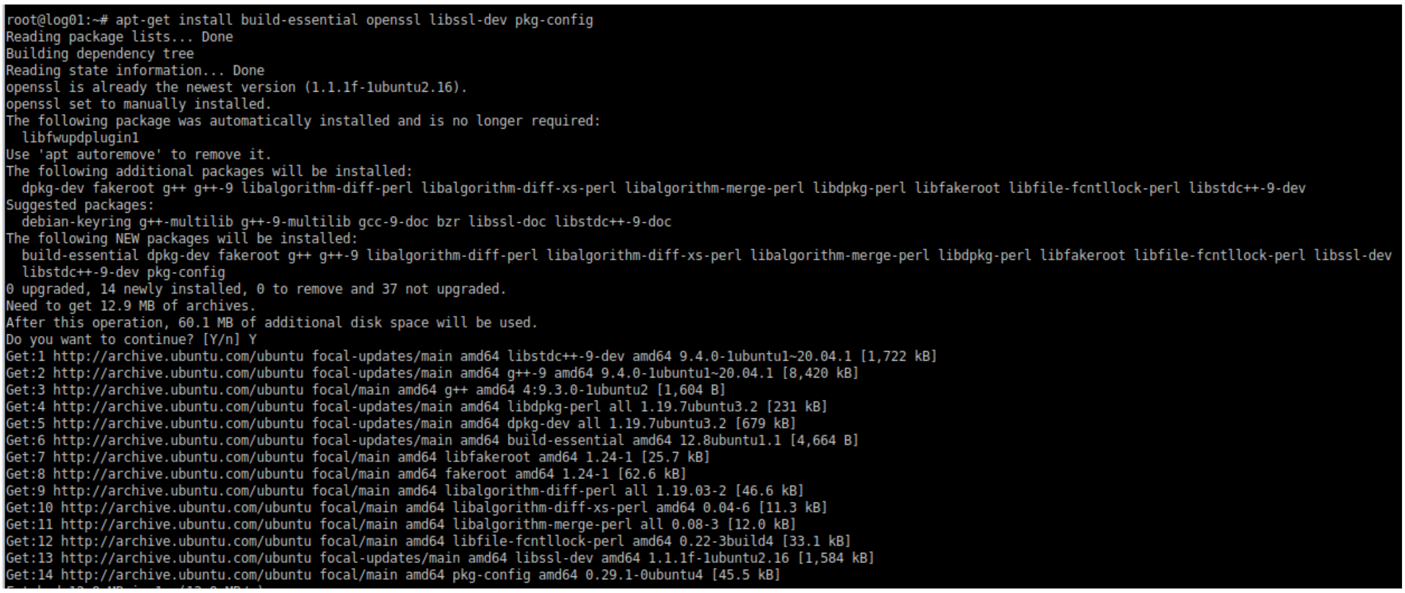
SSL bumping ile Ubuntu 22.04 üzerinde Squid Server 5.7

Squid 5.7 yi ubuntu 20.04 üzerine compile ederek kaynaktan kurmak için yapılması gerekenler

Derlemek için ön gereklilikler

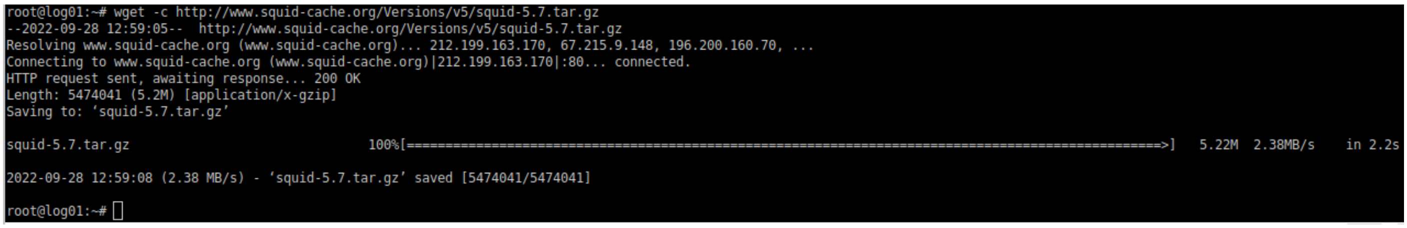
```
apt-get install build-essential openssl libssl-dev pkg-config
```



```
root@log01:~# apt-get install build-essential openssl libssl-dev pkg-config
Reading package lists... Done
Building dependency tree
Reading state information... Done
openssl is already the newest version (1.1.1f-1ubuntu2.16).
openssl set to manually installed.
The following package was automatically installed and is no longer required:
  libfwupdplugin1
Use 'apt autoremove' to remove it.
The following additional packages will be installed:
  dpkg-dev fakeroot g++ g++-9 libalgorithm-diff-perl libalgorithm-diff-xs-perl libalgorithm-merge-perl libdpkg-perl libfakeroot libfile-fcntllock-perl libstdc++-9-dev
Suggested packages:
  debian-keyring g++-multilib g++-9-multilib gcc-9-doc bzr libssl-doc libstdc++-9-doc
The following NEW packages will be installed:
  build-essential dpkg-dev fakeroot g++ g++-9 libalgorithm-diff-perl libalgorithm-diff-xs-perl libalgorithm-merge-perl libdpkg-perl libfakeroot libfile-fcntllock-perl libssl-dev
  libstdc++-9-dev pkg-config
0 upgraded, 14 newly installed, 0 to remove and 37 not upgraded.
Need to get 12.9 MB of archives.
After this operation, 60.1 MB of additional disk space will be used.
Do you want to continue? [Y/n] Y
Get:1 http://archive.ubuntu.com/ubuntu focal-updates/main amd64 libstdc++-9-dev amd64 9.4.0-1ubuntu1~20.04.1 [1,722 kB]
Get:2 http://archive.ubuntu.com/ubuntu focal-updates/main amd64 g++-9 amd64 9.4.0-1ubuntu1~20.04.1 [8,420 kB]
Get:3 http://archive.ubuntu.com/ubuntu focal/main amd64 g++ amd64 4:9.3.0-1ubuntu2 [1,604 B]
Get:4 http://archive.ubuntu.com/ubuntu focal-updates/main amd64 libdpkg-perl all 1.19.7ubuntu3.2 [231 kB]
Get:5 http://archive.ubuntu.com/ubuntu focal-updates/main amd64 dpkg-dev all 1.19.7ubuntu3.2 [679 kB]
Get:6 http://archive.ubuntu.com/ubuntu focal-updates/main amd64 build-essential amd64 12.8ubuntu1.1 [4,664 B]
Get:7 http://archive.ubuntu.com/ubuntu focal/main amd64 libfakeroot amd64 1.24-1 [25.7 kB]
Get:8 http://archive.ubuntu.com/ubuntu focal/main amd64 fakeroot amd64 1.24-1 [62.6 kB]
Get:9 http://archive.ubuntu.com/ubuntu focal/main amd64 libalgorithm-diff-perl all 1.19.03-2 [46.6 kB]
Get:10 http://archive.ubuntu.com/ubuntu focal/main amd64 libalgorithm-diff-xs-perl amd64 0.04-6 [11.3 kB]
Get:11 http://archive.ubuntu.com/ubuntu focal/main amd64 libalgorithm-merge-perl all 0.08-3 [12.0 kB]
Get:12 http://archive.ubuntu.com/ubuntu focal/main amd64 libfile-fcntllock-perl amd64 0.22-3build4 [33.1 kB]
Get:13 http://archive.ubuntu.com/ubuntu focal-updates/main amd64 libssl-dev amd64 1.1.1f-1ubuntu2.16 [1,584 kB]
Get:14 http://archive.ubuntu.com/ubuntu focal/main amd64 pkg-config amd64 0.29.1-0ubuntu4 [45.5 kB]
```

