

# Squid: Proxy-Server unter Linux

› Mit einem Web-Proxy lassen sich nicht nur das Übertragungsvolumen und damit die Kosten in Grenzen halten. Er beschleunigt auch gecachte Webseiten und kann unerwünschte Inhalte komplett sperren.

› VON OLIVER DREES

---

Viele Administratoren schalten zwischen Internet und Firmennetz einen Web-Proxy. Dieser bietet nicht nur die Möglichkeit, den Zugang auf bestimmte Internet-Angebote zu beschränken, sondern verringert auch das Datenvolumen. Bei Internet-Zugängen, die volumenabhängig abgerechnet werden, lassen sich so Kosten sparen. Zudem erhöht sich die Zugriffsgeschwindigkeit auf bereits zwischengespeicherte Daten.

Ein Proxy-Server hat im Netzwerk eine Vermittlungsfunktion. Er nimmt Anfragen von den Anwendern entgegen, lädt Daten aus dem Internet und leitet diese an den User weiter. Dabei können sowohl HTTP- als auch FTP-Inhalte über einen Proxy angefordert werden.

Der Server legt alle angeforderten Daten in einem Cache ab. Bevor der Proxy Daten aus dem Internet holt, wird überprüft, ob diese bereits im Cache vorliegen und noch aktuell sind. Ist dies der Fall, liefert der Server die Internet-Inhalte aus dem lokalen Speicher. Bei Daten, die häufig von verschiedenen Benutzern abgerufen werden, ergibt sich hieraus eine deutliche Reduzierung des Übertragungsvolumens. Zudem liefert der Cache die Inhalte schneller, als wenn man diese aus dem Internet lädt.

Wir beschreiben in diesem Artikel die Installation und Konfiguration des beliebten Linux-Proxys Squid. Dazu dient folgendes Szenario: Der Proxy soll ausschließlich authentifizierten Benutzern den Zugriff auf das Internet gewähren, deren Rechner zusätzlich über die IP-Adresse auf dem Server freigegeben ist. Das Abrufen von Internet-Inhalten soll nur zu bestimmten Zeiten möglich sein und bekannte Direct-Brokering-Seiten werden gesperrt.

## › Proxy-Server: Zugriffskontrollen

Ein Proxy-Server lässt sich so einstellen, dass nur bestimmte Anwender auf diesen zugreifen können und bestimmte Internet-Inhalte gesperrt sind. Es gibt drei Arten der Zugriffskontrolle:

- › Anonymous Proxy: Anfragen werden von beliebigen Rechnern und Anwendern entgegen genommen.
- › Proxy-Server mit Authentifizierung: Erst nach erfolgreicher Authentifizierung des Anwenders durch Benutzername und Passwort werden Anfragen ausgeführt.
- › Proxy-Server mit Zugriffsregeln (ACL / Access Control List): Diese legen beispielsweise fest, zu welchen Zeiten bestimmte Daten aus dem Internet geladen werden dürfen.

Abhängig von den Proxy-Einstellungen werden bei Zugriffsverletzungen bestimmte Aktionen ausgeführt. Eine unerlaubte Anfrage kann so das Umleiten an eine andere Adresse im internen Netzwerk zur Folge haben.



**Forbidden: Versucht ein Anwender, eine gesperrte Seite aufzurufen, erscheint diese Meldung.**

Die Kontrolle der Zugriffsbeschränkungen kann man entweder Squid selbst überlassen oder ein externes Programm wie SquidGuard verwenden.

## › Squid: Installation

Die derzeit aktuelle Version 2.4 von Squid kann man unter [www.squid-cache.org](http://www.squid-cache.org) (<http://www.squid-cache.org>) herunterladen. Nach dem Download entpacken Sie zunächst den Tarball und wechseln in das entsprechende Verzeichnis:

```
» tar xzvf squid-2.4.STABLE1-src.tar.gz
» cd squid-2.4.STABLE1-src
```

Nun müssen Sie Squid konfigurieren. Dazu geben Sie folgenden Befehl ein:

```
» ./configure --prefix=/usr/local/squid --sysconfdir=/etc/squid «
```

