



## openHPI – Sicherheit im Internet Angriffe im World Wide Web – Angriffsziel Web-Browser

Prof. Dr. Christoph Meinel  
Hasso-Plattner-Institut, Potsdam

### Einführung: Angriffe im Web

Das World Wide Web (WWW) ist der am meisten genutzte Dienst im Internet und damit das für Cyberkriminelle attraktivste Angriffsziel

#### Das Web wird auf zwei Ebenen angegriffen:

- Über den **Web-Browser** auf der Nutzerseite des Webs
- Über den **Web-Server** auf der Seite der Dienstanbieter im Web

Nutzer interagieren mit dem Web mittels ihres Web-Browsers:

- Anzeige von Webseiten
- Anzeige der Nutzeroberfläche der verschiedenen Dienstanbieter und Interaktion mit den Diensten, z.B.
  - Suchanfragen, Web-Shops, Online-Banking, usw.

Angreifer nutzt **Unachtsamkeit** und **Ahnungslosigkeit** der Nutzer aus und lockt diese z.B. auf Webseiten mit **bösartigem Inhalt**

Angriffe über Web-Browser machen sich dabei zahlreiche **Schwachstellen von Web-Browsern** zunutze, darunter

- Schwachstellen in **Plugins**
- Schwachstellen in **Skript-Interpretern**
- **Informationslecks** in Browsern

## Angriffe über Web-Browser Schwachstellen in Plugins (1/2)

### Java und ActiveX

- Erlauben Einbettung kleiner Anwendungen in Webseiten
- Viele Schwachstellen bekannt, die eingebetteten böartigen Anwendungen Zugriff auf eigentlich geschützte Ressourcen oder Daten auf dem Nutzersystem erlauben
- **Heute:** kaum noch im Einsatz

### PDF-Plugins

- Plugins erlauben Anzeige von PDF-Dateien direkt im Browser
- In den letzten Jahren wurden viele Sicherheitslücken entdeckt
  - Mit Hilfe speziell präparierter PDF-Dokumente wurde entfernte Code-Ausführung möglich
  - In PDF-Dateien eingebettete dynamische JavaScript-Inhalte ermöglichen Ausführung von Schadcode

### Adobe Flash

- Adobe Flash ist **Multimediaplattform**, die für Animationen, Spiele, Vektorgrafiken, Videos oder aufwändige Benutzeroberflächen im Browser genutzt wird
- In den letzten Jahren wurden **viele Sicherheitslücken** entdeckt
  - Flash-Cookies können Privatsphäre des Nutzers verletzen
  - Schwachstellen in Interpretern für Flash-Programmiersprachen, wie z.B. ActionScript, erlauben Ausführung von Schadcode
- **2015**: Zweimal in Folge wurden gravierende Sicherheitslücken aufgedeckt, für die vom Hersteller nicht rechtzeitig Patches bereitgestellt wurden
  - Sicherheitslücken wurden von Angreifern massiv ausgenutzt
  - Browserhersteller rieten zur Deaktivierung von Flash oder schalteten das Plugin selbst per Browser-Update ab

**Skript-Interpreter** ist Komponente des Browsers, die Skripte – kleine Programme – auf Webseiten ausführt

- Mit Skripten können **dynamische Inhalte** in Webseiten bereitgestellt werden, z.B.
  - Prüfung der Vollständigkeit von Formularen
  - Kontrolle über Präsentation und Grafiken in Webseiten
  - ...
- Skripte werden direkt in den **Code der Website (HTML) integriert**
- **Skripte** werden im Kontext der geladenen Webseite **im Browser ausgeführt**
- Verbreitetste Skript-Sprache im Web ist **JavaScript**
  - hat übrigens nichts mit **Java** zu tun ...

### Sicherheitsrisiko JavaScript

#### Grundsätzlich:

- JavaScript hat Zugriff auf sämtliche Elemente einer Webseite
- Kann auch auf Browser-Historie und Cookies zugreifen

#### Schutzmaßnahme: JavaScript läuft in einer Sandbox

- **Sandbox:** Bereich in Betriebssystem und Arbeitsspeicher, der vom Rest des Systems isoliert ist
- JavaScript erhält daher nur sehr begrenzten Zugriff auf Ressourcen des Nutzersystems
  - kann z.B. nicht auf Dateien oder Geräte zugreifen
- **Aber:** Sandbox wurde in Vergangenheit auch schon teilweise ausgehebelt ...

### Weiteres problematisches Angriffsszenarium:

**Einschleusen von Code**, der über Drittinhalte, z.B. über **Werbeeinblendungen** ungewollt im Browser ausgeführt wird

- Angreifer mietet bei Werbeanbieter Flächen zum Einblenden von Werbebannern
- Banner bestehen typisch aus HTML und JavaScript, die der Werbende bereitstellt
- Angreifer platziert Angriffscode in seiner Werbeeinblendung, die beim Betrachten einer (gutartigen) Webseite vom Browser des Opfers geladen und dann ausgeführt wird

## Angriffe über Web-Browser: Schwachstellen in Skript-Interpretern (4/4)

### Weiteres Angriffsszenarium:

Einschleusen von Code, der über Dritthalte, z.B. über **Werbeeinblendungen** ungewollt im Browser ausgeführt wird

**Schutz:** Spezielle Regeln im Browser, die Möglichkeiten zum Einschleusen limitieren

- **Same-Origin-Policy:** Verbietet JavaScript den Zugriff auf Objekte („DOM-Elemente“), die nicht vom gleichen Webserver stammen
  - **Beispiel:** Skript auf Seite <https://open.hpi.de/courses> darf nicht auf Elemente von <https://www.beispiel.de> (anderer Server) oder <http://open.hpi.de> (anderes Protokoll!) zugreifen

## Angriffe über Web-Browser: Informationslecks im Browser

Viele **Browser-Aktivitäten** werden **automatisch gespeichert**, z.B. die Historie, Bookmarks (dt. Lesezeichen), Cache, Cookies, Formulareingaben, usw.

**Grund:** Nutzung des Browsers soll so bequem wie möglich sein, z.B.

- leichteres Wiederauffinden bereits besuchter Webseiten
- bereits einmal eingegebene Daten bei Bedarf ohne erneute Eingabe wieder zur Verfügung stellen, ...

Aber: **Großes Sicherheitsrisiko**

- Daten können von Angreifern abgegriffen werden (JavaScript)
- Angreifer bekommt Einblick in alle Aktivitäten eines Nutzers im Web, z.B. Hobbies, Interessen, Pläne, ...
- Zugriff auf Browser History ermöglicht u.U. genaue Identifikation eines Nutzer im Web („Profiling“)