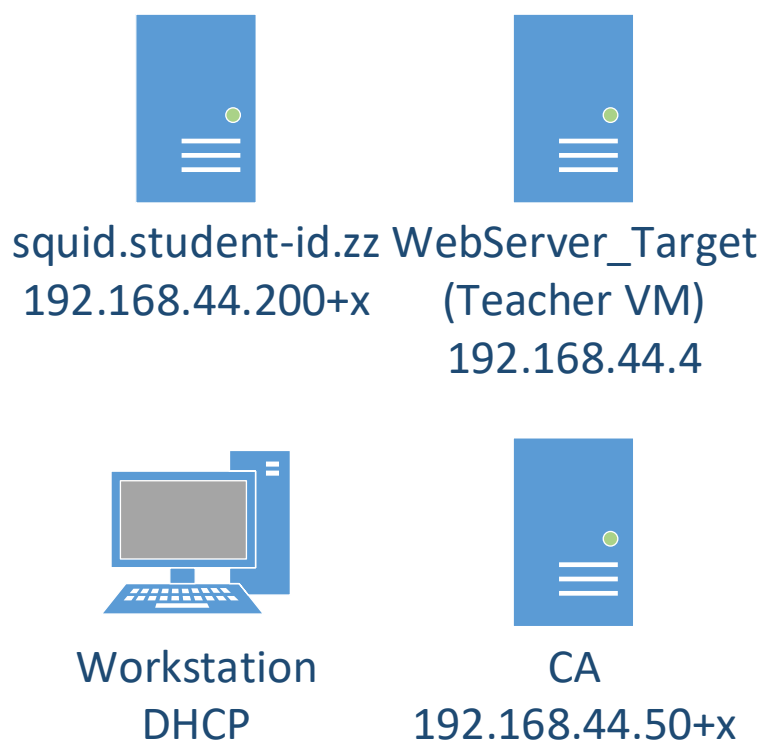


Lab10 – Forward Proxy using Squid

You can use this lab manual for your personal documentation. Use screenshots for your own documentation, there will be questions later on that may point to this lab manual. Take care to check if you need to collect some information from the lab for the answers.

\ at the end of the line is used to mark that the command needs to be on one line. Replace **student-id** with your own student-id and **x** or **y** as your VMs correct IP in the labs.

The labs use the following topology:



All VMs in this lab are in VirtualBox **Bridged** network. The machines that have static IP need to have an offset, check the topology image for reference. USE YOUR WIN7 WORKSTATION IP ADDRESS AS **x**.

All templates for VMs can be found in <\\ghost.labranet.jamk.fi\virtuaalikoneet\TTKS>

NOTE! Reuse CA VM and Workstation from previous labs (Certificates) to save time. If you are working at home, you need the WebServer_Target VM also, set the IP manually and make sure you can connect to it directly.

- **Install Squid**

Retrieve the pre-installed VM image for Centos7, [\ghost.labranet.jamk.fi\virtuaalikoneet\TTKS\](https://ghost.labranet.jamk.fi/virtuaalikoneet/TTKS/). Import it to Virtualbox with the name *Squid* and be sure to set “Reinitialize the MAC address...” tickbox in the import wizard. Remember to check the IP and set the hostname with `hostnamectl` as in previous labs.

Boot up the VM and login (**root/root66**). Check that it has got an IP. First we will install EPEL repo and then squid:

