MODULE:CONCEPTS OF OPERATING SYSTEMS AND ADMINISTRATION

Assignment No. : 10

Assignment Name: SQUID PROXY SERVER CONFIGURATION

Name: Pritish Kshetre

Roll No. : 12

CONFIGURE THE SQUID PROXY SERVER WITH CLI

FORWARD PROXY

Go to Server

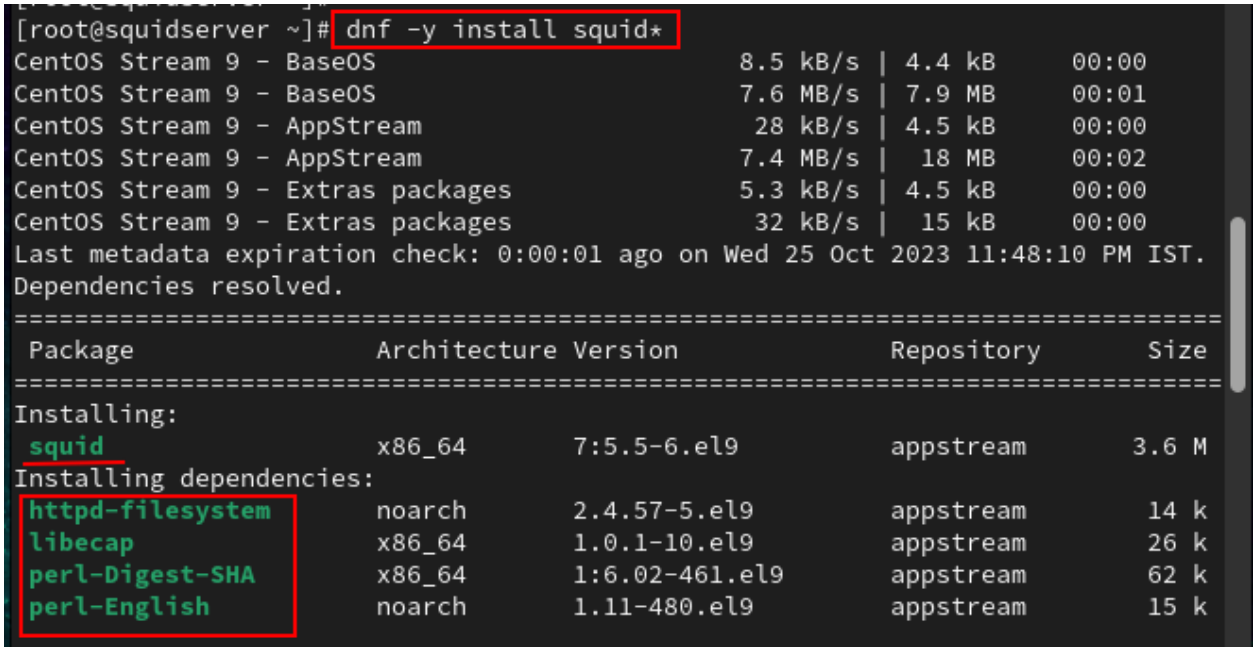
Step 1:- Change the hostname

```
[root@localhost ~]# hostnamectl set-hostname squid_server
[root@localhost ~]# /bin/bash
[root@squidserver ~]# hostname
squidserver
```

Step 2:- repolist shows all the running repos in the repolist

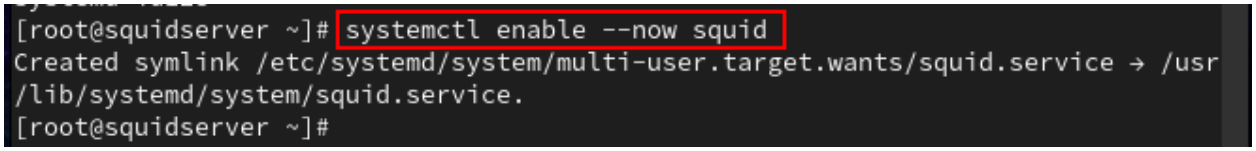
```
[root@squidserver ~]# dnf repolist all
repo id                    repo name                                  status
appstream                  CentOS Stream 9 - AppStream                enabled
appstream-debuginfo        CentOS Stream 9 - AppStream - Debug        disabled
appstream-source           CentOS Stream 9 - AppStream - Source       disabled
baseos                     CentOS Stream 9 - BaseOS                   enabled
baseos-debuginfo           CentOS Stream 9 - BaseOS - Debug           disabled
baseos-source              CentOS Stream 9 - BaseOS - Source          disabled
crb                        CentOS Stream 9 - CRB                      disabled
crb-debuginfo              CentOS Stream 9 - CRB - Debug              disabled
crb-source                 CentOS Stream 9 - CRB - Source             disabled
extras-common              CentOS Stream 9 - Extras packages          enabled
extras-common-source       CentOS Stream 9 - Extras packages - Source disabled
highavailability           CentOS Stream 9 - HighAvailability         disabled
highavailability-debuginfo CentOS Stream 9 - HighAvailability - Debug disabled
```

