Reverse Proxys

Robert Hilbrich

Fakultät für Informatik
Humboldt Universität Berlin

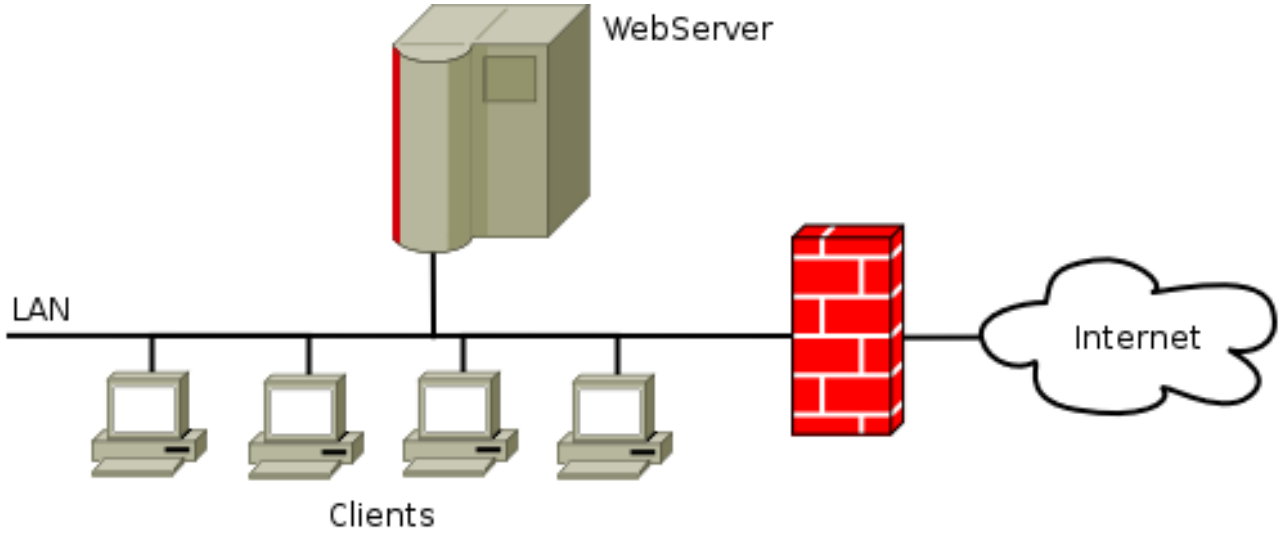
28. September 2006

John von Neumann, 1949

> *It would appear that we have reached the limits of what it is possible to achieve with computer technology, although one should be careful with such statements, as they tend to sound pretty silly in 5 years.*

Übersicht

Eine kleines Firmennetzwerk ...



Wesentliche Elemente

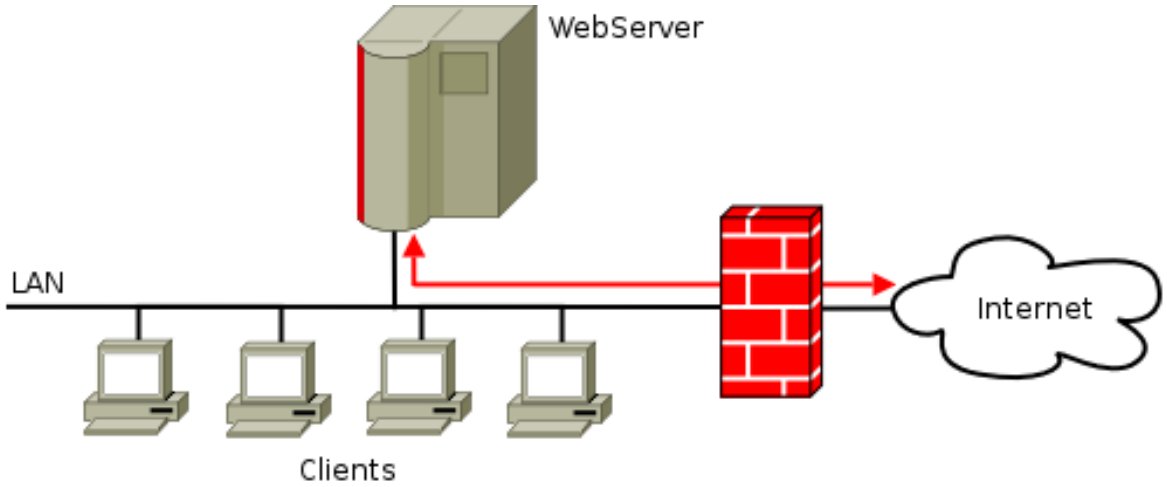
- LAN, Firewall, Clients, WebServer
- (Web-) Applikationen
  (*http://server/app1*, *http://server/app2*, ...)

Public Access

Projektziele

1. Die Web-Applikation *app1* soll vom Internet aus erreichbar sein.
2. Alle anderen Applikation sind vor dem Zugriff aus dem Internet zu schützen.

