PEARSON Education

ADDISON-WESLEY

Hans-Christian Boos

WebFarm

Webtechnologien effektiv einrichten und betreiben

ADDISON-WESLEY

An imprint of Pearson Education

München • Boston • San Francisco • Harlow, England
Don Mills, Ontario • Sydney • Mexico City
Madrid • Amsterdam

3 Proxy-Infrastruktur, Browser-Konfiguration und Zugriffsberechtigung

Dieses Kapitel beschäftigt sich mit dem Design einer Proxy-Infrastruktur für Webanwendungen und Anwendungen der neuen Medien. Es wird eine Infrastruktur aus Proxy-Server beschrieben, die HTTP/S und FTP mit Proxy-Cache-Funktionalität bedienen und direkte Verbindungen als Proxy durchschleusen können. Der Sinn einer solchen Infrastruktur ist es, die Performance von Anwendungen durch die lokale Verfügbarkeit von Seiten zu erhöhen und, noch wichtiger, Bandbreite auf teuren Verbindungen durch das Caching einzusparen.

3.1 Application-Proxy vs. Web-Proxy

Viele Anwendungen haben eigene Proxyservices, die eventuell auch eine Cache-Infrastruktur abbilden können. Es ist aber auf keinen Fall sinnvoll, pro Anwendung eine Proxy-Infrastruktur zu designen, da dies einen enormen Hard- und Software-Betriebsaufwand verursachen würde, der betriebswirtschaftlich und administrativ völlig untragbar wäre. Ähnlich wie beim Anwendungsdesign im Webbereich, das so ausgelegt ist, dass Webanwendungen direkt im Browser ablaufen können, muss bei der Infrastruktur und den Anwendungen darauf geachtet werden, dass eine Standard-Web-Proxy-Infrastruktur für die Abwicklung der Anwendungszugriffe verwendet werden kann. Dabei gilt grundsätzlich, dass Caching aufgrund der Kostenersparnisse berücksichtigt werden und darum ein entsprechendes Anwendungs- und Tool-Design eingehalten werden muss.

Eigene Proxy-Services für Anwendungen und Werkzeuge sind nur im Komplex der Firewall vorgesehen, wo auf dem Application-Level-Gateway entsprechende applikationsspezifische Proxy-Server abgelegt werden können, wenn diese sicherheitszertifiziert sind. Im Bereich der Firewall werden Proxy-Server zur Netzwerktrennung eingesetzt, wozu man auf Applikationsebene eine entsprechende Umsetzung machen muss.

Fazit: Anwendungen müssen für Caching und zum Benutzerzugriff Web-Proxy-Server unterstützen oder zumindest so designed sein, dass cachebare Informationen in einer solchen Web-Proxy-Infrastruktur auch zwischenge-

speichert werden können. Applikationsspezifische Proxy-Server werden nur auf Application-Level-Gateway in Firewalls zur Unterstützung der 100%igen Netzwerktrennung eingesetzt.

3.2 Proxy-Konzept

Viele Informationen, die in Webanwendungen abgelegt sind, werden nicht als dynamisch angesehen, das bedeutet, sie haben eine Gültigkeitsdauer, in der sie sich nicht verändern. Da ein bestimmender Faktor in den IT-Kosten heutzutage die Bandbreite in WAN-Verbindungen ist, ist auf jeden Fall dafür zu sorgen, dass Dokumente, deren Lebensspanne noch nicht überschritten ist, möglichst selten über solche Leitungen übertragen werden. Dies bedeutet, solche Dokumente müssen in einem Cache zwischengespeichert werden. Ein solcher Cache wird verbunden mit einem Proxy-Server, der den zentralen Zugriffspunkt einer Einheit auf Internet- und GAN-Services bietet. Ein solcher Zugriffspunkt hat neben der Informationsspeicherung und damit der Bandbreitenoptimierung auch den Vorteil der Zugangskontrolle auf bestimmte Dienste anhand der Dienste selbst oder anhand eines Berechtigungskonzepts.

Da solche Cache- und Proxy-Server auch kaskadiert werden können, bietet es sich an, eine entsprechende Hierarchie zu schaffen, die eine weitere Optimierung der Bandbreite zulässt. Die Konfiguration für eine solche Zugriffshierarchie auf Clientseite kann manuell umständlich werden, insbesondere wenn Ausfallssicherheitskonzepte ebenfalls implementiert werden sollen. Aus diesem Grund wird zugleich von den Servern eine automatische Browser-Konfiguration vorgenommen werden, die alle notwendigen Konfigurationen für den Benutzer vornimmt.

