

Erratum JM 6.09

Im Artikel „Niemals ungeschützt – Web Application Security“ von Thomas Krautgartner und Marc Hölterhoff wurden im Kasten auf S. 50 die Verweise falsch abgedruckt. Die nachstehende Darstellung zeigt die korrekten Verweise.

Bekannte Angriffsvektoren

Cross-Site Scripting (XSS)

Beim Cross-Site Scripting [5] versucht ein Angreifer JavaScript- oder HTML-Quellcode in die Anwendung einzubringen ohne dass dieser von der Anwendung maskiert wird. Im Moment der Ausführung im Browser des Opfers kann der Angreifer zum Beispiel die Inhalte der Webseite clientseitig überladen oder das „Session-Cookie“ auslesen.

Cross-Site Request Forgery (CSRF)

Dieser Angriffsvektor [12] wird oft auch als „Session Riding“ bezeichnet. Dabei gelingt es dem Angreifer, sein Opfer einen präparierten Link ausführen zu lassen, während das Opfer in der anzugreifenden Anwendung angemeldet ist. Hierdurch führt das Opfer unbemerkt vom Angreifer festgelegte Aktionen unter eigenem Namen und eigenen Berechtigungen aus.

Session Hijacking

Hiermit [26] wird allgemein das Verwenden von fremden Sitzungen bezeichnet, nachdem beispielsweise mittels „Cross-Site Scripting“ das „Session-Cookie“ gestohlen wurde. „Session Hijacking“ ermöglicht es dem Angreifer in der angegriffenen Anwendung unbemerkt die Rolle seines Opfers mit all seinen Rechten einzunehmen.

SQL Injection

Hierbei [6] versucht ein Angreifer bösartige SQL-Befehle über Formular- oder URL-Parameter in eine Anwendung zu injizieren. Ziel dieses Angriffs ist es, ein dahinter liegendes Datenbank-Managementsystem (DBMS) zu kompromittieren beziehungsweise Informationen auszulesen.

Parameter Tampering

Beim „Parameter Tampering“ [13] handelt es sich um eine gezielte Manipulation von Formular- oder URL-Parametern, welche die Anwendung als nicht beliebig manipulierbar vorgesehen hat. Dies gilt beispielsweise für versteckte Formularfelder oder Auswahlboxen mit festgelegten Inhalten sowie eventuell auch spezielle URL-Parameter in Links.

Session Fixation

„Session Fixation“ [28] ist eine Angriffsform, bei welcher ein Angreifer sein Opfer dazu bewegt, sich an einer Anwendung anzumelden, zu welcher sich der Angreifer bereits im Vorfeld eine gültige „Session-ID“ hat erzeugen lassen. Dies kann entweder an gemeinsam genutzten Arbeitsplatzrechnern oder mittels anderer Angriffsvektoren wie etwa „Cross-Site Scripting“ geschehen. Ein Session Hijacking ist die Folge dieser Angriffsform, da das Opfer keine vermeintlich neue Session nutzt sondern die vom Angreifer vorher erzeugte.

Directory Traversal

Diese Angriffsform [2] wird oft auch als Unterform des „Forceful Browsing“ bezeichnet. Ziel des Angreifers ist es durch direkte Manipulation in der URL Zugriff auf Verzeichnisse außerhalb der Dokumentenwurzel des Webservers zu erlangen, um somit vertrauenswürdige Dateien auslesen zu können.

Bruteforcing

Dieser Angriffsvektor [29] wird häufig verwendet, um schwache Passwörter in Anmeldemasken zu erraten. In Verbindung mit einer wörterbuchbasierten Automatisierung kann eine Anwendung angegriffen werden, welche keine Maßnahmen zur Erkennung von „Bruteforce-Angriffen“ anwendet.

Command Injection

Hierbei [9] versucht ein Angreifer Befehle einer Skriptsprache oder Betriebssystemkommandos über Formular- oder URL-Parameter in eine Anwendung zu injizieren. Ziel ist es, diese auf dem Server zur Ausführung zu bringen.