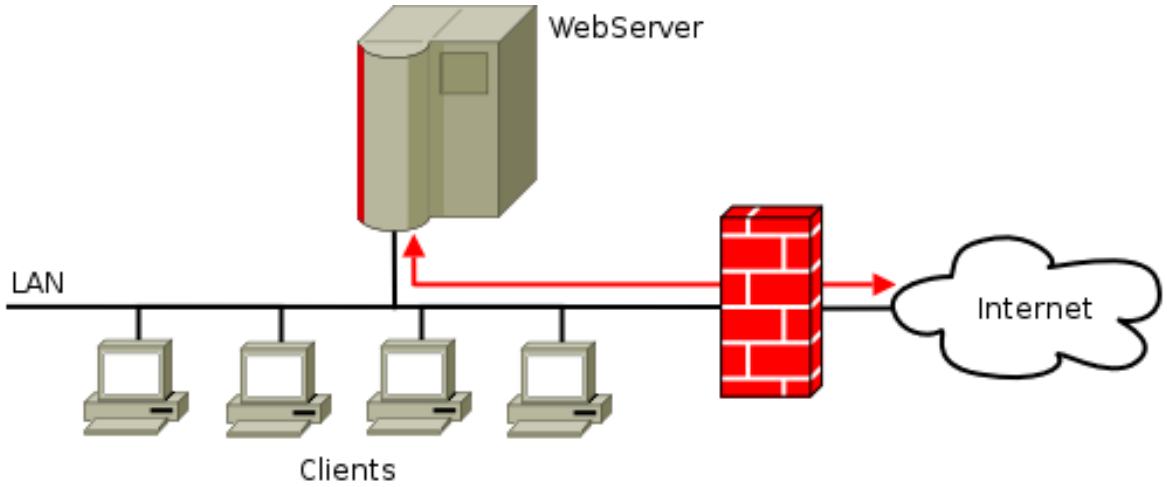
Lösung 1: Firewall öffnen



Ergebnis

- Zugriff auf den Webserver vom Internet möglich
- Problem: Zugriff auf *alle* Web-Applikationen möglich
- Ziel: Filter für HTTP Requests auf URL Ebene

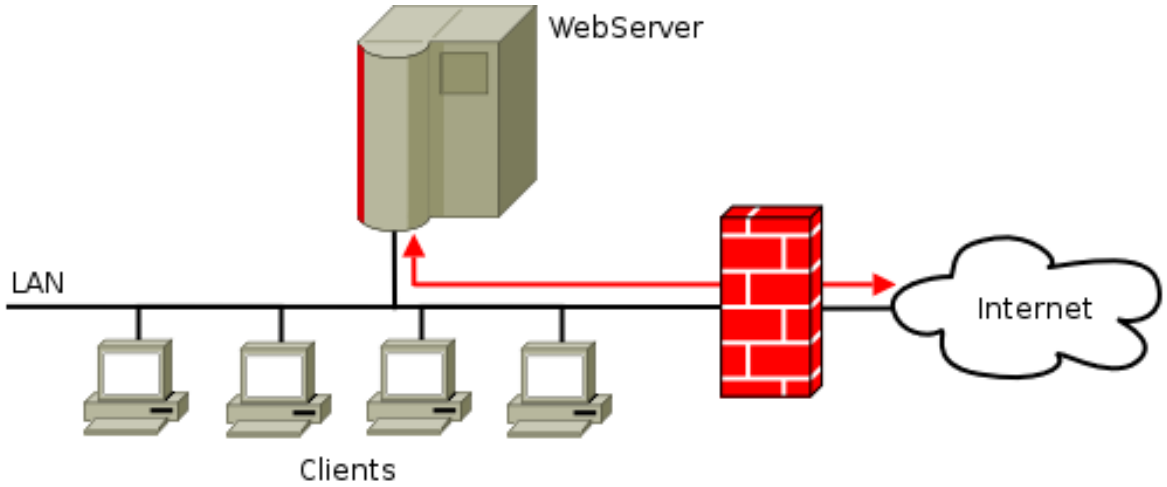
Lösung 1: Firewall öffnen



Ergebnis

- Zugriff auf den Webserver vom Internet möglich
- Problem: Zugriff auf *alle* Web-Applikationen möglich
- Ziel: Filter für HTTP Requests auf URL Ebene