Bu işlem bitince download ediyoruz

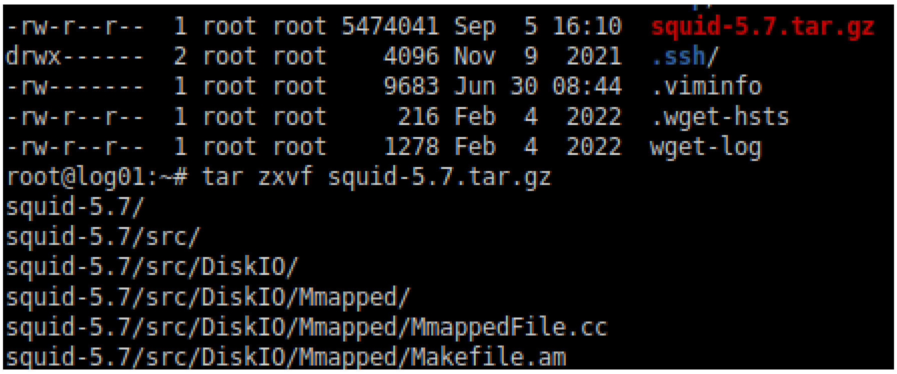
```
wget -c http://www.squid-cache.org/Versions/v5/squid-5.7.tar.gz
```



```
root@log01:~# wget -c http://www.squid-cache.org/Versions/v5/squid-5.7.tar.gz
--2022-09-28 12:59:05--  http://www.squid-cache.org/Versions/v5/squid-5.7.tar.gz
Resolving www.squid-cache.org (www.squid-cache.org)... 212.199.163.170, 67.215.9.148, 196.200.160.70, ...
Connecting to www.squid-cache.org (www.squid-cache.org)|212.199.163.170|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 5474041 (5.2M) [application/x-gzip]
Saving to: ‘squid-5.7.tar.gz’

squid-5.7.tar.gz                              100%[=================================================================================================>]   5.22M  2.38MB/s    in 2.2s

2022-09-28 12:59:08 (2.38 MB/s) - ‘squid-5.7.tar.gz’ saved [5474041/5474041]

root@log01:~# 
```

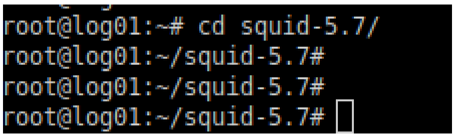
Download edilen kaynak dosyasını açıyoruz.

```
tar zxvf squid-5.7.tar.gz
```



```
-rw-r--r--  1 root root 5474041 Sep  5 16:10  squid-5.7.tar.gz
drwx------  2 root root    4096 Nov  9  2021  .ssh/
-rw-------  1 root root    9683 Jun 30 08:44  .viminfo
-rw-r--r--  1 root root     216 Feb  4  2022  .wget-hsts
-rw-r--r--  1 root root    1278 Feb  4  2022  wget-log
root@log01:~# tar zxvf squid-5.7.tar.gz
squid-5.7/
squid-5.7/src/
squid-5.7/src/DiskIO/
squid-5.7/src/DiskIO/Mmapped/
squid-5.7/src/DiskIO/Mmapped/MmappedFile.cc
squid-5.7/src/DiskIO/Mmapped/Makefile.am
```

