Inhaltsverzeichnis

Ubuntu/Debian-Router mit iptables und Squid-Cache

Um einen Router mit Proxy unter Ubuntu zu realisieren bedarf es einer besondenderen Konfiguration. Zunächst muss aller ankommender Netzwerkverkehr vom System weitergereicht und maskiert werden. (MASQUERADE)

Dabei wird die Quell-IP aller SYN-Pakete durch die Router-IP ersetzt und weitergegeben (FORWARDING).

Alle TCP-Anfragen auf Port 80 sollen jedoch über den Proxy geschoben werden. Dieser lauscht in der Regel auf Port 3128.

Squid-Cache

Zunächst setzen wir den Proxyserver Squid auf. Um diesen nun aber nicht jedem Klienten mitteilen zu müssen betreiben wir diesen als transparenten Proxyserver.
Das bedeutet, dass dieser anonym im Hintergrund arbeitet, ohne das der Nutzer etwas davon mitbekommt.
Somit besteht die Möglichkeit den Datenverkehr mitzuschneiden und über den Squid-Cache zu puffern/beschleunigen.
Hier ein Auszug aus meiner Squid-Konfiguration:

```
sudo vim /etc/squid/
squid.conf# Zum Ausgeben ohne Kommentarzeilen:
cat /etc/squid/squid.conf | egrep -v "(^#.*|^$)"
```

/etc/squid/squid.conf

```
acl all src all
acl manager proto cache_objectacl localhost src 127.0.0.1/32
acl to_localhost dst 127.0.0.0/8 0.0.0.0/32
# Wir erlauben den Internetzugriff von folgenden Netzwerken aus
# bzw deklarieren wir erst eine acl und erlauben sie weiter unten...
acl localnet src 10.0.0.0/8# RFC1918 possible internal network
acl localnet src 172.16.0.0/12# RFC1918 possible internal network
acl localnet src 192.168.1.0/255.255.255.0
# RFC1918 possible internal network
#acl localnet arp 00:00:00:00:00:00 geht auch, dann muss aber der squid mit '--
acl SSL_ports port 443          # https
acl SSL_ports port 563          # snews
acl SSL_ports port 873          # rsync
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
acl Safe_ports port 631         # cups
acl Safe_ports port 873         # rsync
```

```
acl Safe_ports port 901         # SWAT
acl purge method PURGE
acl CONNECT method CONNECT
# Wir können den Netzwerkverkehr auch an einen weiteren Proxy leiten...
# Am Beispiel auf die 192.168.1.254 Port 80
#cache_peer 192.168.1.254 parent 80 7 no-query default
never_direct allow all
http_access allow manager localhost
http_access deny manager
http_access allow purge localhost
http_access deny purgehttp_access deny !
Safe_portshttp_access deny CONNECT !
SSL_ports
http_access allow localnet
http_access allow localhost
http_access deny all
icp_access allow localnet
icp_access deny all
# WICHTIG
# Hier wird deklariert auf welchem Port squid lauscht und die Funktion des tran
# Proxyservers wird hier aktiviert
#
http_port 192.168.1.10:3128
 transparent
hierarchy_stoplist cgi-bin ?
# Wie groß darf ein Objekt im RAM sein
maximum_object_size_in_memory 16
 KB
# Nach welcher Taktik speichern wir: Least Frequently Used with Dynamic Aging
memory_replacement_policy heap LFUDA
cache_replacement_policy heap LFUDA
# WICHTIG
# Man sollte nicht vergessen, dem squid zu sagen wieviel Platz er denn auf der
# benutzen darf. Hier sind es 20 GB. Standardmäßig sind 100MB eingestellt
#
cache_dir ufs /var/spool/squid 20000 16 256
# Wie groß darf ein Objekt auf der Platte sein
maximum_object_size 800
 MB
# Wohin mit dem Log-File?
access_log /var/log/squid/
access.log squid
#anderes
refresh_pattern ^ftp:           1440    20%     10080
refresh_pattern ^gopher:        1440    0%      1440
refresh_pattern -i (/cgi-bin/|\?) 0     0%      0
refresh_pattern (Release|Package(.gz)*)$        0       20%     2880
refresh_pattern .               0       20%     4320
```

```
acl shoutcast rep_header X-HTTP09-First-Line ^ICY.[0-9]
upgrade_http0.9 deny shoutcast
acl apache rep_header Server ^Apache
broken_vary_encoding allow apache
extension_methods REPORT MERGE MKACTIVITY CHECKOUThosts_file /etc/
hostscoredump_dir /var/spool/squid
```

Neustart des squid-Daemons

```
sudo service squid restart
```

Tool zum Auswerten des Squid-Logfile (squidview)

```
sudo squidview
```

Squidview greift immer auf die Logfiles in /root/.squidview/log1 bis log3 zu.

Um ein Logfile in squidview einzubinden müssen wir einen symbolischen Link erstellen:

```
ln -s /path/to/the/access.log /root/.squidview/log1
```

Die Datei, um in Squidview statt den IP-Adressen die Namen anzuzeigen

```
sudo vim /root/.squidview/
aliases
user1 192.168.1.100
user2 192.168.1.101
user3 192.168.1.102
user4 192.168.1.103user5 192.168.1.104
```

Beim Start vin squidview folgende Tasten hintereinander drücken "c" "a" "q", dann nimmt squidview die aliases-Datei.

HTTP-Traffic zum Squid-Cache umleiten (flüchtige und statische Lösung)

Aller eingehender TCP-Datenverkehr muss zum Squid-Port weitergeschoben werden:

```
sudo iptables -t nat -A PREROUTING -i eth0 -p tcp -m tcp --dport 80 -j
 REDIRECT --to-ports 3128
```

Um diese Regel beim Systemstart anzuwenden, speichern wir diese in der rc.local

```
sudo vim /etc/rc.local
```

```
/sbin/iptables --table nat -A PREROUTING -i eth0 -p tcp -m tcp --dport 80 -j
 REDIRECT --to-ports 3128
```

IP-Forward im Kernel aktivieren (statische Lösung)

Natürlich muss in den Systemeinstellung das Weiterleiten vin IP-Paketen aktiviert werden:

```
sudo vim /etc/sysctl.conf
```

```
# Uncomment the next line to enable packet forwarding for IPv4
net.ipv4.ip_forward=1
# Uncomment the next line to enable packet forwarding for IPv6
#net.ipv6.conf.all.forwarding=1
```

Allen anderen Datenverkehr durchreichen (statische Lösung)

```
sudo vim /etc/rc.local
```

```
#Ausgehenden Netzwerkverkehr maskieren
/sbin/iptables --table nat -A POSTROUTING -o eth0 -j
 MASQUERADE
#Ausgehenden neuen Netzwerkverkehr von internen IPs erlauben
/sbin/iptables -A FORWARD -s 192.168.0.0/16 -o eth0 -m conntrack --ctstate
 NEW -j
 ACCEPT
#Ausgehenden Netzwerkverkehr, den wir schon kennen...durchreichen
/sbin/iptables -A FORWARD -m conntrack --ctstate RELATED,ESTABLISHED -j
 ACCEPT
```