13 Web-Seiten im Proxy-Cache zwischenspeichern und filtern

Das World Wide Web wird oft lästerhaft World Wide Wait genannt, weil immer mehr Anwender immer mehr Seiten anfordern, als Netz-Anbieter Bandbreite für nicht bevorzugte Anwender schaffen.

Anwender können Web-Seiten schneller abrufen, wenn sie

- Verträge für schnellere Zugänge, Zugänge mit garantierter Bandbreite oder für Zusatzbandbreite über Satellit abschließen oder
- Seiten, die sie selbst oder andere Anwender der gleichen Gruppe wiederholt anfordern, nicht jedes mal neu laden, sondern aus einem Zwischenspeicher abrufen.

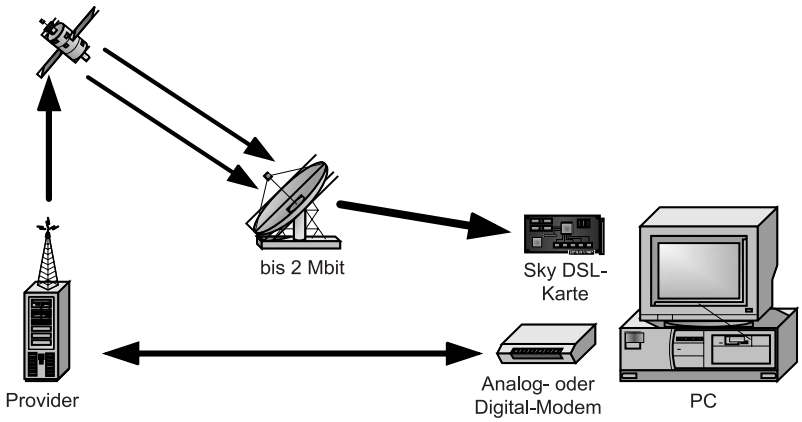

Abbildung 13.1: Mehr Bandbreite, z. B. durch satellitenbasiertes DSL

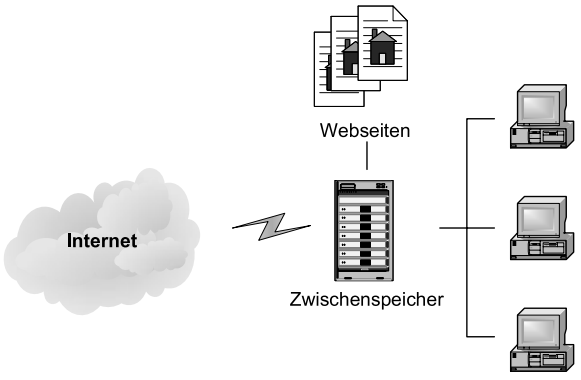

Abbildung 13.2: Web-Seiten im Proxy-Cache

Werden Internetseiten in geschützten Umgebungen, wie in Betrieben, Schulen, Internetcafés mit minderjährigen Besuchern oder Familien, abgerufen, sind Systemverantwortliche gefordert, neben schnellem Seitenabruf auch Filter und Inhaltskontrollen einzurichten.

Web-Zugriffe lassen sich durch verschiedene Zwischenspeicher beschleunigen und filtern:

- Durch lokale Speicher und Filter beim Anwender oder
- zentrale Speicher und Filter zwischen Internet und Clients im Intranet.

Lokale Zwischenspeicher für Internetseiten, Cache genannt, benutzen fast alle Anwender, Anfänger sogar ohne es zu wissen, weil Web-Browser diese Funktion schon in der Grundausstattung bieten. Um in Unternehmen und Bildungseinrichtungen und sonstigen Einrichtungen im Interesse des Jugendschutzes Seiten zu filtern, benötigt man Zusatzprogramme, die Seiten mit unerwünschten Inhalten, wie bestimmten Text- oder Grafikobjekten und von einschlägigen Web-Sites, ignorieren. Löschen Anwender den Verlauf ihrer Web-Sitzungen nicht beim Verlassen des Surfplatzes, können Dritte ausspionieren, welche Web-Sites sie besucht haben.

Diese lokalen Zwischenspeicher legen bereits einmal geladene Internetseiten im Hauptspeicher oder auf der Festplatte ab, so dass bei einem erneuten Zugriff auf die Seite keine weiteres Laden aus dem Internet erforderlich ist, es sei denn, die Seite hätte sich geändert.

Im Netscape Communicator finden Sie die Cache-Einstellungen unter *Bearbeiten • Einstellungen • Erweitert • Cache*.

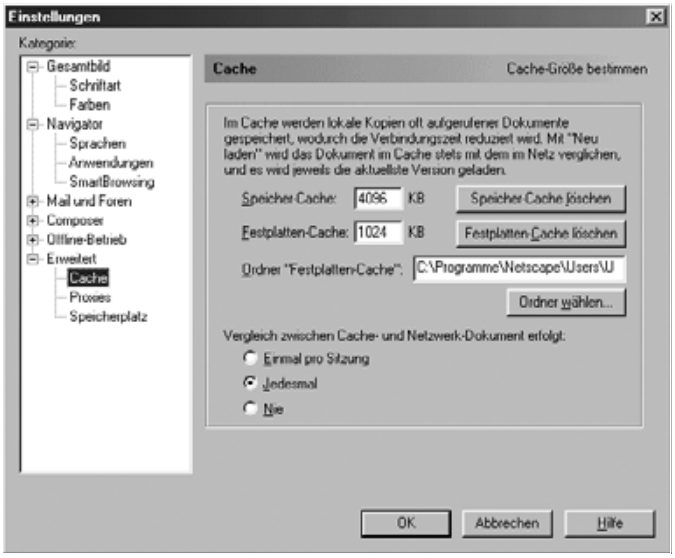

Abbildung 13.3: Cache-Einstellungen im Netscape Communicator

Das entsprechende Menü finden Sie beim Internet-Explorer unter *Extras • Internetoptionen • Temporäre Internetdateien • Einstellungen.*

Abbildung 13.4: Cache-Einstellungen im Internet Explorer

Beim immer beliebter werdenden Browser Opera finden Sie die Cache-Einstellungen unter *Datei • Einstellungen • Verlauf und Puffer*.

