INF 460 – Network Design and Administration

Lecture 3

Proxy Server and Configuration

Nicholas Kiget
www.kiget.me.ke

Proxy servers

- A **proxy** server is a machine which acts as an intermediary between the computers of a local area network (sometimes using protocols other than TCP/IP) and the Internet

Proxy servers

- Most of the time the proxy server is used for the web, and when it is, it's an HTTP proxy. However, there can be proxy servers for every application protocol (FTP, etc.).

Proxy Server

- It is a specialized HTTP Server.
- Functions as a firewall.
  - Protects client computers from Hackers by limiting outside access to clients.
- Allows all clients connected to Web Proxy Server to access Internet from behind "firewall."
- Client computer(s) are allowed access past firewall with minimum effort and without compromising security.

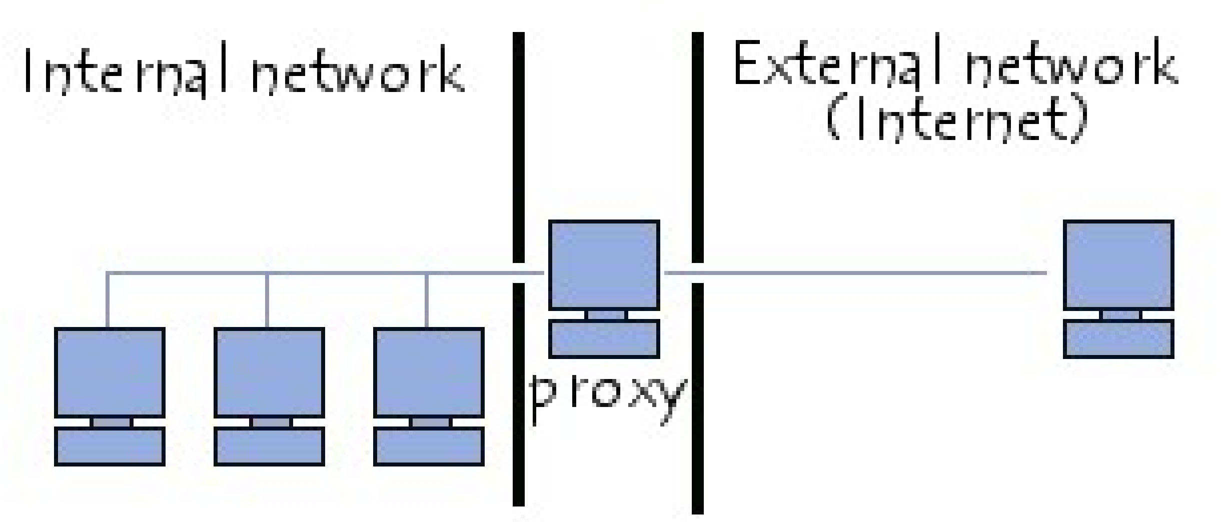
How Proxy Server Work

- Web Proxy Server listens for any request from clients.
- All requests are forwarded to remote internet servers outside firewall.
- Also listens for responses or request from outside the firewall (external servers) and sends to them to internal client computers.

How Proxy Server Work

- Usually, all clients with a subnet use the same proxy server.
- This makes it possible for the proxy server to cache documents that are requested by one or more clients (repeatedly).
- For clients using a web proxy server, it is as if they are getting responses directly from a remote server.

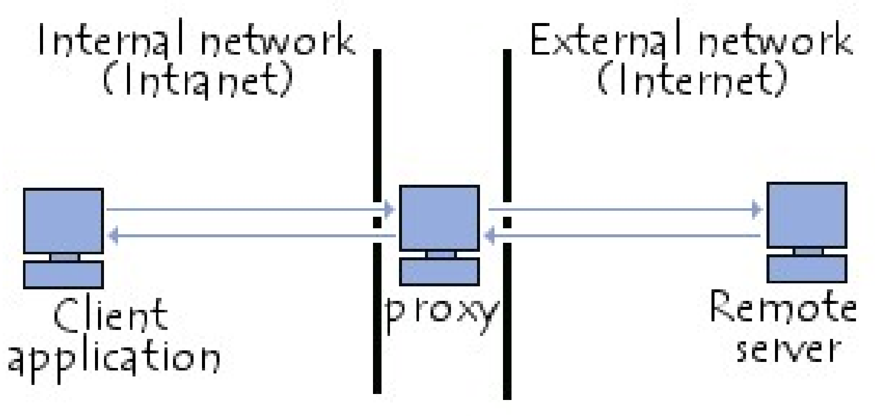
Proxy servers



The operating principle of a proxy server

- It is server which acts as a "proxy" for an application by making a request on the Internet in its stead.
- This way, whenever a user connects to the Internet using a client application configured to use a proxy server, the application will first connect to the proxy server and give it its request.
- The proxy server then connects to the server which the client application wants to connect to and sends that server the request.
- The server then gives its reply to the proxy, which then finally sends it to the application client