Step 3:- Install SQUID with all the dependencies



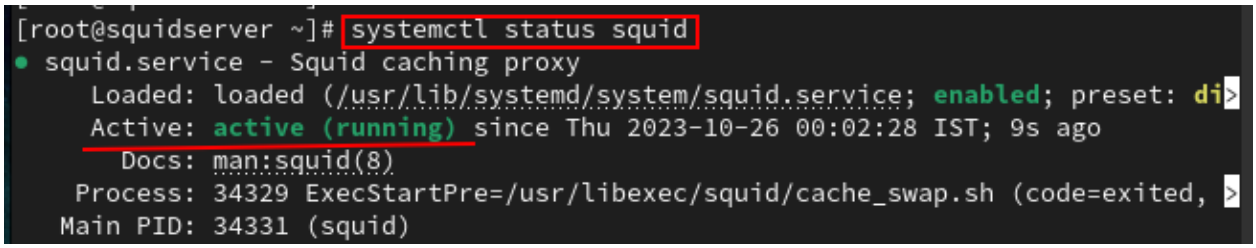
```
[root@squidserver ~]# dnf -y install squid*
CentOS Stream 9 - BaseOS                        8.5 kB/s | 4.4 kB     00:00
CentOS Stream 9 - BaseOS                        7.6 MB/s | 7.9 MB     00:01
CentOS Stream 9 - AppStream                      28 kB/s | 4.5 kB     00:00
CentOS Stream 9 - AppStream                     7.4 MB/s |  18 MB     00:02
CentOS Stream 9 - Extras packages               5.3 kB/s | 4.5 kB     00:00
CentOS Stream 9 - Extras packages                32 kB/s |  15 kB     00:00
Last metadata expiration check: 0:00:01 ago on Wed 25 Oct 2023 11:48:10 PM IST.
Dependencies resolved.
================================================================================
 Package                Architecture Version              Repository       Size
================================================================================
Installing:
 squid                  x86_64       7:5.5-6.el9          appstream       3.6 M
Installing dependencies:
 httpd-filesystem       noarch       2.4.57-5.el9         appstream        14 k
 libecap                x86_64       1.0.1-10.el9         appstream        26 k
 perl-Digest-SHA        x86_64       1:6.02-461.el9       appstream        62 k
 perl-English           noarch       1.11-480.el9         appstream        15 k
```

Step 4:- Start / enable the squid server



```
[root@squidserver ~]# systemctl enable --now squid
Created symlink /etc/systemd/system/multi-user.target.wants/squid.service → /usr
/lib/systemd/system/squid.service.
[root@squidserver ~]#
```

Step 5:- Check the status of the squid service



```
[root@squidserver ~]# systemctl status squid
● squid.service - Squid caching proxy
     Loaded: loaded (/usr/lib/systemd/system/squid.service; enabled; preset: di>
     Active: active (running) since Thu 2023-10-26 00:02:28 IST; 9s ago
       Docs: man:squid(8)
    Process: 34329 ExecStartPre=/usr/libexec/squid/cache_swap.sh (code=exited, >
   Main PID: 34331 (squid)
```

Step 6:- Check the service for squid in firewall

```
[root@squidserver ~]# firewall-cmd --info-service=squid
squid
  ports: 3128/tcp
  protocols:
  source-ports:
  modules:
  destination:
  includes:
  helpers:
```

Step 7:- Add the quid service to firewall so that the traffic should pass through the firewall

```
[root@squidserver ~]# firewall-cmd --permanent --zone=public --add-service=squid
success
```

Step 8:- Reload the firewall after any changes made in the firewall

```
[root@squidserver ~]# firewall-cmd --reload
success
[root@squidserver ~]#
```