Çıkardığımız klasöre geçiyoruz

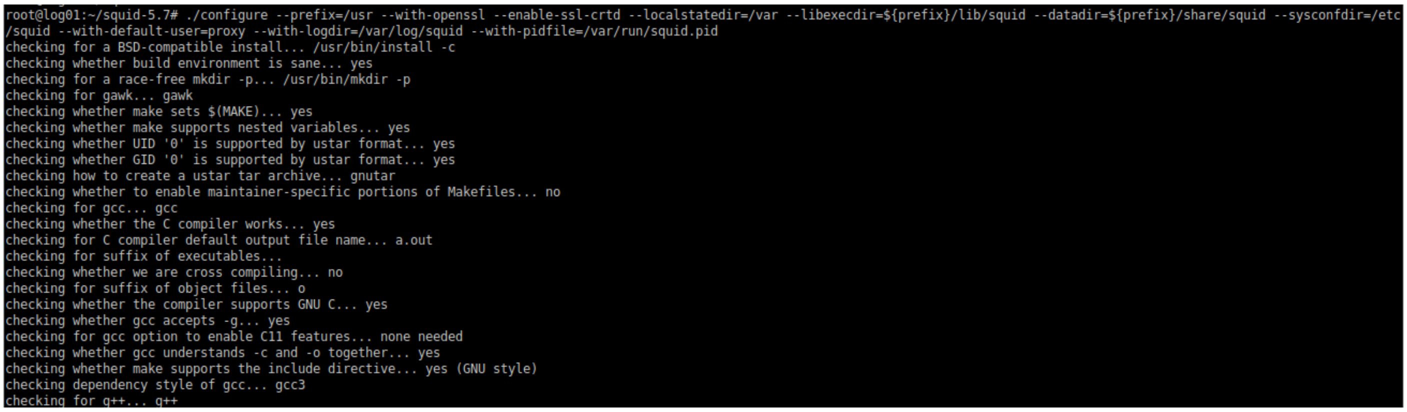


```
root@log01:~# cd squid-5.7/
root@log01:~/squid-5.7#
root@log01:~/squid-5.7#
root@log01:~/squid-5.7#
```

Derleme

Aşağıdaki parametreler ile compile başlatıyoruz

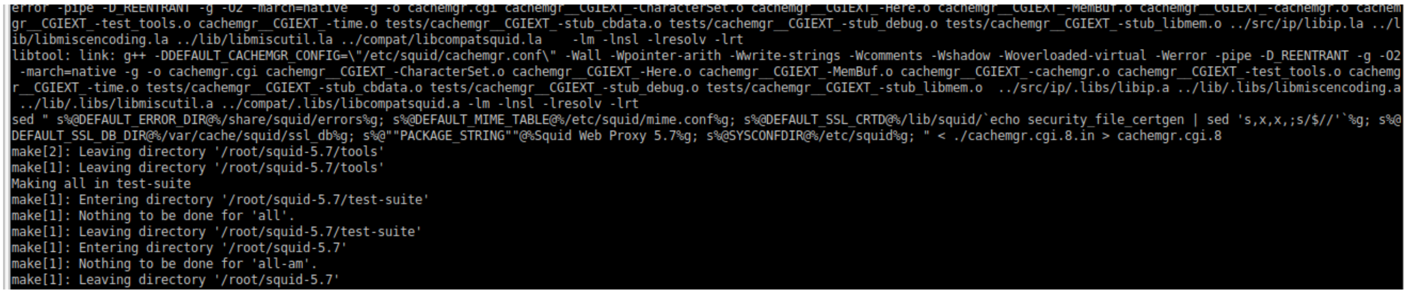
```
./configure --prefix=/usr --with-openssl --enable-ssl-crtd --localstatedir=/var --libexecdir=${prefix}/lib/squid --datadir=${prefix}/share/squid --sysconfdir=/etc/squid --with-default-user=proxy --with-logdir=/var/log/squid --with-pidfile=/var/run/squid.pid
```



```
root@log01:~/squid-5.7# ./configure --prefix=/usr --with-openssl --enable-ssl-crtd --localstatedir=/var --libexecdir=${prefix}/lib/squid --datadir=${prefix}/share/squid --sysconfdir=/etc
/squid --with-default-user=proxy --with-logdir=/var/log/squid --with-pidfile=/var/run/squid.pid
checking for a BSD-compatible install... /usr/bin/install -c
checking whether build environment is sane... yes
checking for a race-free mkdir -p... /usr/bin/mkdir -p
checking for gawk... gawk
checking whether make sets $(MAKE)... yes
checking whether make supports nested variables... yes
checking whether UID '0' is supported by ustar format... yes
checking whether GID '0' is supported by ustar format... yes
checking how to create a ustar tar archive... gnutar
checking whether to enable maintainer-specific portions of Makefiles... no
checking for gcc... gcc
checking whether the C compiler works... yes
checking for C compiler default output file name... a.out
checking for suffix of executables...
checking whether we are cross compiling... no
checking for suffix of object files... o
checking whether the compiler supports GNU C... yes
checking whether gcc accepts -g... yes
checking for gcc option to enable C11 features... none needed
checking whether gcc understands -c and -o together... yes
checking whether make supports the include directive... yes (GNU style)
checking dependency style of gcc... gcc3
checking for g++... g++
```

