в кэше нет, кэш отправляет запрос на сервер. После получения данных, он сохраняет их в своей памяти для следующих запросов. Назначение сетевых кэшей заключается в повышении производительности.

12. Дайте определение ICP и HTCP.

Протокол ICP (Internet Cache Protocol) – протокол, который позволяет осуществлять иерархическое соединение кэшей. Он определяет порядок обмена информацией между кэшами, находящимися в состоянии подчинения. ICP прежде всего используется в иерархии кэшей для поиска определенных объектов в братских кэшах.

Протокол HTCP (Hypertext caching protocol) – протокол, который используется для обнаружения кэшей HTTP и кэшированных данных, управления наборами кэшей HTTP и мониторинга активности кэша.

13. Укажите отличия кэш-сервера от прокси-сервера.

Кэш-серверы и прокси-серверы – не одно и то же. Кэширование по-прежнему остается одной из функций прокси-серверов. Однако повышение спроса на специализированное кэширование приводит к тому, что кэш-серверы все чаще выпускаются в качестве отдельных продуктов.

Кэш-сервер используется для хранения данных, которые часто запрашиваются в сети, в то время как прокси-сервер используется для обработки и пересылки запросов пользователей в сеть.

Прокси-сервер может использоваться для фильтрации трафика и блокирования доступа к нежелательным ресурсам, тогда как кэш-сервер обеспечивает только ускорение доступа к запрашиваемым данным.

Кэш-сервер может хранить большие объемы данных, которые могут быть запрошены несколько раз, в то время как прокси-сервер обычно не хранит данные, за исключением кратковременного кэширования в рамках запроса.

14. Опишите назначение кэш-сервера.

Кэш-серверы изучают активность, перехватывая запросы одним из двух способов: путем прозрачного кэширования или прокси-кэширования. Прозрачный кэш-сервер «просеивает» через себя весь проходящий трафик и поэтому не требует модификации установок конечного клиента. Он устанавливается обычно перед маршрутизатором, соединенным с интернетом.

15. Дайте определение прокси-кэш-сервера и опишите его концепцию.

Прокси-кэш - некое средство в прокси-сервере, которое кэширует поступающие Web-страницы на жестком диске. Если страница, запрашиваемая браузером, уже находится в прокси-кэше, то она отыскивается в нем, а не в интернете. Так случилось, что прокси-кэш-серверами называют практически все устройства кэширования, независимо от их расположения относительно потока информации, а «прозрачное» кэширование стало лишь одним из режимов работы прокси-кэш-сервера.

16. Опишите принцип прозрачного кэширования.

Прозрачный кэш-сервер «просеивает» через себя весь проходящий трафик и поэтому не требует модификации установок конечного клиента. Он устанавливается обычно перед маршрутизатором, соединенным с интернетом.

17. Перечислить архитектуры(модели) прокси-кэш-сервера и описать их суть.

- Прозрачный кэш-сервер «просеивает» через себя весь проходящий трафик и поэтому не требует модификации установок конечного клиента. Он устанавливается обычно перед маршрутизатором, соединенным с интернетом.
- Прямой прокси-кэш. При такой конфигурации запросы пользователей на своем пути к Web-серверу проходят через кэш. Если кэш содержит запрашиваемый документ, этот документ отправляется пользователю. В противном случае сервер работает как прокси, извлекая нужный контент из Web-сервера.
- Обратный прокси-кэш, или «серверный ускоритель». Кэш может быть также сконфигурирован как быстрый Web-сервер для ускорения более медленных традиционных Web-серверов. При этом документы, хранящиеся в кэше, обрабатываются с высокой скоростью, в то время как документы, не занесенные в кэш (обычно динамический контент) запрашиваются при необходимости из

исходных Web-серверов. Такая кэширующая система располагается перед одним или несколькими Web-серверами, перехватывая запросы и действуя наподобие прокси. Эти прокси-кэш-серверы могут размещаться по всей сети, формируя некую распределенную сеть сайтов для хостирования контента. Дополнительное достоинство данной схемы связано с возможностью балансировки нагрузки и динамического зеркалирования.

18. Перечислите фирмы, занимающиеся разработкой и производством прокси-кэш-сервером и охарактеризуйте их продукцию.

CacheFlow специализируется непосредственно на системах кэширования, для чего ею была разработана собственная ОС *CachOS*, оптимизирующая функции кэширования. В продуктах этой фирмы реализована технология *Object Pipelining*, которая позволяет организовать быстрый доступ к контенту с первого раза, ликвидируя значительную часть задержек на пути от Web-браузера клиента до удаленного Web-сервера провайдера.