3.3 Zugang zu internen Netzen

Der WebFarm-Standard definiert das Layout eines Zugangspunkts zu anderen unsicheren Drittnetzen in der Sicherheitsinfrastruktur. Hier wird in der DMZ II ein Ort geschaffen, in dem interne Dienste bereitgestellt werden können, wie z.B. der Dienst eines Zugangs-Proxy-Servers. Es ist davon auszugehen, dass ein solcher Proxy-Server, da es nur eine beschränkte Anzahl an Zugangspunkten zu externen Netzen innerhalb des GAN gibt, stark belastet ist, darum sollte er sofort auf ein Clustering-System oder einen Proxy-Cluster ausgelegt werden. Von Anfang an sollten also in der DMZ II eines Zugangsknotens zwei interne Proxy-Server verfügbar sein. Diese internen Proxy-Server sind für das Routing eventuell eingehender interner Requests zu anderen internen Servern verantwortlich, für die Benutzerberechtigung und für die

Zugriffskontrolle an sich. Ebenso stehen diese Server in Kontakt mit den so genannten PAC(Proxy Automatic Configuration)-File-Servern, um die Möglichkeiten zum Load-Sharing und zur Ausfallsicherheit mit den Proxy-Clustern und anderen Zugangspunkten zu verwalten. In der DMZ I, also in der externen DMZ der Sicherheitskonstruktion, befindet sich ein gleich konzipierter Cluster an Proxy-Servern, die den tatsächlichen Zugang in das externe Netz vornehmen. Diese Server haben die Aufgabe, Content-Security zu implementieren sowie das allgemeine Filtern von ungewollten Seiten (z. B. Pornographie oder Gewalt) vorzunehmen.

Das Routing unter diesen Proxy-Servern ist so ausgelegt, dass sie auch beim Ausfall einzelner Komponenten dieser beiden kaskadierten Cluster verfügbar sind und ggf. lediglich die Performancefluktuationen aufweisen.

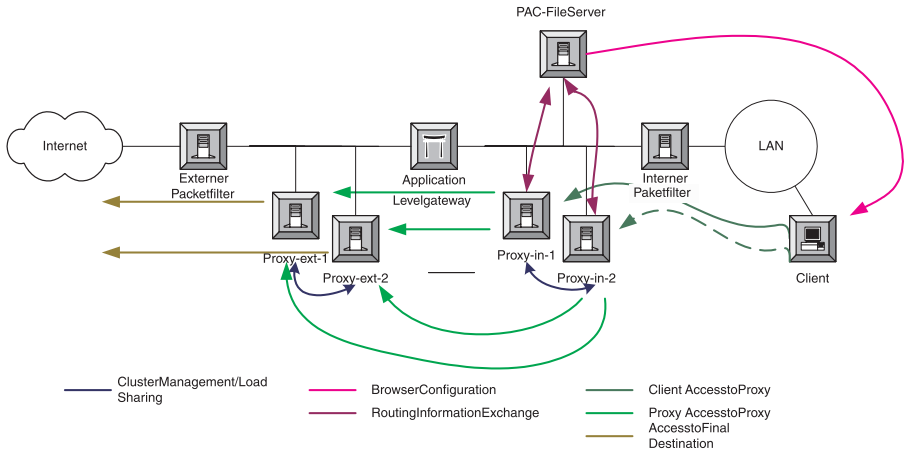

Abbildung 3.1: Routing unter Proxy-Servern

Für den Zugriff auf interne Anwendungen bzw. Anwendungen im GAN wird eine andere Proxy-Konstruktion verwendet.

3.4 Proxy-Hierarchie

Wie bereits kurz in der generellen Idee der Proxy-Infrastruktur beschrieben, ist es sinnvoll, eine Hierarchie von Proxy-Servern aufzubauen, welche die Kommunikation innerhalb des GAN und mit den Verbindungen nach außen so bandbreitenschonend und effizient wie möglich gestalten.

Studien über allgemeine Anwendungsentwicklung besagen, dass in kürzester Zeit 75% aller Anwendungen als Webanwendungen realisiert werden, was auf die offensichtlichen Vorteile dieser Plattform zurückzuführen ist. Aus

diesem Grund muss eine Proxy-Infrastruktur robust sein und auch zukünftigen Anforderungen standhalten bzw. entsprechend erweiterbar sein.