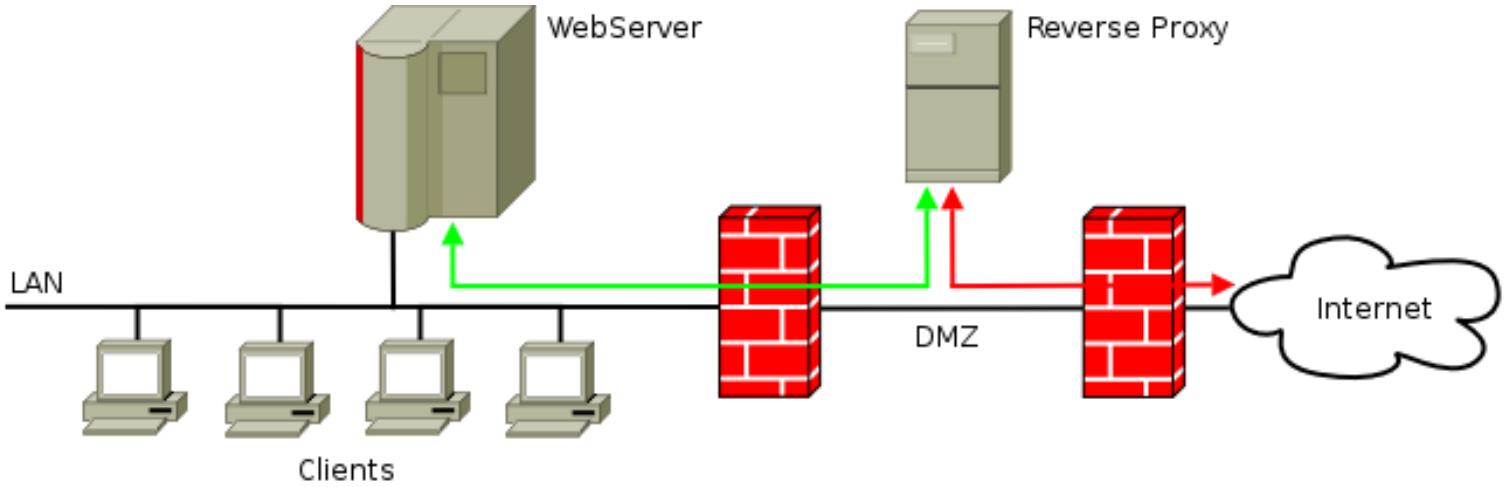
Lösung 1: Firewall öffnen



Ergebnis

- Zugriff auf den Webserver vom Internet möglich
- Problem: Zugriff auf *alle* Web-Applikationen möglich
- Ziel: Filter für HTTP Requests auf URL Ebene

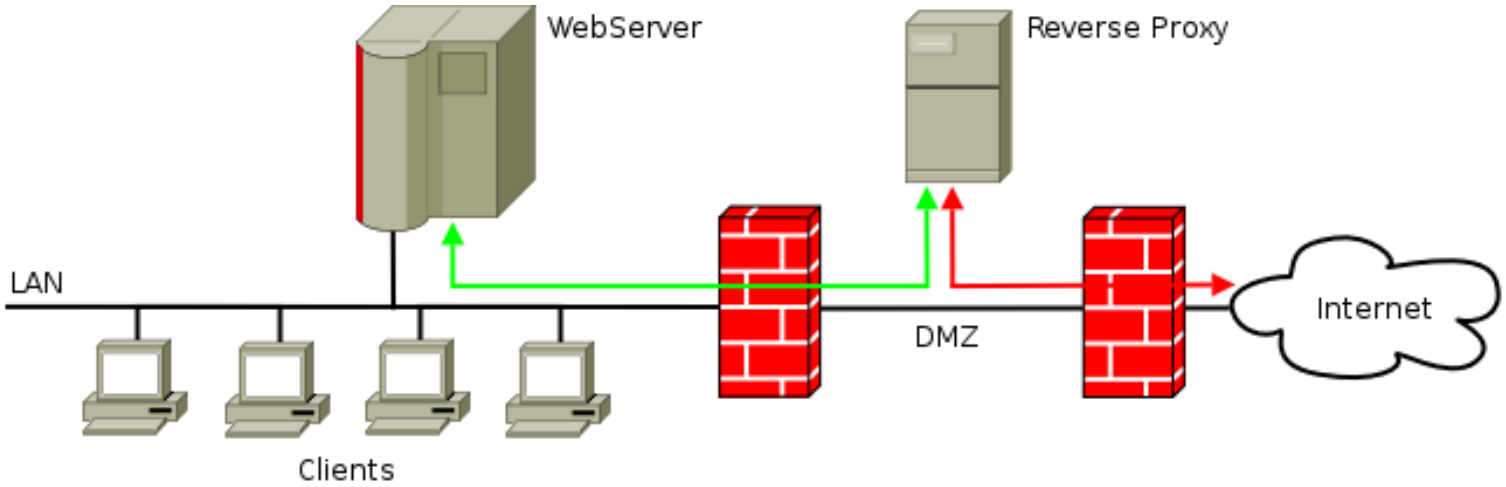
Lösung 2: (Reverse) Proxy



Ergebnis

- Proxy gibt sich als WebServer aus
- effektive Filtermöglichkeiten
- Erhöhung der Performance durch Caching