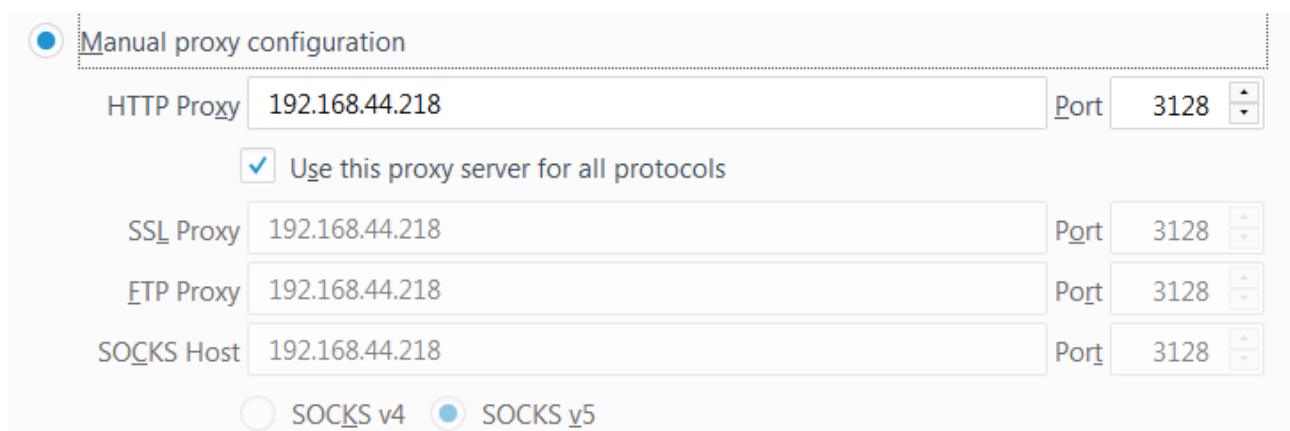
```
yum install epel-release
yum install squid
```

Start and enable squid and allow it through the firewall:

```
systemctl start squid
systemctl enable squid
firewall-cmd --add-service=squid --permanent
firewall-cmd --reload
```

Then edit Firefox proxy settings. On firefox, you can find them in Options - General – Network Proxy. Set HTTP Proxy and your squid VM IP address, port 3128. Set also “Use this proxy server for all protocols”.

NOTE: This will make all traffic go through your proxy. You can use Chrome if this breaks something. Also, please do not visit any important websites when doing this lab and remember to remove this setting from Firefox afterwards.



The screenshot shows the 'Manual proxy configuration' section in Firefox's Network Proxy settings. The 'Manual proxy configuration' radio button is selected. The 'HTTP Proxy' is set to '192.168.44.218' and the 'Port' is '3128'. The checkbox 'Use this proxy server for all protocols' is checked. Below this, the 'SSL Proxy', 'FTP Proxy', and 'SOCKS Host' are all set to '192.168.44.218', each with a 'Port' of '3128'. At the bottom, the 'SOCKS v4' radio button is unselected, and the 'SOCKS v5' radio button is selected.

Let's try the proxy. On Squid VM, run:

```
tail -f /var/log/squid/access.log
```

Now try to access <http://student.labranet.jamk.fi/> in your Workstation Firefox browser. You should see the GET requests in the access log. Refresh the page and try some other pages also.

```
1580984154.770 1120 192.168.44.101 TCP_TUNNEL/200 29489 CONNECT optima.jamk.fi
:443 - HIER_DIRECT/195.148.128.198 -
```

- **Modifying caching**

If you look at the log and browse multiple sites, you can see a lot of TCP_MISS. This means the pages are not cached (cached pages would be TCP_MEM_HIT). Let's force the squid to cache some elements.

Add caching to disk for more persistent cache. Uncomment and modify the following line in squid.conf:

```
cache_dir ufs /var/spool/squid 250 16 256
```

Also add:

```
maximum_object_size 1024 MB
```

Add a refresh-pattern, which forces images to be cached:

```
refresh_pattern -i \.(gif|png|jpg|jpeg|ico|bmp)$ 260000 90% 260009  
override-expire ignore-no-cache ignore-no-store ignore-private
```

```
cache_dir ufs /var/spool/squid 250 16 256  
maximum_object_size 1024 MB  
# Leave coredumps in the first cache dir  
coredump_dir /var/spool/squid  
  
#  
# Add any of your own refresh_pattern entries above these.  
#  
refresh_pattern ^ftp:          1440      20%    10080  
refresh_pattern ^gopher:       1440      0%     1440  
refresh_pattern -i (/cgi-bin/|\?) 0        0%      0  
refresh_pattern .               0         20%    4320  
refresh_pattern -i \.(gif|png|jpg|jpeg|ico|bmp)$ 260000 90% 260009 override-expire ignore-no-cache ignore-no-store ignore-private
```