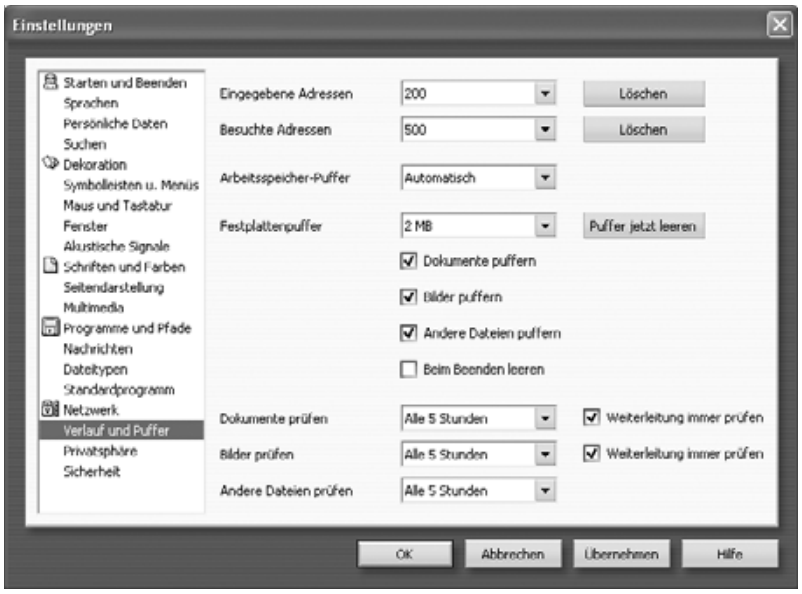

Abbildung 13.5: Cache-Einstellungen bei Opera

Neben diesen lokalen Speichern kann man in lokalen Netzen die von den Anwendern besuchten Seiten zentral speichern. So kommt der Geschwindigkeitsvorteil für das erneute Laden allen zugute, da die Ladezeiten im lokalen Netz vergleichsweise kurz sind. Systemverwalter können damit das Surfverhalten der Anwender auch überwachen, wenn diese die lokalen Speicher nach Arbeitsende löschen.

Für diesen Zweck setzt man auf Kommunikations-Servern einen Proxy-Server, bei Linux meist *Squid*, ein.

Hinweis: Wenn Sie einen Proxy-Server, wie Squid nutzen, dann können Sie den lokalen Cache im Browser deaktivieren oder einen möglichst kleinen Wert reduzieren.

Zusätzlich zu der Cache-Funktion verfügt Squid über eine Stellvertreter- (Proxy-) Funktion. Bei der Einwahl ins Internet stellt der Provider nur eine einzige offizielle IP-Adresse zur Verfügung, die der Linux-Server bekommt. Die anderen Rechner im Netz verfügen nur über lokale IP-Adressen, an die Web-Server keine Antworten schicken können. Diese lokalen Rechner fordern WWW-Seiten indirekt vom Squid an, welcher sie mit der IP-Adresse des Linux-Servers aus dem Internet abruft, sofern er sie nicht schon lokal gespeichert hat.

13.1 Wann lohnt sich ein Proxy-Cache?

Ein Proxy-Cache hat mehrere Aufgaben und Vorteile:

- Er beschleunigt den Internet-Zugriff, da schon einmal geladene Seiten nicht erneut übertragen werden,
- hat eine Stellvertreterfunktion für die Rechner im Netz;
- er kann kontrollieren, welche Inhalte Benutzer im lokalen Netz anfordern dürfen, und
- er dokumentiert, wer wann von welchem Endgerät welche Web-Seiten angefordert hat.

Wie sehr der Proxy-Server das Laden von Web-Seiten dadurch beschleunigt, dass er mehrfach angeforderte Seiten aus dem lokalen Netz statt aus dem Internet bereitstellt, hängt in der Praxis davon ab, wie viele Nutzer die gleichen Seiten anfordern und sich eine vielleicht nur schmalbandige Internetanbindung teilen müssen.

Die Proxy- (Stellvertreter-) Funktion ist die einfachste Möglichkeit, beliebig vielen Rechnern im Intranet den Zugriff auf WWW-Seiten zu ermöglichen. Da dabei nur der Proxy Anfragen ins Internet stellt, kommt man mit einer einzigen offiziellen IP-Adresse aus.

Will man den lokalen Rechnern erlauben, selbst direkt unter Umgehung des Proxy auf Web-Server zuzugreifen, muss der Server die lokalen IP-Adressen jeweils durch seine eigene ersetzen (IP-Masquerading). Um auch dann noch Sperr- und Kontrollmöglichkeiten zu garantieren, muss man jedoch eine Firewall einrichten und betreiben (s. Kapitel 14).

Proxies können gezielt einzelne Seiten oder ganze Internet-Domains sperren, damit kein Browser, der sie anfordert, solche Seiten überhaupt sehen oder laden kann.

Da ein Proxy alle Zugriffe protokollieren kann, lässt sich überwachen, wer welche Seiten aufgerufen hat.

13.2 So funktioniert ein Proxy-Cache

Anfragen von Client-Browsern gehen nicht mehr direkt ins Internet, sondern zum Proxy-Server. Dieser prüft, ob er eine aktuelle Version der angeforderten Seite gespeichert hat. Wenn die Seite vorliegt und noch aktuell ist, liefert er sie direkt aus dem lokalen System heraus an den Browser.

Hat er die Seite nicht im Speicher oder ist sie nicht mehr aktuell, so lädt der Proxy sie aus dem Internet, speichert sie im lokalen Netz und stellt sie dann den Browsern der Clients zur Verfügung.

13.3 Squid installieren und konfigurieren

Da alle Linux-Distributionen Squid enthalten, lässt er sich einfach durch Auswahl des zugehörigen Pakets einrichten. Bei SuSE befindet sich die bewährte Version des Squid in der Selektion `Netzwerk/Server` im Paket `squid` bzw. in der entsprechenden rpm-Datei auf der `CD2`.

Datei	*Bedeutung*
`/usr/sbin/squid`	Binärdatei des Squid-Servers.
`/etc/init.d/squid`	Start/Stop-Script für Squid.
`/etc/squid.conf`	Squid-Konfigurationsdatei.

Tabelle 13.1: Die Dateien zu Squid

Nach der Installation muss man dafür sorgen, dass Squid automatisch startet. Dazu rufen Sie im YaST-Kontrollzentrum unter *System* den *Runlevel-Editor* auf und klicken auf *Runlevel Eigenschaften*. Dort markieren Sie in der Liste die Zeile für den Squid und aktivieren die Runlevel *3* und *5*. Anschließend können Sie YaST wieder beenden.