Der Nutzen einer Proxy-Infrastruktur wird durch die Anwendungsentwicklung an sich festgelegt, denn die Anwendungen müssen ein Caching von nicht veränderten Daten unterstützen und können auf diese Weise die Kommunikation im GAN wesentlich verbessern sowie enorme IT-Kosten einsparen, die anfallen würden, wenn Bandbreitenerweiterungen für ganze GANs im größeren Umfang notwendig werden. Die entsprechenden Vorgaben sind im Anwendungsentwicklungsteil des WebFarm-Standards hinterlegt.

Generell ist davon auszugehen, dass es unterschiedliche Größen von Arbeitsgruppen gibt, die mit unterschiedlichen Anforderungen Anwendungen innerhalb ihrer eigenen Arbeitsgruppe und innerhalb des GAN beanspruchen. Hierbei ist dann auch eine entsprechende Proxy-Infrastruktur erforderlich.

Die Anbindungspunkte an externe Netze bieten gute Knotenpunkte in der Infrastruktur, da sie ohnehin technisch und GAN-verbindungstechnisch am besten ausgestattet sind.

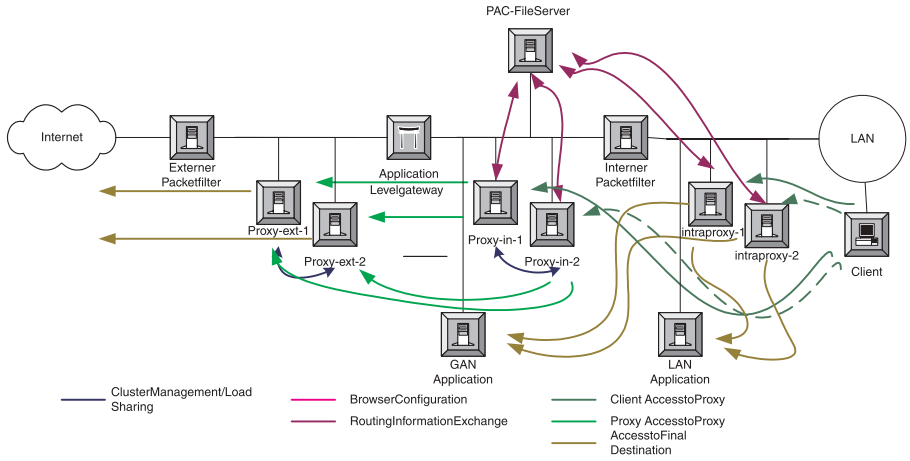

Abbildung 3.2: Proxy-Infrastruktur

Darum werden an den Anbindungspunkten so genannte Intra-Proxy-Cluster errichtet, die das interne Routing der Proxy-Infrastruktur übernehmen und den Zugriff auf interne Applikationen steuern. Die automatische Konfiguration der Clients unterscheidet anhand des eingeführten DNS, welche Zugriffe intern und welche über die Außenanbindung zu erfolgen haben, und verwendet also entsprechend die Proxy-int- oder Intra-Proxy-Server beim Zugriff.

Für den Aufbau der eigentlichen Hierarchie sind vier verschiedene Arbeitsumgebungen, an Größe der Umgebung gemessen, vorgesehen. Es handelt sich nach kleiner werdender Größe sortiert um die Typen

- Zentrale
- Niederlassung
- Geschäftsstelle
- Außenstelle

Generell kann aber gesagt werden, dass ab einem Team von vier Benutzern, die in ihrer täglichen Arbeit mehr als 25% Webanwendungen nutzen sollen, ein Proxy auf jeden Fall sinnvoll ist. Betrachtet man den Anstieg am Nutzen von Webanwendungen muss man also von einer Proxy-Infrastruktur ab einer Außenstelle von vier Benutzern rechnen, die mit großer Wahrscheinlichkeit über eine sehr langsame Leitung angebunden sind und daher die Investition in einen Proxy-Server eher rechtfertigen, als die Leitung zu erweitern.

Ohne Proxy-Server muss von einer Leitungsverfügbarkeit von 64 KBps bis 128 KBps pro Benutzer ausgegangen werden.