Network Appliance - крупнейший поставщик NAS (Network Attached Storage), и, естественно, ее коньком являются технологии предоставления быстрого, надежного доступа по сети к большим объемам данных. Подходы, отработанные при проектировании устройств доступа, фирма перенесла и на свои прокси-кэш-серверы:

- отказоустойчивая высокопроизводительная архитектура (поддержка RAID, fibre channel, горячая замена блоков питания и вентиляторов);
- оптимизированная под RAID4 файловая система WAFL плюс кэширование запросов на запись в памяти типа NVRAM.

Cisco Systems. В продукции применена ОС *Cisco IOS*, оптимизированная для организации телекоммуникаций, а также технология *Cisco Network Caching*, которая минимизирует избыточный трафик, передаваемый по каналам WAN. Она позволяет повысить производительность сети за счет того, что большинство запросов к внешним ресурсам выполняется локально, а не за счет передачи этих запросов к удаленным серверным группам. Такое решение защищает внутреннюю сеть от неконтролируемых перегрузок в интернете или в корпоративной сети, что позволяет повысить качество предоставляемых услуг и доступность информации, хранящейся на внешних серверах.

Cobalt Networks. Эта компания поставляет недорогие кэширующие продукты для предприятий малого и среднего бизнеса.

Compaq Computer. В представленных в обзоре продуктах *Compaq*, образующих линейку для потребителей разных классов, используется ПО кэширования *Novell ICS Caching*. По утверждению компании используемая полоса уменьшается на величину до 30%, сокращение времени доставки контента пользователю (или провайдеру) или времени отклика сети достигает десятикратного, Разгрузка Web-серверов составляет до 80% запросов.

Intel. Для изделий этой корпорации характерна высокая отказоустойчивость. Имеется порт аварийного управления *Emergency Management Port* (EMP), позволяющий управлять устройством даже при отказе программного обеспечения или сети. Применяется архитектура *DataFlow*, которая обеспечивает одновременную буферизацию и передачу потоков данных для повышения пропускной способности и бесперебойного ввода/вывода.

19. Перечислите детали, на которые стоит обратить внимание при покупке прокси-кэш-сервера.

- Технические характеристики,
- Простота установки и использования,
- Возможность изменения оптимизируемой характеристики,
- Гибкость конфигурирования,
- Масштабируемость и надежность,
- Размеры, масса, энергопотребление.

20. Опишите принцип активного и пассивного кэширования и их отличия.

Современные кэши используют пассивное или активное кэширование. При пассивном кэшировании кэш-сервер проверяет свежесть контента. Обычно кэш-сервер посылает команду `get` (в HTTP) для запрашивания объекта от контент-сервера. В тех случаях, когда объект уже был сохранен, кэш-сервер использует модифицированную команду `get if`, в соответствии с которой объект скачивается, если он был изменен после последнего запроса. Затем кэш-сервер сравнивает даты изменения объекта, поступившего от сервера, и объекта, хранящегося в кэше, и направляет пользователю самый последний вариант.

При активном кэшировании улучшение характеристик достигается с использованием эвристических методов оценки срока жизни объекта. Сервер при этом проводит вычисления, используя такие данные, как дата занесения объекта в кэш, продолжительность его пребывания в кэше, IP-адрес источника и множество подобных сведений. При таком подходе не нужно проверять каждый запрос. Вместо этого кэш-сервер делает определенные предположения о времени жизни объекта, скажем два дня. В течение этого интервала все запросы объекта немедленно обслуживаются из кэша, однако по истечении этого срока кэш обновляет объект

21. Опишите принцип каскадной настройки прокси серверов.

Каскадная настройка прокси-серверов, также известная как проксирование через цепочку, является способом организации более надежной и эффективной работы прокси-серверов.

Кратко принцип каскадной настройки прокси серверов заключается в следующем:

1. Настройка первого прокси-сервера на перенаправление трафика на другой прокси-сервер.
2. Второй прокси-сервер, в свою очередь, перенаправляет трафик на следующий прокси-сервер.
3. Такой каскад продолжается, пока трафик не достигнет конечного сервера.
4. Каждый прокси-сервер в цепочке может выполнять свои функции, например, фильтрацию трафика или блокировку нежелательных сайтов.

Каскадная настройка прокси-серверов позволяет улучшить производительность и безопасность сети, так как она повышает надежность и устойчивость системы. Если один прокси-сервер не может обработать запрос, то остальные могут продолжить работу и предоставить пользователю доступ к ресурсу.