=

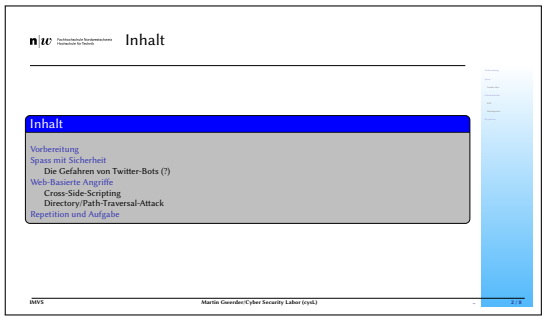


Abbildung 1: Folie “Inhaltsverzeichnis”

Inhaltsverzeichnis

1 Vorbereitung

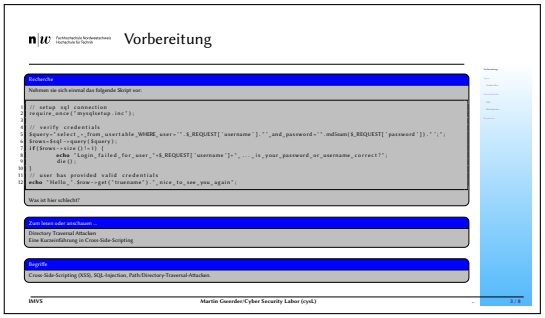


Abbildung 2: Folie “Vorbereitung”

Bitte arbeiten sie jeweils die Vorbereitungs-Slide vor(!) einer Lektion durch. Sie erlauben es damit, dass alle mit einem bestimmten Vorwissen in die Lektion kommen. Dies führt zu interessanteren, abwechslungsreicheren Diskussionen.

2 Spass mit Sicherheit

2.1 Die Gefahren von Twitter-Bots (?)

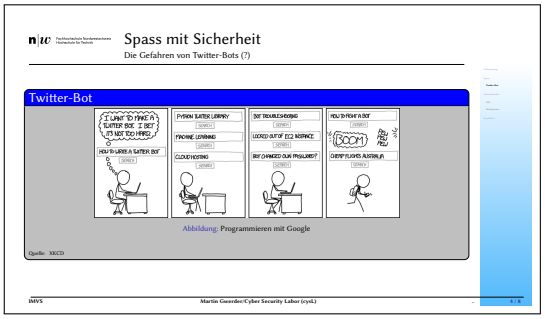


Abbildung 4: Folie “Spass mit Sicherheit”/“Die Gefahren von Twitter-Bots (?)”

Die Gefahr bei kopiertem Code ist häufig nicht im Code sondern in den Weglassungen. Ist ihnen schon einmal aufgefallen wieviele Leute Fragen “Wie kann ich …in …machen?”. Wieviele Leute haben sie angetroffen, die sich Gedanken um die Sicherheit ihres Codes gemacht haben? Oder welche Hintertüren sie mit ihrem Code eröffnen. Speziell im Web-Umfeld werden blindwütig irgendwelche Libraries eingesetzt zum Verwalten von sicherheitskritischen Informationen. Eine Schwäche in einer Library reicht üblicherweise aus um tausende von Websiten zu gefährden.

3 Web-Basierte Angriffe

3.1 Cross-Side-Scripting

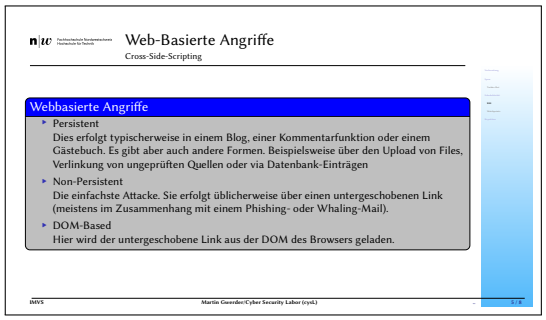


Abbildung 5: Folie “Web-Basierte Angriffe”/“Cross-Side-Scripting”

- Persistent
  Dies erfolgt typischerweise in einem Blog, einer Kommentarfunktion oder einem Gästebuch. Es gibt aber auch andere Formen. Beispielsweise über den Upload von Files, Verlinkung von ungeprüften Quellen oder via Datenbank-Einträgen.
- Non-Persistent
  Die einfachste Attacke. Sie erfolgt üblicherweise über einen untergeschobenen Link (meistens im Zusammenhang mit einem Phishing- oder Whaling-Mail).
- DOM-Based
  Hier wird der untergeschobene Link aus der DOM des Browsers geladen.

3.2 Directory/Path-Traversal-Attack

Abbildung 6: Folie "Web-Basierte Angriffe"/"Directory/Path-Traversal-Attack"

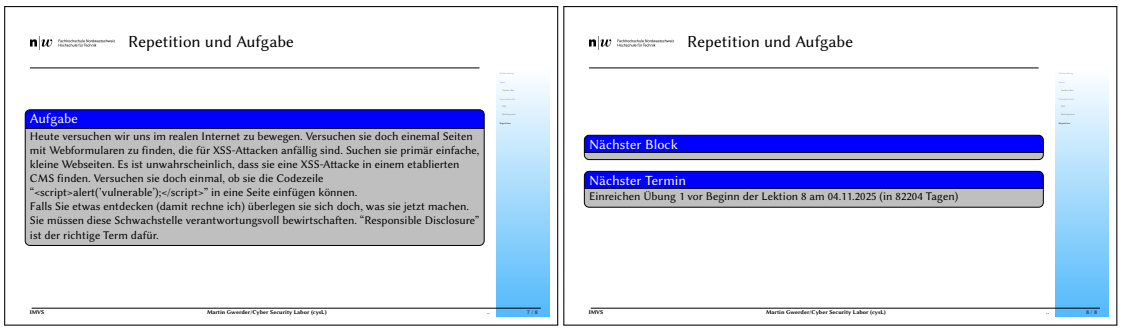
Lokale Files werden ausserhalb des Server-Roots geladen und übermittelt. Dies erfolgt über…

- Backreferencing
  Normalerweise über verwendung des ".."-Hardlinks
- Logische Links
  Diese können auch Files (oder Verzeichnisse) ausserhalb des Roots referenzieren.

Häufig wird diese Attacke mit Obfuscation eingesetzt.

- http://www.foo.bar/index.php?getFile=../../../../etc/passwd
  Eine Traversal-Attacke ohne obfuscation.
- http://www.foo.bar/index.php?getFile=..%2F..%2F..%2F..%2Fetc%2Fpasswd
  Eine Traversal-Attacke mit leichter obfuscation (URL-Encoding von Slashes).
- http://www.foo.bar/%69%6E%64%65%78%2E%70%68%70%3F%67…
  Eine Traversal-Attacke mit hoher obfuscation (URL-Encoding der Dokumente).

4 Repetition und Aufgabe



(a) "Repetition und Aufgabe"/"" (b) "Repetition und Aufgabe"/""

Abbildung 7

Heute versuchen wir uns im realen Internet zu bewegen. Versuchen sie doch einemal Seiten mit Webformularen zu finden, die für XSS-Attacken anfällig sind. Suchen sie primär einfache, kleine Webseiten. Es ist unwahrscheinlich, dass sie eine XSS-Attacke in einem etablierten CMS finden. Versuchen sie doch einmal, ob sie die Codezeile "<script>alert('vulnerable');</script>" in eine Seite einfügen können.

Falls Sie etwas entdecken (damit rechne ich) überlegen sie sich doch, was sie jetzt machen. Sie müssen diese Schwachstelle verantwortungsvoll bewirtschaften. "Responsible Disclosure" ist der richtige Term dafür.