



# Die 7 klassischen Denkfehler zum Thema Web Application Security

**Obwohl es sich inzwischen herumgesprochen hat, dass Web-Applikationen das derzeit beliebteste Einfallstor für Hacker sind, halten sich auch unter erfahrenen Administratoren hartnäckig einige Fehleinschätzungen. Auf den folgenden Seiten werden wir mit den sieben klassischen Denkfehler aufräumen.**

## Denkfehler 1

„Über unsere Web-Applikationen erhält man keinen Zugang zu unseren Systemen.“

Zielgerichtete oder professionelle Datenzugriffe laufen oftmals im Verborgenen und über verdeckte Pfade ab - gerade Web-Applikationen bieten Hackern vielfältige Ansatzpunkte zum Datendiebstahl und gehören daher heute zu ihren bevorzugten Angriffszielen. Dies ist kein Wunder, da Web-Applikationen per Definition eine elektronische Schnittstelle zu Daten und Transaktionen darstellen.

Waren früher sensible Daten und kritische Transaktionen erst hinter mehrfachen Verteidigungslinien abrufbar, bieten heute Web-Applikationen direkte Zugriffe bis hin zu den wichtigsten Informationen eines Unternehmens. Nirgends sind Hacker daher so nah am gewünschten Ziel wie bei den Web-Applikationen. Nur eine Schwachstelle auf irgendeiner Ebene genügt, um erfolgreich zu sein.

Anders als früher sind deshalb zielgerichtete Informationsdiebstähle an der Tagesordnung. Es werden keine Exploits mehr geschrieben, die Sicherheitslücken von beliebigen Zielen ansteuern, sondern ein interessantes Ziel wird mit technischen Manipulationen gezielt aus dem Tritt gebracht, bis die gewünschten Daten extrahiert werden können. Schon die klassischen Manipulationsmethoden wie Forceful Browsing, Cross-Site-Scripting, SQL/Command Injections oder die Ausnutzung von Business Logik Schwächen führen bei drei Viertel aller Web-Applikationen zum Erfolg.

Diese zielgerichteten Datenzugriffe sind nicht über die Signatur erkennbar und kein reaktiver Sicherheitsschutz wie ein IDS (Intrusion Detection System) oder IPS (Intrusion Prevention System) kann sie verhindern.

Eines der größten Missverständnisse hierbei ist, dass bei einem erfolgreichen Angriff nicht nur die Daten in Gefahr sind, welche die Web-Applikation selbst verwendet. Alle an die Web-Applikation angeschlossenen Systeme und Schnittstellen sind potenziell betroffen.

Es ist nicht unüblich, dass über eine vermeintlich harmlose Web-Applikation, wie zum Beispiel eine Hilfeseite oder ein Firmen-Telefonbuch, die gesamten internen Unternehmensdaten gestohlen werden können.

Die Verbreitung von solchen Angriffen auf der Applikationsebene hat durch die populäre Web 2.0-Technologie noch zugenommen. Diese hoch dynamischen Web-Applikationen, welche Inhalte von anderen Benutzern tausendfach verteilen, werden immer häufiger von organisierten Cybercrime-Gruppen verwendet.

Dass dieses Problem ernst genommen werden muss, belegt ein aktueller Sicherheitsreport<sup>1</sup> zur Cyberkriminalität. Darin wird berichtet, dass rund 25 Prozent aller Unternehmen bereits einmal verschiedenen Attacken zum Opfer gefallen sind, die von Web-2.0-Sites wie Xing, MySpace oder Facebook ausgingen.

Wirksamen Schutz gegen unerlaubte Datenzugriffe bieten Web Application Firewalls (WAFs) zwischen Anwender und Webanwendung, die nur gültige URLs zulassen und somit Backend-Systeme vor illegalem Zugriff schützen.

---

1... Sophos-Security Threat Report - July 2009 update:  
<http://www.sophos.com/sophos/docs/eng/papers/sophos-security-threat-report-jul-2009-na-wpus.pdf>



# Die 7 klassischen Denkfehler zum Thema Web Application Security

## Denkfehler 2

„Die Sicherheit von Web-Applikationen muss schon bei der Entwicklung sichergestellt werden.“

Natürlich lassen Applikationsentwickler Sicherheitsaspekte wie Logik, Fein-Autorisierung oder Datenverarbeitung in die Entwicklung einfließen. Im späteren Einsatz ist die Lösung aber immer Bestandteil einer komplexeren IT-Landschaft, auf die der Entwickler keinen Einfluss mehr hat.