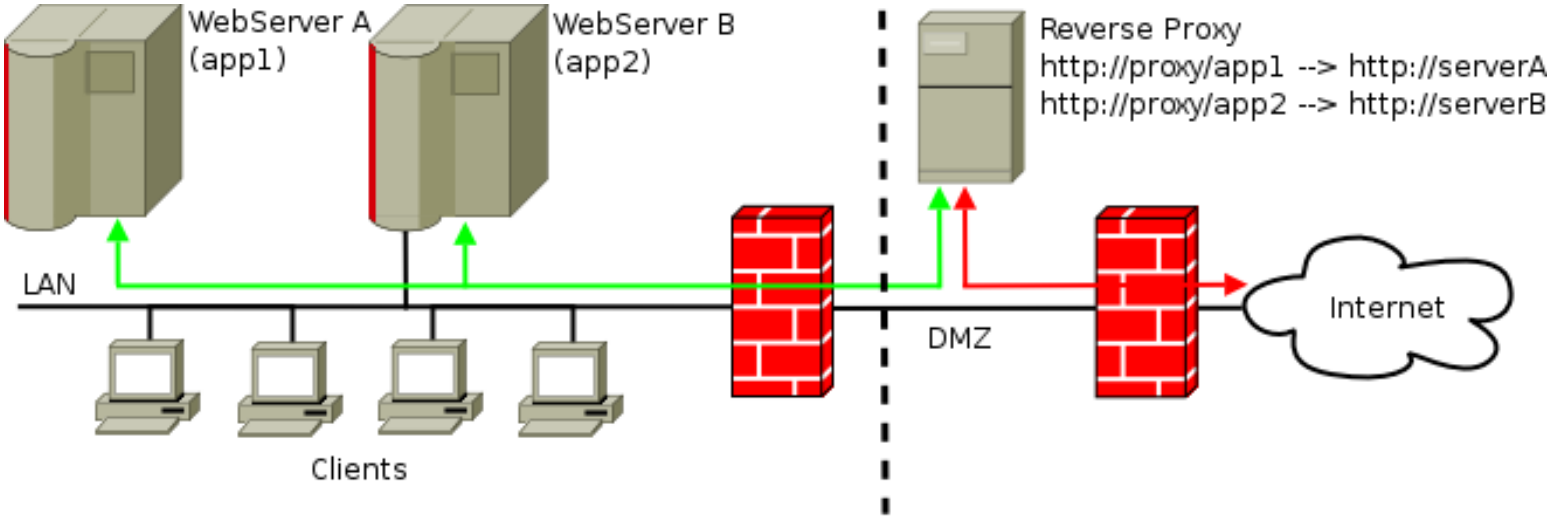
Lösung 2: (Reverse) Proxy



Ergebnis

- Proxy gibt sich als WebServer aus
- effektive Filtermöglichkeiten
- Erhöhung der Performance durch Caching

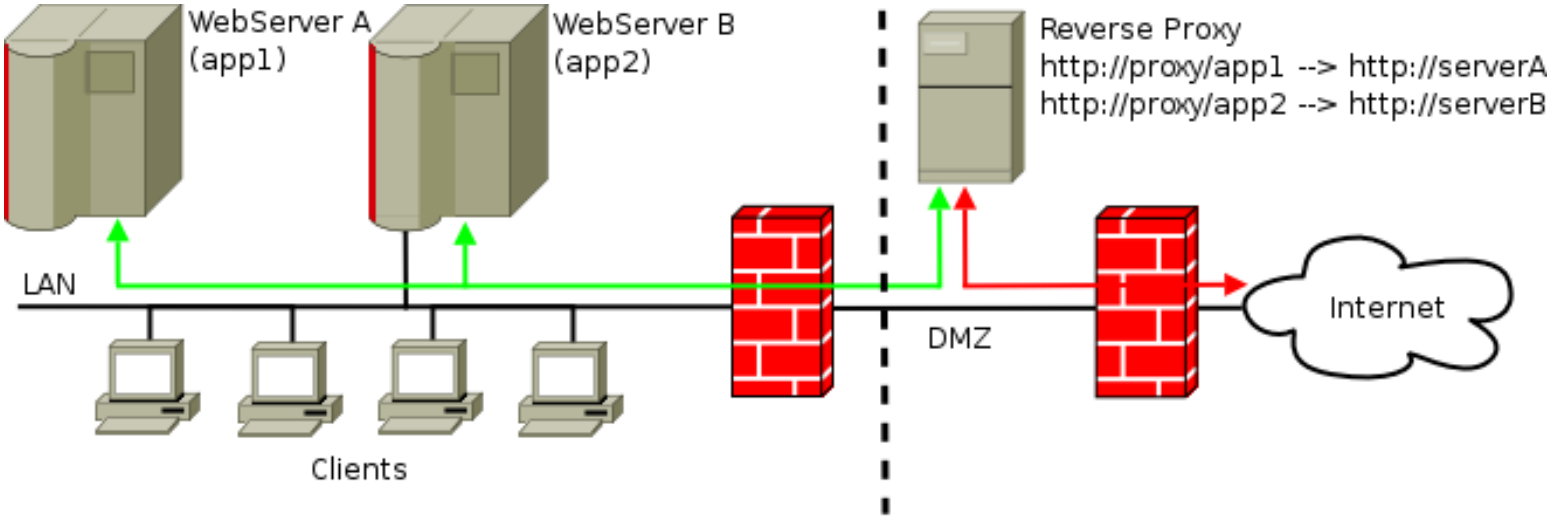
Dienste konsolidieren



Ergebnis

- Proxy Lösung ist sehr flexibel und sicher
- verschiedene Webserver lassen sich konsolidieren
- cool!

Dienste konsolidieren

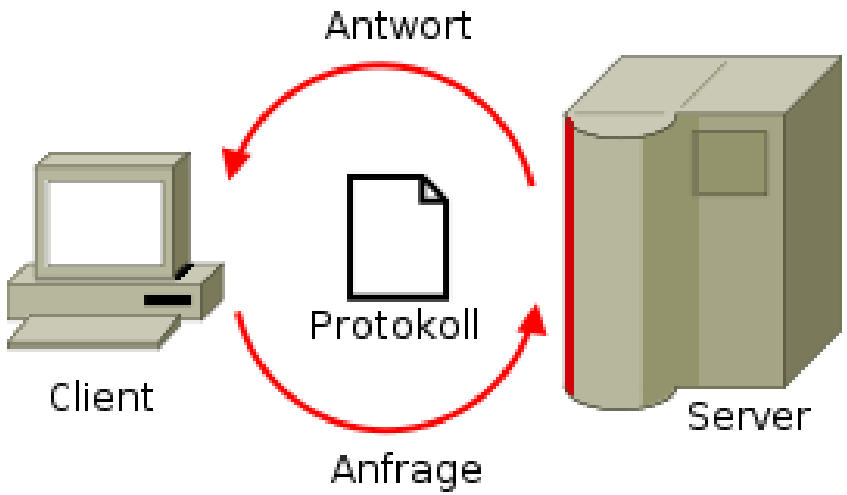


Ergebnis

- Proxy Lösung ist sehr flexibel und sicher
- verschiedene Webserver lassen sich konsolidieren
- cool!