Diese Änderung wird erst beim nächsten Neustart des Netzwerks bzw. des Rechners wirksam. Von Hand starten Sie Squid mit

```
rcsquid start
```

Die für den laufenden Betrieb benötigten Ordner und Dateien legt Squid beim ersten Start selbstständig an.

Squid konfiguriert man über die 3.000 Zeilen große Datei `squid.conf`, die überwiegend aus Kommentaren und Dokumentation besteht. Jeder Schalter ist zuerst ausführlich dokumentiert, dabei ist auch immer die Standardeinstellung angegeben.

`/etc/squid/squid.conf` (Auszug ab Zeile 722):

```
#  TAG: emulate_httpd_log       on|off
#       The Cache can emulate the log file format which many
#       'httpd'
#       programs use.  To disable/enable this emulation, set
#       emulate_httpd_log to 'off' or 'on'.  The default
#       is to use the native log format since it includes useful
#       information that Squid-specific log analyzers use.
#
#Default:
# emulate_httpd_log off
```

Die ersten acht Zeilen sind reiner Kommentartext, erkennbar an dem einleitenden # Zeichen. Der Kommentar erklärt die Schalter. Der Schalter selber ist hier durch ein # deaktiviert, wodurch die Vorgabe `emulate_httpd_log off` gilt. Will man die Vorgabe ändern, so muss man den Schalter durch Entfernen des Kommentarzeichens aktivieren und `off` durch `on` ersetzen.

Um die Vorgaben individuell einzustellen, sollte man die Konfigurationsdatei sorgfältig bearbeiten. Insbesondere sollte man

- die Größe des Cache im laufenden Betrieb beobachten (s. Logdateien des Squid) und
- den tatsächlichen Bedürfnissen anpassen (s. u.).

In der aktuellen Distribution hat SuSE den Squid so weit auf Sicherheit getrimmt, dass er auch Web-Zugriffe aus dem lokalen Netz ablehnt.

`/etc/squid/squid.conf` (Auszug ab Zeile 1668):

```
#Recommended minimum configuration:
acl all src 0.0.0.0/0.0.0.0
acl manager proto cache_object
acl localhost src 127.0.0.1/255.255.255.255
acl to_localhost dst 127.0.0.0/8
acl SSL_ports port 443 563
```

```
acl Safe_ports port 80          # http
acl Safe_ports port 21          # ftp
acl Safe_ports port 443 563     # https, snews
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 CONNECT method CONNECT

#  TAG: http_access
#       Allowing or Denying access based on defined access lists
#
#       Access to the HTTP port:
#       http_access allow|deny [!]aclname ...
#
#       NOTE on default values:
#
#       If there are no "access" lines present, the default is
#       to deny the request.
#
#       If none of the "access" lines cause a match, the default
#       is the opposite of the last line in the list.
#       If the last line was deny, then the default is allow.
#       Conversely, if the last line is allow,
#       the default will be deny.  For these reasons, it is a
#       good idea to have an "deny all" or "allow all" entry at
#       the end of your access lists to avoid potential
#       confusion.
#
#Default:
# http_access deny all
#
#Recommended minimum configuration:
#
# Only allow cachemgr access from localhost
http_access allow manager localhost
http_access deny manager
# Deny requests to unknown ports
http_access deny !Safe_ports
# Deny CONNECT to other than SSL ports
http_access deny CONNECT !SSL_ports
#
# We strongly recommend to uncomment the following to protect
# innocent web applications running on the proxy server who
# think that 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

# Exampe rule allowing access from your local networks. Adapt
# to list your (internal) IP networks from where browsing should
# be allowed
#acl our_networks src 192.168.1.0/24 192.168.2.0/24
#http_access allow our_networks
http_access allow localhost

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

Die Regel in der letzten Zeile aus diesem Ausschnitt verbietet jeglichen Zugriff per HTTP, wenn ihn bis dahin nicht eine andere Regel erlaubt hat.

Um die Regeln zu aktivieren, die Zugriffe aus dem lokalen Netz erlauben, entfernen Sie die Kommentarzeichen # vor den hervorgehobenen Zeilen,

```
acl our_networks src 192.168.1.0/24 192.168.2.0/24
http_access allow our_networks
```

und veranlassen Sie den Squid, seine Konfigurationsdatei neu einzulesen:

```
rcsquid reload
```

Nach dieser Änderung stellt Squid seine Dienste im lokalen Netz zur Verfügung.

13.4 Zugriffskontrolle durch den Proxy-Cache

Squid kann jeglichen Zugriff auf Internetadressen ausschließen, die Systembetreiber als unerwünscht einstufen:

Um einzelne Server, hier unsere Server `www.mues.li` und `www.wapbu.ch` vollständig zu sperren, richtet man in `squid.conf` eine Zugriffsregel (`Access List=acl`) ein:

```
acl  heutegesperrt  dstdomain  www.mues.li  www.wapbu.ch
```

Hinter dem Schlüsselwort `acl` folgt erst ein frei definierbarer Name für diese Regel, dann deren Gültigkeitstyp und danach eine Aufzählung der zu sperrenden Adressen.

Den in `squid.conf` bereits voreingestellten `acl`-Zeilen, fügt man eigene einfach hinzu.

Die so definierte Regel muss man noch aktivieren:

```
http_access deny  heutegesperrt
```

Dadurch verweigert Squid Zugriff auf alle Seiten, auf die die Regel zutrifft. Diese Zeile muss vor der Zeile

```
http_access allow our_networks
```

stehen.

Nach diesen Änderungen muss der Squid mit

