

## Sicherheit im Web Hosting Umfeld

Web-Server und webbasierte Applikationen sind inzwischen das beliebteste Ziel von Hackern. In der *CSI / FBI Computer Crime and Security Survey* von 2004 berichtete jeder Befragte von Incidents auf seinem Webserver. Es ist also nicht die Frage, ob hier überhaupt ein Problem vorliegt, sondern allenfalls, wie groß das Problem ist und wie man die Sicherheit seines Webserver besser in den Griff bekommen kann. Eine Variante, die gelegentlich in Betracht gezogen wird, ist den Webserver aus dem Haus zu geben und bei einem externen Hosting-Anbieter betreiben zu lassen. Dadurch, so hoffen manche Organisationen, hat man sich des Problems erledigt und das interne Unternehmensnetz ist vollständig von dem potentiell gefährdeten Webserver getrennt.

Nicht immer ist dieser Weg jedoch machbar oder sinnvoll und in einigen Fällen kann das das sogenannte Outsourcing des Webserver sogar zu einer deutlichen Verschlechterung der Gesamtsicherheit führen. Im folgenden Artikel werden einige Vorteile, Herausforderungen und Probleme bei einer Hosting-Entscheidung in Bezug auf die IT-Sicherheit diskutiert.

Es gibt viele Gründe, die Web-Hosting interessant machen können. Angefangen von den offensichtlichen Fällen bei kleinen Firmen oder Privatpersonen, die keine eigene performante Internet-Anbindung besitzen und deshalb eine eigene Web-Präsenz bei einem Host mit direkter Backbone-Anbindung aufbauen wollen bis hin zu Firmen, die den Betriebsaufwand für einen eigenen Webserver scheuen und diese Tätigkeiten gerne einem externen Partner überlassen, der den Server vollständig in einem speziellen Rechenzentrum betreibt.

Aber auch wenn es um die Sicherheit des Webserver geht, gibt es gerade bei kleineren Firmen häufig die Motivation, dass man mit der Sicherheit oder potentiellen Angreifbarkeit des eigenen Webserver am liebsten nichts zu tun hätte. Die Abgabe des Servers an einen Outsourcing-Anbieter beendet dieses Thema zumindest gedanklich für den Verantwortlichen.

Andererseits bringt ein extern gehosteter Webserver auch neue Bedrohungen mit sich. Die Entscheidung, ob im Einzelfall der Aufbau eines neuen Webserver in einer eigenen Firewall-DMZ des Unternehmens oder bei einem externen Hosting-Anbieter die bessere Lösung ist, hängt von vielen Faktoren ab. Um sich der Frage langsam zu nähern, sollte man zunächst überlegen, welchen Bedrohungen der Webserver als solches unabhängig von seiner späteren Positionierung ausgesetzt ist.

Die erste und offensichtlichste Bedrohung ist immer der direkte Einbruch in den Webserver durch einen Angreifer. Die Ursachen für einen solchen Einbruch können auf verschiedenen Ebenen liegen. Während es früher meist Verwundbarkeiten im Betriebssystem bzw. im Webserver-Prozess waren, die durch das rechtzeitige Einspielen von Patches bzw. Service-Packs behoben werden konnten, sind es heute bevorzugt Probleme auf der Applikationsebene in Form von Programmierfehlern in den CGI- und PHP-Skripten oder Java-Servlets. Die eigenen Programmierer oder die Entwickler eines externen Dienstleisters, die für die interaktive Funktionalität des Web-Auftritts sorgen, vergessen häufig, dass es nicht nur brave Anwender gibt, sondern auch Angreifer, die gezielt versuchen, jeden Aspekt und jede Funktionalität einer Web-Anwendung zu missbrauchen, um in den Server einzubrechen, um aus dem Server ein Spam-Relay zu machen oder einfach um die Homepage zu verunstalten und damit angeben zu können, dass man es geschafft hat, dort einzubrechen.

Moderne Angriffe auf Applikationsebene nutzen vor allem Fehler in der individuellen Applikation aus, um beispielsweise Befehle in Backend-Datenbanken zu injizieren („SQL-Injection“), um URL-Parameter oder Session IDs zu manipulieren oder um per „Cross-Site-Scripting“ Java-Script-Befehle in dynamisch aufgebaute Seiten zu platzieren, die andere Anwender täuschen oder ihre Session-Informationen stehlen.

Ein externer Web-Hosting-Anbieter kümmert sich in der Regel um das ständige Einspielen von Service-Packs bzw. Patches. Dadurch können klassische Angriffspunkte auf der Ebene des Betriebssystems und der Webserver-Plattform minimiert werden, ohne dass der Auftraggeber damit belastet wird. Je höher die Verwundbarkeiten bzw. Angriffe jedoch im ISO/OSI Referenzmodell einzuordnen sind, umso weniger kann ein Hosting-Anbieter die Probleme beheben, ohne dass der Besitzer der Inhalte davon betroffen ist. Wenn zum Beispiel im Extremfall eine Verwundbarkeit in einem anwendungsspezifischen PHP-Script des Servers vorliegt, wird der Hoster kaum die von seinem Kunden gelieferten Programme umprogrammieren wollen oder können. Ein Grenzfall sind Verwundbarkeiten in Plattform-Elementen, die wenn sie gepatcht werden, anders funktionieren und deshalb die Scripte bzw. Programme des Kunden betreffen.