Übersicht

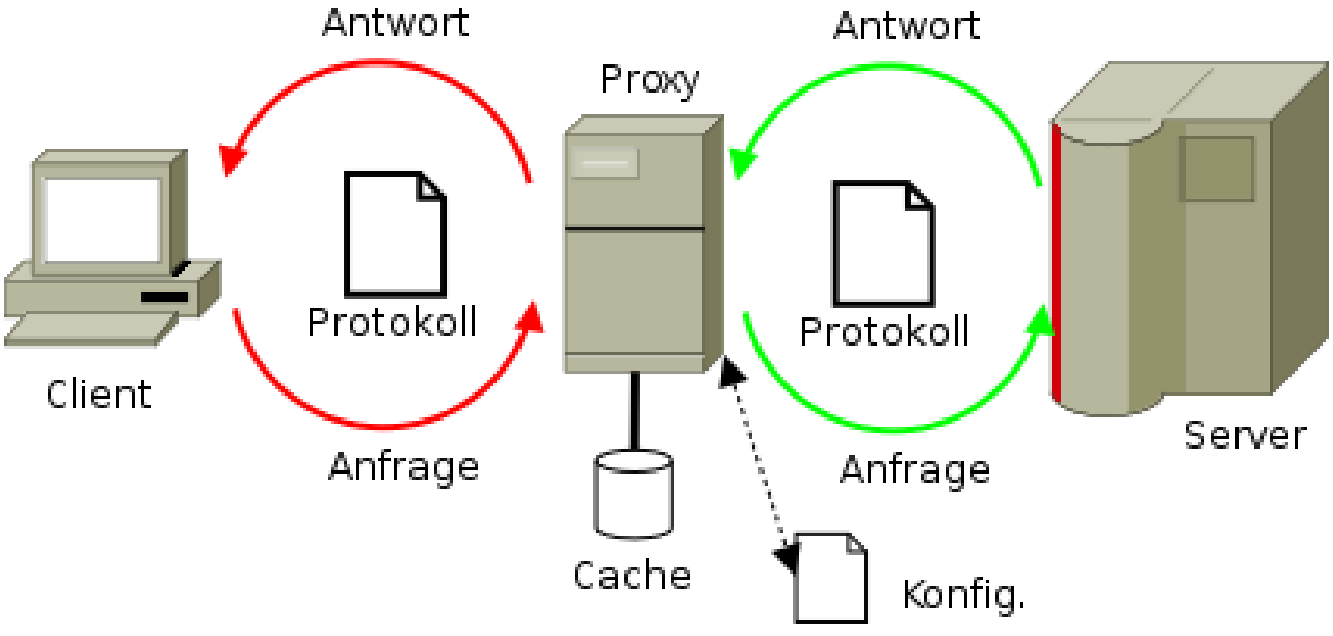
Klassisches Client- und Server-Modell



Eigenschaften

- Server bietet einen Dienst für Clients an
- Client kommuniziert mit Server, um Dienst zu nutzen
- Protokoll regelt Kommunikation