```
rcsquid reload
```

seine Konfigurationsdatei neu einlesen.

Sobald die Sperren aktiv sind, zeigt der Browser des Clients beim Versuch, gesperrte Seiten aufzurufen, eine Fehlermeldung.

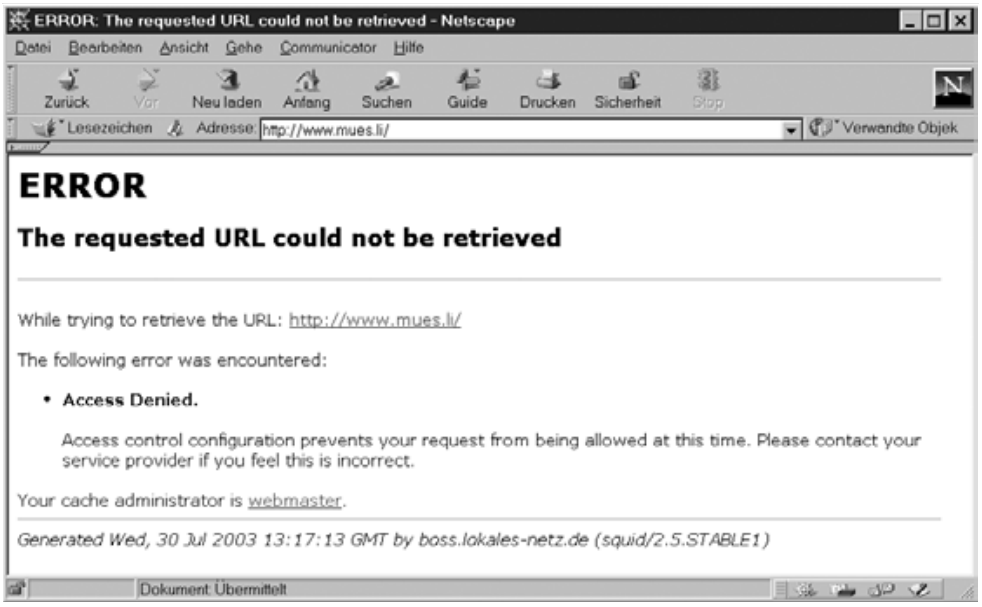

Abbildung 13.6: Zugriffsverweigerung bei gesperrten Seiten

Nach diesen Änderungen hat der besprochene Abschnitt der Konfigurationsdatei folgendes Aussehen:

`/etc/squid.conf` (Auszug ab Zeile 1686 nach Veränderungen):

```
#  TAG: http_access
#       Allowing or Denying access based on defined access lists
#
#       Access to the HTTP port:
#       http_access allow|deny [!]aclname ...
#
#       NOTE on default values:
#
```

```
#       If there are no "access" lines present, the default is
#       to deny the request.
#
#       If none of the "access" lines cause a match, the default
#       is the opposite of the last line in the list.
#       If the last line was deny, then the default is allow.
#       Conversely, if the last line is allow, the default will
#       be deny.  For these reasons, it is a good idea to have
#       an "deny all" or "allow all" entry at the end
#       of your access lists to avoid potential confusion.
#
# Default:
# http_access deny all
#
# Recommended minimum configuration:
#
# Only allow cachemgr access from localhost
http_access allow manager localhost
http_access deny manager
# Deny requests to unknown ports
http_access deny !Safe_ports
# Deny CONNECT to other than SSL ports
http_access deny CONNECT !SSL_ports
#
# We strongly recommend to uncomment the following to protect
# innocent web applications running on the proxy server who
# think that 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

# Exampe rule allowing access from your local networks. Adapt
# to list your (internal) IP networks from where browsing should
# be allowed
acl   heutegesperrt  dstdomain  www.mues.li   www.wapbu.ch
http_access  deny  heutegesperrt

acl our_networks src 192.168.1.0/24 192.168.2.0/24
http_access allow our_networks
http_access allow localhost

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

Sie sollten auf alle Fälle in den Logdateien überprüfen, ob die Sperren auch wie geplant arbeiten. Hinweise zur Auswertung der Logdateien finden Sie im Abschnitt 13.6.

13.5 Browser der (Windows)-Clients einstellen

Damit Client-Browser den Proxy-Cache nutzen können, müssen sie ihn aktivieren. Dazu muss man im jeweiligen Browser die IP-Adresse des Proxy-Servers und seine Portnummer (voreingestellt 3128) eintragen.

Den Netscape Communicator konfiguriert man unter *Bearbeiten* • *Einstellungen* • *Erweitert* • *Proxies* • *Manuelle Proxy-Konfiguration*

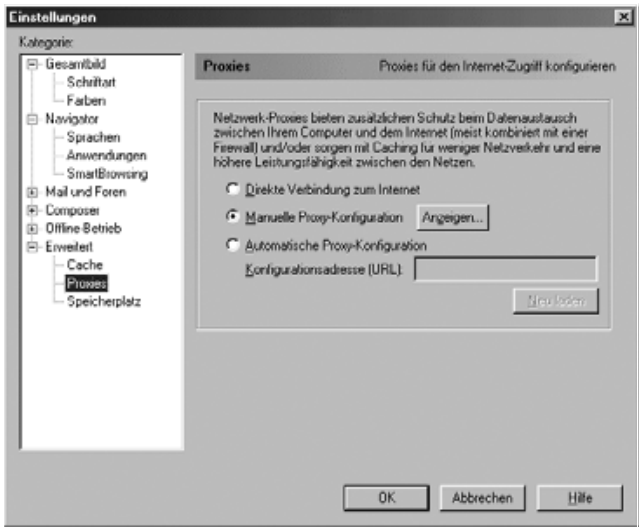

Abbildung 13.7: Einstellungen im Netscape Communicator

Er öffnet ein Formular, in dem Sie Parameter für verschiedene Protokolle einstellen können.

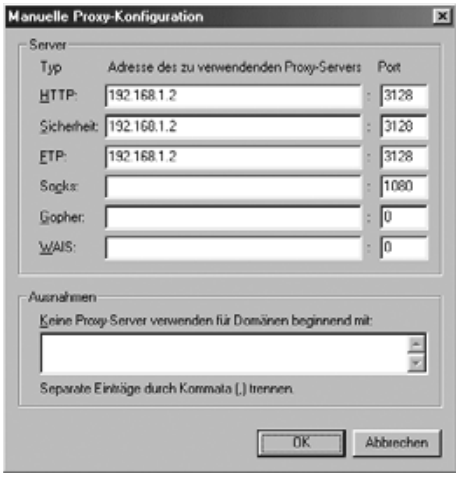

Abbildung 13.8: Manuelle Proxy-Konfiguration im Netscape Communicator

Für HTTP, HTTPS (Sicherheit) und FTP gibt man die IP-Nummer oder den Namen des Kommunikations-Servers und den Port `3128`, die Voreinstellung von Squid an.

Die restlichen Zeilen bleiben wie voreingestellt. In dem großen Eingabefeld kann man Adressen (im lokalen Netz) angeben, für die der Browser den Proxy nicht benutzen soll.

Beim Microsoft Internet Explorer finden sich die gleichen Einstellmöglichkeiten unter *Extras* • *Internetoptionen* • *Verbindungen* • *LAN-Einstellungen*.

Geht man hier auf *Erweitert*, so öffnet der Explorer einen weiteren Dialog mit der praktischen Einstellmöglichkeit *Für alle Protokolle denselben Server verwenden*.