Der Kunde kann bei interaktiven Inhalten seines Webserver gerade nicht davon ausgehen, dass ein Hoster sich um die Sicherheit des gesamten Servers kümmert. Sobald es um Verwundbarkeiten in den Inhalten bzw. in der Eingabeverarbeitung auf dem Webserver geht, ist der Kunde involviert und der Programmierer der Webseiten und Verarbeitungen muss selbst bei der Problemlösung mitarbeiten.

Angriffe auf einen Webserver müssen nicht unbedingt so aussehen, dass der Angreifer sofort vollständig in den Server einbricht und ihn auf Betriebssystem-Ebene unter seine Kontrolle bringt. Ein ganz anderes Beispiel sind Web-Applikationen, die nur für eine geschlossene Benutzergruppe gedacht sind und deshalb zunächst eine Benutzeranmeldung mit Name und Passwort oder sogar mit einer starken Authentisierung erfordern. In einer solchen Umgebung besteht das erste Ziel des Angreifers darin, die Zugriffskontrolle und Benutzeranmeldung zu umgehen. Derartige Angriffe finden heute fast immer auf Applikationsebene statt und haben ihre Ursachen in Fehlern in der kundenspezifischen Programmierung. Auch hier leisten Hosting-Anbieter derzeit kaum einen Beitrag zur Sicherheit.

Hinterfragt man die Leistungen eines Hosting-Anbieters in Bezug auf die Sicherheit der von ihm zu hostenden Webserver, so bekommt man meist Informationen zur physischen Sicherheit des Rechenzentrums, zur Überwachung oder zum Brandschutz. Diese Themen sind sicher ein wichtiger Teil der gesamten Sicherheit, haben aber mit dem typischen Angriff auf Webserver wenig zu tun. In diesen Bereichen beschränken sich die Anbieter meist auf SSL-Verschlüsselung, das regelmäßige Patchen der Systeme, gelegentlich noch auf regelmäßiges Scannen der Server nach bekannten Verwundbarkeiten und in manchen Fällen auf optionale vorgeschaltete Firewalls.

Alle diese Maßnahmen sind wichtig und sinnvoll, behindern aber heute keinen Hacker mehr, da die Angriffe bevorzugt über die Applikationsebene erfolgen und damit nicht vom Patch-Stand des Betriebssystems oder Webserver und nicht von bekannten Verwundbarkeiten abhängen. Auch Netzwerk-Firewalls, die Zugriffe auf den Port 80 oder 443 beschränken, haben keine Auswirkungen auf das Problem der Applikationssicherheit, da derartige Angriffe über den erlaubten Zugriff auf den Server erfolgen.

Erschwerend kommt hinzu, dass gerade ein Anbieter von Web-Hosting nicht einen einzelnen Webserver betreibt, sondern in der Regel sehr viele. Selbst wenn ein Kunde davon ausgeht, dass seine eigene Web-Anwendung kaum angreifbar ist und dass die vom Hoster vor den Servern vorgeschaltete Firewall für ihn selbst als Schutz ausreicht, so muss er bedenken, dass hinter der selben Firewall viele weitere und potentiell sehr unsichere Webserver angeschlossen sind. Der erfolgreiche Einbruch in den eventuell unsicheren Nachbar-Server eines anderen Kunden eröffnet dem Angreifer womöglich einen direkten Weg zu dem vermeintlich sichereren Server. Eine Firewall, die vor allen gehosteten Server steht und diese gemeinsam schützen soll, lässt in der Praxis meist beliebige Angriffe zwischen den einzelnen Servern zu.

Besonders kritisch wird die Situation bei einem extern gehosteten Webserver, wenn der Webserver nicht autark funktioniert, sondern auf Daten aus dem internen Netz des Kunden angewiesen ist. Ein Beispiel für ein solches Szenario wäre ein Webserver, der

einen Produktkatalog mit aktuellen Preisen und Verfügbarkeiten anbietet und bei dem der gehostete Webserver die benötigten Daten per SQL-Verbindung von einem Datenbankserver aus dem Netz des Kunden abrufen. Durch eine solche Struktur wird das Hosting meist unsinnig und neben der direkten Gefahr für den Webserver selbst besteht jetzt ein zusätzliches Risiko durch potentielle Zugriffe aus dem Netz des Hosters in das Netz des Kunden. Da der Hosting-Anbieter typischerweise sehr viele verschiedene Webserver mit sehr unterschiedlichem Sicherheitsniveau in einem gemeinsamen Netz betreibt, kann es sein, dass nicht der eigene Server abfragen schickt, sondern ein anderer gehackter Server im selben Netz mit gespoofter IP-Quell-Adresse versucht, die Kommunikationsverbindung auszunutzen.