The operating principle of a proxy server



Caching

- Most proxies have a **cache**, the ability to keep pages commonly visited by users in memory (or "in cache"), so they can provide them as quickly as possible.

  cache" is used often in computer science to refer to a temporary data storage space (also called "buffer.")
- A proxy server with the ability to cache information is called a "**proxy-cache** server".
- The feature, implemented on some proxy servers, is used both to reduce Internet bandwidth use and to reduce document loading time for users.
- Nevertheless, to achieve this, the proxy must compare the data it stores in cached memory with the remote data on a regular basis, in order to ensure that the cached data is still valid.

Filtering

- Because of this, Internet connections can be filtered, by analysing both client requests and server replies.
- When filtering is done by comparing a client's request to a list of authorised requests, this is called *whitelisting*, and when it's done with a list of forbidden sites, it's called *blacklisting*.
- Finally, analysing server replies that comply with a list of criteria (such as keywords) is called *content filtering*.

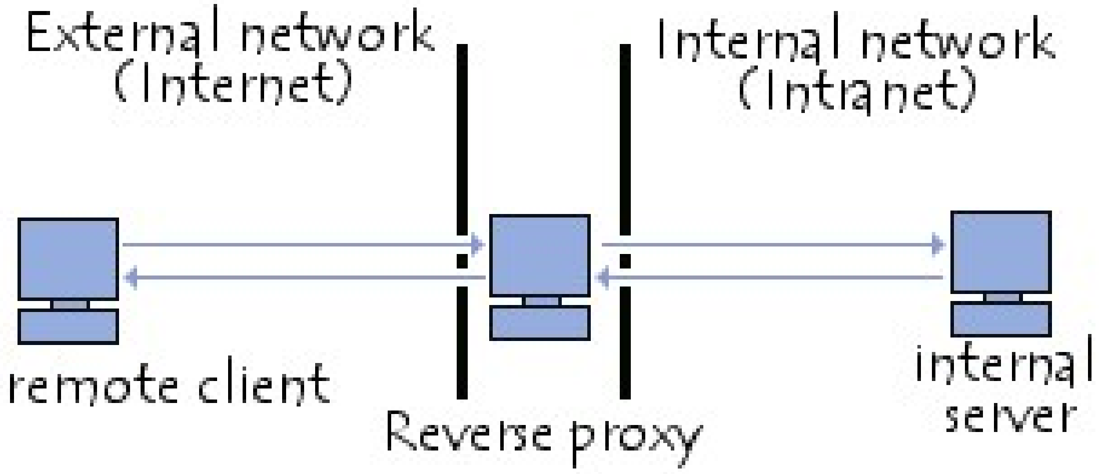
Authentication

- Proxy can sometimes be used to authenticate users, meaning to ask them to identify themselves, such as with a username and password.
- It is also easy to grant access to external resources only to individuals authorised to do so, and to record each use of external resources in log files.
- This type of mechanism, when implemented, obviously raises many issues related to individual liberties and personal rights.

Reverse-proxy servers

- A *reverse-proxy* is a "backwards" proxy-cache server; it's a proxy server that, rather than allowing internal users to access the Internet, lets Internet users indirectly access certain internal servers.

Reverse-proxy servers



Conclusion

- Both firewall and proxy server are used for net work security and facility
- Proxy server can be a part of firewall

Setting up a proxy server

- The most widely used proxy, without a doubt, is Squid, a free software program available for several platforms, including Windows and Linux.
- In Windows, there are several programs for setting up a local area network proxy server at a low cost:
- Wingate is the most common solution (but isn't free of charge)
- Configuring a proxy with Jana server is becoming more and more common
- Windows includes Microsoft Proxy Server (MSP), which works with Microsoft Proxy Client

Transparent proxying

- Router forwards all traffic to port 80 to proxy machine using a route policy
- Pros
  - Requires no explicit proxy configuration in the user’s browser

Transparent proxying

- Recommendation: Don’t use it!
  - Create a proxy auto-configuration file and instruct users to point at it
  - If you want to force users to use your proxy, either
    - Block all traffic to port 80
    - Use a route policy to redirect port 80 traffic to an origin web server and return a page explaining how to configure the various web browsers to access the proxy