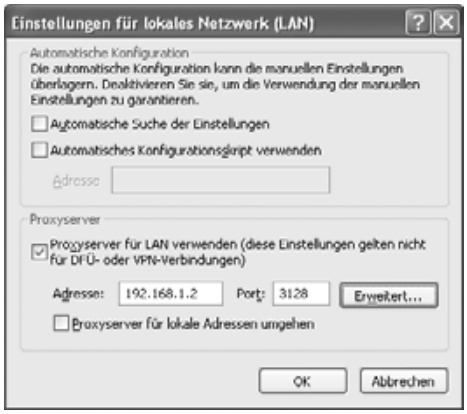

Abbildung 13.9: Menü Verbindung im Internet Explorer

Ähnlich wie beim Netscape Communicator gibt es auch hier ein weiteres Formular für die verschiedenen Protokolle.

Abbildung 13.10: Menü Proxy-Einstellungen im Internet Explorer

Auch hier kann man wieder die lokalen Adressen ausnehmen.

Bei Opera finden Sie die entsprechenden Einstellungen in einem einzigen Formular unter *Datei • Einstellungen • Netzwerk • Proxyserver*.

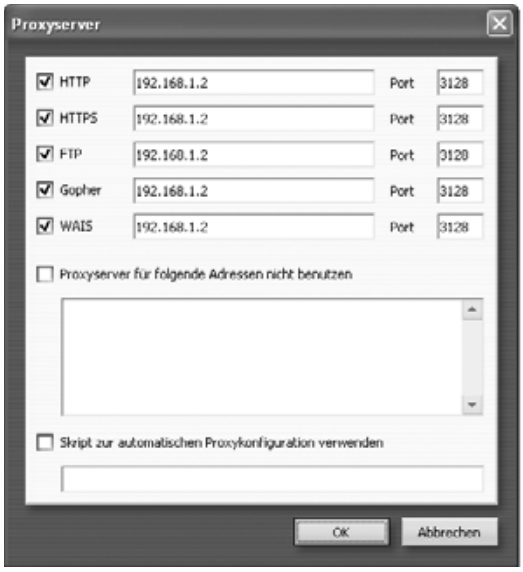

Abbildung 13.11: Menü Proxy-Einstellungen bei Opera

Achtung: Wenn auf dem Kommunikations-Server IP-Masquerading aktiviert ist, können Anwender den Proxy umgehen, indem sie im Browser die Proxy-Einstellungen deaktivieren.

13.6 Die Logdateien des Squid

Die folgenden Logdateien helfen Systembetreuern, Squid zu überwachen. Die angegebenen Pfade beziehen sich auf SuSE 8.2 und können bei anderen Distributionen abweichen.

Datei	*Bedeutung*
`/var/log/squid/rcsquid.log`	Eventuelle Startmeldungen
`/var/run/squid.pid`	Prozess-ID
`/var/logs/squid/cache.log`	Sehr ausführliche Meldungen und Statistik-Informationen des Squid
`/var/log/squid/access.log`	Hier protokolliert Squid jeden einzelnen Zugriff auf den Proxy. Das Format der Datei ähnelt dem der HTTP-Logdatei.

Tabelle 13.2: Logdateien des Squid

Datei	*Bedeutung*
`/var/log/squid/store.log`	Verzeichnis der gespeicherten Dateien mit Speicherort und Web-Quelle
`/var/cache/squid/*`	Vielzahl nummerierter Verzeichnisse, die den eigentlichen Cache bilden

Tabelle 13.2: Logdateien des Squid (Forts.)

Normalerweise interessiert es weniger, wo der Squid welche Datei abgelegt hat. Um bestimmten Zugriffen nachzugehen, will man feststellen können, wer welche Internetseiten aufgerufen hat. Dazu braucht man sich nur die Datei `accesss.log` anzuschauen. Eine typische Zeile sieht folgendermaßen aus:

```
1059570953.020   1577 192.168.1.56 TCP_MISS/200 1611 GET
↵ http://www.linuxbu.ch/ - DIRECT/62.134.48.2 text/html
```

In der ersten Spalte stecken Datum und Uhrzeit, leider nicht in einem menschenlesbaren Format, sondern als UNIX-Zeit, d. h. als Sekunden seit der Geburt der Programmiersprache C, (dem 1.1.1970). In der dritten Spalte steckt die Information, von welchem Rechner aus die Seite aufgerufen wurde und in der sechsten Spalte die URL. Will man diese Datei häufiger kontrollieren, sollte man das Logfile-Format für die Zeitangabe ändern. Aktiviert man in der `squid.conf` den Schalter

```
emulate_httpd_log on,
```

und startet den Squid neu, so legt er die Zeitangaben lesbar ab. Achten Sie darauf, dass Ihr Server seine eigene Zeit richtig synchronisiert (siehe Kapitel 12.9).

```
192.168.1.56 - - [30/Jul/2003:16:41:33 +0200]
↵ "GET http://www.linuxbu.ch/ HTTP/1.0" 200 1610
↵ TCP_CLIENT_REFRESH_MISS:DIRECT
```

Über die Datei `access.log` können Sie alle über den Proxy vermittelten Web-Zugriffe aus Ihrem Netz heraus nachvollziehen. In der ersten Spalte eines Eintrages steht immer die IP-Adresse des Rechners, der eine Seite aufgerufen hat. Danach folgen Datum und Uhrzeit, sowie die URL des angeforderten Dokumentes. Zuletzt kommen dann noch der Statuscode des Web-Servers, die Dateigröße und ob Squid das Dokument bereits im Cache vorgefunden hat oder nicht. Im Abschnitt 13.8 lesen Sie, wie Sie das Auswerten der Protokolle mit Webalizer automatisieren.

Bei den umfangreichen Möglichkeiten der Überwachung darf man die geltenden Gesetze, Vorschriften und Vereinbarungen nicht aus dem Auge verlieren. Dazu gehören:

- Bundesdatenschutzgesetz (`www.datenschutz.de/recht/gesetze/`),
- Landesdatenschutzgesetz des jeweiligen Bundeslandes,

- Betriebsverfassungsgesetz und
- Telekommunikationsgesetz sowie
- eventuelle Betriebsvereinbarungen.