Step 9:- Check the squid service gets added to the firewall

```
[root@squidserver ~]# firewall-cmd --list-all
public (active)
  target: default
  icmp-block-inversion: no
  interfaces: enp0s3
  sources:
  services: cockpit dhcpv6-client squid ssh
  ports:
  protocols:
  forward: yes
  masquerade: no
  forward-ports:
  source-ports:
  icmp-blocks:
  rich rules:
```

Step 10:- Configure the squid.conf file for setting up the proxy server

```
[root@squidserver ~]# vi /etc/squid/squid.conf
[root@squidserver ~]#
```

Step 11:- In this add the range of the network from which your client gets access to internet through the proxy server

Make sure the client and server must be on the same network range

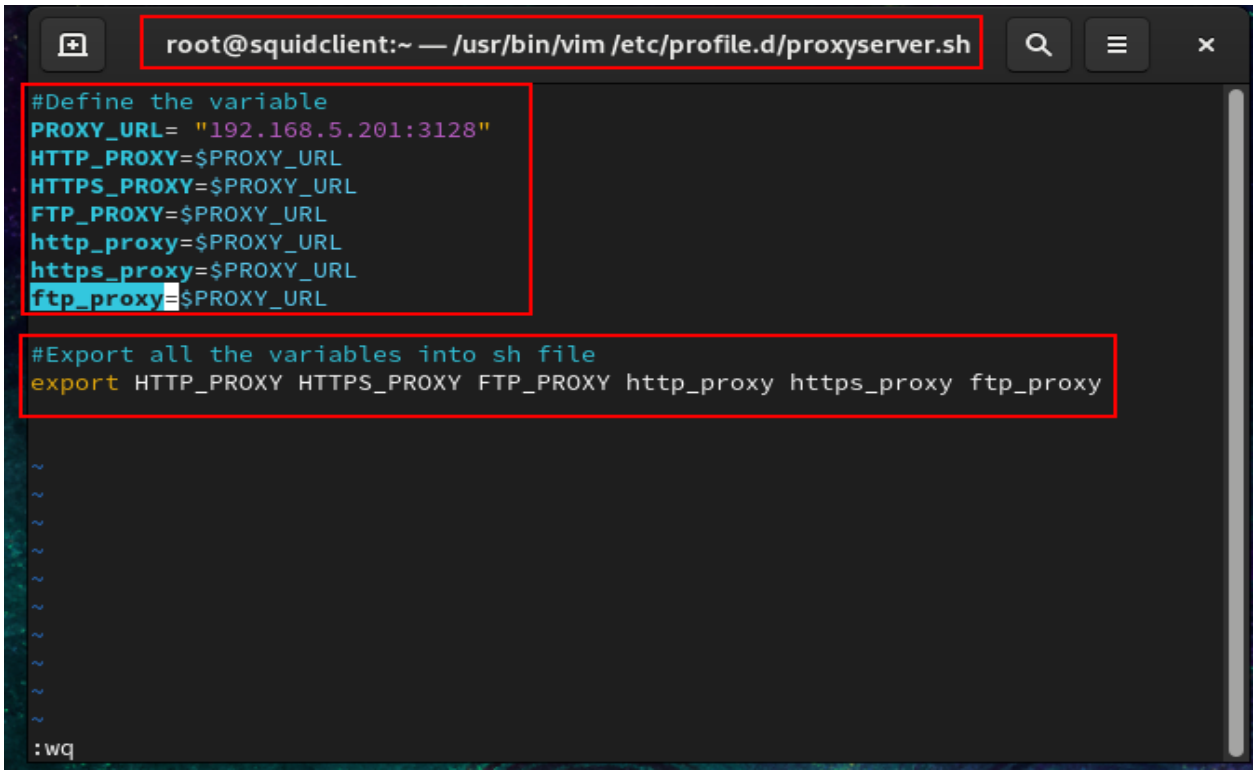
```
root@server:~ — /usr/bin/vim /etc/squid/squid.conf
50 #
51 # INSERT YOUR OWN RULE(S) HERE TO ALLOW ACCESS FROM YOUR CLIENTS
52 #
53 acl newlocalnet src 192.168.5.0/23
54 http_access allow newlocalnet
55
56 # Example rule allowing access from your local networks.
57 # Adapt localnet in the ACL section to list your (internal) IP networks
58 # from where browsing should be allowed
59 http_access allow localnet
60 http_access allow localhost
61
62 # And finally deny all other access to this proxy
63 http_access deny all
64
65 # Squid normally listens to port 3128
66 http_port 3128
67
68 # Uncomment and adjust the following to add a disk cache directory.
69 #cache_dir ufs /var/spool/squid 100 16 256
70
71 # Leave coredumps in the first cache dir
72 coredump_dir /var/spool/squid
                                                    64,0-1        85%
```