Hiçbir warning yada hata almadan bitmeli. Ardından derleme işlemini aşağıdaki komut ile başlatıyoruz. Bu 15 dakika kadar sürebiliyor.

```
make all
```



```
error -pipe -D_REENTRANT -g -O2 -march=native  -g -o cachemgr.cgi cachemgr__CGIEXT_-CharacterSet.o cachemgr__CGIEXT_-Here.o cachemgr__CGIEXT_-MemBuf.o cachemgr__CGIEXT_-cachemgr.o cachem
gr__CGIEXT_-test_tools.o cachemgr__CGIEXT_-time.o tests/cachemgr__CGIEXT_-stub_cbdata.o tests/cachemgr__CGIEXT_-stub_debug.o tests/cachemgr__CGIEXT_-stub_libmem.o ../src/ip/libip.la ../l
ib/libmiscencoding.la ../lib/libmiscutil.la ../compat/libcompatsquid.la    -lm -lnsl -lresolv -lrt
libtool: link: g++ -DDEFAULT_CACHEMGR_CONFIG=\"/etc/squid/cachemgr.conf\" -Wall -Wpointer-arith -Wwrite-strings -Wcomments -Wshadow -Woverloaded-virtual -Werror -pipe -D_REENTRANT -g -O2
 -march=native -g -o cachemgr.cgi cachemgr__CGIEXT_-CharacterSet.o cachemgr__CGIEXT_-Here.o cachemgr__CGIEXT_-MemBuf.o cachemgr__CGIEXT_-cachemgr.o cachemgr__CGIEXT_-test_tools.o cachemg
r__CGIEXT_-time.o tests/cachemgr__CGIEXT_-stub_cbdata.o tests/cachemgr__CGIEXT_-stub_debug.o tests/cachemgr__CGIEXT_-stub_libmem.o  ../src/ip/.libs/libip.a ../lib/.libs/libmiscencoding.a
 ../lib/.libs/libmiscutil.a ../compat/.libs/libcompatsquid.a -lm -lnsl -lresolv -lrt
sed " s%@DEFAULT_ERROR_DIR@%/share/squid/errors%g; s%@DEFAULT_MIME_TABLE@%/etc/squid/mime.conf%g; s%@DEFAULT_SSL_CRTD@%/lib/squid/`echo security_file_certgen | sed 's,x,x,;s/$//'`%g; s%@
DEFAULT_SSL_DB_DIR@%/var/cache/squid/ssl_db%g; s%@""PACKAGE_STRING""@%Squid Web Proxy 5.7%g; s%@SYSCONFDIR@%/etc/squid%g; " < ./cachemgr.cgi.8.in > cachemgr.cgi.8
make[2]: Leaving directory '/root/squid-5.7/tools'
make[1]: Leaving directory '/root/squid-5.7/tools'
Making all in test-suite
make[1]: Entering directory '/root/squid-5.7/test-suite'
make[1]: Nothing to be done for 'all'.
make[1]: Leaving directory '/root/squid-5.7/test-suite'
make[1]: Entering directory '/root/squid-5.7'
make[1]: Nothing to be done for 'all-am'.
make[1]: Leaving directory '/root/squid-5.7'
```

Make işlemi sorunsuz şekilde bittikten sonra

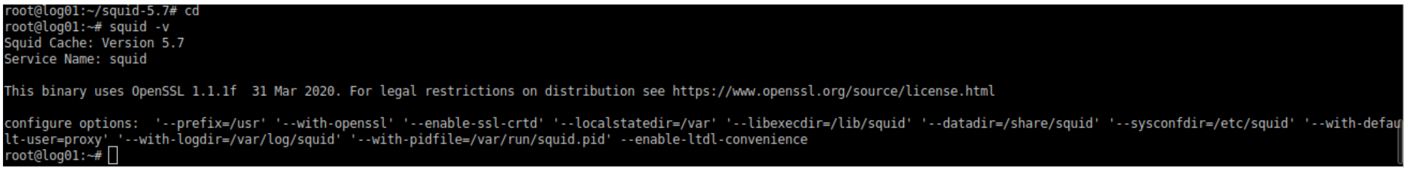
```
make install
```

yapıyoruz ve kurulum bitiyor.

Kontrol edelim.

```
squid -v
```

Aşağıdaki gibi görünmeli



```
root@log01:~/squid-5.7# cd
root@log01:~# squid -v
Squid Cache: Version 5.7
Service Name: squid

This binary uses OpenSSL 1.1.1f  31 Mar 2020. For legal restrictions on distribution see https://www.openssl.org/source/license.html

configure options:  '--prefix=/usr' '--with-openssl' '--enable-ssl-crtd' '--localstatedir=/var' '--libexecdir=/lib/squid' '--datadir=/share/squid' '--sysconfdir=/etc/squid' '--with-defau
lt-user=proxy' '--with-logdir=/var/log/squid' '--with-pidfile=/var/run/squid.pid' --enable-ltdl-convenience
root@log01:~# 
```

Systemd dosyası

Systemd servisi Ubuntu 22.04 versiyon linux kurulumlarının sistem servis yöneticisidir. Bu yöneticiye squid için gerekli detayların eklenmesi gerekir.