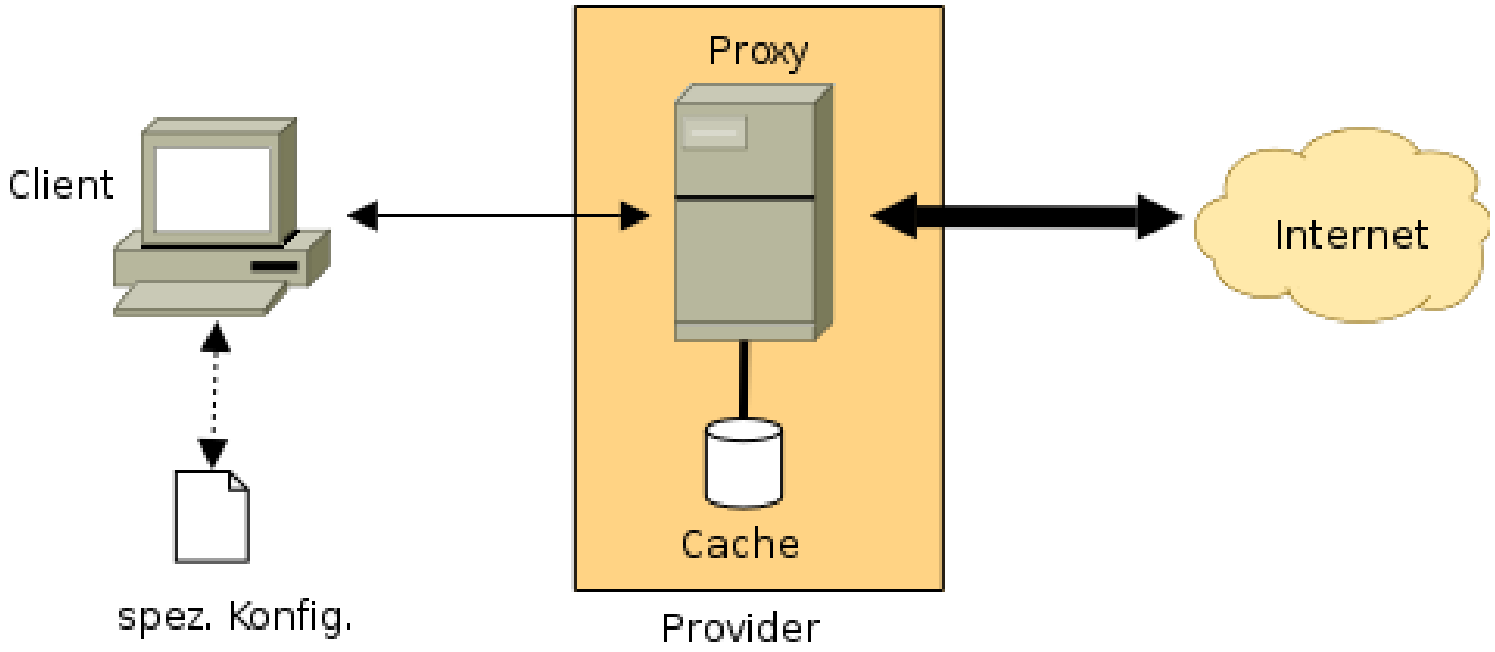
Proxy Funktionsweise



Merkmale

- Proxy ist zwischen Client und Server
- Cache beschleunigt Zugriff und spart Bandbreite
- Unterschiedliche Protokolle möglich (POP3, FTP, HTTP, ...)

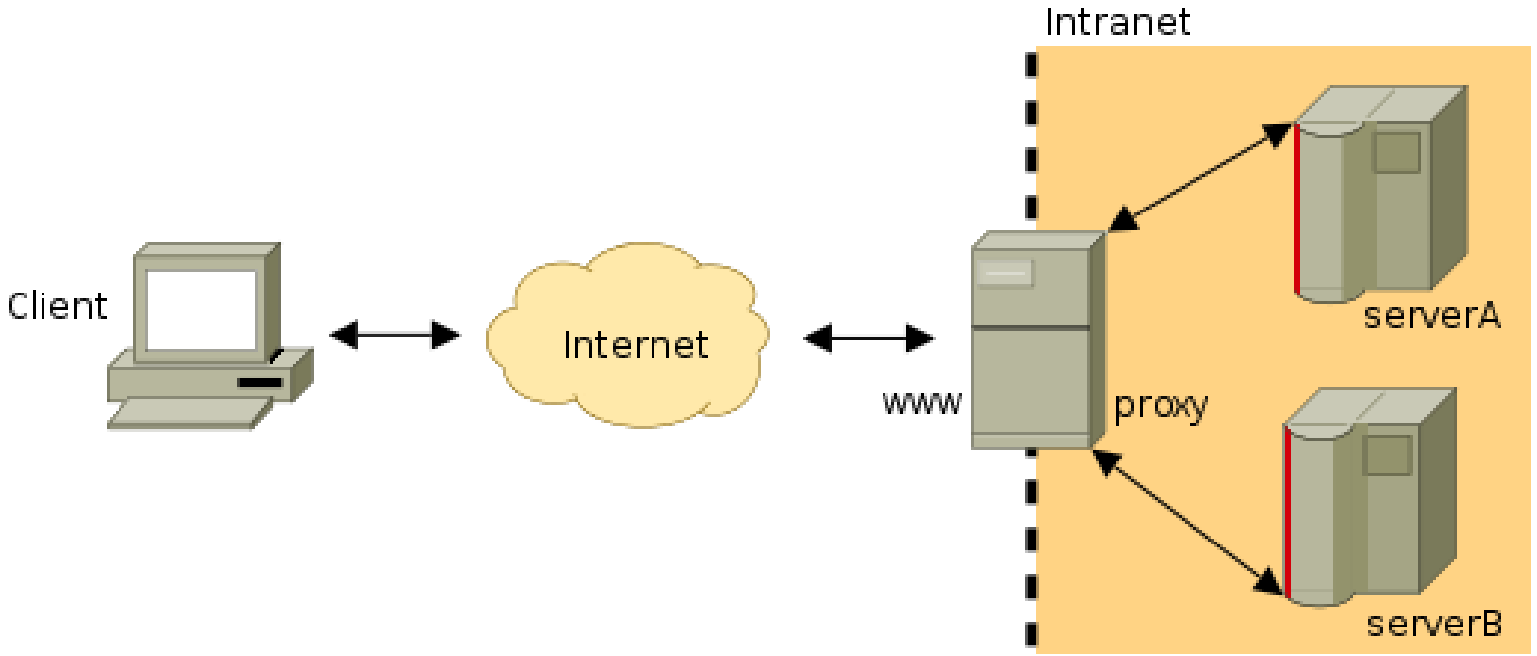
Forward Proxy



Merkmale

- *Alle* Anfragen gehen über den Proxy
- besondere Konfiguration des Clients notwendig
- Ziel: beschleunigter Zugriff oder sicherer Zugriff

Reverse Proxy

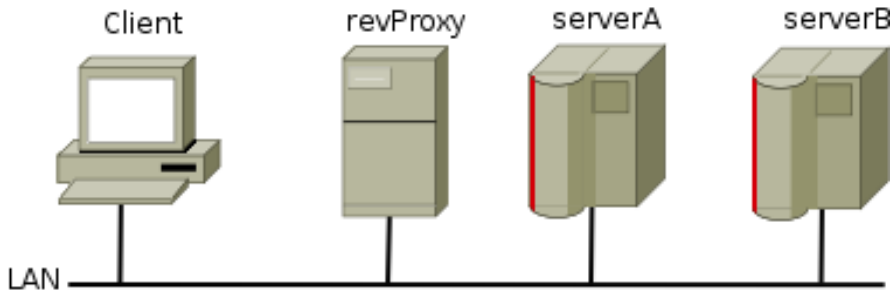


Merkmale

- Client sieht nur den Proxy (alias www)
- Proxy kapselt den Zugriff auf die Server
- Ziel: sicherer Zugriff aufs Intranet vom Internet