Daher ist es zwingend erforderlich, mit allen Benutzern genaue Regelungen für die Internetnutzung und die mögliche Überwachung dieser Regeln zu vereinbaren. Unternehmen sollten in Arbeitsverträgen klar regeln, welche privaten Nutzungen des Internet sie ihren Mitarbeitern erlauben. Dies ist ein sehr problematischer Bereich, da Unternehmen damit die Rolle von Internet-Providern übernehmen. Wenn Betriebe hingegen ihren Mitarbeitern jegliche private Nutzung des Internet vom Firmenarbeitsplatz/Account aus untersagen, sparen sie sich viele rechtliche Risiken.

13.7 Cache-Dateien überwachen

In sehr aktiven Umgebungen kann es gelegentlich zu Problemen bei vielen gleichzeitig offenen Dateien kommen. Voreingestellt sind 8 MB Hauptspeicher und etwa 100 MB Festplattenspeicher für den Squid. Wird der Festplattenplatz wirklich ausgenutzt, kommt der Squid gelegentlich mit der maximalen Zahl gleichzeitig offener Dateien in Schwierigkeiten. Bei überlasteten Verbindungen ins Internet kann es auch dazu kommen, dass Squid unvollständig geladene Dateien im Cache speichert.

Sollte einer dieser Effekte auftreten, oder finden sich in der Datei `/var/log/warn` vermehrt Fehlermeldungen des Squid, so kann man einfach den kompletten Cache löschen. Dazu geht man folgendermaßen vor:

```
rcsquid stop
```

beendet den Squid. Man sollte ihm aber zum Beenden mindestens 30 Sekunden Zeit lassen, bevor man weitermacht. Die Zeile

```
rm  -r  /var/squid/cache/*
```

löscht einfach vollständig alle Cache-Ordner.

Sie müssen darauf achten, als Benutzer Squid angemeldet zu sein, wenn Sie die Cache-Dateien neu einrichten. Vom Root-Account aus richtet man die Cache-Ordner neu ein mit:

```
su  squid
/usr/sbin/squid -z
exit
```

Danach kann man Squid wieder starten

```
rcsquid start
```

13.8 Auswertung mit Webalizer

Im Abschnitt 13.6 haben Sie gelesen, wie die Logdatei des Squid aufgebaut ist und wie Sie diese analysieren können. Manchmal ist man aber an statistischen Aussagen über die Squid-Nutzung interessiert. Es kann z. B. interessant sein festzustellen, welche Internetseiten die Nutzer am häufigsten aufrufen, eine Auswertung ähnlich wie die Auswertung der Zugriffe auf den Web-Server Apache.

Die Datei `/var/log/squid/access.log` ähnelt in ihrem Aufbau der Logdatei des Web-Servers Apache, vor allem wenn Sie wie beschrieben die http-Emulation aktivieren.

Daher können Sie auch diese Datei mit dem Programm Webalizer auswerten. Webalizer haben Sie bereits in Kapitel 6 kennen gelernt und vermutlich auch installiert.

Die folgende Beschreibung geht davon aus, dass Sie die Squid-Auswertung zusätzlich zu einer eventuell vorhandenen Web-Server-Auswertung nutzen wollen.

Sie müssen ein Verzeichnis einrichten, in das Webalizer die Squid-Statistik ablegen kann. Eine Möglichkeit wäre `/srv/www/htdocs/squidalizer`:

```
mkdir  /srv/www/htdocs/squidalizer
```

Squid und Apache speichern ihre Logdateien an verschiedenen Orten. Daher müssen Sie für die Squid-Auswertung auch eine spezielle Konfigurationsdatei erstellen.

Zum Erzeugen dieser zweiten Konfigurationsdatei sollten Sie einfach die vorhandene Datei kopieren, z. B. als squidalizer.conf:

```
cp  /etc/webalizer.conf  /etc/squidalizer.conf
```

Nun müssen Sie diese Datei für die Pfade des Squid anpassen, damit der Webalizer die richtige Logdatei bearbeitet.

`/etc/squidalizer.conf (Auszug ab Zeile 23)`

```
# LogFile defines the web server log file to use.  If not
# specified here or on on the command line, input will default
# to STDIN.  If the log filename ends in '.gz' (ie: a gzip
# compressed file), it will be decompressed on the fly as it
# is being read.

LogFile        /var/log/squid/access.log

# LogType defines the log type being processed.  Normally, the
# Webalizer expects a CLF or Combined web server log as input.
# Using this option, you can process ftp logs as well (xferlog
# as produced by wu-ftp and others), or Squid native logs.
```

```
# Values can be 'clf', 'ftp' or 'squid', with 'clf' the
# default. LogType clf

# OutputDir is where you want to put the output files.
# This should be a full path name, however relative ones
# might work as well.
# If no output directory is specified, the current
# directory will be used.

OutputDir      /srv/www/htdocs/squidalizer
```

Damit ist die Konfiguration bereits funktionsfähig und Sie können den Webalizer mit dieser Konfigurationsdatei starten

```
webalizer  -c  /etc/squidalizer.conf
```

Natürlich können Sie auch diese Squid-Daten automatisch auswerten, indem Sie den Programmaufruf in die Cron-Tab von root aufnehmen.

13.9 Benutzer authentifizieren

Im Abschnitt 13.8 konnten Sie lesen, wie Sie die Logdateien des Squid statistisch auswerten können. Manchmal kann es aber auch wichtig sein, zu überprüfen, welche Seiten einzelne Nutzer aufrufen.

Speziell bei der Nutzung in Betrieben, Schulen oder Jugendeinrichtungen lässt sich so feststellen, ob jemand und wer irgendwelche strafbaren oder jugendgefährdenden Seiten aufruft.

Achtung: Bevor Sie beginnen, Logdateien personenbezogen auszuwerten, sollten Sie diese Schritte rechtlich absichern. Dazu kann eine Vereinbarung mit den Nutzern bzw. dem Personalrat über die Nutzung des Internets und über die Auswertung solcher Nutzungen erforderlich sein.

Die Zuordnung zwischen einer aufgerufenen Seite und dem Nutzer, der sie aufgerufen hat, ist auch mit der bisherigen Konfiguration schon möglich, aber aufwändig. Die Datei `access.log` enthält für jede aufgerufene Seite die IP-Adresse des Rechners, von dem aus jemand die Seite aufgerufen hat. In der Datei `lastlog` bzw. den Samba-Logdateien können Sie dann feststellen, welcher Benutzer sich zu diesem Zeitpunkt an dem Rechner angemeldet hatte.

