

Sonderdruck für Blue Coat Systems

Beschleunigung von SSL-Verkehr

Schneller durch den Tunnel

Der Anteil von SSL-verschlüsseltem (Secure Sockets Layer) Webverkehr wächst pro Jahr um rund 30 Prozent. Viele Hersteller von Unternehmenssoftware migrieren ihre Anwendungen auf Webtechniken, zudem kommen neue Anwendungen auf den Markt, die von Haus aus für die Nutzung via Internet konzipiert sind. Interne wie externe Webanwendungen nutzen SSL, um die Vertraulichkeit übertragener Daten zu gewährleisten. Wer als Administrator diesen Verkehr beschleunigen möchte, sieht sich mit neuen Herausforderungen konfrontiert.

Durch die Verlagerung von immer mehr geschäftskritischen Prozessen nach außen stehen die IT-Verantwortlichen in Unternehmen vor einem Dilemma: Einerseits haben sie extern gehostete Anwendungen wie beispielsweise das beliebte CRM-System von Salesforce.com nicht mehr vollständig unter ihrer Kontrolle. Andererseits müssen sie aber so weit wie möglich sicherstellen, dass diese Anwendungen ebenso performant arbeiten wie intern gehostete geschäftskritische Applikationen. Gleichzeitig muss der Austausch sensibler Daten zwischen Benutzer und Anwendung vertraulich und sicher bleiben. Limitierender Faktor für die Antwortzeit so-

wohl interner wie externer Webanwendungen ist in den meisten Fällen die WAN-Verbindung, über die die Außenstellen ebenso wie Homeoffices und mobile Mitarbeiter auf zentrale IT-Ressourcen zugreifen. Dezentrale Unternehmen führen heute in der Regel den gesamten Internetverkehr von Außenstellen und externen Mitarbeitern zunächst über das WAN in die Zentrale und übergeben ihn erst dort ins Internet, um den Datenfluss an zentraler Stelle kontrollieren zu können. Damit unterliegen aber auch alle geschäftskritischen Anwendungen den Bedingungen auf der WAN-Verbindung und müssen sich beispielsweise die verfü-

bare Bandbreite mit dem restlichen Verkehr zwischen Zweigstelle und Zentrale teilen. An dieser Stelle können Appliances zur WAN-Optimierung zum Einsatz kommen, um die verfügbare Bandbreite effektiver zu nutzen, die Latenz zu verringern, Protokolle zu optimieren und Anwendungen entsprechend ihrer Wichtigkeit zu priorisieren. Bei der Beschleunigung von SSL-Verkehr im WAN gibt es dabei unterschiedliche Ansätze, die jeweils ihre Vor- und Nachteile aufweisen.

SSL-Verbindung

Um die verschiedenen Ansätze voneinander abgrenzen zu können, sollte man sich zunächst kurz den Aufbau einer SSL-Verbindung vor Augen führen. Zunächst schickt ein Client-Computer eine so genannte „Client-Hello“-Nachricht an den gewünschten Server, die unter anderem alle vom Client unterstützten Verschlüsselungsalgorithmen enthält. Dies beantwortet der Server mit einer „Server-Hello“-Nachricht, in der er dem Client unter anderem den höchsten von ihm unterstützten Kryptalgorithmus mitteilt. Ab SSLv3 schickt der Server dem Client im Anschluss noch ein Zertifikat. Dort sind unter anderem der öffentliche Schlüssel des Servers sowie Informationen über die Stelle enthalten, die das Zertifikat ausgestellt und digital signiert hat.

Der Client kann nun prüfen, ob das Zertifikat des Servers gültig ist und ob eine vertrauenswürdige Certificate Authority (CA)

es unterzeichnet hat. Ist dies nicht der Fall, geben Browser typischerweise eine Warnung an den Benutzer aus und fragen nach der weiteren Vorgehensweise. Ist das Zertifikat vertrauenswürdig und gültig, verschlüsselt der Client mit dem öffentlichen Schlüssel des Servers eine zufällig ausgewählte Zahlensequenz – das so genannte Premaster Secret – und schickt es an den Server, der es mit seinem privaten Schlüssel entschlüsselt. Da nur der Server den privaten Schlüssel besitzt, ist auch nur er in der Lage, das Premaster Secret zu entschlüsseln. Auf Basis dieser Daten leiten sowohl Server wie Client anschließend mithilfe von zwei Hash-Funktionen das Master Secret und daraus einen einmaligen symmetrischen Session Key ab. Dieser symmetrische Schlüssel kommt nun bei der Ver- und Entschlüsselung der im Folgenden ausgetauschten Nutzdaten zum Einsatz.

