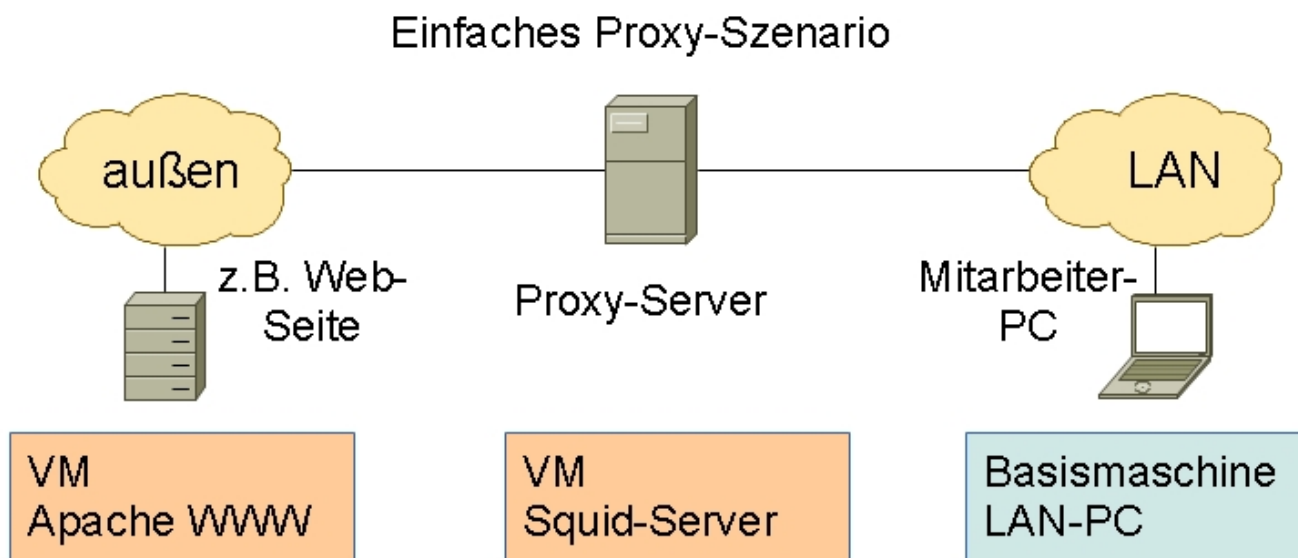


# Proxy-Laborübung

## Überblick

Die folgende Abbildung zeigt den prinzipiellen Laboraufbau für die Proxy-Übung.



Die Übung wird u.a. auf virtuellen Ubuntu Maschine durchgeführt. Insbesondere der Proxy selbst wird virtualisiert betrieben. HINWEIS: Bei der Arbeit mit virtuellen Maschinen sind im [Labor einige Details](#) zu beachten.

## Vorbereitung

Als Proxy-Server wird das Paket `squid` eingesetzt. Diese Software bietet einen weiten Funktionsumfang im Bereich Proxy-Server an. Sollte die Software noch nicht installiert sein, so kann dies mittels:

```
sudo apt-get install squid3
```

nachgeholt werden. Die Version 2.7 ist veraltet und sollte für Neuinstallationen nicht mehr eingesetzt werden. Die Default-Konfiguration findet standardmäßig in `/etc/squid/squid.conf` statt. Der squid-Server wird mittels der folgenden Befehle gestartet bzw. gestoppt.

```
sudo service squid3 [start|stop|--full-restart]
```

**ACHTUNG:** Es kann immer nur einer der Befehle in den eckigen Klammern verwendet werden! Wenn eine eigene Konfiguration statt der default-config verwendet werden soll, so kann dies squid mittels des Parameters `-f` mitgeteilt werden.

```
sudo squid3 -f /PFAD/squid.conf
```

Alternativ kann der Proxy auch im Vordergrund -N gestartet werden und mit -d der Debug-Level eingestellt werden.

```
sudo squid3 -N -d 1
```

## Weiterführende Informationen zu squid:

Die Konfigurationsoptionen sind hier zu finden: ([gekürzte Version](#))

Die vollständige Übersicht befindet sich hier: <http://www.squid-cache.org/Doc/config/>

Auf derselben Seite findet man auch sehr viele Beispiele zu unterschiedlichen Szenarien:

<http://wiki.squid-cache.org/ConfigExamples>

Sehr ausführliches deutsche Handbuch: <http://www.squid-handbuch.de/hb/>

Man-Page zu squid: <http://linux.die.net/man/8/squid>

Anleitung für Cache: <http://www.gulp.de/kb/pt/techexpert/tintenfisch.html>

Eine sehr verkürzte Konfiguration (nicht lauffähig) sieht wie folgt aus:

### [squid\\_basic.conf](#)

```
http_port 192.168.10.4:8088
cache_mem 16 MB
cache_dir ufs /PFAD/cache 10016 256
cache_access_log /PFAD/logs/access.log
cache_log /PFAD/logs/cache.log

# Zugriffe mit loggen
access_log /var/log/squid3/access.log squid

# Einfach Access-Regeln
# Alle anderen Quellen aus dem angegebenen Netz dürfen alles
acl localnet src 192.168.0.0/16
http_access allow localnet
```

### [squid\\_medium.conf](#)

```
http_port 192.168.16.202:8080
access_log /var/log/squid3/access.log squid

cache_peer 192.168.16.91 parent 3128 0 no-query default

client_netmask 255.255.0.0

acl SSL method CONNECT
never_direct allow SSL

acl no_access src 192.168.16.101
http_access deny no_access
```

```
#acl bad_sites url_regex -i "/etc/squid3/bad_sites"  
acl bad_sites url_regex -i kopfload.de google.de  
http_access deny bad_sites  
  
acl localnet src 192.168.0.0/16  
http_access allow localnet
```

Eine aufwendigere Konfiguration, die allerdings noch angepasst werden muss.

### squid\_full.conf

```
# Üblicherweise wartet Squid auf dem Port 3128 auf Anfragen.  
#http_port 3128  
# Binden an ein internes Interface hier LISTEN_IP  
# ACHTUNG: Die ACL (s.u.) muss entsprechend angepasst werden.  
http_port LISTEN_IP:3128  
  
# TAG: cache_mem zusätzlicher Speicherverbrauch für aktivierten Cache  
in (bytes)  
cache_mem 32 MB  
  
# TAG: maximum_object_size Max. Größe gecachter Dateien in (bytes)  
DEFAULT 4096 KB  
maximum_object_size 10000 KB  
  
# TAG: maximum_object_size_in_memory (bytes)  
maximum_object_size_in_memory 32 KB  
  
# TAG: cache_replacement_policy Verfahren um Speicherplatz freizugeben  
cache_replacement_policy heap LFUDA  
  
# TAG: memory_replacement_policy Verfahren um Speicherplatz  
freizugeben  
memory_replacement_policy heap LFUDA  
  
# TAG: cache_dir Speicherort des Caches  
cache_dir ufs /var/spool/squid3 2000 16 256  
  
# TAG: client_netmask Datenschutz vollständige IP wird gespeichert  
client_netmask 255.255.255.255  
# TAG: client_netmask Datenschutz letztes Oktett=0 (anonym)  
#client_netmask 255.255.255.0  
  
# TAG: forwarded_for IP Adresse verbergen  
forwarded_for off  
  
#Squid VIA Header ausschalten  
#via off  
via on
```

```
# Vorgelagerter Proxy (Master-Proxy)
# TAG: cache_peer
#
#           hostname          type      proxy  icp
#           hostname          type      port   port  options
cache_peer IP_ADDR_MASTER    parent    80    0    no-query default

# DNS-Server
dns_nameservers IP_ADRESSE:80

# Pfad zur Process-ID Datei
pid_filename /var/run/squid.pid
# Deutsche Fehlermeldungen
error_directory /usr/share/squid3/errors/de

# ACL Access Control List
# Form: acl <frei_definierbarer_Name> <acl_Typ> <Werte>
# Die Reihenfolge der Freigaben ist entscheidend!
# Würde zuerst ein http_access deny all gesetzt, kann man darunter
# keinen Zugriff mehr einrichten.
# Daher sollten Freigaben möglichst am Anfang der squid.conf stehen.

# Gezielt Seiten sperren. Alles was in der Datei bad-sites.squid steht
# wird gesperrt. z.B. Facebook, Youtube
acl bad_url dstdomain "/etc/squid3/bad-sites.squid"
http_access deny bad_url

# Alternative Proxys blocken
acl anon-prox-sites url_regex -i "/squid3/blocked/keywords"
http_access deny anon-prox-sites

# https Anfragen nicht selbst beantworten, sondern über parent (Schul)
# Proxy laufen lassen
acl SSL method CONNECT
never_direct allow SSL

# Beispiel: Alles erlauben, dass auch dem entsprechenden Subnetz kommt.
#acl all src 192.168.0.0/255.255.255.0
acl all src 192.168.XX.0/255.255.255.0
http_access allow all

# Beispiel 2: Alles aus dem Bereich freigeben.
#acl freigegeben2 src 192.168.0.1-192.168.0.11
#http_access allow freigegeben2

# Beispiel 3: Genau eine Maschine freigeben.
#acl testpc src 192.168.30.1
#http_access allow testpc
```

