

Freitag, 03.03.2006
IT Security Day
IT Security Day



Webprogrammierung? Aber sicher!

Chris Mair

<http://www.l006.org>



Übersicht



Grundbegriffe:

- World Wide Web
- "statische" Webseiten
- "dynamische" Webseiten


Angriffe:

- Code einschleußen (Code/SQL injection)
- Cross Site Scripting



Wer Wie Was

- 🔒 WWW Client (Browser)
- 🔒 Web Server
- 🔒 Anfrage: GET /index.html
- 🔒 Antwort: HTTP/1.1 200 OK
- 🔒 > "Statische" Webseiten



 

index.html:

Hallo, liebes Publikum :)

Programmierer sind faul!

- 🔒 Das ist auch gut so :)
- 🔒 Antwort hängt von Eingabe Parametern ab!
- 🔒 Anfrage: GET /index.php?user=**chris**
- 🔒 Antwort: HTTP/1.1 200 OK
- 🔒 > "dynamische" Webseiten

index2.html:

```
<a href="index2.php?user=chris">Startseite von Chris</a>  
<br>  
<a href="index2.php?user=peter">Startseite von Peter</a>
```

index2.php:

```
Hallo,  
<?php  
$user = $_GET["user"];  
switch ($user) {  
    case "chris":  
        echo "liebes";  
        break;  
    case "peter":  
        echo "<b>liebstes</b>";  
        break;  
}  
?>  
Publikum.
```

index3.html:

wie index2.html, aber mit index3.php als Linkziel

index3.php:

wie index2.php, aber mit folgenden Zeilen zusätzlich:

```
<br>
Hier Informationen &uuml;ber mich:<br>
<pre>
<?php
system("finger $user"); # VORSICHT!
?>
</pre>
```

Probleme

-  **Mangelhafte Überprüfung der Eingabe-Parameter:**
 - Manipulation der Anwendung selbst
 - Code Injection
 - SQL Injection
-  **Beispiele**

index3.php cracken -> Code Injection:

```
index3.php?user='chris;ls'
```

index4.php:

```
<?php
$login = $_GET["login"];
$conn = pg_connect("host=localhost port=5432 dbname=lbs user=chris password=");
if (!$conn) { echo "ERROR\n"; exit; }
$result = pg_query($conn, "SELECT vorname, nachname FROM benutzer where login='$login'");
if (!$result) { echo "ERROR\n"; exit; }
while ($row = pg_fetch_row($result)) {
    echo "Daten zu login '$login': $row[0] $row[1]<br>";
}
?>
```

index4.php cracken -> SQL Injection:

```
index4.php?login=chris'+union+select+vorname+,+nachname+from+geheim+--
```

Lösungen



- 🔒 Rigorose Überprüfung der Eingabe Parameter:
- Parameter anhand von Liste erlaubter Parameter überprüfen
 - bestimmte Zeichen herausfiltern (engl: "to escape")
 - SQL: "prepared statements"

🔒 Beispiele



Cross Site Scripting



- 🔒 Beispiel ("session hijacking"):
- Forum
 - Benutzer loggen sich ein und erhalten eine geheime Session Nummer zugewiesen
 - Benutzer dürfen Nachrichten hinterlassen
 - ein Angreifer kann in seiner Nachricht Client Code verstecken (etwa Java Script) um die Session Nummer nach aussen zu schicken
 - Ein Opfer, das die Nachricht des Angreifers liest, teilt ihm damit seine Session Nummer mit
 - der Angreifer kann sich als Opfer anmelden.

