Social Hacking mit technischen Hilfsmitteln stoppen

Wie mit technischen Schutzmassnahmen die psychologischen Tricks der Hacker vereitelt werden können. **VON MARC RUEF ***

Social Hacking ist nicht erst seit dem Medienrummel um Amerikas «Superhacker» Kevin Mitnick eine klassische Disziplin der Computerkriminalität. Psychologische Tricks werden gut und gerne als erfolgsversprechende Alternative zu technischen Attacken auf Computersysteme eingesetzt. Mit vorgespielten Telefonanrufen, gefälschten E-Mails oder manipulierten Webauftritten sollen Anwender zu kompromittierenden Handlungen – wie zum Beispiel die Herausgabe sensitiver Benutzerdaten – verleitet werden [1].

Derlei Angriffsformen sind besonders dann gefährlich, wenn sie technisch gestützt und in legitime Bereiche eingebettet umgesetzt werden können. Eine externe Kopie einer Webseite ist im Normalfall spätestens dann zu erkennen, wenn man die befremdliche URL in der Adresszeile des Webbrowsers genauer betrachtet. Kann jedoch der Inhalt auf der Originalseite so manipuliert werden, dass sich durch einen Angreifer eigene Daten einschleusen lassen, behält die Attacke ihre vermeintliche Authentizität.

Eine Grundlage solcher technisch gestützter Angriffe ist eine entsprechende Anfälligkeit des Original-Webangebots. Ein Angreifer versucht die Darstellung der Seiten so zu manipulieren, dass diese die von ihm gewünschten Informationen bereithält oder zusätzliche Verarbeitungen initiiert.

Das Problem der HTML-Injection

Die einfachste Methode einer solchen Manipulation ist in der klassischen HTML-Injection gegeben. Dabei nutzt der Angreifer einen Fehler in der Eingabeüberprüfung einer Applikation aus [2]. Benutzt die Anwendung die übermittelten Daten ohne

* Marc Ruef ist Buchautor und arbeitet als Security Consultant bei der Zürcher Scip AG.

zusätzliche Überprüfung in der Ausgabe, können eigene Meldungen generiert werden. Der Angreifer könnte eine Meldung ausgeben, die da lautet: «Der Dienst ist zur Zeit nicht verfügbar. Bitte schicken Sie zwecks Recovery Ihr Passwort an die folgende Adresse.»

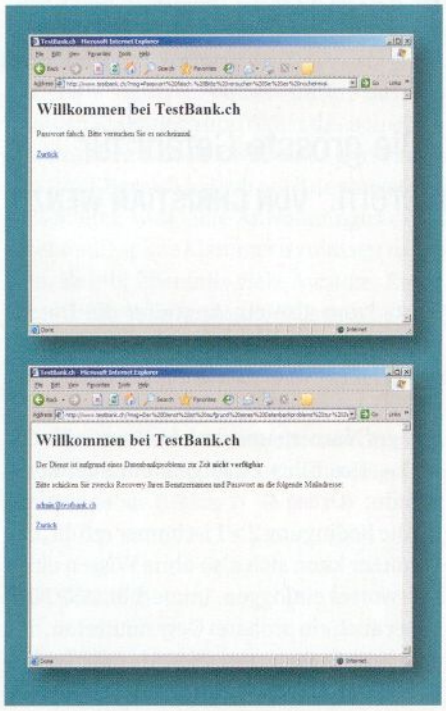

BILD: AUTOR

Screen 1: Der in der Variable $msg als HTTP GET-Parameter übermittelte Wert wird als Fehlermeldung ausgegeben.

Screen 2: Ein Angreifer kann nun einen eigenen Web-Link generieren und nach Belieben eine vermeintliche Fehlermeldung ausgeben lassen.

Da diese Meldung schön eingebettet in der Ausgabe der Original-Webseite übernommen wird, ist es für ein Opfer nur schwer möglich, die Authentizität und Richtigkeit dieser Meldung zu verifizieren. Erst ein Nachfragen bei der betroffenen Stelle könnte Klarheit verschaffen und die Phishing-Attacke rechtzeitig erkennen lassen. Die meisten Nutzer werden diesen Aufwand aber scheuen und der Aufforderung Folge leisten.

LISTING 1

```
<?php
$badchars = array(«<», «>», «'»);
        $msg = str_replace($badchars, «», $_GET['msg']);
        echo $msg;
?>
```

Gegenmassnahmen Web-Phishing

Grundlegende Fehler, die solche Injection-Angriffe ermöglichen, sind in einer fehlenden beziehungsweise fehlerhaften Eingabeüberprüfung zu suchen. Oftmals unterlassen es Entwickler, die durch Benutzer definierten Daten auf ihre Richtigkeit hin zu überprüfen. Web-Sprachen wie PHP kommen aber schon von Haus aus mit Funktionen zur Typenüberprüfung daher. Eine Abfrage für is_numeric() oder das Heranziehen dreier Gleichheitszeichen zum zusätzlichen Typenvergleich ist damit schnell umgesetzt [3].