Mit dem Schalter `--prefix` legen Sie das Zielverzeichnis fest, mit `--sysconfdir` den Speicherplatz für die Konfigurationsdatei. In unserem Beispiel dürfen Anwender ausschließlich mit einem gültigen Benutzernamen und Passwort auf den Proxy-Server zugreifen. Dafür stehen in den Unterverzeichnissen von `auth_modules` folgende Authentifizierungs-Mechanismen zur Verfügung: LDAP, MSNT, NCSA, [PAM](http://www.kernel.org/pub/linux/libs/pam/) (<http://www.kernel.org/pub/linux/libs/pam/>), SMB und getpwnam. Wir verwenden die NCSA-Webserver-Authentifizierung. Dazu führen Sie im entsprechenden Verzeichnis (NCSA) folgende Befehle aus:

```
» make
» su
» make install
» exit
```

Das kompilierte Authentifizierungs-Modul finden Sie im Ordner `bin`. Allerdings kann Squid mit eigenen Bordmitteln nicht alle Zugriffsbeschränkungen aus unserem Beispiel abdecken. Daher installieren wir zusätzlich das Programm SquidGuard.

## › SquidGuard: Berkeley DB Library

Für die Installation von SquidGuard wird die Version 2.x der Berkeley-DB-Library benötigt. Befindet sich diese nicht auf Ihrem Rechner, kann man sie unter [www.sleepycat.com](http://www.sleepycat.com) (<http://www.sleepycat.com>) herunterladen. Entpacken Sie nach dem Download das Archiv und wechseln Sie in das Unterverzeichnis `build_unix` des Berkeley-DB-Verzeichnisses. Nun wird das Skript `configure` wie folgt ausgeführt:

```
» ../dist/configure --prefix=/usr/local --sysconfdir=/etc/BerkeleyDB «
```

Übersetzen und installieren Sie das Programm mit folgenden Befehlen:

```
» cd ../dist/configure
» make
```

- » *su*
- » *make install*
- » *exit*

Alle notwendigen Dateien finden Sie unter *usr/local/BerkeleyDB*, die Konfigurationsdateien unter */etc/BerkeleyDB*.

### › SquidGuard: Installation

Zur Installation von [SquidGuard](http://www.squidguard.org) (<http://www.squidguard.org>) entpacken Sie den Tarball, wechseln in das neue Verzeichnis und konfigurieren das Programm:

- » *tar xzvf squidGuard-1.1.4.tar.gz*
- » *cd squidGuard-1.1.4*
- » *./configure --prefix=/usr/local/squidGuard --with-sg-config=/etc/squidGuard.conf --with-sg-logdir=/var/squidGuard/log*

SquidGuard wird in das Verzeichnis */usr/local/squidGuard* installiert, die Konfigurationsdatei */etc/squidGuard.conf* eingerichtet und als Log-Verzeichnis */var/squidGuard/log* festgelegt. SquidGuard können Sie nun kompilieren und installieren:

- » *make*
- » *su*
- » *make install*
- » *exit*

### › Konfiguration: Squid

Alle notwendigen Einstellungen wie beispielsweise den Proxy-Port (Default: 3128) oder die Zugriffsbeschränkungen nehmen Sie in einer zentralen Konfigurationsdatei vor: */etc/squid/squid.conf*. In der Regel sind jedoch wenige Änderungen nötig, eine weiter gehende Anpassung ist mit Hilfe der Erklärungen einfach zu realisieren.

Für die spätere Authentifizierung der Anwender ist es wichtig, dass Sie das dafür zuständige Programm angeben und eine ACL vom Typ *proxy\_auth* festlegen. Entsprechend den in der Konfiguration angegebenen Optionen müssen Sie als User *root* einen Benutzer und entsprechende Verzeichnisse anlegen sowie die Rechte vergeben:

- » *useradd -g nogroup -d /var/squid -c "Proxy-Squid" -s /bin/bash squid*
- » *mkdir /var/squid/cache*
- » *mkdir /var/squid/logs*
- » *chown -R squid.root /var/squid*
- » *htpasswd -c /etc/squid/passwd Benutzername*

Der letzte Befehl erstellt eine Passwortdatei. Dabei ist es wichtig, dass weitere Benutzernamen mit dem gleichen Befehl ohne die Option *-c* angelegt werden. Nun kann der Cache initialisiert werden:

- » *su squid*
- » */usr/local/squid/bin/squid -z*
- » *exit*

Die Zugriffsbeschränkungen richten Sie mit Hilfe des Programms SquidGuard ein.