Step 12:- Then configure the proxyserver.sh file

```
[root@squidclient ~]# vi /etc/profile.d/proxyserver.sh
[root@squidclient ~]#
```

Go to Client

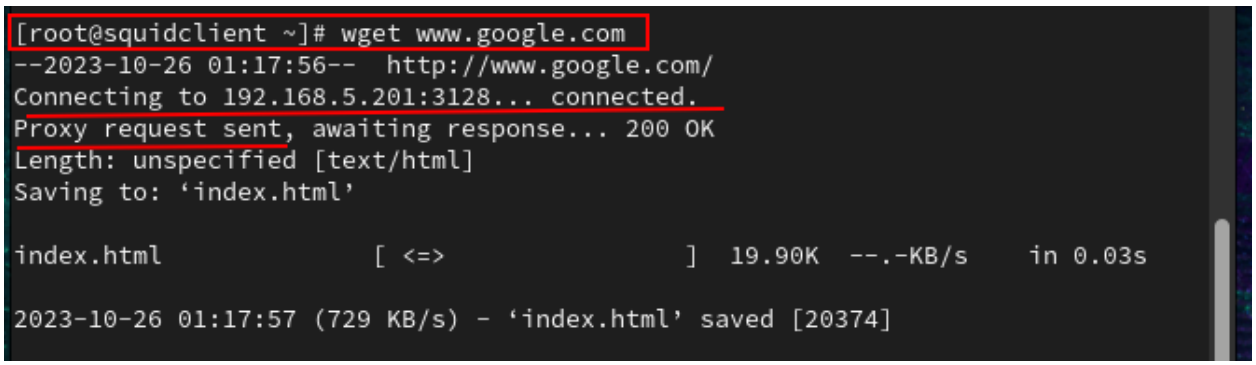
Step 13:- In this file you've to specify the PROXY_URL in that you specify the SQUID SERVER IP and SQUID PORT 3128 , so that all the client request should pass from the Proxy server