This will force the images on the page to be cached as they usually are not. Run the following to create cache directories and restart squid:

```
systemctl stop squid  
squid -z  
systemctl start squid
```

```
[root@localhost.localdomain squid]# systemctl stop squid  
[root@localhost.localdomain squid]# squid -z  
[root@localhost.localdomain squid]# 2020/02/06 10:26:51 kid1| Set Current Directory to /var/spool/squid  
2020/02/06 10:26:51 kid1| Creating missing swap directories  
2020/02/06 10:26:51 kid1| /var/spool/squid exists  
2020/02/06 10:26:51 kid1| Making directories in /var/spool/squid/00  
2020/02/06 10:26:51 kid1| Making directories in /var/spool/squid/01  
2020/02/06 10:26:51 kid1| Making directories in /var/spool/squid/02  
2020/02/06 10:26:51 kid1| Making directories in /var/spool/squid/03  
2020/02/06 10:26:51 kid1| Making directories in /var/spool/squid/04  
2020/02/06 10:26:51 kid1| Making directories in /var/spool/squid/05  
2020/02/06 10:26:51 kid1| Making directories in /var/spool/squid/06  
2020/02/06 10:26:51 kid1| Making directories in /var/spool/squid/07  
2020/02/06 10:26:51 kid1| Making directories in /var/spool/squid/08  
2020/02/06 10:26:51 kid1| Making directories in /var/spool/squid/09  
2020/02/06 10:26:51 kid1| Making directories in /var/spool/squid/0A  
2020/02/06 10:26:51 kid1| Making directories in /var/spool/squid/0B  
2020/02/06 10:26:51 kid1| Making directories in /var/spool/squid/0C  
2020/02/06 10:26:51 kid1| Making directories in /var/spool/squid/0D  
2020/02/06 10:26:51 kid1| Making directories in /var/spool/squid/0E  
2020/02/06 10:26:51 kid1| Making directories in /var/spool/squid/0F  
systemctl start squid  
[root@localhost.localdomain squid]#
```

Now clear the browser cache or open a private browsing window. Now see the log again and try refreshing various web pages (iltasanomat, ampparit, telkku.com etc.). You should get at least some TCP_MEM_HIT results.

```
1  
1580985142.812      0 192.168.44.101 TCP_MEM_HIT/200 1818 GET http://info.cern.ch/favicon.ico - HIER NONE/- image/vnd.microsoft.icon  
1580985145.012      0 192.168.44.101 TCP_TUNNEL/200 5211 CONNECT url.ampparit.com:443 - HIER DIRECT/104 20 82 82
```

• Bypassing certain pages

Try accessing the TestWebServer and see that it gets cached. We can control the squid settings so local content does not get cached. Add a rule that forces direct access in squid.conf:

```
acl webserver dst 192.168.50.4
always_direct allow webserver
cache deny webserver
```

Restart squid and see how this changes the caching. Visits to the teachers TestWebServer should now go bypass the proxy.

- **Configure SSL**

If you haven't already, configure a CA like in *Lab1 Certificate*. Make sure the CA is installed as a trusted root CA in the Workstation.

Try to access www.jamk.fi or any other page that uses HTTPS. Squid cannot cache this kind of connection by default as it is SSL protected. We can however make squid act like a CA and write certificates on the fly.

QUESTIONNAIRE: How is a SSL connection shown in the Squid access.log when the proxy is NOT SSL-capable?

