Running Jenkins behind Squid

In situations where you want a user friendly url to access Jenkins (Not port 8080), it makes sense to have Jenkins runs behind Squid, so that you can access Jenkins on port 80 or 443. This document discusses some of the approaches for doing this.

**Squid 2.6**

Using Squid 2.6 (default on Centos 5.3):

```plaintext
acl all src 0.0.0.0/0.0.0.0
acl localhost src 127.0.0.1/255.255.255.255
acl manager proto cache_object
acl to_localhost dst 127.0.0.0/8
acl valid_dst dstdomain .YOUR_DOMAIN ci

cache_replacement_policy heap LFUDA
memory_replacement_policy heap GDSF

cache_dir ufs /var/spool/squid 512 16 256
cache_mem 512 MB
maximum_object_size 12000 KB

## http --> https redirect
## don't forget to update "Jenkins URL" on https://ci.YOUR_DOMAIN/configure
#acl httpPort myport 80
#http_access deny httpPort
#deny_info https://ci.YOUR_DOMAIN/ httpPort

cache_peer localhost parent 8080 0 originserver name=myAccel
coredump_dir /var/spool/squid
hierarchy_stoplist cgi-bin
http_access allow localhost
http_access allow manager localhost
http_access allow valid_dst
http_access deny all
http_access deny manager

## mkdir /etc/squid/ssl/ && cd /etc/squid/ssl/ # to generate your self-signed certificate
## openssl genrsa -out jenkins.key 1024
## openssl req -new -key jenkins.key -x509 -out jenkins.crt -days 999
http_port 80 vhost
https_port 443 cert=/etc/squid/ssl/jenkins.crt key=/etc/squid/ssl/jenkins.key vhost

http_reply_access allow all
icp_access allow all

refresh_pattern -i \.jp(e?g|gif|png|ico) 300 20% 600 override-expire

logformat combined %>a %ui %un \{%tl\} "$rm $ru HTTP/$rv" $Hs $<st "{%Referer}>h" "${%User-Agent}>h" $Ss $Sh
strip_query_terms off
access_log /var/log/squid/access.log combined

visible_hostname ci.YOUR_DOMAIN
```

This assumes that you run Jenkins on localhost port 8080. But you can have it on an other server / different port (adjust line starting with cache_peer)

Of course replace YOUR_DOMAIN with your domain.

**With ssl**

Remove one level of comment
sed s/^#// /etc/squid/squid.conf

#!/ If you use the swarm client plugin, the nodes will complain about:

SunCertPathBuilderException: unable to find valid certification path to requested target
  at sun.security.validator.PKIXValidator.doBuild(PKIXValidator.java:285)
  at sun.security.validator.PKIXValidator.engineValidate(PKIXValidator.java:191)
  at sun.security.validator.Validator.validate(Validator.java:218)
  at com.sun.net.ssl.internal.ssl.X509TrustManagerImpl.validate(X509TrustManagerImpl.java:126)
  at com.sun.net.ssl.internal.ssl.X509TrustManagerImpl.checkServerTrusted(X509TrustManagerImpl.java:209)
  at com.sun.net.ssl.internal.ssl.X509TrustManagerImpl.checkServerTrusted(X509TrustManagerImpl.java:249)
  at com.sun.net.ssl.internal.ssl.ClientHandshaker.serverCertificate(ClientHandshaker.java:1014)
... 13 more
Caused by: sun.security.provider.certpath.SunCertPathBuilderException: unable to find valid certification path to requested target

Fix:

Notes

We use Active Directory internally to authenticate our users, it was mandatory to allow https access only.