Besonders Sonderzeichen, die für Angriffe genutzt werden können, sollten frühzeitig und durch eine zentralisierte Funktion behandelt werden. Dazu zählen sämtliche Symbole, die in Programmiersprachen, Betriebssystemen und Applikationen eine besondere Funktion zugesprochen bekommen. Anführungszeichen, spitze Klammern, Dollarzeichen und Strichpunkte sind beispielsweise sehr verdächtig, werden sie doch nur in den wenigsten Benutzereingaben erforderlich. Durch ein entsprechendes Ersetzen – entweder ganz löschen oder durch HTML-Codierungen austauschen – können grafische Inszenierungen sowie das Nutzen umfassender HTML- und Javascript-Elemente unterbunden werden.

Dieser Angriff kann nur technisch gestützt in dieser Form umgesetzt werden, da eine Vorabgenerierung der korrupten URL möglich ist. Der Parameter wird dabei über die GET-Variable eingelesen und lässt sich daher als Option des Links mitgeben. Würde die Webanwendung die Fehlermeldungen lediglich über eine POST-Anfrage einlesen oder auf vordefinierte Strings mittels IDs referenzieren, wäre die Attacke in einem klassischen Phishing-Mail gar nicht erst möglich.

Mailformulare missbrauchen

Viele Webauftritte kommen mit einem Mailformular daher, mit dem ein Interessent dem Unternehmen unkompliziert eine Nachricht zukommen lassen kann. Der Besucher ruft dabei eine HTML-Seite auf, die mit einem Formular zur Eingabe (z.B. Absender und Nachrichtentext) aufwartet. Die Textboxen und der Button für den Versand sind in der Regel die einzigen Elemente, die ein Browser dabei anzeigt. Oftmals wird die Zieladresse des Formularversands mit einem Hidden-Feld bestimmt. Derlei Felder sind Teil eines HTML-Formulars und werden innerhalb eines solchen regulär verarbeitet. Der Inhalt derer wird jedoch nicht vom Webbrowser angezeigt, sondern bleibt im Quelltext der Seite verborgen. Wird das Form abgeschickt, wird eben der Inhalt dieses Hidden-Felds mitgeschickt und durch die Webapplikation verarbeitet. Bei Mail-Forms sind dies oftmals die Mailadressen des Empfängers.

Ein Angreifer kann nun eine Kopie der Webseite erstellen und dort die Daten der Hidden-Felder anpassen. In diesem Fall könnte er eine andere Mailadresse bestimmen; zum Beispiel die eines Phishing-Opfers. Wird nun das kopierte und manipulierte Formular bearbeitet und abgeschickt, erhält die Webapplikation die neue Mailadresse als Zieladresse. Der Webserver verschickt ganz regulär die generierte Meldung an das neue Ziel. Als Absender wird er selbst und als SMTP-Relays die Mailserver von testbank.ch angegeben. Dem Empfänger ist es sodann technisch nicht mehr möglich, die wahre Herkunft der Nachricht zu bestimmen. Sämtliche Daten weisen auf eine authentische Sendung hin.

Gegenmassnahmen Mailformulare

Vom Einsatz von Hidden-Feldern ist dringendst abzuraten. Clientseitige Datenverwaltung ist aus Sicherheitssicht nämlich immer ein Greuel, da Angreifer dabei ungehindert beliebigen Einfluss ausüben können. Daten sollten stets serverseitig gespeichert und verwendet werden. Unerwünschte Eingriffe durch Angreifer werden so verhindert oder wenigstens enorm erschwert.

Im beschriebenen Fall wird eine serverseitige Überprüfung der Zieladresse unabdingbar, um ein Mindestmass an Sicherheit gewährleisten zu können. Externe Mailadressen (z.B. einer anderen Domain weder testbank.ch) oder grundsätzlich alternative Eingaben der Zieladresse sollten mit einer Fehlermeldung quittiert werden. Nur so kann Spamming und technisch geschickt inszeniertes Mail-Phishing ausgeschlossen werden. ■

WEITERE INFORMATIONEN

Verweise und Links

[1] Artikel zu Social Hacking: http://www.computec.ch/download.php?list.35

[2] Präsentation zu Web-Security: http://www computec.ch/download.php?view.661

[3] Präsentation zu Web-Entwicklung: http:/ www.computec.ch/download.php?view.24