Derleyerek kurduğumuz squid 5.7 systemd dosyasyı olmadan gelir, oluşturmak için aşağıdaki şablonu kullana bilirsiniz

```
## Copyright (C) 1996-2022 The Squid Software Foundation and contributors
##
## Squid software is distributed under GPLv2+ license and includes
## contributions from numerous individuals and organizations.
## Please see the COPYING and CONTRIBUTORS files for details.
##

[Unit]
Description=Squid Web Proxy Server
Documentation=man:squid(8)
```

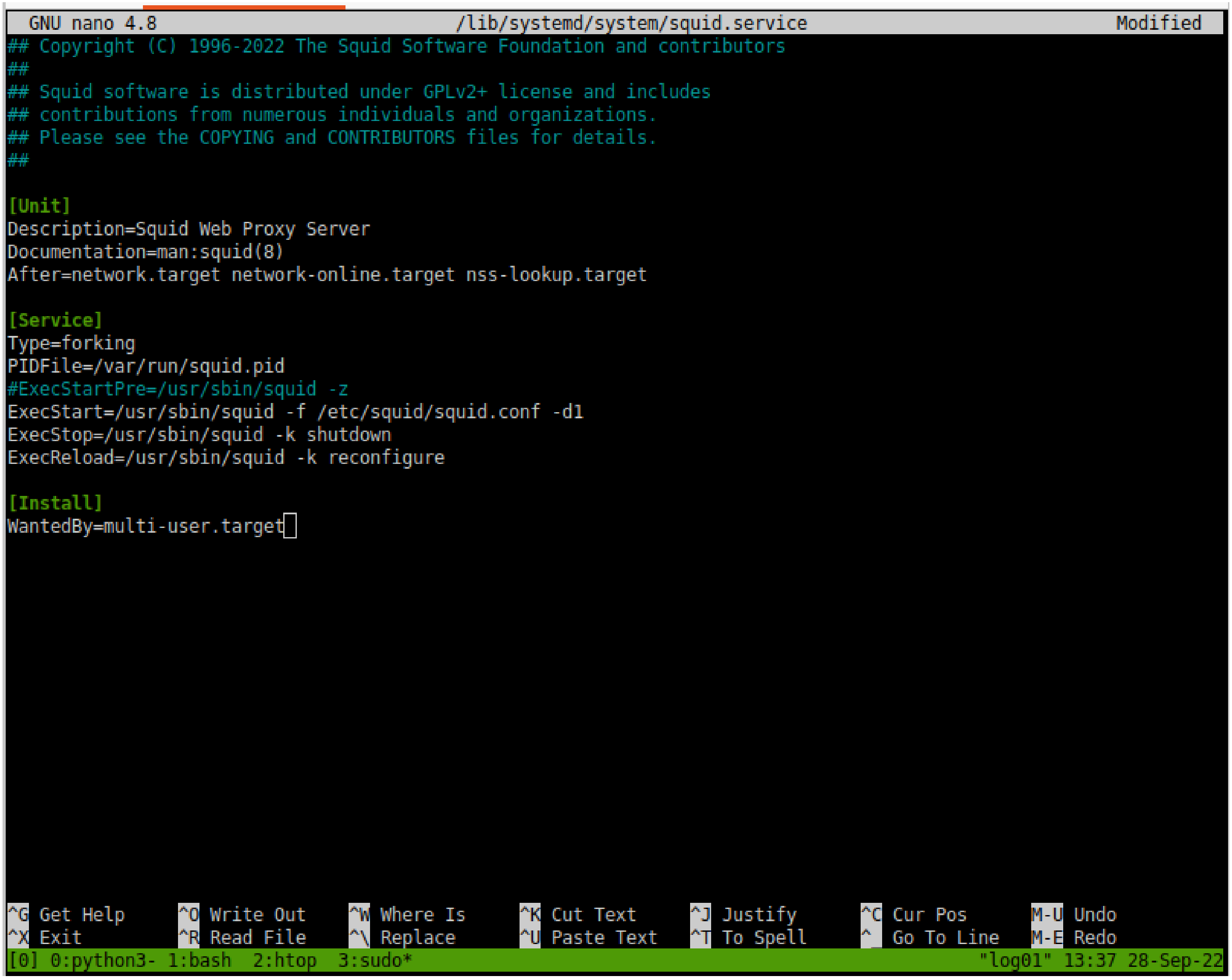
```
After=network.target network-online.target nss-lookup.target

[Service]
Type=forking
PIDFile=/var/run/squid.pid
#ExecStartPre=/usr/sbin/squid -z
ExecStart=/usr/sbin/squid -f /etc/squid/squid.conf -d1
ExecStop=/usr/sbin/squid -k shutdown
ExecReload=/usr/sbin/squid -k reconfigure

[Install]
WantedBy=multi-user.target
```

Bu şablonu kopyalayıp aşağıdaki komutu kullanarak yeni systemd servis dosyasının içine kopyalayalım

```
sudo nano /lib/systemd/system/squid.service
```

```
root@log01:~#
root@log01:~# sudo nano /lib/systemd/system/squid.service
```

kopyaladıktan sonra aşağıdaki gibi görünmelidir.



```
  GNU nano 4.8                          /lib/systemd/system/squid.service                            Modified
## Copyright (C) 1996-2022 The Squid Software Foundation and contributors
##
## Squid software is distributed under GPLv2+ license and includes
## contributions from numerous individuals and organizations.
## Please see the COPYING and CONTRIBUTORS files for details.
##

[Unit]
Description=Squid Web Proxy Server
Documentation=man:squid(8)
After=network.target network-online.target nss-lookup.target

[Service]
Type=forking
PIDFile=/var/run/squid.pid
#ExecStartPre=/usr/sbin/squid -z
ExecStart=/usr/sbin/squid -f /etc/squid/squid.conf -d1
ExecStop=/usr/sbin/squid -k shutdown
ExecReload=/usr/sbin/squid -k reconfigure

[Install]
WantedBy=multi-user.target
```