Http Response Splitting

Durch Manipulation von Headern oder Parametern [30] soll die Ausgabe der Anwendung dahingehend verändert werden, dass in den „Response-Headern“ ein zweiter „Response“ eröffnet wird, dessen Inhalt der Angreifer komplett kontrollieren kann.

Information Disclosure

Bei diesem Angriffsvektor [7] gibt die Webanwendung dem Angreifer wertvolle technische Informationen, beispielsweise in Form ausgegebener „Java Stack Traces“, für weitere Angriffe preis. Dies können unter anderem Typen eingesetzter Web-Frameworks sein. Der Angreifer ruft die ungewollten Ausgaben durch manipulierte Eingaben hervor.

Redirect Injection

Im Rahmen der „Redirect Injection“ [10] injiziert ein Angreifer einer Anwendung Links, zum Beispiel über Formular- oder URL-Parameter, welche bei Redirects innerhalb der Anwendung ausgeführt werden. Damit leitet er seine Opfer auf bösartige Webseiten. Diese Angriffstechnik wird zum Teil auch für „Phishing-Attacken“ eingesetzt.

Links & Literatur

- [1] <http://tomcat.apache.org/security-6.html>
<http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2008-2938>
<http://sunsolve.sun.com/search/document.do?assetkey=1-66-245246-1>
- [2] http://en.wikipedia.org/wiki/Directory_traversal
http://www.owasp.org/index.php/Testing_for_Path_Traversal
- [3] <http://www.webappsec.org/>
- [4] <http://www.webappsec.org/projects/statistics/>
- [5] http://en.wikipedia.org/wiki/Cross-site_scripting
http://www.owasp.org/index.php/Cross_Site_Scripting
<http://www.cgisecurity.com/xss-faq.html>
- [6] http://en.wikipedia.org/wiki/SQL_injection
http://www.owasp.org/index.php/SQL_injection
- [7] http://www.webappsec.org/projects/threat/classes/information_leakage.shtml
- [8] https://www.owasp.org/index.php/Top_10_2007
- [9] http://www.owasp.org/index.php/Command_Injection
http://en.wikipedia.org/wiki/Code_injection
- [10] <http://www.milw0rm.com/papers/289>
- [11] <http://www.owasp.org/index.php/HTTPOnly>

- [12] http://en.wikipedia.org/wiki/Cross-site_request_forgery
http://www.owasp.org/index.php/Cross-Site_Request_Forgery
<http://www.cgisecurity.com/csrf-faq.html>
- [13] http://www.owasp.org/index.php/Web_Parameter_Tampering
<http://www.cgisecurity.com/owasp/html/ch11s04.html>
- [14] http://www.owasp.org/index.php/Category:OWASP_WebScarab_Project
- [15] <http://www.owasp.org/>
- [16] <http://www.w3.org/TR/html4/sgml/entities.html>
- [17] <http://www.bsi.de>
- [18] <http://www.bsi.bund.de/literat/studien/websec/WebSec.pdf>
- [19] <http://cwe.mitre.org/top25/>
- [20] http://www.owasp.org/index.php/Category:OWASP_Stinger_Project
- [21] <http://www.hdiv.org/>
- [22] Matthias Rohr: Einbruchssichere Webanwendungen, Java Magazin 02.08
- [23] <http://www.phion.com/DE/Pages/airlockstart.aspx>
- [24] <http://www.modsecurity.org/>
- [25] <http://www.webcastellum.org/>
- [26] http://en.wikipedia.org/wiki/Session_hijacking
http://www.owasp.org/index.php/Session_hijacking_attack
- [27] <http://www.fortify.com/products/sca/>
- [28] http://en.wikipedia.org/wiki/Session_fixation
http://www.owasp.org/index.php/Session_Fixation
<http://shiflett.org/articles/session-fixation>
- [29] http://www.owasp.org/index.php/Brute_force_attack
- [30] http://en.wikipedia.org/wiki/HTTP_response_splitting
http://www.owasp.org/index.php/HTTP_Response_Splitting
http://www.webappsec.org/projects/threat/classes/http_response_splitting.shtml