Step 14:- Start/ Enable the file

Step 15:- Try to access to the www.google.com

Here the request goes from the proxy server



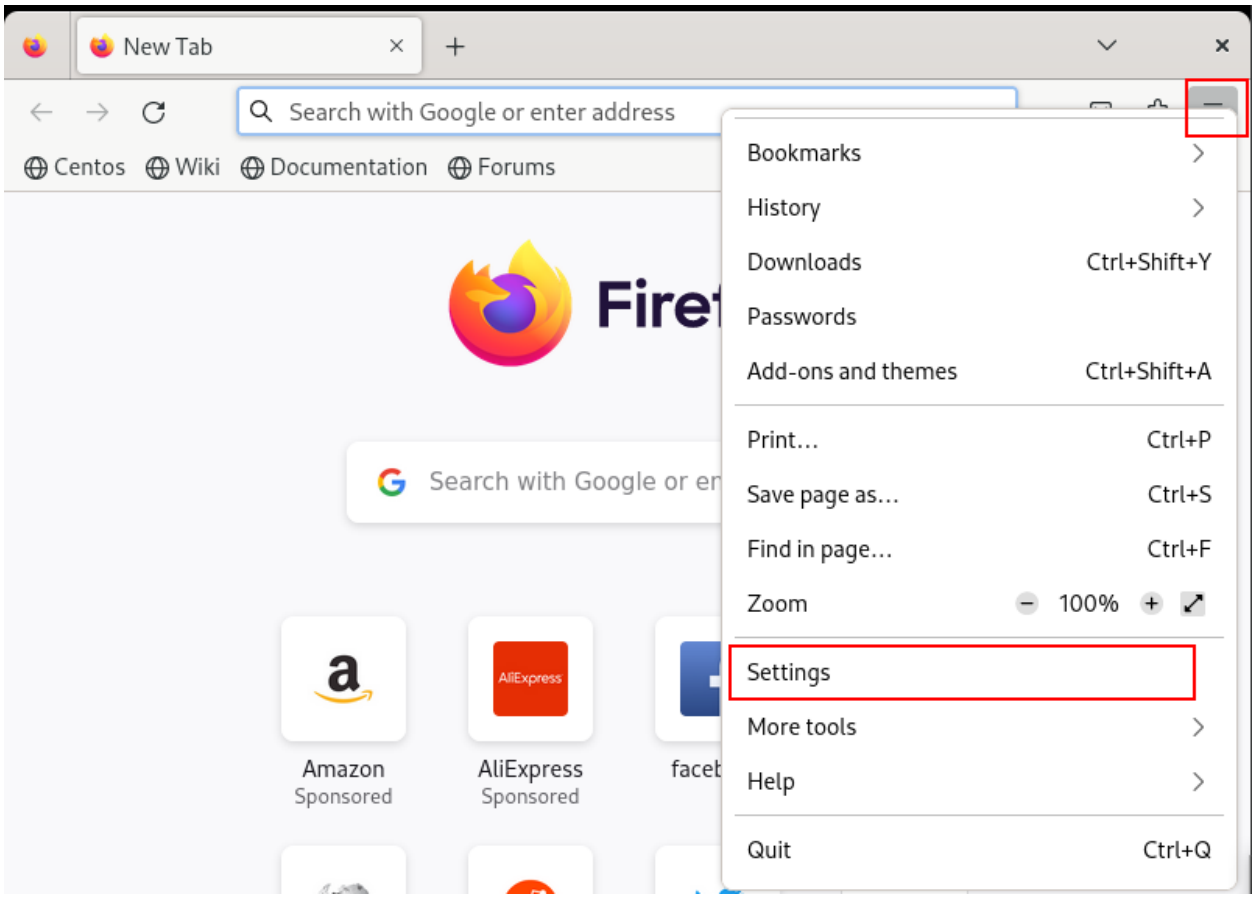
Step 16:- Check the proxy server getting the request from the client

```
[root@squidserver ~]#
[root@squidserver ~]# tail /var/log/squid/access.log
1698263267.867    173 192.168.5.206 TCP_MISS/301 699 GET http://www.amazon.com/ - HIER_DIRECT/108.
158.65.52 text/html
1698263268.776    766 192.168.5.206 TCP_TUNNEL/200 9090 CONNECT www.amazon.com:443 - HIER_DIRECT/1
08.158.65.52 -
1698263277.453    487 192.168.5.206 TCP_MISS/200 21723 GET http://www.google.com/ - HIER_DIRECT/14
2.250.182.4 text/html
[root@squidserver ~]#
[root@squidserver ~]#
```

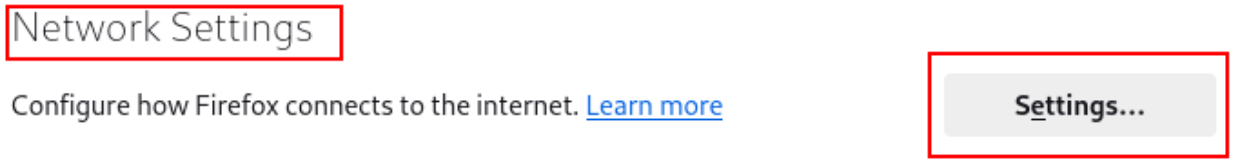
CONFIGURE THE SQUID PROXY SERVER WITH GUI