First, create a certificate request for the squid server:

```
cd /etc/squid
mkdir ssl_cert
chown squid:squid ssl_cert
chmod 700 ssl_cert
cd ssl_cert
openssl req -new -newkey rsa:2048 -sha256 -days 365 -nodes -extensions \
v3_ca -keyout squidCA.key -out squidCA.csr
```

Set the CN as squid.student-id.zz and Organisation again as ZZ-Test.

Then again transfer this file to the CA and sign it. Note that this will now be an intermediary CA, signed by your previously generated root CA and so it will be trusted by default.

```
openssl ca -config ca.cnf -extensions v3_ca -days 365 -keyfile ca.key \
-cert ca.pem -in squidCA.csr -out squidCA.pem
```

```
1 out of 1 certificate requests certified, commit? [y/n]y
Write out database with 1 new entries
Data Base Updated
[root@localhost.localdomain ca]# _
```

Transfer the signed squidCA.pem back to the correct /etc/squid/ssl_cert folder.

```
root@localhost.localdomain ssl_cert]# scp root@192.168.44.80:/root/ca/squidCA.pem /etc/squid/ssl_cert/
The authenticity of host '192.168.44.80 (192.168.44.80)' can't be established.
ECDSA key fingerprint is SHA256:I7Titguz9FMvruD0lwFyLLQG9ZKsFxZTvPe0lorSbjc.
ECDSA key fingerprint is MD5:e8:ae:cd:d3:dd:ca:3e:95:a2:89:d4:6a:fb:8d:97:67.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.44.80' (ECDSA) to the list of known hosts.
root@192.168.44.80's password:
squidCA.pem
root@localhost.localdomain ssl_cert]#
```

Next, modify the http_port-line and configure squid to use this CA certificate and do a "SSL-bump" in squid.conf:

```
http_port 3128 ssl-bump cert=/etc/squid/ssl_cert/squidCA.pem \
```

```
key=/etc/squid/ssl_cert/squidCA.key generate-host-certificates=on \
dynamic_cert_mem_cache_size=4MB
acl step1 at_step SslBump1
ssl_bump peek step1
ssl_bump bump all
```

```
# Squid normally listens to port 3128
http_port 3128 ssl_bump cert=/etc/squid/ssl_cert/squidCA.pem key=/etc/squid/ssl_cert/squidCA.key generate-host-certificates=on dynamic_cert_mem_cache_size=4MB
acl step1 at_step SslBump1
ssl_bump peek step1
ssl_bump bump all
```

Lastly, create the folder used to store generated certificates:

```
/usr/lib64/squid/ssl_crt -c -s /var/lib/ssl_db
chown squid:squid -R /var/lib/ssl_db
restorecon -R /var/lib/ssl_db
```

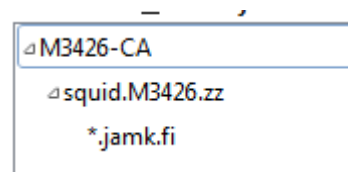
```
[root@localhost.localdomain ssl_cert]# /usr/lib64/squid/ssl_crt -c -s /var/lib/ssl_db
Initialization SSL db...
Done
[root@localhost.localdomain ssl_cert]#
```

```
[root@localhost.localdomain ssl_cert]# chown squid:squid -R /var/lib/ssl_db/
[root@localhost.localdomain ssl_cert]# restorecon -R /var/lib/ssl_db/
[root@localhost.localdomain ssl_cert]# systemctl restart squid
[root@localhost.localdomain ssl_cert]#
```

Restart squid. Try to browse to <https://www.jamk.fi>. Check the logs that squid sees the traffic (it will not cache it on the first try). Try some other pages too and notice how you won't get a certificate error.

```
1580987950.831 0 192.168.44.101 TCP_MEM_HIT/200 7867 GET https://www.jamk.fi/favicon.ico - HIER_NONE/- image/x-icon
```

When you are finished, check the certificate of the page and take a screenshot of the Certificate Hierarchy path (shown in View Certificate - Details)



QUESTIONNAIRE: How can you distinguish the certificate signed by Squid (other than the issuer field) from the legitimate one?