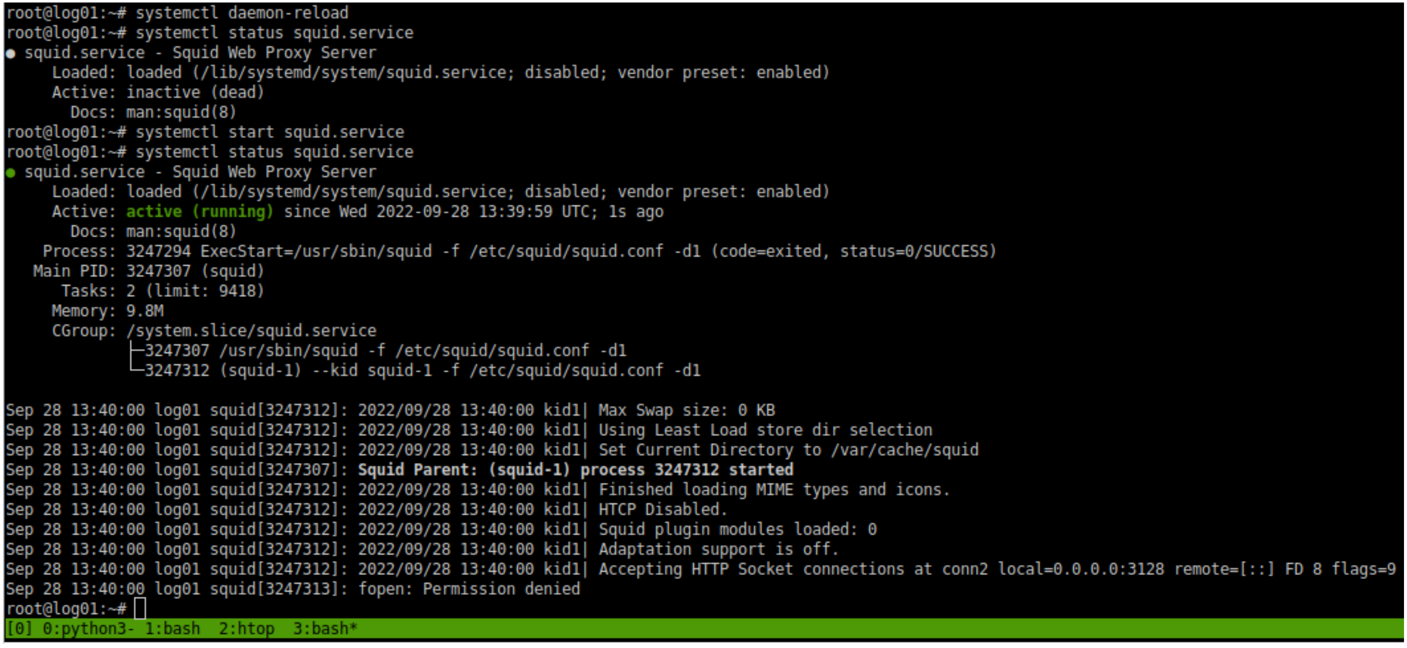
ctrl+x, yes, yes yapıp çıkıyoruz

```
root@log01:~# systemctl daemon-reload
```

yapıp dameon dosyalarının yeniden yüklendiğine emin oluyoruz

```
systemctl status squid.service
```

yaptıktan sonra squid servisinin çalıştığını görebilirsiniz.



```
root@log01:~# systemctl daemon-reload
root@log01:~# systemctl status squid.service
● squid.service - Squid Web Proxy Server
     Loaded: loaded (/lib/systemd/system/squid.service; disabled; vendor preset: enabled)
     Active: inactive (dead)
       Docs: man:squid(8)
root@log01:~# systemctl start squid.service
root@log01:~# systemctl status squid.service
● squid.service - Squid Web Proxy Server
     Loaded: loaded (/lib/systemd/system/squid.service; disabled; vendor preset: enabled)
     Active: active (running) since Wed 2022-09-28 13:39:59 UTC; 1s ago
       Docs: man:squid(8)
    Process: 3247294 ExecStart=/usr/sbin/squid -f /etc/squid/squid.conf -d1 (code=exited, status=0/SUCCESS)
   Main PID: 3247307 (squid)
      Tasks: 2 (limit: 9418)
     Memory: 9.8M
     CGroup: /system.slice/squid.service
             ├─3247307 /usr/sbin/squid -f /etc/squid/squid.conf -d1
             └─3247312 (squid-1) --kid squid-1 -f /etc/squid/squid.conf -d1

Sep 28 13:40:00 log01 squid[3247312]: 2022/09/28 13:40:00 kid1| Max Swap size: 0 KB
Sep 28 13:40:00 log01 squid[3247312]: 2022/09/28 13:40:00 kid1| Using Least Load store dir selection
Sep 28 13:40:00 log01 squid[3247312]: 2022/09/28 13:40:00 kid1| Set Current Directory to /var/cache/squid
Sep 28 13:40:00 log01 squid[3247307]: Squid Parent: (squid-1) process 3247312 started
Sep 28 13:40:00 log01 squid[3247312]: 2022/09/28 13:40:00 kid1| Finished loading MIME types and icons.
Sep 28 13:40:00 log01 squid[3247312]: 2022/09/28 13:40:00 kid1| HTCP Disabled.
Sep 28 13:40:00 log01 squid[3247312]: 2022/09/28 13:40:00 kid1| Squid plugin modules loaded: 0
Sep 28 13:40:00 log01 squid[3247312]: 2022/09/28 13:40:00 kid1| Adaptation support is off.
Sep 28 13:40:00 log01 squid[3247312]: 2022/09/28 13:40:00 kid1| Accepting HTTP Socket connections at conn2 local=0.0.0.0:3128 remote=[::] FD 8 flags=9
Sep 28 13:40:00 log01 squid[3247313]: fopen: Permission denied
root@log01:~# 
[0] 0:python3- 1:bash  2:htop  3:bash*
```