Go to client

Step 1:- Go to browser > Settings

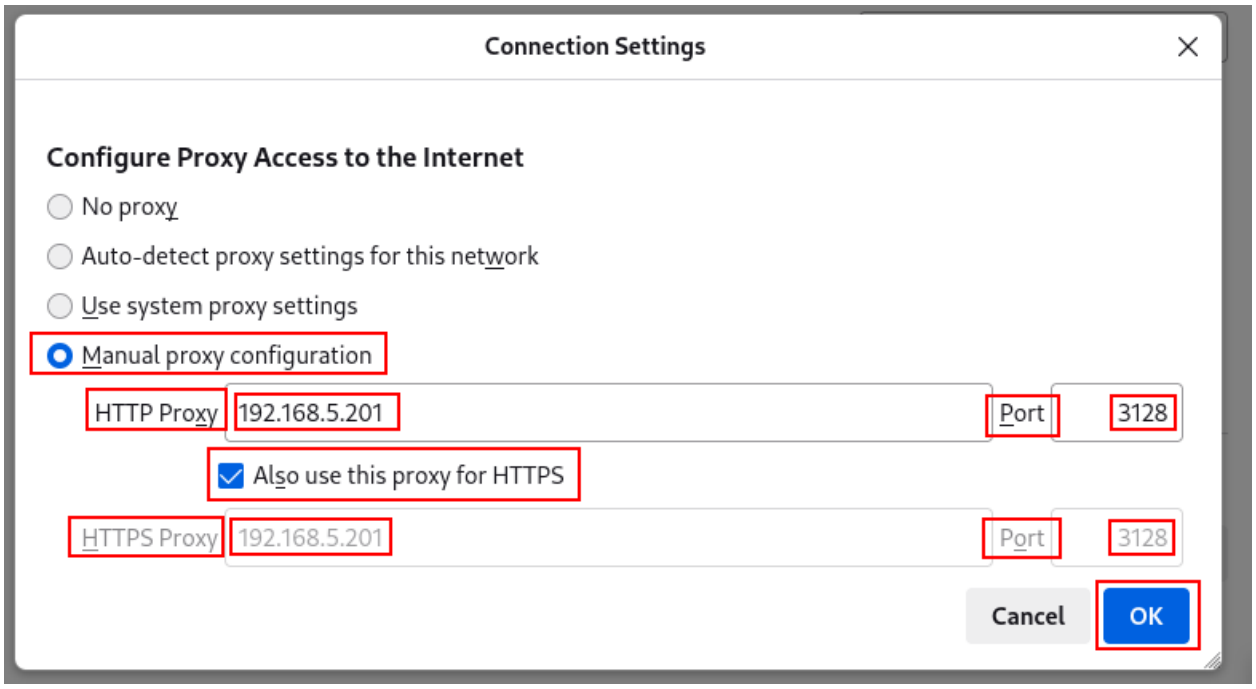


Step 2:- Settings > Scroll down > Network Settings

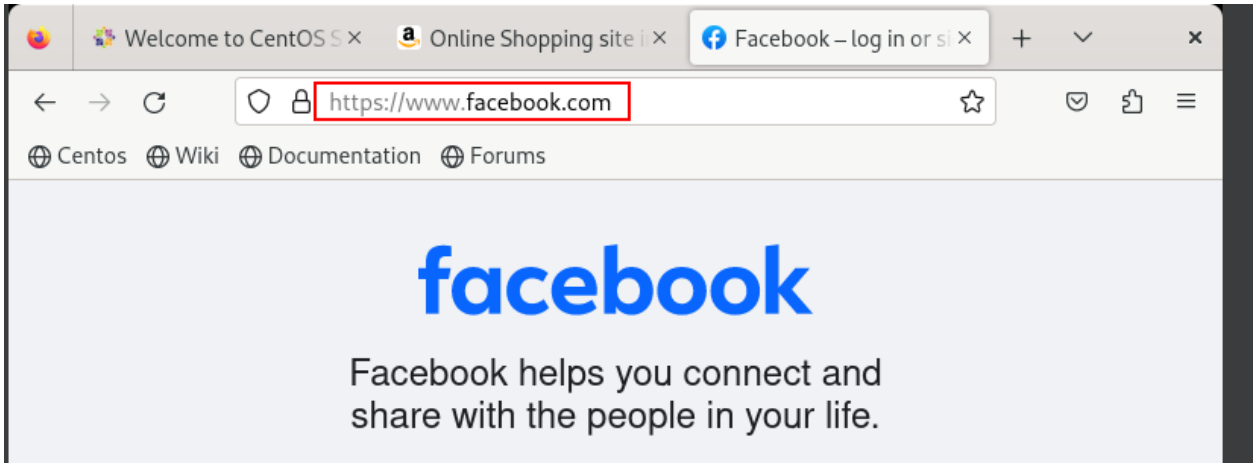


Step 3:- Manually configure proxy settings

Here put the Proxy Server IP, Add port of Squid Proxy server



Step 4:- Access the websites



BLOCK THE WEBSITES WITH SQUID PROXY SERVER

Go to Server

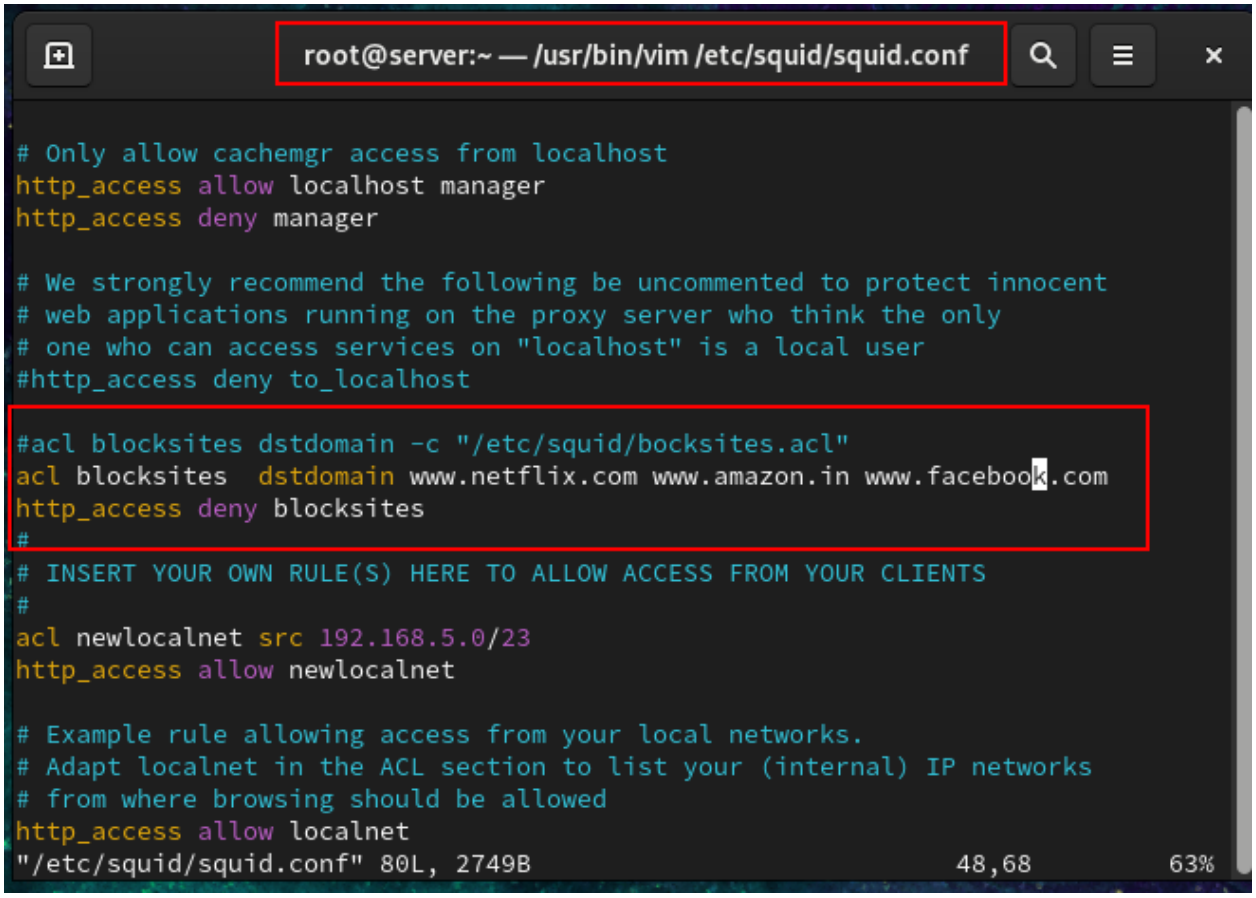
Step 1:- Go to /etc/squid > Then create a blank file with .acl extension

```
[root@server ~]# cd /etc/squid
[root@server squid]# touch blocksites.acl
[root@server squid]#
[root@server squid]# ls
blocksites.acl          errorpage.css          mime.conf.default
cachemgr.conf           errorpage.css.default  squid.conf
cachemgr.conf.default   mime.conf              squid.conf.default
```