SSL-Offloading

Der hier beschriebene SSL-Handshake und der damit verbundene Austausch von Schlüsseln zwischen Client und Server ist ein rechenintensiver Vorgang, der bei vielen gleichzeitigen Verbindungen den Prozessor eines Webserver deutlich belastet.

PCI-Karten und Netzwerkbeschleuniger

Um die Server-CPU von dieser Aufgabe zu befreien, können so genannte SSL-Accelerator-Cards zum Einsatz kommen. In Form von PCI-Steckkarten übernehmen sie meist den asymmetrischen Teil des SSL-Handshakes, während die symmetrische – und weitaus weniger rechenintensive – Verschlüsselung der Nutzdaten beim Server verbleibt. SSL-Beschleunigungskarten eignen sich also dazu, einzelne Server von Teilen ihrer SSL-Aufgaben zu entlasten – daher die Bezeichnung „Offload“. So genannte Network Accelerators arbeiten nach einem ähnlichen Prinzip. Jedoch sitzt dieser Typ von SSL-Offloadern als eigenständige Appliance im Netzwerk vor einem oder mehreren Webservern und übernimmt für diese die komplette SSL-

Verarbeitung. Beide Arten von SSL-Offloadern dienen dabei rein der Entlastung von Webservern. Bandbreitenengpässe und Latenzprobleme lösen sie ebenso wenig, wie sie den SSL-Verkehr inspizieren und kontrollieren können.

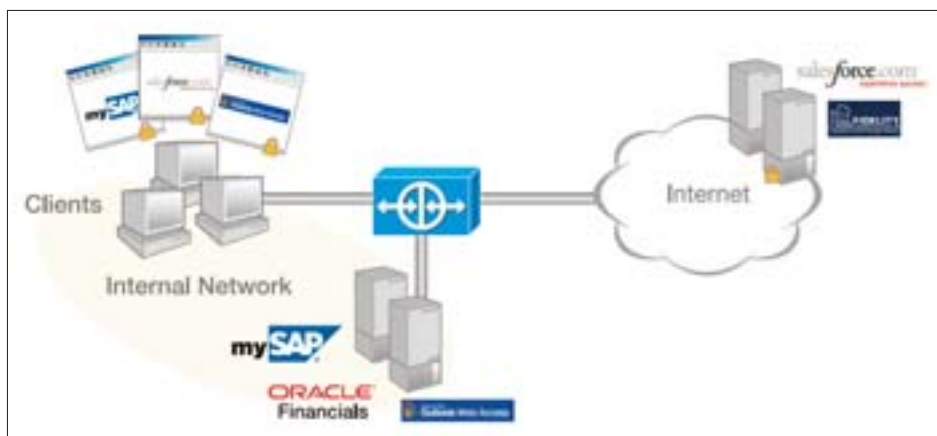
SSL-Proxies

Um neben der reinen Entlastung der Webserver-CPU den SSL-Verkehr optimieren und beschleunigen zu können, setzen die Hersteller von Appliances zur WAN-Optimierung auf so genannte SSL-Proxies. Grundsätzlich führen solche Proxies dabei einen gutwilligen Man-in-the-middle-Angriff auf die Verbindung zwischen Client und Server durch: Dem Client gegenüber verhält sich ein Proxy wie ein Server, dem

Handshakes den symmetrischen Schlüssel nicht generieren kann. Um dieses Problem zu lösen, verfolgen die Hersteller von WAN-Optimierern zwei unterschiedliche Wege.