Konfigürasyon

squid konfigürasyon dosyalarının olduğu yere gidiyoruz

```
cd /etc/squid/
```

mevcut konfigürasyonu yedekliyoruz.

```
root@log01:/etc/squid# mv squid.conf squid.conf.disabled
```

Yeni konfigürasyonu yapıştıracağımız dosyayı nano kullanarak açıyoruz

```
root@log01:/etc/squid# nano squid.conf
```

Aşağıdaki şablonu yeni açtığımız dosya içerisine kaydediyoruz.

```
# Recommended minimum configuration:
#
##
# NTLM
##
#auth_param ntlm program /usr/bin/ntlm_auth --diagnostics --helper-protocol=squid-2.5-ntlmssp --domain=BANKA
#auth_param ntlm children 10
```

```
#auth_param ntlm keep_alive off
#icap_send_client_username on
#acl lan proxy_auth REQUIRED



# Example rule allowing access from your local networks.
# Adapt to list your (internal) IP networks from where browsing
# should be allowed
acl localnet src 0.0.0.1-0.255.255.255□# RFC 1122 "this" network (LAN)
acl localnet src 10.0.0.0/8□□# RFC 1918 local private network (LAN)
acl localnet src 100.64.0.0/10□□# RFC 6598 shared address space (CGN)
acl localnet src 169.254.0.0/16 □# RFC 3927 link-local (directly plugged) machines
acl localnet src 172.16.0.0/12□□# RFC 1918 local private network (LAN)
acl localnet src 192.168.0.0/16□□# RFC 1918 local private network (LAN)
acl localnet src fc00::/7       □# RFC 4193 local private network range
acl localnet src fe80::/10      □# RFC 4291 link-local (directly plugged) machines

acl SSL_ports port 443
acl Safe_ports port 80□□# http
acl Safe_ports port 21□□# ftp
acl Safe_ports port 443□□# https
acl Safe_ports port 70□□# gopher
acl Safe_ports port 210□□# wais
acl Safe_ports port 1025-65535□# unregistered ports
acl Safe_ports port 280□□# http-mgmt
acl Safe_ports port 488□□# gss-http
acl Safe_ports port 591□□# filemaker
acl Safe_ports port 777□□# multiling http

#
# Recommended minimum Access Permission configuration:
#
# Deny requests to certain unsafe ports
http_access deny !Safe_ports

# Deny CONNECT to other than secure SSL ports
http_access deny CONNECT !SSL_ports

# Only allow cachemgr access from localhost
```

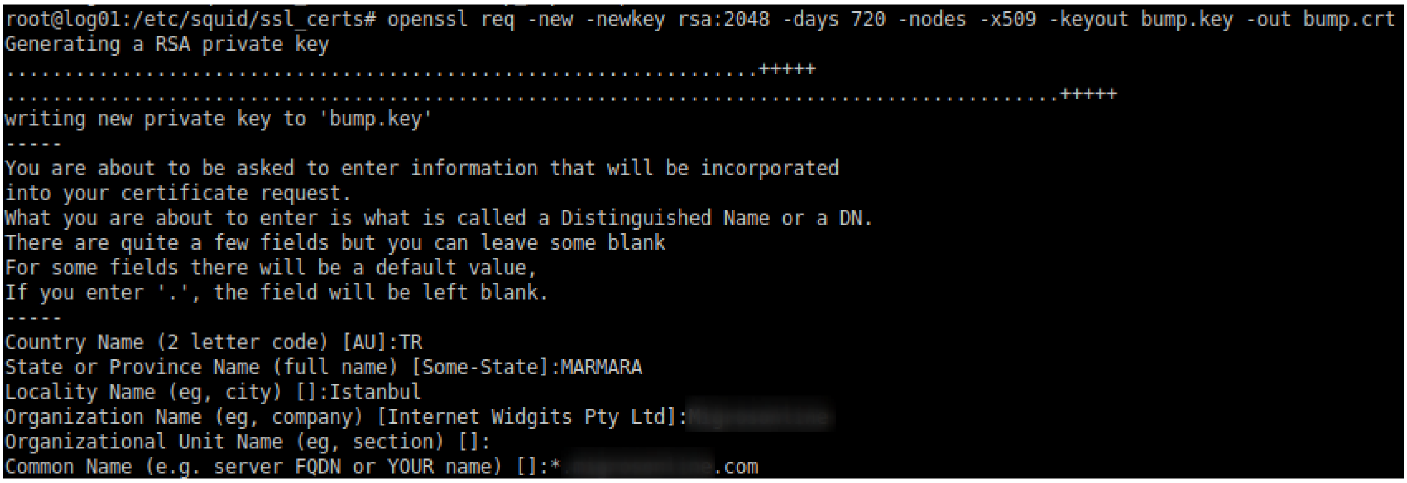
```
http_access allow localhost manager
http_access deny manager

# We strongly recommend the following be uncommented to protect innocent
# web applications running on the proxy server who think the only
# one who can access services on "localhost" is a local user
#http_access deny to_localhost

#
# INSERT YOUR OWN RULE(S) HERE TO ALLOW ACCESS FROM YOUR CLIENTS
#

# Example rule allowing access from your local networks.
# Adapt localnet in the ACL section to list your (internal) IP networks
# from where browsing should be allowed


#http_access allow localnet
http_access allow localhost


# And finally deny all other access to this proxy
http_access allow all

# Squid normally listens to port 3128
#http_port 3128
http_port 3128 tcpkeepalive=60,30,3 ssl-bump generate-host-certificates=on dynamic_cert_mem_cache_size=20MB cert=/etc/squid/bump.crt key=/etc/squid/bump.key cipher=HIGH:MEDIUM:!LOW:!RC4:!SEED:!IDEA:!3DES:!MD5:!EXP:!PSK:!DSS options=NO_TLSv1,NO_SSLv3,NO_SSLv2,SINGLE_DH_USE,SINGLE_ECDH_USE tls-dh=prime256v1:/etc/squid/bump_dhparam.pem

# Uncomment and adjust the following to add a disk cache directory.
#cache_dir ufs /var/cache/squid 100 16 256

# Leave coredumps in the first cache dir
coredump_dir /var/cache/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
sslcrtd_program /usr/lib/squid/security_file_certgen -s /var/lib/squid/ssl_db -M 20MB
sslproxy_cert_error allow all
ssl_bump stare all
```