## Aufgabe 1

Die Mitarbeiter-PC sollen über den squid-Server auf einen Internetauftritt zugreifen können. Zunächst soll dies transparent erfolgen, d.h. ohne Eingriffe durch den Proxy selbst. Lesen Sie dazu die `access.log` Datei aus bzw. geben Sie den Inhalt mittels des folgenden Befehls in der Kommandozeile aus:

```
tail -f /PFAD/logs/access.log
```

Der Pfad muss demjenigen entsprechen, den Sie in der Konfiguration unter `access_log` vorgegeben haben.

Setzen Sie sich mit den Angaben im `access.log` auseinander.

**Warum taucht TCP\_MISS/200 in den Zeilen auf? Wie kommt es, dass der Proxy MISS angibt, obwohl die Seite ausgeliefert wurde?**

## Aufgabe 2

Nun soll der Zugriff kontrolliert werden. Folgende Einschränkungen sollen ausgetestet werden:

- Es sollen nur Maschinen aus dem vorher definierten IP-Bereich zugreifen.
- Es sollen bestimmte URLs nicht erreichbar sein. Hier soll statt dessen eine Hinweis-Seite erscheinen
- Es soll nur EINE bestimmte Maschinen (ZIEL) erreichbar sein.

Betrachten Sie dazu die TAGs `acl` und `http_access` mit den dazugehörigen Parametern

## Zusatzaufgabe

1. Schalten Sie zwei Proxy hintereinander. Betrachten Sie dazu den Parameter `cache_peer`.
2. Ändern Sie die Fehlermeldungen, die der Proxy liefert. Betrachten Sie dazu den Parameter `error_directory`.

## Einrichten der VM

`/etc/apt/apt.conf`

```
Acquire::http::proxy "http://raumserver:3128/";  
Acquire::https::proxy "https://raumserver:3128/";  
Acquire::ftp::proxy "ftp://raumserver:3128/";  
Acquire::socks::proxy "socks://raumserver:3128/";
```

`/etc/apt/sources.list`

```
deb http://ftp.halifax.rwth-aachen.de/ubuntu/ trusty main restricted

deb http://ftp.halifax.rwth-aachen.de/ubuntu/ trusty-updates main restricted
deb-src http://ftp.halifax.rwth-aachen.de/ubuntu/ trusty-updates main
restricted

deb http://ftp.halifax.rwth-aachen.de/ubuntu/ trusty universe
deb-src http://ftp.halifax.rwth-aachen.de/ubuntu/ trusty universe
deb http://ftp.halifax.rwth-aachen.de/ubuntu/ trusty-updates universe
deb-src http://ftp.halifax.rwth-aachen.de/ubuntu/ trusty-updates universe

deb http://ftp.halifax.rwth-aachen.de/ubuntu/ trusty multiverse
deb-src http://ftp.halifax.rwth-aachen.de/ubuntu/ trusty multiverse
deb http://ftp.halifax.rwth-aachen.de/ubuntu/ trusty-updates multiverse
deb-src http://ftp.halifax.rwth-aachen.de/ubuntu/ trusty-updates multiverse

deb http://ftp.halifax.rwth-aachen.de/ubuntu/ trusty-backports main
restricted universe multiverse
deb-src http://ftp.halifax.rwth-aachen.de/ubuntu/ trusty-backports main
restricted universe multiverse
```

From:  
<https://www.kopfload.de/> - **kopfload - Lad Dein Hirn auf!**

Permanent link:  
[https://www.kopfload.de/doku.php?id=lager:oeff\\_netze:proxy\\_labor&rev=1682862422](https://www.kopfload.de/doku.php?id=lager:oeff_netze:proxy_labor&rev=1682862422)

Last update: **2025/11/19 16:13**