### › Konfiguration: SquidGuard

Legen Sie zunächst eine Konfigurationsdatei mit den Namen *squidGuard.conf* im Verzeichnis */etc* an. [Hier](http://www.tecchannel.de/download/798/squidguard_konfigurationsbeispiel.txt) ([http://www.tecchannel.de/download/798/squidguard\\_konfigurationsbeispiel.txt](http://www.tecchannel.de/download/798/squidguard_konfigurationsbeispiel.txt)) können Sie sich eine Beispielkonfiguration herunterladen. Diese arbeitet mit einer Negativliste: "Alles was nicht verboten ist, ist erlaubt." Wenn Sie eine Positivliste verwenden wollen, muss die Erlaubnis vor den Verboten stehen. Alle Konfigurationsänderungen aktivieren Sie mit dem Befehl:

- » */usr/local/squid/bin/squid -k reconfigure* «

SquidGuard arbeitet anhand von Datenbanken oder Textdateien: Domain-Listen zum Sperren ganzer Domains, URL-Listen zum Sperren spezieller URLs oder reguläre

Ausdrücke, mit deren Hilfe Sie bestimmte Muster in den URLs und Domains verbieten. Diese Dateien und auch die Logfiles sollte man in einem zentralen Verzeichnis speichern:

```
» mkdir /var/squidGuard
» mkdir /var/squidGuard/db
» mkdir /var/squidGuard/log
```

Sie können verschiedene Datenquellen parallel verwenden, etwa um die Daten thematisch zu gliedern:

```
» mkdir /var/squidGuard/db/finance «
```

Vor einem Vergleich wandelt SquidGuard alle Zeichen in Kleinbuchstaben um. Daher muss darauf keine Rücksicht genommen werden. Legen Sie nun eine Textdatei mit dem Namen *finance.domains* unter */var/squidGuard/db/finance* an. Der Dateiname ist frei wählbar und muss nicht den hier verwendeten gleichen. In dieser Datei fügen Sie nun alle Domainnamen ein, die für den Anwender gesperrt werden sollen, für jede Domain eine separate Zeile, beispielsweise:

```
» comdirect.de
» dab.de
```

### › Zugriffskontrolle: Filtern von URLs

Sollen nur bestimmte URLs und keine kompletten Domains gesperrt werden, legen Sie im Datenbank-Ordner die Datei *finance.urls* an. Verwenden Sie für jede URL eine eigene Zeile, wobei Sie das Protokoll (http, ftp, etc.) nicht mit angeben müssen. Wollen Sie zum Beispiel den Zugriff auf den Börsenticker *http://financial.domain.de:3456/applets/ticker.class* sperren, wird daraus folgender Eintrag:

```
» financial.domain.de/applets/ticker.class «
```

Es kann vorkommen, dass eine Webseite unter mehreren URLs erreichbar ist, beispielsweise *www.financial.domain.de* und *web.financial.domain.de*. Hier vereinfacht SquidGuard dem Administrator die Arbeit, denn das Verbot der oben genannten URL schließt auch alle Subdomains ein. Ein Verbot aller Objekte einschließlich der Unterverzeichnisse erreichen Sie durch folgende Zeile:

```
» financial.domain.de/verbotenes_Verzeichnis «
```

### › Zugriffskontrolle: Filtern von Stichwörtern

Möchten Sie alle Domains oder URLs sperren, welche die Buchstabenkombination *depot* enthalten, legen Sie dazu die Datei *finance.regex* an. Verwenden Sie für jedes zu sperrende Stichwort eine eigene Zeile. Allerdings sind auch Suchanfragen bei Suchmaschinen betroffen, da in der Regel die Anfrage in der URL steht. Es besteht jedoch immer die Gefahr, dass auch erwünschte URLs gesperrt werden. Ferner reduziert sich die System-Performance, da jede URL gegen diese Stichwörter geprüft wird. Aus diesen Gründen sollten Sie reguläre Ausdrücke nur sehr sparsam verwenden.