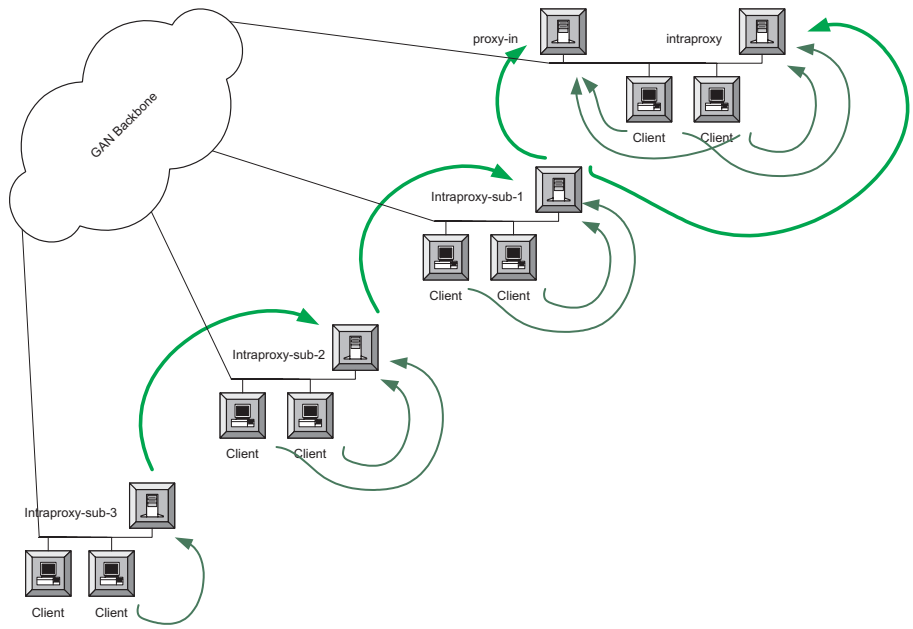

Abbildung 3.3: Konfiguration Proxy-Routing

Die Skizze zeigt die Konfiguration des Proxy-Routings sowie der Client-Konfiguration. Sowohl das Routing als auch vor allem die Client-Konfiguration wird durch PAC-Files generiert, um auf eventuelle Systemausfälle oder Netzwerkbesonderheiten dynamisch reagieren zu können.

Bis auf die Zugriff-Cluster für externe Netze benötigen die Proxy-Server keine Zugriffskontrolle, da das GAN ein offener Transaktionsraum ist. Einzelne Anwendungen im GAN werden nicht durch die Proxy-Server, sondern durch deren Benutzerverifikation bzw. Freischaltungen in der Firewall an sich geschützt.

3.5 Browser-Konfiguration

Alle Browser-Konfigurationen werden automatisch generiert und vom Browser bei jedem Start vom so genannten *PAC-File-Server* heruntergeladen, wenn dieser verfügbar ist. Sollte der PAC-File-Server nicht verfügbar sein, wird eine Standardeinstellung verwendet, die ebenfalls durch den PAC-File-Server verteilt werden kann, wenn eine Verbindung zu diesem besteht.

Eine solche automatisierte Konfiguration der Browser ist sehr sinnvoll, da auf diese Weise auch bei Netzwerk- und Infrastrukturveränderungen keine weitere Interaktion mit Clients und Client-Arbeitsplätzen erforderlich wird, was die Kosten für solche Änderungen und Erweiterungen erheblich verringert. Zusätzlich werden lokale Konfigurationsfehler als Fehlerquelle ausgeschlossen, was den Benutzersupport erheblich entlastet.

3.5.1 Zentrale

In zentralen Standorten ist die Konfiguration der Browser komplexer als in anderen Standorten. Dies liegt daran, dass hier der Browser unterscheidet, ob es sich um einen GAN-Zugriff oder um einen Zugriff auf externe Netze handelt.

Im Falle eines Zugriffs auf GAN-Applikationen werden Zugriffe auf den Cluster der Intra-Proxy-Server gesendet und direkt dort verarbeitet. Da hier auch keine Benutzerkontrolle vorgenommen wird, ist es nicht erforderlich, in der Zentrale selbst eine Schicht zwischen Proxy-Proxy-Verbindungen und Client-Proxy-Verbindungen einzuziehen.

Im Falle eines Zugriffs auf externe Adressen wird der Proxy-In-Cluster verwendet, der direkt über die Sicherheitskonstruktion und unter Verwendung der Proxy-Ext-Cluster eine Verbindung zum gewünschten externen Server herstellen kann, wenn der verwendete Client über die entsprechende Berechtigung verfügt und die gewünschte Zieladresse nicht generell gefiltert wird.

