Studiengang Informatik
apsi 2020

Fachhochschule Nordwestschweiz
A. Wagner, M. Schläpfer, R. Wagner

Lösungsblatt 6 — 26. Oktober 2020

Aufgabe 6-1: Secure Session Management

(a) HTTP ist ein zustandsloses Protokoll, d.h. aufeinanderfolgende Requests sind völlig zusammenhangslos. Sollen die Requests eines Benutzers also innerhalb eines Kontextes betrachtet werden, muss dies durch ein Session Management entsprechend realisiert werden. Ein Beispiel ist ein Webshop, bei welchem der Benutzer Artikel in einem Warenkorb ablegen kann.

(b) Grundsätzlich muss ein Session Identifier erstellt und mit den Requests an den Server gesendet werden. Dies erfolgt oft mittels Cookies oder einem anderen HTTP Request Header (z.B. Authorization HTTP Header). Alternativ kann dies auch als Parameter (`GET` oder `POST` erfolgen.

(c) Unter Session Hijacking versteht man die Übernahme einer Benutzersession. In der Regel durch Verwendung des entsprechenden Session Identifiers. Eine XSS Schwachstelle könnte zur Ausführung eine Scriptes führen, welches das Session Cookie ausliest und dessen Inhalt an den Angreifer übermittelt. Der Session Identifier muss geschützt werden. Bei Cookies sollte bspw. das HttpOnly Flag gesetzt werden, welches verhindert, dass Scripts auf das Cookie zugreifen können.

(d) Der Session Identifier darf nicht in den Besitz des Angreifers gelangen. Somit muss der Zugriff darauf verhindert, aber gleichzeitig auch die Entropie hoch gehalten werden. Der Angreifer soll also keine Möglichkeit erhalten, den Session Identifier zu erraten.

(e) Der Session Identifier darf unter keinen Umständen erraten werden. Ansonsten kann ihn der Angreifer verwenden. Timestamp und Benutzername sind ableitbar und haben demnach nur eine geringe Entropie. D.h. der Session Identifier kann erraten und die Session so übernommen werden.

Aufgabe 6-2: HTTP Parameter

(a) HTTP stellt für die Übermittlung von Parametern die Methoden `GET` und `POST` zur Verfügung.

(b) `GET` ermöglicht es, Parameter direkt in der URL als *Query String* anzugeben. Dies stellt eine sehr einfache Möglichkeit dar. Allerdings muss man sich drüber im Klaren sein, dass Requests in vielerlei Logfiles gelogged werden. Dabei werden die `GET` Parameter natürlich mitgelogged. Ganz im Gegensatz zu `POST` parametern. Dort wird der *Query String* im HTTP Body übermittelt.

(c) Die Verbindung ist zwar geschützt und ein Angreifer kann die Login Credentials nicht direkt im Netzwerk lesen. Hingegen werden aber die Login Credentials in den Logfiles mitgelogged. Wer Zugriff auf die Logfiles erhält, kann demnach die Login Credentials auslesen. Sie sollten nicht als `GET` Parameter übertragen werden.

Aufgabe 6-3: CGI

Siehe Musterlösung.