Bei großen Domain- und URL-Listen sollte der Zugriff auf die einzelnen Sätze nicht über eine Textdatei erfolgen. Die Zugriffsgeschwindigkeit erhöht sich durch die Verwendung von Berkeley-DB-Datenbanken erheblich. Als Vorlage für die Datenbank verwenden Sie die angelegten Textdateien, die Sie mit dem Befehl:

```
» /usr/local/squidGuard/bin/squidGuard -C Dateiname «
```

umwandeln. SquidGuard verwendet automatisch die Datenbank-Version, wenn eine solche neben einer Textdatei vorhanden ist. Sollte SquidGuard beim Erzeugen der Datenbanken nicht zum Prompt zurückkehren, drücken Sie die Tastenkombination [Strg]+[C] um das Programm abzubrechen. Dem Fehler kommen Sie auf die Spur, indem Sie das Logfile einsehen, das nach der hier vorgestellten Konfiguration unter */var/squidGuard/log/squidGuard.log* zu finden ist. In den meisten Fällen handelt es sich um einfache Tippfehler in der Konfigurationsdatei.

### › Zugriffskontrolle: Administration

Zum Verwalten der Datensätze wird ein so genanntes Diff-File verwendet. Geben Sie in diesem die Datensätze in folgender Form an:

```
» +domain.hinzufügen.de
» -domain.aus.Datenbank.entfernen.de
```

Die Diff-Datei muss im gleichen Verzeichnis wie die betreffende Datenbank liegen und zwingend *Name-der-Datenbank.diff* lauten. Mit dem Befehl:

```
» /usr/local/squidGuard/bin/squidGuard -u «
```

werden alle Datenbanken aktualisiert. Die Änderungen gelten sofort, so dass Sie Squid nicht neu starten müssen. Nach der Aktualisierung löschen Sie das Diff-File. Die Verzeichnisse und Dateien gehören dem Benutzer *squid* und die Datenbank-Dateien sollten auch nur von diesem Benutzer und der entsprechenden Gruppe lesbar sein. Dadurch wird vermieden, dass jeder Anwender die komplette Liste der verbotenen Sites einsehen kann.

```
» chown -R squid.root /var/squidGuard
» chmod 640 /var/squidGuard/db/*/*
```

Zuletzt kopieren Sie das CGI-Skript *squidGuard.cgi* aus dem Verzeichnis *sample* des Quellverzeichnisses in den CGI-Ordner des Webservers. Geben Sie der Datei zudem ausführbare Rechte. Ferner sind im Skript wenige Einstellungen anzupassen, wie die E-Mail-Adresse, an die sich Benutzer wenden können, falls sie eine Sperrung für ungerechtfertigt halten. Sie haben die Möglichkeit, ein eigenes Skript zu schreiben, das Ihren Anforderungen besser entspricht.

## › Proxy-Server ins System einbinden

Als User *root* starten Sie den Proxy-Server mit folgendem Befehl:

```
» /usr/local/squid/bin/squid «
```

Über den syslog-Daemon in der Datei */var/squid/logs/cache.log* wird der Start mitprotokolliert. Bei der Fehlersuche liefert die Datei hilfreiche Informationen. Den Server beenden Sie mit

```
» /usr/local/squid/bin/squid -k shutdown «
```

Um den Proxy-Server bei jedem Systemstart zu aktivieren, können Sie ein Initscript verwenden. Ein Beispiel dafür finden Sie [hier](#)

([http://www.tecchannel.de/download/798/init\\_script\\_proxy.txt](http://www.tecchannel.de/download/798/init_script_proxy.txt)) . Unter SuSE speichern Sie dieses unter dem Namen */sbin/init.d/squid* und setzen Links aus den Runlevel-Verzeichnissen auf diese Datei:

```
» cd /sbin/init.d/rc2.d
» ln -s ../squid ./S20squid
» ln -s ../squid ./K10squid
```

## › Webalizer: Überwachen des Proxys

Nachdem Sie den Proxy-Server installiert und fertig konfiguriert haben, stellt sich die Frage nach der Überwachung. Dazu gibt es verschiedene Ansätze, die jeweils ihre Vor- und Nachteile haben. Die einfachste Art der Überwachung ist, sich die neuen Einträge des Squid-Logfiles auf einer Textkonsole ausgeben zu lassen. Dazu verwenden Sie das Kommando:

```
» tail -f /var/squid/logs/access.log «
```