3.5.2 Nicht-Zentrale

In allen anderen Standorten, sprich in allen Standorten, die nicht selbst über eine Außenanbindung verfügen, werden die Clients so konfiguriert, dass sie auf jeden Fall ihren lokalen Proxy verwenden. Die Proxy-Konfiguration wird so erweitert, dass die in der Hierarchie höher liegenden Proxy-Server als Fall-Backs verwendet werden, wenn der lokale Proxy-Server nicht verfügbar ist, so dass der Client in fast allen Fällen eine Verbindung zur gewünschten Anwendung erhalten wird.

Aus diesem Grund muss auch der Intra-Proxy-Cluster so konfiguriert sein, dass er ggf. (im Fall-Back-Fall) Anfragen an den Proxy-In-Cluster weiterleiten kann.

3.6 Cache

Eine der wichtigsten Funktionen der Proxy-Server ist der Cache, also das Zwischenspeichern von Daten, damit diese nicht über externe Verbindungen wieder und wieder herangeholt werden müssen. Die verwendeten Protokolle unterstützen im Caching eine Nachfrage an den Originalserver, ob sich das Dokument verändert hat – wenn nicht, wird das Dokument nicht neu übertragen, sondern der Client, der den Proxy nach dem Dokument gefragt hat, bekommt das zwischengespeicherte Dokument; wenn sich das Dokument verändert hat, wird es heruntergeladen und auf dem Proxy-Cache-Server zwischengespeichert und steht somit auch anderen Clients zum Abruf zur Verfügung.

Generell ist die Art des Caching und die Strategie des Caching abhängig von der Software zu sehen. Es sollte aber ein moderner Priority-Queue-Cache-Ansatz gewählt werden, der auch von allen Produkten, die im WebFarm-Standard eingesetzt werden, verwendet wird.

Zusätzlich ist die Frage nach *Garbage-Collection*, also dem Sammeln alter Daten, die im Zwischenspeicher liegen und nie oder nur sehr selten abgerufen werden und wichtigeren Daten den verfügbaren Speicherplatz wegnehmen, zu stellen; eine solche Garbage-Collection oder eine Reorganisation des Cache-Space sollte auf jeden Fall definierbar sein.

3.6.1 Was kann gecached werden?

Die Grundsatzfrage nach der Effizienz von Anwendungen oder der Effizienz von Proxy-Servern an sich ist, was kann eigentlich alles gecached werden, und was muss eins zu eins durchgeschleust werden.

Die Antwort ist relativ einfach. Gecached werden können alle Informationen, die eine gleich bleibende URL, also eine gleich bleibende Lokation in einer Anwendung, haben und nicht direkten Programmcode oder Verschlüsselung beinhalten. Nicht gecached werden können alle Informationen, die dynamische URLs haben, Anwendungen, die auf dem Server ausgeführt werden müssen, um Programmresultate zu übertragen, oder Anwendungen, die dem Client übertragen werden und dort nicht abgelegt werden. Nicht gecached werden also z.B. Dokumente, die von einem Such-CGI zurückgeliefert werden oder von einem datenbankbasierten Content-Management-Werkzeug, das jeden Zugriff auf die Informationsbasis dynamisch generiert; genauso wenig gecached werden JAVA-Applets, weil diese Informationen enthalten, die sich von Release zu Release ändern können. Ebenfalls nicht zwischengespeichert werden verschlüsselte Informationen, also Informationen, die per SSL übertragen werden.

Caching von Applets und SSL-Übertragungen

Applets werden, weil es ja Applikationen sind, die für das einmalige Ausführen gedacht sind, genauso wenig wie verschlüsselt übertragene Daten nicht zwischengespeichert, da Applets und auch verschlüsselte Daten durchaus cachebar sein sollten, wenn dies der Sicherheitsstufe der Infrastruktur entspricht, was bei internen Netzen gegeben ist.

Es ist darum möglich, Applets mit einem Zertifikat bei der Entwicklung zu versehen, das die Version des Applets und dessen Authentizität kennzeichnet. Der Proxy-Server kann angewiesen werden, dass korrekt identifizierte und versionierte Applets zwischengespeichert werden. In einigen Situationen kann der Proxy auch mit einem eigenen Verschlüsselungszertifikat ausgerüstet werden und kann dann auch gemäß des WebFarm-Standards angewiesen werden, mit SSL übertragene HTML- oder andere Dateien zwischenzuspeichern.