Angeschlossene Komponenten wie Betriebssystem, Bibliotheken, Middleware Web Server oder Datenbank stellen jeweils eigene Sicherheitsrisiken dar. Hinzu kommt, dass Entwickler auch nur die Risiken berücksichtigen können, die zum Zeitpunkt der Applikationsentwicklung bekannt sind. Beim Projektstart können aber Attacken auf die Web-Applikation treffen, die zum Entwicklungszeitpunkt unbekannt sind.

Zwar schaffen Software-Updates und neue Applikationsversionen Abhilfe, doch lassen sich die Sicherheitsvorkehrungen oft nicht schnell genug nachrüsten. Um schnell und sicher auf unvermittelte Bedrohungen reagieren zu können, sollten daher vorinstallierte Sicherheitsmaßnahmen der Web-Applikationen mit einer vorgelagerten WAF kombiniert werden. Leider werden häufig sinnvolle Maßnahmen in der Applikationsentwicklung und der Einsatz einer WAF gegeneinander ausgespielt.

Unternehmen müssen aber erkennen, dass nur die Kombination aus beidem zu effektivem und effizientem Schutz führt.

## Denkfehler 3

„Wir verschlüsseln den gesamten Datenverkehr mit SSL (HTTP/S) und das reicht.“

Das SSL-Netzwerkprotokoll gewährleistet den sicheren Datenverkehr zwischen dem Anwender beziehungsweise Web-Browser und dem Server, nicht die Absicherung des Servers selbst. Die Wahrung von Vertraulichkeit und Integrität übertragener Daten im öffentlichen, nicht-vertrauenswürdigem Internet ist enorm wichtig.

Auch Hacker nutzen diesen Schutz, und so gelangen ihre Angriffe über diesen Weg auch 'sicher' und verschlüsselt bis zum Firmen-Web-Server.

Um diese Attacken früh genug zu erkennen, müssen SSL-verschlüsselte Verbindungen spätestens an den Unternehmensgrenzen enden - leistungsfähige WAFs verschaffen an diesem Punkt die nötige Kontrolle.

Als Wächterinstanz zwischen Anwender und Applikation stoppt die WAF zunächst den einströmenden Datenverkehr, um ihn danach mehrstufig gefiltert weiterzuleiten.

Über diesen Zwischenschritt erreichen nur autorisierte und mehrfach geprüfte Benutzeranfragen den Web-Server. Nach einmal erfolgter Autorisierung kann die WAF die Daten wieder SSL-verschlüsselt weiterschicken, sollte der Backend-Server SSL-Anfragen erwarten.

Übernimmt eine WAF diese Sicherheitsfunktionen, wird der Server entlastet und dadurch beispielsweise die Performance der Anwendungen auf dem Server erhöht.



# Die 7 klassischen Denkfehler zum Thema Web Application Security

## Denkfehler 4

„Unsere Systeme sind immer aktuell gepatcht und wir lassen regelmäßig einen automatischen Scanner laufen, da ist alles auf Grün.“

Automatische Scanner geben einen grundlegenden Überblick über Schwachstellen in einer Unternehmens-IT - die meisten der heutigen Angriffe auf Web-Applikationen erkennen sie jedoch nicht.

Trotz unauffälligem Scanergebnis können Hacker unbemerkt in die Web-Applikation eingedrungen sein. Um gezielten Datendiebstahl aufzudecken, empfiehlt es sich daher, einen professionellen Penetration-Test einzusetzen. Er misst das Sicherheitslevel der Applikationsumgebung, sollte aber immer auch die Prüfung manueller Angriffe umfassen, die auf Reverse Engineering (dem Ausnutzen der Kenntnis interner Systemstrukturen) basieren und ein bis zwei Mal pro Jahr durchgeführt werden.

Automatische Security Scanner und Penetration-Tests prüfen den aktuellen Zustand einer Systemarchitektur und sind immer nur eine Momentaufnahme, das heißt, sie bieten keinen proaktiven Schutz der Systeme.

Häufig stehen für kundenspezifische Schwachstellen entsprechende Updates oder Patches nicht unmittelbar zur Verfügung.

Mit einer WAF kann ein Unternehmen im Sinne von „virtuellem Patching“ unmittelbar auf ein erkanntes Leck reagieren und unerlaubte Server-Anfragen verhindern.

Das gibt dem Unternehmen die Zeit, im Hintergrund geordnet eine Aktualisierung zu implementieren.