Sobald ein neuer Eintrag im Logfile erzeugt wird, erscheint er auch auf der Konsole. Dadurch erhalten Sie einen aktuellen Überblick über den Verkehr. Entdecken Sie dabei Adressen, die gesperrt werden sollten, können Sie diese umgehend in die Datenbanken aufnehmen. Diese Form der Auswertung ist aber bezüglich der Proxy-Nutzung nur begrenzt aussagekräftig. Abhilfe schaffen der Cachemanager von Squid oder externe Tools wie [Webalizer](http://www.webalizer.com) (<http://www.webalizer.com>) . Das Programm dient hauptsächlich zur Auswertung von Webserver-Logfiles. In der aktuellen Version kann es auch Logfiles von Squid auswerten. Es erstellt unter anderem Übersichten über die Anzahl der Anfragen

und übertragenen Bytes sowie Tagesstatistiken.

Verwenden Sie das Programm *configure* mit den Schaltern `--enable-dns` und `--with-language=german`, um für die Statistiken DNS-Namen anstatt IP-Adressen zu erhalten. In der Datei `/etc/webalizer.conf.sample` sind alle Konfigurationsoptionen beschrieben. Kopieren Sie die Datei nach `/etc/webalizer.conf` und führen Ihre Änderungen durch. Ein Beispiel dazu finden Sie [hier](#)

([http://www.tecchannel.de/download/798/webalizer\\_konfigurationsbeispiel.txt](http://www.tecchannel.de/download/798/webalizer_konfigurationsbeispiel.txt)).

Kopieren Sie nun die Datei `/var/squid/logs/access.log` nach `/var/squid/logs/access_log` und lassen Sie Squid mit `/usr/local/squid/bin/squid -k rotate` eine neue Logdatei anlegen. Führen Sie schließlich das Kommando *webalizer* aus und die Auswertungen stehen Ihnen zur Verfügung, die Sie mit einem Webbrowser ansehen können. Es ist empfehlenswert einen Cron-Job zu erstellen, der die obigen Schritte periodisch ausführt. Dieser Schritt wird hier jedoch nicht weiter beschrieben. Um Schwierigkeiten zu vermeiden, sollten Sie in der Konfiguration von Squid die automatische Rotation mit `logfile_rotate 0` ausschalten und die Umbenennung der Logdateien selbst vornehmen.

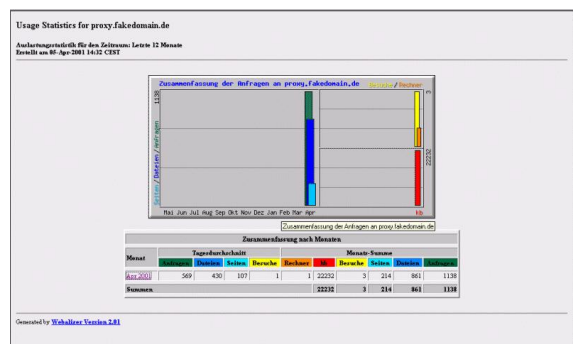
## › Webalizer: Auswertung der Log-Files

Die Auswertung über Webalizer zählt alle Anfragen. Dabei werden auch die Anfragen berücksichtigt, die der Cache selber ausgeliefert hat. Eine Übersicht über die aktuellen Daten von Squid liefert das CGI-Programm *Cachemanager*. Kopieren Sie die Datei `/usr/local/squid/bin/cachemgr.cgi` in das CGI-Verzeichnis des Webserver und setzen Sie die Dateirechte entsprechend der Webserver-Konfiguration. Vergessen Sie dabei nicht das Execute-Recht. Anschließend passen Sie die Datei `/etc/squid/squid.conf` an, damit der Cachemanager Zugriff auf Squid erhält:

- » `http_access allow manager`
- » `cachemgr_passwd password all`

Über einen Webbrowser greifen Sie über die URL

`http://proxy-server/cgi-bin/cachemgr.cgi` auf die Auswertungen zu, wobei der Benutzername *cachemgr* und das Passwort im obigen Beispiel *password* lautet.



Übersichtlich: Webalizer präsentiert Ihnen die Proxy-Statistiken kompakt und übersichtlich.

Sie können Informationen von der Auslastung des Festplatten-Caches bis hin zu den Antwortzeiten des Caches ermitteln. Interessant ist vor allem der Punkt "General/Runtime Information". Die Punkte "Request Hit Ratios" und "Byte Hit Ratios" geben Aufschluß darüber, wie viele Anfragen beziehungsweise Bytes aus dem Cache geliefert wurden. Für diese Objekte wurde keine Verbindung nach außen aufgebaut. Neben diesen Möglichkeiten finden Sie über die Webseiten von Squid verschiedene andere Programme, die jeweils ihre Vor- und Nachteile haben.