Sertifikalandırma

kullanıcılara gönderilecek sertifikayı generate ediyoruz

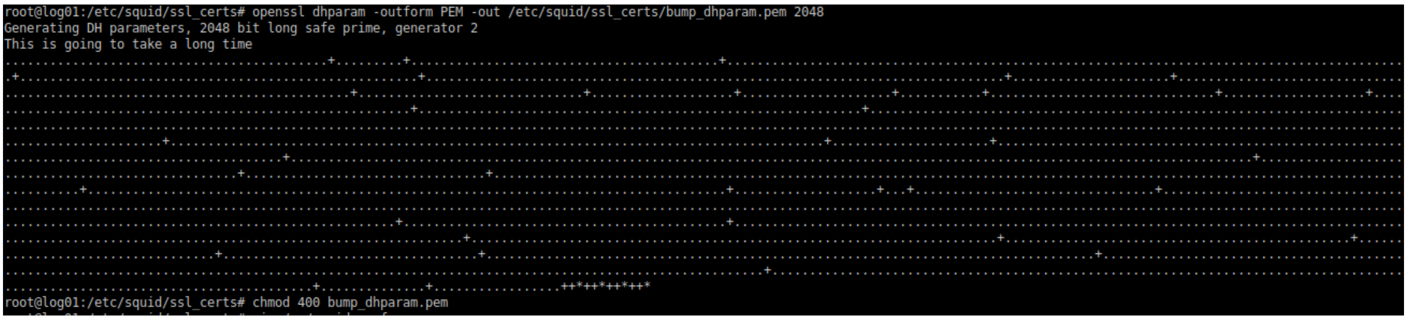
```
openssl req -new -newkey rsa:2048 -days 720 -nodes -x509 -keyout bump.key -out bump.crt
```



```
root@log01:/etc/squid/ssl_certs# openssl req -new -newkey rsa:2048 -days 720 -nodes -x509 -keyout bump.key -out bump.crt
Generating a RSA private key
.................................................................+++++
..........................................................................................+++++
writing new private key to 'bump.key'
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:TR
State or Province Name (full name) [Some-State]:MARMARA
Locality Name (eg, city) []:Istanbul
Organization Name (eg, company) [Internet Widgits Pty Ltd]:[redacted]
Organizational Unit Name (eg, section) []:
Common Name (e.g. server FQDN or YOUR name) []:*[redacted].com
```

Otomatik sertifika jenerasyonu için kullanacağımız parametrelere örnek dosyayı oluşturuyoruz.

```
openssl dhparam -outform PEM -out /etc/squid/ssl_certs/bump_dhparam.pem 2048
chmod 400 bump_dhparam.pem
```



```
root@log01:/etc/squid/ssl_certs# openssl dhparam -outform PEM -out /etc/squid/ssl_certs/bump_dhparam.pem 2048
Generating DH parameters, 2048 bit long safe prime, generator 2
This is going to take a long time
root@log01:/etc/squid/ssl_certs# chmod 400 bump_dhparam.pem
```

Sertifikanını güvenlik ayarlarını yapıyoruz

```
chown proxy:proxy /etc/squid/bump*
chmod 400 /etc/squid/bump*
```

Ubuntu için çalışacak olan ssl sertifikalarını içeren klasör ve veri tabanını oluşturuyoruz

```
mkdir -p /var/lib/squid/ssl_db
/usr/lib/squid/ssl_crtd -c -s /var/lib/squid/ssl_db
```

devreye alma

Yukarıdaki adımların tamamını yaptıktan sonra,

```
systemctl restart squid.services
```

ve kontrol için

```
systemctl status squid.service
```

Bu noktadan sonra ;

- oluşturduğumuz ssl sertifikasını clientlara gönderip trusted root authorities klasörüne import ediyoruz
- client üzerinde proxy ayarlarını sunucu_ip _adresi:3128 gösterecek şekilde yapıyoruz.

test için

```
curl -kv -x http://172.17.21.7:3128 https://docs.mikronet.tech
```

Revision #1
Created 9 November 2022 11:55:13 by Mesut Bayrak
Updated 9 November 2022 12:00:52 by Mesut Bayrak