Das ist mühsam und funktioniert auch nur dann, wenn Sie eine Benutzeranmeldung am Netzwerk erzwingen (siehe Kapitel 9).

Mit der eigenen Benutzer-Authentifizierung von Squid kann man erreichen, dass Benutzer nur nach Angabe ihres Benutzernamens und ihres Passworts auf Webseiten zugreifen können. Squid trägt dann in die Datei `access.log` auch den Benutzernamen ein, was das Zuordnen und Auswerten der Aufrufe sehr erleichtert.

Im folgenden Auszug aus der Datei `access.log` ist die Benutzer-Authentifizierung aktiviert :

```
192.168.1.56 - - [30/Jul/2003:19:14:15 +0200]  "GET http://
↵ www.linuxbu.ch/ HTTP/1.0" 407 1764 TCP_DENIED:NONE
192.168.1.56 - debacher [30/Jul/2003:19:14:21 +0200]
↵ "GET http://www.linuxbu.ch/ HTTP/1.0" 200 1610
↵ TCP_SWAPFAIL_MISS:DIRECT
192.168.1.56 - debacher [30/Jul/2003:19:14:21 +0200]
↵ "GET http:// www.linuxbu.ch/links.htm HTTP/1.0" 200 2313
↵ TCP_HIT:NONE
```

Beim ersten Zugriff auf das Internet blockt Squid ab und erzwingt eine Benutzeranmeldung. Nach erfolgreicher Anmeldung liefert er die ursprünglich angeforderte Seite, im vorliegenden Fall sogar aus seinem Speicher. In jeder Zeile steht nun hinter der IP-Adresse des Rechners auch der Name des angemeldeten Benutzers.

Um die Authentifizierung zu aktivieren, benötigen Sie ein externes Modul, das für den Squid Benutzernamen und Passwort überprüft. Außerdem müssen Sie die Konfigurationsdatei `squid.conf` ein wenig bearbeiten.

Die externen Module funktionieren recht einfach. Nach dem Start erwarten sie ständig die Angabe von Benutzernamen und Passwort und liefern dann `OK` oder `ERR` zurück. Ein derartiges Modul können Sie daher auch leicht selber schreiben; die folgenden Abschnitte beschreiben die Module `smb_auth`, `ncsa_auth` und `pam_auth`. Alle drei Module sind Bestandteil des Pakets `squid`.

Sie sollten sich je nach Ihrer Situation für eines der Module entscheiden. Wenn in Ihrem Netz Squid auf dem gleichen Rechner läuft, auf dem Sie Ihre Benutzer verwalten, sollten Sie `ncsa_auth` oder `pam_auth` benutzen, da beide sehr schnell arbeiten. Läuft Squid nicht auf Ihrem Anmelde-Server, so müssen Sie `smb_auth` benutzen, da dieses auf einen beliebigen Samba-Server im Netz zugreifen kann. Sie benötigen dann keine Passwortdatei auf Ihrem Squid-Rechner und authentifizieren Benutzer über das Netz.

13.9.1 Das Modul smb_auth

Mit diesem Modul können Sie die Benutzeranmeldung von einem Samba-Server bestätigen lassen. Benutzen Sie dort ältere SuSE-Versionen als 6.4, so müssen Sie Samba auf dem Anmelde-Server gegebenenfalls aktualisieren. Das Modul `smb_`

`auth` haben Sie bereits zusammen mit dem Paket squid installiert und es steht Ihnen unter `/usr/sbin/smb_auth` zur Verfügung.

Leider ist die im Sommer 2003 ausgelieferte Version von smb_auth nicht funktionsfähig, sie sucht die benötigten Hilfsprogramme in einem falschen Verzeichnis. Mit einer kleinen Änderung an der Datei /usr/sbin/smb_auth.sh können Sie das korrigieren.

`/usr/sbin/smb_auth.sh` (Dateianfang)

```
#!/bin/sh
#
# smb_auth - SMB proxy authentication module
# Copyright (C) 1998  Richard Huveneers
# <richard@hekkihek.hacom.nl>
#
# This program is free software; you can redistribute it
# and/or modify it under the terms of the GNU General Public
# License as published by the Free Software Foundation;
# either version 2 of the License, or
# (at your option) any later version.
#
# This program is distributed in the hope that it will be
# useful, but WITHOUT ANY WARRANTY;
# without even the implied warranty of
# MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
# See theGNU General Public License for more details.
#
# You should have received a copy of the GNU General Public
# License along with this program; if not, write to the Free
# Software Foundation, Inc., 675 Mass Ave, Cambridge,
# MA 02139, USA.

# Korrektur laut www.linuxbu.ch 4. Auflage
SAMBAPREFIX="/usr/"

read DOMAINNAME
read PASSTHROUGH
```

Fügen Sie einfach die hervorgehobene Zeile am Ende der Kommentarzeilen ein, damit *smb_auth* seine Hilfsprogramme findet.

Bevor Sie das Modul testen können, müssen Sie auf Ihrem Anmelde-Server noch eine Datei `proxyauth` erstellen, die nur das Wort `allow` enthält. Diese Datei muss über die Freigabe `netlogon` erreichbar sein. Gemäß der Beschreibung aus Kapitel 9 wäre dies das Verzeichnis `/home/netlogon/`; schauen Sie in der Samba-Konfigurationsdatei Ihres Netzwerks nach, was Sie eingestellt haben.

Sie können das Modul ohne Squid testen. Sie müssen nur die Arbeitsgruppe wissen, für die die Anmeldung erfolgen soll. Gemäß der Beschreibung aus Kapitel 9 heißt diese einfach *ARBEITSGRUPPE*. Damit ergibt sich folgender Aufruf:

```
/usr/sbin/smb_auth  -W  ARBEITSGRUPPE
```

Der Eingabeprompt erscheint nicht wieder, weil das Programm auf eine Eingabe wartet.

Geben Sie nun Ihren Benutzernamen und nach einem Leerzeichen Ihr Passwort ein, gibt das Modul `OK aus und bei` einem falschen Benutzernamen oder einem falschen Passwort `ERR` .

Wenn das so klappt, ist Ihr Prüfmodul einsatzbereit. Falls es Probleme gibt, können Sie `smb_auth` mit dem zusätzlichen Parameter `-d` (debug) aufrufen.

```
root@boss:~ > /usr/sbin/smb_auth  -W  ARBEITSGRUPPE  -d

gast gast
Domain name: ARBEITSGRUPPE
Pass-through authentication: no
Query address options:
Domain controller IP address: 192.168.1.2
Domain controller NETBIOS name: BOSS
Contents of //BOSS/NETLOGON/proxyauth: allow
OK
```

Beenden können Sie den Dialog mit dem Modul über den Tastendruck auf Strg + D.