Zwei Lösungsansätze für SSL-Probleme

Ein Ansatz besteht darin, das Zertifikat des Webserver inklusive seines privaten Schlüssels vom Server auf den Proxy zu verlagern. Dies entspricht weitestgehend der Arbeitsweise von SSL-Offloadern, die mit dem originalen Serverzertifikat den SSL-Handshake durchführen. Allerdings hat dieser Weg einige entscheidende Nachteile. Denn zunächst setzt er voraus, dass die für den Proxy verantwortliche IT-Ab-



Extern gehostete Webanwendungen wie Salesforce.com sind heute häufig für Unternehmen genauso geschäftskritisch wie interne Applikation à la Mysap Quelle: Blue Coat Systems

Server gegenüber wie ein Client. Auf diese Weise erhält ein Proxy Einblick in die ausgetauschten Daten und kann auf Basis dieser und anderer Informationen wie Uhrzeit oder Benutzer Entscheidungen treffen sowie die ausgetauschte Datenmenge beispielsweise durch Kompression verringern. SSL-Proxies stehen dabei zusätzlich vor der Herausforderung, den Datenverkehr von beiden Seiten zunächst entschlüsseln zu müssen, bevor sie ihn kontrollieren und optimieren. Anschließend muss ein SSL-Proxy die Daten wiederum verschlüsseln und an die Gegenstelle weiterleiten. Das Problem dabei ist, dass der Proxy den privaten Schlüssel des Webserver nicht kennt und daher bei der Analyse des SSL-

teilung überhaupt Zugriff auf die privaten Schlüssel der Anwendungsserver hat, deren Verkehr zu optimieren ist. Bei ausgelagerten Anwendungen ist dies naturgemäß unmöglich, da der Anbieter seine Serverzertifikate niemals herausgeben wird. Und selbst bei unternehmensinternen Anwendungen kann es schwierig werden, wenn die Anwendungen in einer anderen Abteilung angesiedelt sind. Ein Vorteil dieses Ansatzes ist hingegen, dass an den Clients im Unternehmen keinerlei Änderungen notwendig sind. Denn der Proxy mit dem privaten Serverzertifikat verhält sich gegenüber einem Client wie der ursprüngliche Server. Allerdings sollte man dabei bedenken, dass der Proxy dann auch wie

zuvor der Webserver gegen Angriffe auf das Zertifikat geschützt werden sollte. Wer also nur ausgewählte interne Anwendungen optimieren will, für den kann dies der richtige Weg sein. Extern gehostete SSL-verschlüsselte Webapplikationen bleiben bei diesem Design aber immer außen vor. Im zweiten Ansatz nutzt der Proxy nicht das Zertifikat des Servers für den SSL-Verbindungsaufbau, sondern generiert auf Basis des ursprünglichen Serverzertifikats selbst ein emuliertes Serverzertifikat pro SSL-Handshake. Dort sind bis auf den öf-

nen wie auch zu externen Anwendungen kontrollieren und optimieren kann. Der Nachteil ist, dass Unternehmen bestimmten SSL-Verkehr unter Umständen gar nicht verarbeiten wollen oder dürfen – beispielsweise wenn Mitarbeiter online ihren Kontostand abfragen. Hier sollte der Proxy daher zudem in der Lage sein, je nach Benutzer, URL oder Kategorie der angesteuerten Website zu unterscheiden, ob die jeweilige Session über den Proxy läuft oder unberührt – aber auch unoptimiert – durchgelassen wird.

nächsten Schritt können WAN-Optimierer bei webbasierten Anwendungen die Protokolle HTTP und TCP optimieren. Bei HTTP versucht der Proxy beispielsweise, möglichst viele Übertragungen zu parallelisieren und so den Aufbau einer Webseite zu beschleunigen. Liegen die Daten von SSL-verschlüsselten Webanwendungen auf dem Proxy im Klartext vor, kann dieser einzelne Objekte oder Teile davon auch zwischenspeichern (Object Caching oder Byte Caching).