Squid hardware requirements

- UNIX operating system (NT is not currently supported, nor has anyone announced work on a port)
- 128M RAM minimum recommended (scales by user count and size of disk cache)
- Disk
  - 512M to 1G for small user counts
  - 16G to 24G for large user counts
  - Squid 2.x is optimized for JBOD, not RAID

Installing Squid (overview)

- Get distribution from http://squid.nlanr.net/
- Increase maximum file descriptors available per process *before* configuring Squid
- Run configure script with desired compile-time options
- Run make; make install
- Edit squid.conf file
- Run Squid -z to initialize cache directory structure
- Start Squid daemon
- Test
- Migrate users over to proxy

Squid distributions (versions)

- 2.0, 2.1, 2.2
  - Redesigned disk storage algorithm much improved
  - Understands Cache-Control: tag
  - Better LRU/refresh rule engine
  - Supports proxy authentication
  - See documentation for full list of enhancements
- Recommendation: 2.1, 2.2 is fairly stable,

Squid compile-time configuration

- --prefix=/var/squid
- --enable-asyncio
  - Only stable on Solaris and bleeding edge Linux
  - Can actually be slower on lightly loaded proxies
- --enable-dlmalloc
- --enable-icmp
- --enable-ipf-transparent for transparent proxy support on some systems (*BSD)

Squid compile-time configuration

- --enable-snmp if desired
- --enable-delay-pools if desired
- --enable-cachemgr-hostname=<hostname> if using an alias for proxy or building on a different machine from the target proxy machine
- --enable-cache-digest and/or --enable-carp if using cache hierarchies

squid.conf runtime settings

- Default squid.conf file is heavily commented! Read it!
- Must set
  - cache_dir (one per disk)
  - cache_peer (one per peer) if participating in a hierarchy
  - cache_mem (8-16M preferred, even for large caches)
  - acl rules (default rules mostly work, but must reflect your address space)

squid.conf runtime settings

- Recommendations
  - ipcache_size, fqdncache_size to 4096
  - log_fqdn off (use Apache's logresolve offline)
  - Increase dns_children, redirect_children, authenticate_children based on usage statistics (see cachemgr.cgi front-end)
  - Tweak refresh_pattern rules (Danger Will Robinson! -- I suggest starting with examples found in the squid mailing list archives)

squid.conf runtime settings

- Recommendations (continued)
  - quick_abort_min 128 KB, quick_abort_max 4096 KB, quick_abort_pct 75
    - Tailor based on your bandwidth to the Internet
    - By default, squid will complete retrieval of any object requested, regardless of size; can burn considerable amounts of bandwidth!
- Too many other options in squid.conf to cover here; you really should read all the embedded comments!

squid.conf ACL example

- acl manager proto cache_object
- acl localhost src 127.0.0.1/32
- acl managerhost src 204.248.51.34/32
- acl managerhost src 204.248.51.39/32
- acl managerhost src 204.248.51.40/32
- acl cawtech src 204.248.51.0/24
- acl cawtech-internal src 172.16.0.0/16
- acl all src 0.0.0.0/0.0.0.0

squid.conf ACL example

- acl SSL_ports port 443 563
- acl gopher_ports port 70
- acl wais_ports port 210
- acl whois_ports port 43
- acl www_ports port 80 81
- acl ftp_ports port 21
- acl Safe_ports port 1025-65535

- acl CONNECT method CONNECT
- acl FTP proto FTP
- acl HTTP proto HTTP
- acl WAIS proto WAIS
- acl GOPHER proto GOPHER
- acl WHOIS proto WHOIS

squid.conf ACL example

- http_access deny manager !localhost !managerhost
- http_access deny CONNECT !SSL_ports
- http_access deny HTTP !www_ports !Safe_ports
- http_access deny FTP !ftp_ports !Safe_ports
- http_access deny GOPHER !gopher_ports !Safe_ports
- http_access deny WAIS !wais_ports !Safe_ports
- http_access deny WHOIS !whois_ports !Safe_ports

- http_access allow localhost
- http_access allow cawtech
- http_access allow cawtech-internal
- http_access deny all

Managing Squid

- I recommend the Calamaris.pl logfile analysis script, available at http://calamaris.cord.de/
- Use modified MRTG with Squid’s SNMP support (see SNMP section in Squid FAQ for details)

END

Nicholas Kiget
www.kiget.me.ke