3.7 Reverse-Proxy als Anwendungsschutz

Um einen besonderen Schutz für den Zugriff auf bestimmte Anwendungen zu schaffen, ist es eventuell wünschenswert, den Zugriff auf Anwendungen über einen zentralen Punkt zu etablieren. Hierbei kann ein Proxy-Server verwendet werden, der in entgegengesetzter Richtung arbeitet – als Reverse-Proxy.

3.8 Normaler Zugriff

Bei einer Webanwendung, die in einem geschützten Segment einer Firewall steht, werden die Zugriffe der externen Benutzer meist durch einen Packetfilter auf einen bestimmten Port der Webanwendung (üblicherweise 80 oder 443) beschränkt. Die Anwendung selbst muss sich um Verschlüsselung, Zugriffsberechtigungen und Content-Berechtigungen sowie potenzielle Angriffe kümmern.

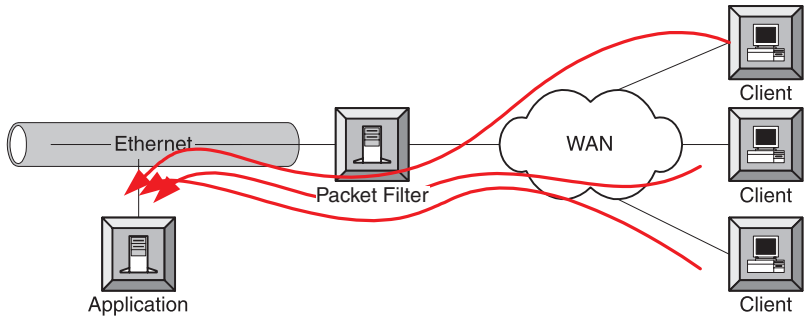

Abbildung 3.4: Normaler Zugriff

Diese Installation bietet üblicherweise ein ausreichendes Maß an Sicherheit, da die Berechtigungsstrukturen in Webanwendungen sehr granulär definiert werden können. Soll aber der Zugriff auf die Anwendung ohne Kontakt zur eigentlichen Anwendung eingeschränkt werden, soll bei externen Zugriffen auf eine interne Anwendung Verschlüsselung zum Einsatz kommen oder auch die Trennung von Zugriffssegmenten erzeugt werden, so ist die normale Lösung nicht ausreichend.

3.9 Zugriff über einen Reverse-Proxy

Konstruktionen mit Reverse-Proxy-Servern können diese Anforderungen erfüllen. Sie bieten einen Zugriffspunkt für zumeist externe Benutzer auf bestimmte Anwendungsgruppen, der vom Unternehmen selbst kontrolliert werden kann und noch nicht dem Sicherheitsniveau der Anwendungen selbst entsprechen muss.

3.9.1 Reverse-Proxy zur Zugriffskontrolle

In der Konfiguration des Reverse-Proxy-Servers kann eine Zugriffskontrolle auf die Anwendung selbst erfolgen. Es stehen hierbei alle Kontrollmechanismen des HTTP-Protokolls zur Verfügung. Grundsätzlich ist die beste Lösung

eine Zugangskontrolle über Zertifikate, aber auch Passwortverfahren sind möglich. Die im WebFarm-Standard eingesetzte Proxy-Software unterstützt auch Side-Includes, über die andere Methoden eingebracht werden können, die aber für spezifische Anwendungen ausgewählt werden müssen. Zusätzlich können bereits bestimmte Teile von Webanwendungen, die über URLs gekennzeichnet werden, durch die Identifikation des Benutzers am Proxy-Server für bestimmte Benutzer überhaupt zugreifbar bzw. bereits am Proxy-Server verweigert werden.

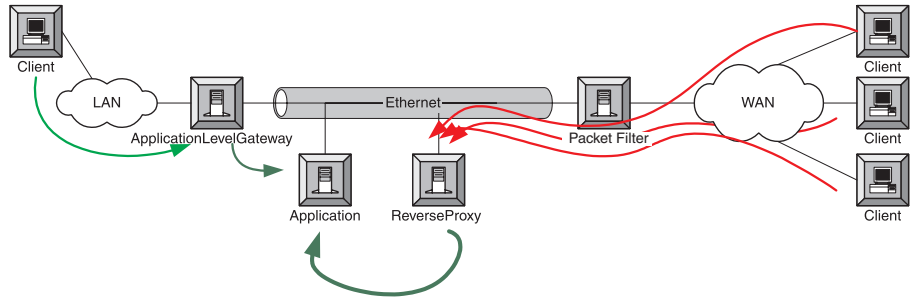

Abbildung 3.5: Zugriff über Reverse-Proxy

Das Vorhandensein einer Zugriffskontrolle über den Proxy-Server sollte die Anwendung nicht davon entbinden, selbst eine Berechtigungsschematik zu implementieren. Es kann so ein durchaus sinnvoller doppelter Schutz geschaffen werden, vor allem wenn der Reverse-Proxy für differenzierte Benutzergruppen eingesetzt wird.