Sogar in der jüngeren Vergangenheit gab es immer wieder Beispiele, in denen nicht nur einzelne Server bei einem bekannten Hoster gehackt wurden, sondern wo die Administrationswerkzeuge des Anbieters generelle Schwachstellen hatten, so dass man jeden dort betriebenen Server relativ einfach manipulieren konnte.

Um das Problem des Zugriffs vom Hoster in das Netz des Kunden zu umgehen, kommen manche Firmen auf die Idee, die benötigten Daten regelmäßig aktiv aus dem Firmennetz zum externen Server zu replizieren und eventuell auf dem Server zwischengespeicherte Anfragen abzuholen und intern zu verarbeiten. Solche Mechanismen erwecken häufig spontan ein Gefühl der stark verbesserten Sicherheit, erweisen sich jedoch bei näherer Betrachtung oft als noch größeres Sicherheitsproblem. Neben den schon existierenden Risiken des Angriffs von einem schon gehackten Nachbarserver kommen jetzt zusätzliche Verwundbarkeiten auf dem Kommunikationsgegenstück im Netz des Kunden hinzu. Der Aufwand, der jetzt zur Absicherung des Gegenstücks beim Kunden betrieben werden muss, wiegt oft den Vorteil durch das Hosting wieder auf. Wenn der Kunde ohnehin eine schnelle Leitung für die Kommunikation zwischen internen Datenbanken und dem externen Webserver beim Hoster benötigt und wenn dazu noch aufwendige Sicherheitsmechanismen zum Schutz der Kommunikationsbeziehung zwischen Webserver und interner Datenbank aufgebaut werden müssten, dann kann der Webserver auch gleich in einer demilitarisierten Zone der Firewall beim Kunden stehen. So kann man sich die Kosten für das Hosting sparen.

Technisch gesehen wäre es kein größeres Problem für die Anbieter von Web-Hosting, neben dem typischen Angebot bezüglich Patchen und Firewalls auch spezielle Sicherheitsmechanismen für Schutz auf der Applikationsebene gegen einen entsprechenden Aufpreis anzubieten. Allenfalls würde einigen Anbietern das dazu nötige Personal fehlen. Entsprechende Produkte sind jedenfalls seit mehreren Jahren am Markt und werden von großen Unternehmen, die die Sicherheit ihrer Web-Applikationen selbst in die Hand genommen haben und die das Problem der Sicherheit auf Applikationsebene verstanden haben, erfolgreich eingesetzt.

Sogenannte Web-Applikations-Filter (WAFs) arbeiten meist auf der Basis eines Reverse-Proxies und kontrollieren jede URL, jedes Cookie und jede Formular-Eingabe gegen zuvor definierte oder gelernte Wertebereiche und maximale Längen. Dadurch können im Gegensatz zu klassischen Firewalls nahezu alle Angriffe auf Webserver verhindert werden. Der Aufwand, um solche Mechanismen aufzubauen, mag zwar zunächst dramatisch klingen, beschränkt sich in der Praxis aber durch intelligente Lern-Mechanismen der Produkte auf wenige Tage bei der Inbetriebnahme und auf wenige Stunden bei typischen Änderungen der Web-Applikationen.

Leider werden entsprechende Mechanismen bisher von nahezu keinem Hoster angeboten, was vermutlich an der bis jetzt fehlenden Nachfrage und an dem fehlenden Verständnis der Problematik auf Seite der Kunden liegt. Die bevorzugten Verkaufsargumente beim Hosting sind eher der Preis und die Performanz als das Thema Sicherheit, woran man auch erkennen kann, dass Kunden, die großen Wert auf Sicherheit legen, oft erst gar keine Hoster anfragen, sondern ihre Server selbst betreiben und schützen.

Zusammenfassend kann man sagen, dass Web-Hosting gerade bei kleineren Firmen ohne geeignete eigene Internet-Anbindung und bei statischen Web-Präsentationen ohne dahinter liegende Transaktionen sicherlich eine sinnvolle Lösung darstellt.

Sobald der Webserver jedoch nicht mehr nur eine statische Präsentation liefert, sondern interaktive Applikationen anbietet, die Transaktionen auf internen Systemen beim Kunden auslösen, entstehen zusätzliche Risiken, die die Attraktivität des externen Hostings deutlich mindern. In jedem Fall bieten die Hoster heute kaum Sicherheitsmechanismen an, die Angriffe auf Applikations-Ebene abwehren könnten. Wer hier eine angemessene Sicherheit wünscht, muss dies selbst in die Hand nehmen und sich beispielsweise aus dem Angebot der kommerziell verfügbaren „Web-Applikations-Filter“ bedienen.

Ein komplettes Abgeben der Verantwortung für die Sicherheit wird ohnehin nie funktionieren, da kein Anbieter von Web-Hosting die Verantwortung für Programme übernehmen kann, die der Kunde selbst oder eine dritte Firma entwickelt hat. Auch wenn es „nur“ zu einem Website Defacement kommt, kann der imaginäre Schaden beträchtlich sein und der Hosting-Anbieter wird dafür nicht aufkommen.

***Stefan Strobel, Geschäftsführer der cirosec GmbH und Buchautor***