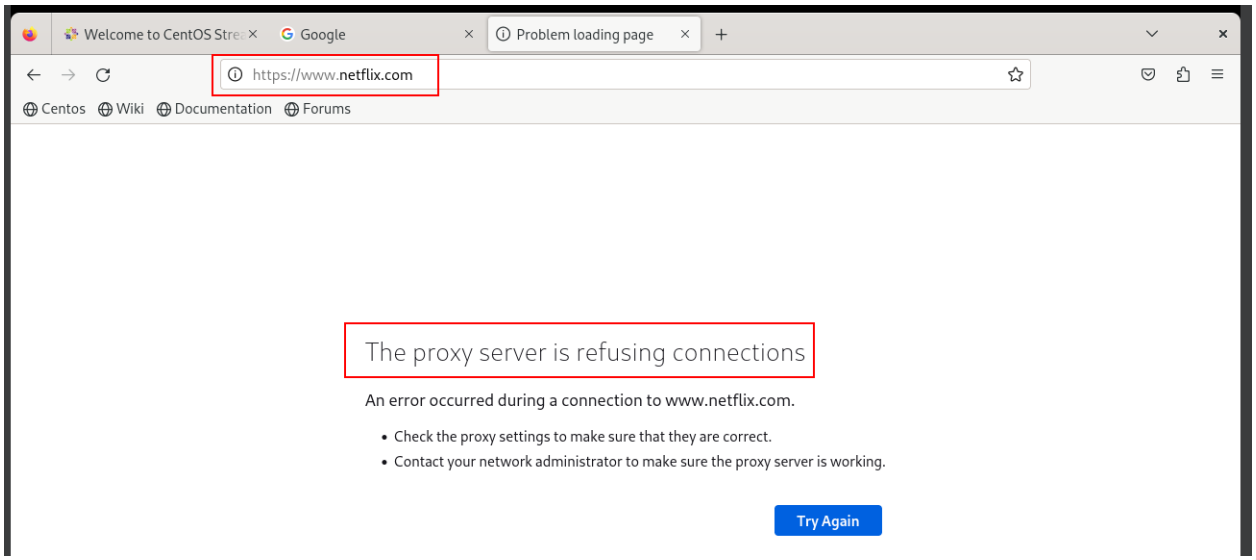
Step 2:- Configure the squid.conf file

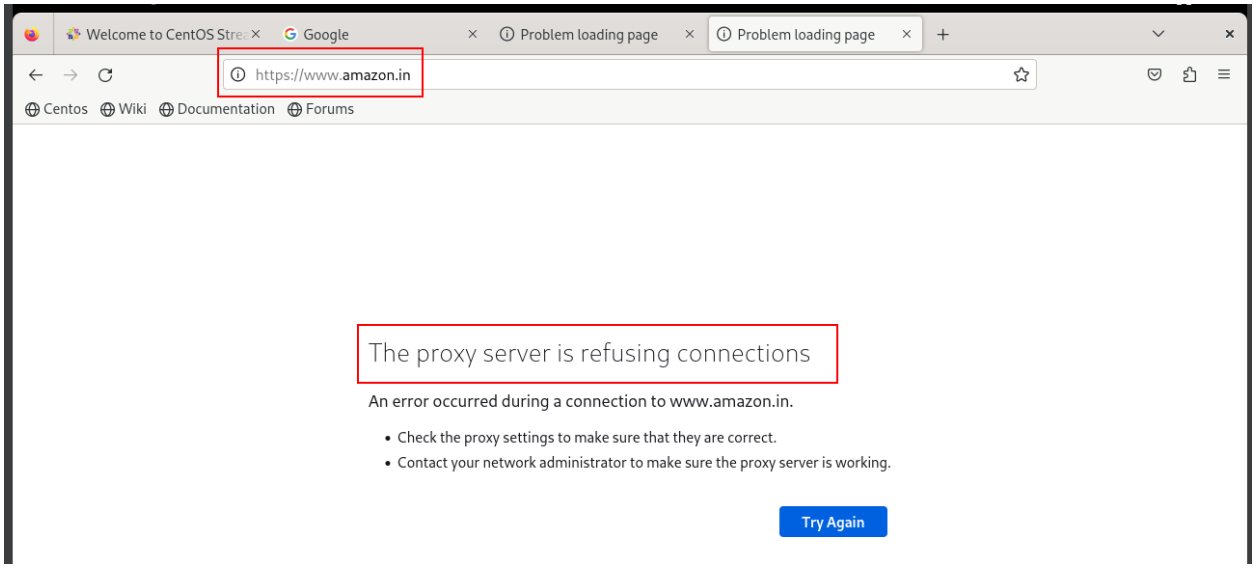
```
[root@server ~]#
[root@server ~]# vi /etc/squid/squid.conf
[root@server ~]#
[root@server ~]#
```

Step 3:- In this set the websites with their domains and specify the deny http access for the client



Step 5:- You can check at client that the websites have no access permission





But the websites other than block are able to access by the client