13.9.2 Das Modul ncsa_auth

Das Modul `ncsa_auth` ist deutlich schneller als `smb_auth`, kann aber nicht über das Netz arbeiten. Das Modul gehört zu Squid und wird mit ihm zusammen installiert.

Auch bei diesem Modul können Sie die Funktionsfähigkeit ohne Squid testen. Das Modul erwartet als Aufruf-Parameter den Namen der Passwortdatei, gegen die es prüfen soll.

```
/usr/sbin/ncsa_auth   /etc/httpd/passwort
```

Sie können hierbei Passwortdateien angeben, die Sie mit dem *htpasswd*-Programm des Apache erzeugt haben (siehe Kapitel 6). Damit können Sie nur ausgewählten Nutzern den Internet-Zugriff erlauben. Wenn Sie allen Benutzern die Anmeldung erlauben wollen, dann sollten Sie besser das Modul `pam_auth` verwenden.

Auch dieses Modul erwartet wieder in einer Eingabezeile einen Benutzernamen und das dazugehörige Passwort, getrennt durch ein Leerzeichen, und liefert `OK` bzw. `ERR` zurück.

13.9.3 Das Modul pam_auth

Am sichersten ist das Modul `pam_auth`. Es braucht keinen direkten Zugriff auf die Passwortdatei, sondern versucht mit den übergebenen Benutzerdaten ein Login. Wenn das Login klappt, sind die übergebenen Daten in Ordnung und das Modul liefert `OK` zurück. Wenn das Login nicht klappt, dann liefert das Modul `ERR` zurück.

Sie können auch dieses Modul von der Eingabezeile aus testen mit:

```
/usr/sbin/pam_auth
```

13.9.4 squid.conf anpassen

Nachdem Sie eines der Authentifizierungs-Module installiert haben, können Sie es in den Squid einbinden. Dazu müssen Sie ein paar Zeilen der `squid.conf` verändern, z. T. einfach durch Auskommentieren. Zuerst geben Sie das Authentifizierungs-Programm an.

`/etc/squid.conf` (Auszug ab Zeile 1154)

```
#Recommended minimum configuration:
#auth_param digest program <uncomment and complete this line>
#auth_param digest children 5
#auth_param digest realm Squid proxy-caching web server
#auth_param digest nonce_garbage_interval 5 minutes
#auth_param digest nonce_max_duration 30 minutes
#auth_param digest nonce_max_count 50
#auth_param ntlm program <uncomment and complete this line
#to activate>
#auth_param ntlm children 5
#auth_param ntlm max_challenge_reuses 0
#auth_param ntlm max_challenge_lifetime 2 minutes

auth_param basic program /usr/sbin/pam_auth
#auth_param basic program /usr/sbin/ncsa_auth  /etc/shadow
#auth_param basic program /usr/sbin/smb_auth -W ARBEITSGRUPPE
auth_param basic children 5
auth_param basic realm Squid proxy-caching web server
auth_param basic credentialsttl 2 hours
```

In der hier gedruckten Erweiterung finden Sie die Einträge für alle drei Module, es darf aber immer nur eine der Zeilen aktiviert sein.

Nun müssen Sie noch Zugriffsregeln einfügen, die nur autorisierten Benutzern einen Zugriff erlauben. Dazu gehören je eine zusätzliche `acl`-Zeile (im Listing hervorgehoben) und je eine `http_access` Zeile:

`/etc/squid.conf` (Auszug ab Zeile 1710):

```
# Recommended minimum configuration:
#
# Only allow cachemgr access from localhost
http_access allow manager localhost
http_access deny manager
# Deny requests to unknown ports
http_access deny !Safe_ports
# Deny CONNECT to other than SSL ports
http_access deny CONNECT !SSL_ports
#
# We strongly recommend to uncomment the following to protect
# innocent web applications running on the proxy server who
# think that 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

# Exampe rule allowing access from your local networks. Adapt
# to list your (internal) IP networks from where browsing should
# be allowed
acl heutegesperrt dstdomain www.mues.li  www.wapbu.ch
http_access deny heutegesperrt

acl  domainuser  proxy_auth  REQUIRED
http_access deny  !domainuser

acl our_networks src 192.168.1.0/24 192.168.2.0/24
http_access allow our_networks
http_access allow localhost

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

Nach einem Neustart des Squid ist Ihre Benutzer-Authentifizierung aktiviert und Ihre Benutzer müssen sich beim ersten Zugriff auf eine Internetseite beim Squid anmelden.

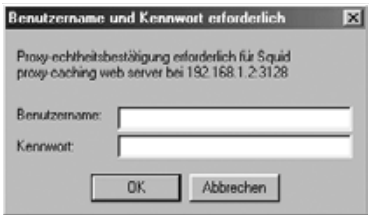

Abbildung 13.12: Authentifizierung für Squid

13.9.5 Feintuning

Das hier beschriebene Verfahren der Benutzeranmeldung am Squid hat einen Haken. Programme wie der Realplayer, die einen Proxy benutzen, aber keine Passwörter speichern können, bekommen keinen Internet-Zugriff mehr.

Hier können Sie über Firewall-Regeln (siehe Kapitel 14) den Zugriff auf den zugehörigen Port gezielt freigeben.

```
/usr/sbin/iptables  -I FORWARD --dport rtsp -p tcp -j ACCEPT
/usr/sbin/iptables  -I FORWARD --dport rtsp -p udp -j ACCEPT
```

Hiermit bekommt der Realplayer einen direkten Internet-Zugriff und muss nicht mehr über den Squid aufs Internet zugreifen.

Sollte Ihnen der Text *Squid proxy-caching web server* im Anmeldefenster nicht gefallen, so können Sie auch einen individuelleren Text vorgeben, z. B. Internetzugriff.

`/etc/squid.conf` (Auszug ab Zeile 1166)

```
auth_param basic program /usr/sbin/pam_auth
#auth_param basic program /usr/sbin/ncsa_auth /etc/shadow
#auth_param basic program /usr/sbin/smb_auth -W ARBEITSGRUPPE
auth_param basic children 5
auth_param basic realm Internetzugriff
auth_param basic credentialsttl 2 hours
```

Sie können den Text natürlich nach Ihren Vorstellungen gestalten.