## › Einsatz eines Proxy-Servers und Datenschutz

Mit der Anleitung in diesem Artikel haben Sie einen Proxy-Server komplett konfiguriert, der weit reichende Beschränkungsmöglichkeiten bietet. Dabei ist das System einfach zu administrieren.

Neben der Verwendung Ihrer eigenen Datenbank können Sie so genannte "Blacklists"

<http://ftp.ost.eltele.no/pub/www/proxy/squidGuard/contrib/>) über die Homepage von SquidGuard herunterladen. Dabei handelt es sich um vorgefertigte Datenbanken mit Domainnamen zu Seiten mit zweifelhaften Inhalten. Die Listen können Sie analog zu den hier erklärten Schritten in Ihr System integrieren. Jedoch sollte man beachten, dass die Blacklists von einem Programm erzeugt werden und daher Domainnamen enthalten können, die der Administrator nicht gesperrt haben möchte. Ferner sind die Listen nicht allumfassend, bieten aber eine hervorragende Basis.

Der Datenschutz ist in diesem Beitrag außer Acht geblieben und wirft entsprechende Fragen auf. So wird die IP-Adresse jeden Clients und der Benutzername für die Authentifizierung mitprotokolliert. Diese Informationen lassen sich personenbezogen auswerten und verwenden. Vor allem der Benutzername in Verbindung mit aufgerufenen Seiten in den Auswertungen ist im Sinne des Datenschutzes bedenklich. Daher sollten Sie Ihre Anwender gegebenenfalls über Nutzungsrichtlinien darauf hinweisen.

Für weiter führende Informationen zum Proxy-Server Squid sind die [FAQs](#) (<http://www.squid-cache.org/Doc/FAQ/FAQ.html>) der Programmierer sehr empfehlenswert. Eine ausführliche Anleitung finden Sie [hier](#) (<http://squid-docs.sourceforge.net/latest/html/book1.htm>) , zu einer detaillierten Konfigurationsübersicht zu SquidGuard gelangen Sie [hier](#) (<http://www.squidguard.org/config/>) . (kpf)

## tecCHANNEL Buch-Shop

Literatur zum Thema Linux	Bestell-Link
Titel von Pearson Education	<a href="#">Bestellung</a>
PDF-Titel (50% billiger als Buch)	<a href="#">Downloads</a>

## › Listing: Squidguard-Konfiguration

```

» # Datei /etc/squidGuard.conf
» logdir /var/squidGuard/log
» dbhome /var/squidGuard/db
»
» dest financial {
» # Die nachstehenden Datenbanken werden unter diesem ACL-Namen
» # zusammengefasst.
» domainlist finance/finance.domains
» urllist finance/finance.urls
» expressionlist finance/finance.regexp
» }
»
» src locals { # hierhin alle erlaubten IP-Adressen
» ip 192.168.1.100/32 # der Proxy aus einem anderen Subnetz darf
» ip 192.168.0.0/24 # das ganze Netzwerk
» }
»
» time working_hours {
» weekly Mondays Tuesdays Wednesdays Thursdays Fridays 06:30-20:00
» }
»
» acl {
» locals within working_hours {
» # Während der Bürozeiten dürfen die festgelegten lokalen
» # Computer alle Seiten im Internet aufrufen, die nicht
» # in den unter financial angelegten Datenbanken stehen
» pass !financial all
» }
» # Alle abgewiesenen Anfragen werden an ein CGI-Skript
» # umgeleitet, das eine Fehlermeldung erzeugt.
» redirect
» http://prefect.tellerrand.de/cgi-bin/squidGuard.cgi?clientaddr=%a&clientname=%n&clientuser=%i&clientgroup=%s&url=%u
» }

```

```

»
» default {
» # Die Standardregel lautet: "Alles was nicht erlaubt ist
» # wird verboten".
» pass none
»
» # Alle abgewiesenen Anfragen werden an ein CGI-Skript
» # umgeleitet, das eine Fehlermeldung erzeugt.
» redirect
http://prefect.tellerrand.de/cgi-bin/squidGuard.cgi?clientaddr=%a&clientname=%n&clientuser=%i&clientgroup=%s&url=%u
» log anonymous default-deny.log
» }
» }