Übersicht

Geplanter Aufbau



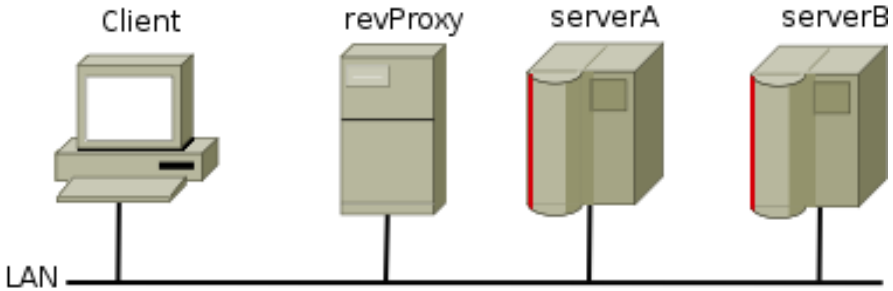
Proxy-Mapping

- http://revproxy/dienst1 → http://serverA
- http://revproxy/dienst2 → http://serverB
- http://revproxy/* → http://revproxy/*

Beispiel

http://revproxy/dienst2/foo/bar → http://serverB/foo/bar

Geplanter Aufbau



Proxy-Mapping

- http://revproxy/dienst1 → http://serverA
- http://revproxy/dienst2 → http://serverB
- http://revproxy/* → http://revproxy/*

Beispiel

http://revproxy/dienst2/foo/bar → http://serverB/foo/bar

Besonderheiten

Nicht enthalten

- Firewalls
- DNS Alias Vergabe
- Caching

Schwierigkeiten

- Servernamen in HTTP Headern (Error Codes)
- Links in der HTML Seite (¡a¿...¡/a¿)
- Links in JavaScript, ...
- Cookies (Cookiedomain ...)

Besonderheiten

Nicht enthalten

- Firewalls
- DNS Alias Vergabe
- Caching

Schwierigkeiten

- Servernamen in HTTP Headern (Error Codes)
- Links in der HTML Seite (¡a¿...¡/a¿)
- Links in JavaScript, ...
- Cookies (Cookiedomain ...)

Squid


Squid Web Proxy Cache - Mozilla Firefox
Datei Bearbeiten Ansicht Gehe Lesezeichen Extras Hilfe
Documentation
FAQ · Wiki
Configuration Guide
Users guide
Non-English
Advisories More...
Versions
STABLE: 2.6
DEVEL: 3.0
Older Versions
Mirror Sites
FTP files
Web site
Support
Bugzilla Database
Mailing lists
Contacting us
Commercial services
Project Sponsors
Squid-based products
Donations
Related
IRCache project
Logfile Analysis
Software
Writings
Binary distributions
Misc
Team Squid
www.squid-cache.org
Squid Web Proxy Cache
Squid is...
a full-featured Web proxy cache
free, open-source software
the result of many contributions by unpaid (and paid) volunteers
Squid supports...
proxying and caching of HTTP, FTP, and other URLs
proxying for SSL
cache hierarchies
ICP, HTCP, CARP, Cache Digests
transparent caching
WCCP (Squid v2.3 and above)
extensive access controls
HTTP server acceleration

Squid

Vorteile

- bekannteste OpenSource Proxy Implementation
- sehr stabil, sehr gute Cache Verwaltung
- sehr gute Dokumentation

Nachteile

- Kein Mapping von Unterverzeichnissen zu verschiedenen Servern möglich
- d.h.: nur 1:1 Mapping von Proxy-Hostnamen zu Server-Hostnamen möglich
- d.h.: kein URL-based Mapping möglich
- für unsere Anforderungen ungeeignet

Squid

Vorteile

- bekannteste OpenSource Proxy Implementation
- sehr stabil, sehr gute Cache Verwaltung
- sehr gute Dokumentation

Nachteile

- Kein Mapping von Unterverzeichnissen zu verschiedenen Servern möglich
- d.h.: nur 1:1 Mapping von Proxy-Hostnamen zu Server-Hostnamen möglich
- d.h.: kein URL-based Mapping möglich
- für unsere Anforderungen ungeeignet

Apache Module

- Apache 2.x common Distribution enthält proxy-Module
- Apache 1.x proxy-Module unterstützen nur HTTP 1.0

mod_proxy

- mod_proxy_http
- mod_proxy_ftp
- mod_proxy_connect
- mod_cache, mod_disk_cache, mod_mem_cache
- mod_proxy_html
- mod_headers
- mod_deflate

1. Schritt

- Laden der entsprechenden Module im Apache

Allgemein

in httpd.conf:

```
LoadModule proxy_module /usr/lib...
LoadModule proxy_http_module /usr/lib...
...
```

Ubuntu

Symlinks zu den `*.load/*.conf` Dateien anlegen:
`/etc/apache2/mods-available` → `/etc/apache2/mods-enabled`