3.9.2 Reverse-Proxy zur Trennung der Zugriffssegmente

Ist diese Trennung sicherheitstechnisch noch nicht ausreichend, können die Segmente, in denen die Anwendung platziert ist und in denen der tatsächliche Zugriff der Benutzer erfolgt, über den Reverse-Proxy getrennt werden.

Dabei steht der Reverse-Proxy in einem anderen Segment des gefilterten Netzwerks als die Anwendung. Ein Zugriff auf dieses Segment ist dann über den Packetfilter nur noch vom Reverse-Proxy aus möglich.

Auf diese Weise entsteht ein zusätzlicher Schutz für den Anwendungsserver selbst, und die Benutzerkontrolle auf bestimmte Serverbereiche kann auf dem Reverse-Proxy vorgezogen werden. Wie bereits erwähnt, kann dies nicht die Anwendung von einer eigenen Benutzerkontrolle entbinden.

Eine zusätzliche Möglichkeit des Reverse-Proxy-Servers ist das Ändern des Namens der Anwendung, wenn z.B. eine Anwendung in der DMZ über einen DMZ-Strukturnamen angesprochen wird, vom GAN oder Internet aus

aber über einen anderen Namen erreichbar oder zusätzlich über einen anderen Namen erreichbar sein soll, kann ein entsprechendes Mapping auf dem Proxy-Server konfiguriert werden. Bei einer solchen Konfiguration ist darauf zu achten, dass entweder die Anwendung so designed ist, dass sie nur relative Links in der eigenen Anwendung verwendet oder dass der Reverse-Proxy-Server, was mit der im WebFarm-Standard vorgeschlagenen Proxy-Umgebung möglich ist, ein Content-Scanning vornimmt und eventuelle Links und Verweise auf den internen Server, der ja nicht angesprochen werden soll, erkennt und in die Adresse des Reverse-Proxy-Servers einträgt.

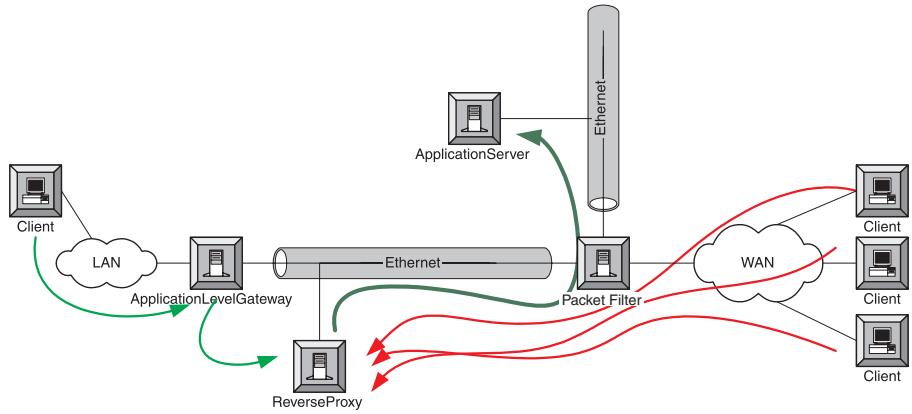

Abbildung 3.6: Reverse-Proxy mit Trennung der Zugriffssegmente

3.9.3 HTTPS-Zugriff auf HTTP-Anwendungen

In vielen Fällen macht es Sinn, eine interne Anwendung, die innerhalb des GAN einen Sicherheitsstandard hat, der keine Verschlüsselung vorschreibt, für den externen Gebrauch verschlüsselt anzubieten, da hier im Gegensatz zum GAN keinerlei Kontrolle über die Netzwerkwege besteht.

