Express-Guide
~to~
Basic Setup of
SQUID-CACHE
Proxy Chaining

by, **ABK** ~ http://www.twitter.com/**aBionic**

::Task Detail::

- Setting up a secured Chained-Proxy between different offices using Squid for a specific URL set.

::Background::

Links: *http://www.squid-cache.org/*

- Some background information about Squid Cache Proxy Server:
  - its a high performance proxy caching server for web clients, supporting FTP, Gopher and HTTP data objects i.e. normally text-based protocols
  - keeps metadata and hot-objects cached in RAM, supports non-block DNS and SSL
  - it can be implemented as a Normal Proxy which needs to be configured at User's end or even as a Interception Proxy

---

::Execution Method::

- Installing Squid was really easy, its available at YUM Repositories so use #yum -y install squid

- Open 'squid.conf' file in an editor to edit squid's configuration {might find it at '/etc/squid/squid.conf' or '/usr/local/etc/squid/squid.conf'}

  - *NOTE: Remember, the settings here are interpreted as per their occurrence in file as a filter above another. So, if you block "A,B,C" first and then allows "C,D,E"; then C will remain blocked. So, to be on safer side Squid.Conf has a section defined for each configuration along-with its detail. For every line of configuration go in its section and then add it.*

- Now the most basic setting required to edit is enabling access from clients for that just add settings as per following 2 lines
  - following lines
    acl myClientNetwork src *192.168.0.0/16*
    http_access allow myClientNetwork
- Suppose you wanna set rules for URLs of "A.com" and "Z.com" domain
  - make its ACL as
    acl egurls url_regex *.A.com .Z.com*
  - Denying proxy of this URL set
    always_direct allow egurls
    never_direct deny egurls
  - Allowing proxy of this URL set
    always_direct deny egurls
  - Denying direct access of URLs if proxy not possible
    never_direct allow egurls
- Check if line with 'http_port' is
  - **http_port 3128**
- To stop caching queries
  - **acl Query urlpath_regex cgi-bin \?**
  - **cache deny Query**
- Setting a hostname for Proxy, just don't reveal any info
  - **visible_hostname *ANYHOSTNAME***
- To setup a Parent Squid Server to set Proxy Chaining
  - **cache_peer *parent1IPorName* parent 3128 0 no-query default**
  - **cache_peer *parent2IPorName* parent 3128 0 no-query**
- To provide sibling Squid Server for cache checks
  - **cache_peer *parent2IPorName* sibling 3128 3130**
- To setup Squid Proxy only for Fail-over, preferring direct connection otherwise
  - **prefer_direct on**
- To deny caching, just keep it to proxy
  - **cache deny all**
- To open support for more ports (say 1234)
  - **acl safe_ports port *1234***

◆ Check correctness of squid.conf and apply changes
#squid k parse

◆ Creating Swap directories for Squid Cache
#squid -z

◆ Starting service
#squid -Ncd1
or
#service squid restart

::Tools/Technology Used::

- Squid Cache-Proxy Server:
- BurpSuite Proxy Tool:

::Inference::

- Squid can be used for multiple uses like Standard Proxy, Interception Proxy, Reverse Proxy, Cache Service, and even as a Load Balancer for Web Service running on that server.
- Its a great utility being developed from great time and still has great scope to be developed.
- Its just that its configuration styling is a bit buggy, sometimes shows weird results due to some self-unhanded issues.

::Troubleshooting/Updates::

- ***Problem:*** The web-service we were supposed to proxy was generating HTTP Request to several other domain names registered to same organization and sometimes it's IP addresses. This re-occurred several times.
  Solution:
  So, I tried to figure out all the URLs involved in correct functioning of Web-Service by analyzing it's request using BurpSuite Proxy tool.
  But this results into just the URLs requested at that time. So, to be on more safer side I analyzed the source code of parts of service giving error and en-listed the remaining URLs.

- ***Problem:*** Configurations of Squid Box were copied onto a newer box for similar results, but it resulted in blocking of sites supposed to go via Proxy.
  Solution:
  Initially, it was really absurd as the same settings worked over other box. But, Squid is somewhat popular for such results so it wasn't a worry. We were just trying different tweaks not changing the meaning of it but stating same things in different manner.
  It resulted into a revelation that the behavior was specific to certain Query URLs, other were working fine. Now, it should have worked because even these URLs matched RegEx.
  Some more tweaking of settings made it worked when we explicitly added

the 'always_direct' line to it; now normally that shouldn't have mattered... but for Query based URLs it explicitly required that setting
{no documentation found though}

acl egurls url_regex *.A.com .Z.com*
always_direct allow egurls
never_direct deny egurls

- ***Requirement:***
  Squid Proxy was connected to two ISPs via two Ethernet Cards, and we required to find a way of load-balancing between both service providers.
  Solution:
  Reading about it showed that load-balancing configuration provided by Squid is only for Parent Cache Proxies which is not based on Ethernet-Based load-balancing. We found Ethernet-Bonding with load-balancing module to implement the same. It has been discussed under one of the articles on this portal itself.