```

### › Listing: init\_script\_proxy

```

» #!/bin/sh
» #
» # Author: Oliver Drees
» #
» # Startscript für squid
» #
»
» ./etc/rc.config
»
»
» case "$1" in
» start)
» echo -n "Starting WWW-proxy squid:"
» startproc -l /var/squid/squid.out /usr/local/squid/bin/squid || return="\tfailed"
» sleep 1
» echo -e "\tdone"
» ;;
» stop)
» echo -n "Shutting down WWW-proxy squid:"
» /usr/local/squid/bin/squid -k shutdown 2>/dev/null
» echo -e "\tdone"
» ;;
» status)
» echo -n "Checking for WWW-proxy squid: "
» checkproc /usr/local/squid/bin/squid && echo OK || echo No process
» ;;
» *)
» echo "Usage: $0 {start/stop/status}"
» exit 1
» esac
»
» exit 0

```

### › Listing: Webalizer Konfiguration

```

» # Datei: webalizer.conf
» LogFile /var/squid/logs/access_log
» LogType squid
» OutputDir /home/www/htdocs/usage
» HistoryName webalizer.hist
» Incremental yes
» IncrementalName webalizer.current
» ReportTitle Usage Statistics for
» HostName proxy.fakedomain.de
» HTMLExtension html
»
» PageType htm*
» PageType cgi

```

- » *PageType phtml*
- » *PageType php3*
- » *PageType pl*
- »
- » *DNSCache dns\_cache.db*
- » *DNSChildren 2*
- »
- » *CountryGraph no*
- » *# keine Benutzer-Angaben in der Auswertung zeigen*
- » *TopUsers 0*
- » *#TopUsers 20*
- »
- » *AllSites yes*
- » *AllURLs yes*
- » *AllReferrers yes*
- » *AllAgents yes*
- » *AllSearchStr yes*
- » *AllUsers yes*
- »
- » *HideURL \*.gif*
- » *HideURL \*.GIF*
- » *HideURL \*.jpg*
- » *HideURL \*.JPG*
- » *HideURL \*.png*
- » *HideURL \*.PNG*
- » *HideURL \*.ra*
- »
- » *HideUser \**

## › Weitere Themen zu diesem Artikel:

Test: Linux für den Server (<http://www.tecchannel.de/betriebssysteme/487/>)

Linux als Firewall (<http://www.tecchannel.de/betriebssysteme/695/index.html>)

Hypertext Transfer Protocol (<http://www.tecchannel.de/internet/208/index.html>)

So funktioniert TCP/IP (<http://www.tecchannel.de/internet/209/index.html>)

Linux Firewall mit ipchains (<http://www.tecchannel.de/betriebssysteme/704/index.html>)

Masquerading mit Linux (<http://www.tecchannel.de/betriebssysteme/707/index.html>)

Linux als Windows-Server (<http://www.tecchannel.de/betriebssysteme/248/index.html>)

Java Virtual Machine unter Linux (<http://www.tecchannel.de/betriebssysteme/776/index.html>)

Linux als Windows-Server (<http://www.tecchannel.de/betriebssysteme/248/index.html>)

---

Copyright © 2001  
IDG Interactive GmbH  
Alle Rechte vorbehalten. Jegliche Vervielfältigung oder Weiterverbreitung in jedem Medium in Teilen oder als Ganzes bedarf der schriftlichen Zustimmung der IDG Interactive GmbH. DPA-Texte und Bilder sind urheberrechtlich geschützt und dürfen weder reproduziert noch wiederverwendet oder für gewerbliche Zwecke verwendet werden. Für den Fall, dass in tecChannel unzutreffende Informationen veröffentlicht oder in Programmen oder Datenbanken Fehler enthalten sein sollten, kommt eine Haftung nur bei grober Fahrlässigkeit des Verlages oder seiner Mitarbeiter in Betracht. Die Redaktion übernimmt keine Haftung für unverlangt eingesandte Manuskripte, Fotos und Illustrationen. Für Inhalte externer Seiten, auf die von tecChannel aus gelinkt wird, übernimmt die IDG Interactive GmbH keine Verantwortung.