Anwendungen, die für den internen Gebrauch entwickelt wurden und dann doch einem sicheren Umfeld zugeführt werden sollen, bzw. Anwendungen, die nachträglich durch Verschlüsselung abgesichert werden sollen, können mit einem Reverse-Proxy-Server von HTTP- auf HTTPS-Übertragung, die SSL zur Verbindungsabwicklung verwendet, umgestellt werden.

Dabei erfolgt der verschlüsselte Zugriff vom Client bis hin zum Reverse-Proxy, der in diesem Falle ein eigenes Zertifikat benötigt. Der Reverse-Proxy selbst setzt dann auf den eigentlichen Application-Server um. Der WebFarm-Standard sieht hier zwei verschiedene Konfigurationen vor.

Erstens kann ein ganz normaler Reverse-Proxy-Server verwendet werden, der den Zugriff auf einen internen Application-Server umsetzt, wie das bei

den vorhergehenden Modellen bereits der Fall war. Zusätzlich zu den bisher beschriebenen Aufgaben des Reverse-Proxy-Servers kommt in diesem Fall noch das Umsetzen von HTTPS nach HTTP hinzu.

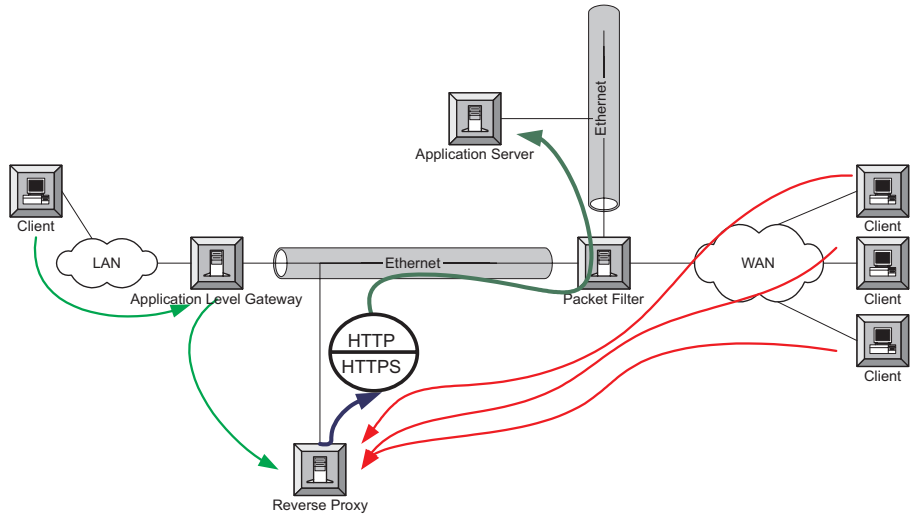

Abbildung 3.7: Reverse-Proxy mit Umsetzung HTTP/HTTPS

Im anderen Falle wird der Reverse-Proxy-Server in den WWW-Server des Application-Servers integriert, und er setzt den Zugriff auf die lokale Maschine des Application-Servers um, eben nur auf den anderen Port. Hierzu kann entweder eine HTTPS nach HTTP-Umsetzung verwendet werden oder eben der Zugriff auf den Webtree des Application-Servers. Letztes ist nur sinnvoll möglich, wenn keine Namenumsetzung oder zusätzliche Benutzerberechtigung erfolgen soll und wenn der Aufwand, einen zweiten, identisch konfigurierten Webserver aufzusetzen, vertretbar ist.

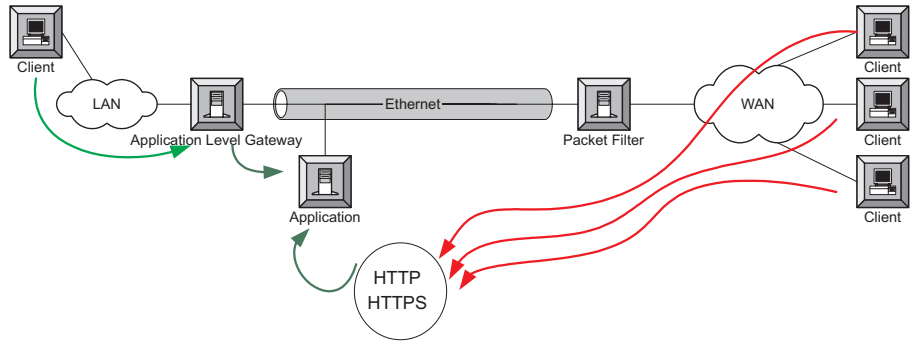

Abbildung 3.8: Integrierter Proxy