2. Schritt - reverse Proxy

- Änderungen in der Datei `proxy.conf`
- einfachstes Proxying

Sicherheitswarnung

Die Option `ProxyRequests On` ist für reverse Proxys nicht notwendig!

proxy.conf

```
...
ProxyPass /dienst1/ http://serverA
ProxyPass /dienst2/ http://serverB
...
```

3. Schritt - HTTP Header

- Fehlermeldungen enthalten Hostnamen im HTTP Header

302: Moved Temporarily

```
...
HTTP/1.1 302 Found
Location: http://serverA
...
```

proxy.conf

```
...
ProxyPassReverse /dienst1/ http://serverA/
ProxyPassReverse /dienst2/ http://serverB/
...
```

4. Schritt - Cookies

- Cookies enthalten auch eine Domain und einen Cookiepfad

proxy.conf

```
...
ProxyPassReverseCookieDomain serverA revProxy
ProxyPassReverseCookieDomain serverB revProxy
ProxyPassReverseCookiePath / /dienst1
ProxyPassReverseCookiePath / /dienst2
...
```

5. Schritt - Simple HTML Links

- Links in HTML Dateien: `<a href="...">...</a>`

mod_proxy_html

- SAX Parser für HTML4 / XHTML1
- arbeitet wie ein regelbasierter Ausgabefilter
- `SetOutputFilter proxy-html` aktiviert das Modul
- `ProxyHTMLURLMap <S> <D>` definiert eine Regel

Mapping Regeln

Es gilt last match!

5. Schritt: Simple HTML Links

proxy.conf

```
...
SetOutputFilter proxy-html
...
<Location /dienst1/>
        ProxyHTMLURLMap / /dienst1/
        ProxyHTMLURLMap /dienst1 /dienst1
</Location>
<Location /dienst2/>
        ProxyHTMLURLMap / /dienst2/
        ProxyHTMLURLMap /dienst2 /dienst2
</Location>
```

Achtung

Zweite Regel ist ein Infinite-Rewrite Blocker.

5. Schritt: Simple HTML Links

proxy.conf

```
...
SetOutputFilter proxy-html
...
<Location /dienst1/>
        ProxyHTMLURLMap / /dienst1/
        ProxyHTMLURLMap /dienst1 /dienst1
</Location>
<Location /dienst2/>
        ProxyHTMLURLMap / /dienst2/
        ProxyHTMLURLMap /dienst2 /dienst2
</Location>
```

Achtung

Zweite Regel ist ein Infinite-Rewrite Blocker.

6. Schritt: Erweiterte HTML Links

- Links in JavaScripts, ...
- im Prinzip einfacher Text, der URLs enthält
- Search-and-Replace Algorithmen notwendig

proxy.conf

```
ProxyHTMLExtended On
...
ProxyHTMLURLMap <RA> <RA>
```

ProxyHTMLExtended

- Angabe von komplexen regulären Ausdrücken möglich
- Debugmodus: `ProxyHTMLLogVerbose On`

7. Schritt: Compressed HTML

HTTP Header

```
Content-Type: text/html
Content-Encoding: gzip
```

1. Lösung: mod_deflate

```
...
SetOutputFilter INFLATE;proxy-html;DEFLATE
...
```

2. Lösung: no compressed HTML

```
...
RequestHeader unset Accept-Encoding
...
```

7. Schritt: Compressed HTML

HTTP Header

```
Content-Type: text/html
Content-Encoding: gzip
```

1. Lösung: mod_deflate

```
...
SetOutputFilter INFLATE;proxy-html;DEFLATE
...
```

2. Lösung: no compressed HTML

```
...
RequestHeader unset Accept-Encoding
...
```

7. Schritt: Compressed HTML

HTTP Header

```
Content-Type: text/html
Content-Encoding: gzip
```

1. Lösung: mod_deflate

```
...
SetOutputFilter INFLATE;proxy-html;DEFLATE
...
```

2. Lösung: no compressed HTML

```
...
RequestHeader unset Accept-Encoding
...
```

Übersicht

Fehlende Dinge / Optimierungsmöglichkeiten

- Weiterführendes Link-rewriting
- Fortführung des Gedanken zur Content-Transformation zum barrierefreien Web $\longrightarrow$ mod_accessibility
- Caching und sonstige Beschleunigung des Zugriffs
- Content-Filter

John von Neumann, 1949

> *It would appear that we have reached the limits of what it is possible to achieve with computer technology, although one should be careful with such statements, as they tend to sound pretty silly in 5 years.*

Aber

- Was ist mit JvN Architektur?
- Proxy ist eher ein allgemeines Prinzip
- *Multiplexen / Demultiplexen* von Client Anfragen

John von Neumann, 1949

> *It would appear that we have reached the limits of what it is possible to achieve with computer technology, although one should be careful with such statements, as they tend to sound pretty silly in 5 years.*

Aber

- Was ist mit JvN Architektur?
- Proxy ist eher ein allgemeines Prinzip
- *Multiplexen / Demultiplexen* von Client Anfragen

Quellen

- Apache - `www.apache.org`
- ApacheWeek - ein gutes online Journal zum Server und Umfeld `www.apacheweek.com/features/reverseproxies`
- Dokumentation zum Modul mod_proxy `httpd.apache.org/docs/2.0/mod/mod_proxy.html`
- Squid - `www.squid-cache.org`
- Wikipedia, Wikiquote