Object Caching und Byte Caching

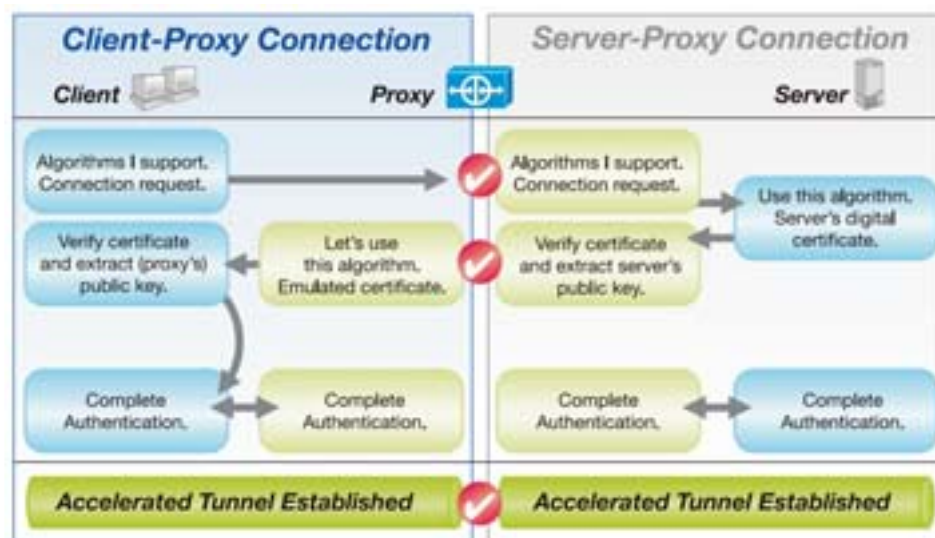
Während Object Caching komplette Dateien zwischenspeichert und bei mehrfacher Anforderung nur einmal über das WAN überträgt, wendet Byte Caching diesen Mechanismus auf Dateifragmente an. Sind beispielsweise bei einer Powerpoint-Präsentation auf einem Sharepoint-Server seit dem letzten Download noch zwei Folien hinzugekommen, überträgt der WAN-Optimierer nur noch die beiden neuen Folien über die WAN-Strecke und holt die restlichen Daten aus seinem Byte Cache. Der Text von HTML-Seiten schließlich lässt sich mithilfe von Datenkompression in der Regel um den Faktor drei verkleinern, was ebenfalls Bandbreite auf der WAN-Strecke spart.

Fazit

Immer mehr SSL-verschlüsselte Webanwendungen zwingen Unternehmen mittelfristig dazu, sich auch mit der SSL-Beschleunigung im WAN auseinanderzusetzen. Wer lediglich bestimmte interne Anwendungen beschleunigen will und Zugang zu deren Serverzertifikat hat, für den reichen Lösungen mit einer SSL-Offload-ähnlichen Architektur aus. Wer hingegen jede externe wie interne Anwendung beschleunigen will oder muss, der kommt nicht um eine Lösung herum, die Serverzertifikate emuliert.

Martin Walzer/wg

Martin Walzer ist Senior Network Consultant bei Blue Coat Systems.



Um interne wie externe SSL-verschlüsselte Anwendungen optimieren zu können, muss ein WAN-Beschleuniger das Serverzertifikat emulieren und mit seinem eigenen öffentlichen Schlüssel versehen
Quelle: Blue Coat Systems

fentlichen Schlüssel des Servers dieselben Daten enthalten wie im Serverzertifikat. Der öffentliche Schlüssel stammt aber jetzt vom Proxy. Anschließend signiert dieser noch das temporäre Zertifikat und präsentiert es dann dem Client. Auf Serverseite baut der Proxy parallel eine ganz normale SSL-Verbindung zum Zielservers auf. Damit der Client nun dem Benutzer nicht meldet, dass das emulierte Zertifikat des Proxies nicht vertrauenswürdig sei, muss der Administrator den Proxy zur Liste der vertrauenswürdigen Certificate Authorities (CAs) auf allen Clients hinzufügen. Dies geschieht in der Praxis meist einmalig über die jeweils im Unternehmen etablierten Mechanismen zur Softwareverteilung. Der Vorteil dieser Architektur ist, dass der Proxy ab sofort den kompletten SSL-Verkehr zu inter-

Doch welche Möglichkeiten haben Proxies in WAN-Optimierern, um SSL-verschlüsselte Inhalte zu beschleunigen?

SSL-Optimierung

Zunächst können die Geräte, da sie jetzt den Inhalt des Verkehrs kennen, bestimmten internen und externen Webanwendungen mehr Bandbreite einräumen als dem übrigen Verkehr auf der WAN-Verbindung. Bandbreitenoptimierung sorgt also dafür, dass unerwünschte Protokolle und Anwendungen, die ebenfalls SSL nutzen, erst gar nicht auf die WAN-Strecke gelangen. Wichtige Anwendungen erhalten zudem eine garantierte Mindestbandbreite, während nicht geschäftsrelevante Anwendungen gegebenenfalls ausgebremst werden. Im