## Denkfehler 5

„Unsere Web-Applikationen sind sicher, bei uns ist noch nie etwas passiert.“

Diese Annahme ist riskant, weil sich der Anwender oftmals in falscher Sicherheit wähnt:

Denn laut Gartner sind drei von vier Web-Applikationen angreifbar, wobei drei Viertel aller Angriffe heute auf Web-Applikationen zielen.

Dabei hinterlassen Hackerzugriffe auf Web-Anwendungen oft keine Spuren und werden nicht entdeckt, weil die Daten nicht verschwinden oder verändert werden - alle Anwendungen funktionieren normal weiter und es werden auch keine Systemzugriffe verzeichnet.

Die heutige Wahrnehmung scheint immer noch geprägt zu sein von den Viren und Trojanern aus früheren Jahren, in denen ein Angriff stets auffällige Folgen hatte.

Das Ziel jetziger Attacken besteht darin, möglichst unbemerkt Daten zu stehlen. Für solche zielgerichteten Angriffe existieren keine vorher bekannten Signaturen. Um auch gegen diese Angriffe gewappnet zu sein, ist ein dynamischer Schutz nötig, der sich an der konkreten Applikation orientiert und nicht an Tausenden (reaktiver, häufig sogar veralteter) Signaturen.

Neben dem allgemeinen Datenschutz erfüllen Security-Lösungen mit PCI DSS-Schutz zusätzlich die gesetzlich vorgeschriebenen Compliance-Anforderungen in der Abwicklung von Kreditkartentransaktionen von Dienstleistern im Finanz- oder eCommerce-Bereich.



# Die 7 klassischen Denkfehler zum Thema Web Application Security

## Denkfehler 6

„Bei uns gab es Angriffe, aber es sind keine Daten gestohlen worden.“

Solche Aussagen von Unternehmen tauchen immer dann in den Medien auf, wenn eine Schwachstelle in einem System publik wurde.

Das Problem ist, dass elektronischer Datendiebstahl meist nicht von normalen Anwendungszugriffen zu unterscheiden ist. Somit kann das Unternehmen nicht wissen, ob schon lange jemand sehr viele Daten über eine Schwachstelle elektronisch kopiert hat - diese Art des Angriffs hinterlässt schließlich keinerlei Spuren.

Anders als bei Viren oder Trojanern, die noch vor einigen Jahren im Umlauf waren, werden die Systeme auch bei gezielten Angriffen nicht beeinträchtigt und laufen für den Benutzer wie gewohnt weiter.

Am wirksamsten ist daher der proaktive Schutz der Systeme mit Security-Lösungen mit mehreren Sicherheitsstufen. Der wichtigste Filter ist die Authentisierungsabfrage an den Benutzer, die den Anwendungen vorgelagert ist. Sie stellt sicher, dass nur Befugte Zugang erhalten und überhaupt mit dem Applikationsserver interagieren dürfen.

An zweiter Stelle kommt die dynamische Filterung, welche nur gültige Server-Anfragen zulässt, ohne sich auf Signaturen zu stützen. Zudem lassen sich über ein Reporting die Zugriffe und abgefragten Daten der angemeldeten ID-Nummern genau zurückverfolgen. Voraussetzung: Ein Reverse Proxy Server, der vor den Web-Applikationsservern installiert ist und Netzwerkverbindungen und Netzwerkprotokolle wie SSL abfängt.

## Denkfehler 7

„Wir nutzen bereits einen Reverse Proxy Server und setzen die besten und teuersten Firewalls ein, sogar zwei verschiedene hintereinander.“

Netzwerk-Firewalls prüfen möglichst in Echtzeit den Datenverkehr zum Web-Server beziehungsweise die Signaturen und Protokolle der Nutzeranfragen an den Server.

Die Firewall erkennt aber bestenfalls einfache Angriffe mittels vordefinierter Signaturen. Um die meist getarnten Hackerangriffe zu identifizieren, muss aber eine Firewall wenigstens auch auf den verschlüsselten Datenverkehr zugreifen können, was meist nicht der Fall ist.

Ein wirksamer Schutz von Web-Applikationen, der über die reine Filterung von URL-Signaturen hinausgeht, muss sich insbesondere auch mit applikatorischen Themen wie zum Beispiel der vorgelagerten Authentisierung, der Cookie Protection und dem Schutz von URL-Adressen und HTML-Formularen auseinandersetzen. Auch muss der Zugangsweg über manipulierte URLs versperrt werden. Nur Web-Application-Security-Lösungen, die mehrstufig - sowohl statisch und dynamisch - alle Abfragen und Daten beim Zugriff auf die Web-Anwendungen filtern, bieten auch vor noch unbekanntem Angriffen proaktiven Schutz.