Online-Testat zu D: Web Security

> Beachtet bitte die Allgemeinen Hinweise zu den Online-Testaten.
>
> **Spezielle Hinweise:**
>
> - Vergesst nicht, die Abschlussaufgabe zu Gruyere vor dem Testat zu bearbeiten. Die Dokumentation der Abschlussaufgabe muss in der letzten Aufgabe des Testats hochgeladen werden.

Leistungsübersicht

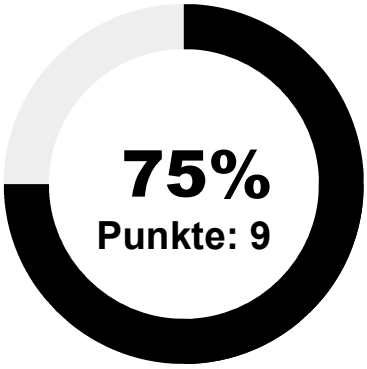
Maximale Anzahl Lösungsversuche	1
Anzahl gemachter Versuche	1
Erreichte Punktzahl	9
Status	**Bestanden**

Resultate

Kurs	**Praktikum IT-Sicherheit SS21 (080042)**
Test	**IT-Sicherheit SS 20 - Versuch Web Security - Online-Testat**

Dies sind Ihre Testresultate

Dauer	1h 29m 49s
Beantwortet	16 von 16 Fragen (100%)
Erreichte Punktzahl	9 von 12 Punkten (75%)
Benötigte Punktzahl	9.0



Kleingruppe 1

Zur Sektion springen ›

1.1 Angriffstypen 1

Zur Sektion springen ›

1.2 Konzepte 1

Zur Sektion springen ›

1.3 Cookies 1

Zur Sektion springen ›

2.1 Cross-Site Scripting 2

Zur Sektion springen ›

2.2 Weitere Angriffe 2

Zur Sektion springen ›

3.1 Daten in Gruyere 1

Zur Sektion springen

3.2 Angriffe auf Gruyere 1

Zur Sektion springen

Elevation of Privilege 1

Zur Sektion springen

3.3 Schwachstellen in Gruyere 1

Zur Sektion springen

Teil 4: Abschlussaufgabe 2

Zur Sektion springen

Feedback 2

Zur Sektion springen

Kleingruppe 0 von 0 Punkten (0%)

Gruppenmitglieder

Status	Beantwortet		
Erreichte Punktzahl	0 / 0		0%

Antwort

Dieses Testat kann pro Kleingruppe nur einmal abgelegt werden. Bitte gebt dazu hier die Namen und Matrikelnummern aller Gruppenmitglieder an.

```
Lara Quitte, 1031494; Tino Jeromin, 1151903;
```

6 Wörter

Zurück zur Übersicht

Teil 1: Begriffe und Konzepte

Zurück zur Übersicht

1.1 Angriffstypen 0 von 1 Punkten (0%)

Cross-Site Script Inclusion

Status	Beantwortet		
Erreichte Punktzahl	0 / 1		0%

Antwort

Wodurch zeichnet sich **Cross-Site Script Inclusion** (XSSI) aus Sicht des Angreifers aus?

- (●) Eigener Code wird in fremde Webseiten eingebunden
- ( ) Aktionen des Opfers werden im Namen des Angreifers ausgeführt
- ( ) Fremder Code wird in eine eigene Webseite eingebunden
- ( ) Eigene Aktionen werden im Namen des Opfers ausgeführt

Zurück zur Übersicht

1.2 Konzepte 1 von 1 Punkten (100%)

Hidden Input Fields

Status	Beantwortet		
Erreichte Punktzahl	1 / 1	[redacted]	100%

Antwort

Die Verwendung von POST-Requests mit versteckten Feldern (*hidden input fields*) bietet einen signifikanten Schutz gegen Angriffe, die versuchen, die Request-Parameter auszuspähen.

- ◉ falsch
- ○ wahr

Zurück zur Übersicht

1.3 Cookies 1 von 1 Punkten (100%)

Cookies

Status	Beantwortet		
Erreichte Punktzahl	1 / 1	[redacted]	100%

Antwort

Was hat es zur Folge, wenn eine Webanwendung das "Secure"-Flag eines Cookies setzt?

- ○ Das Cookie läuft nach einmaliger Verwendung ab.
- ○ Die Schlüssel-Werte-Paare des Cookies werden vom Server verschlüsselt.
- ○ Das Cookie kann clientseitig nicht mittels JavaScript gelesen werden.
- ◉ Das Cookie kann vom Client nur über eine HTTPS-Verbindung geschickt werden.

Zurück zur Übersicht

Teil 2: Fragen zu Angriffen

Zurück zur Übersicht

2.1 Cross-Site Scripting 1 von 2 Punkten (50%)

Datendiebstahl mittels XSS

Status	Beantwortet		
Erreichte Punktzahl	0 / 1		0%

Antwort

Angenommen, ein Angreifer kann eine XSS-Schwäche einer Website ausnutzen, um per JavaScript die Startseite zu manipulieren, die unter anderem das folgende Login-Formular enthält:

```
<form method="POST" action="login">
  <input type="text" name="username">
  <input type="password" name="password">
  <input type="submit" value="Login">
</form>
```

Wie könnte der Angreifer die Login-Daten eines Benutzers stehlen, ohne dass dieser es merkt?

```
Der Angreifer kann den opfern eine Phishing-Email schicken, in der er die
Url der Website so verändert, dass der username und Passwort an eine Email
vom Angreifer geschickt werden. Über die veränderte Url kann so Javascript
mit in die Website eingebaut werden, die dieses Funktion ausführt.
```

50 Wörter (max. {1})

Reflected XSS

Status	Beantwortet		
Erreichte Punktzahl	1 / 1	[redacted]	100%

Antwort

In einer Webanwendung auf `www.database.com` werden Fehlermeldungen angezeigt, wenn der Benutzer eine unbekannte Seite aufruft, z. B. führt der Aufruf von `https://www.database.com/animal/unicorn` zu der Fehlermeldung:

> **Sorry, but animal/unicorn does not exist!**

Dazu wird serverseitig das folgende Template verwendet (`path` ist der Pfad aus der URL):

```
<html>
  <head>
    <title>Error</title>
  </head>
  <body>
    <div class='errormessage'>Sorry, but {{path}} does not exist!</div>
  </body>
</html>
```

Für welche URL wird beim Aufruf im Browser des Benutzers eine JavaScript-Funktion `doEvilStuff()` aufgerufen?

```
https://www.database.com/<script>doEvilStuff()</script>
```

7 Wörter (max. {1})

Zurück zur Übersicht

2.2 Weitere Angriffe 2 von 2 Punkten (100%)

☑ Cross-Site Request Forgery

Status	Beantwortet		
Erreichte Punktzahl	1 / 1	[redacted]	100%

Antwort

Welche Voraussetzungen müssen gegeben sein, damit ein **XSRF-Angriff** auf eine Website durchgeführt werden kann?

Achtung: Gegebenfalls sind mehrere Antworten richtig.

- ☐ Die Website verwendet einen GET-Request für die Transaktion, die der Angreifer auslösen will.
- ☐ Der Browser des Opfers lässt die Ausführung von JavaScript zu.
- ☐ Die Website ist anfällig für XSS-Angriffe.
- ☑ Das Opfer ist bei der Website eingeloggt.

Client-State Manipulation

Status	Beantwortet		
Erreichte Punktzahl	1 / 1	[redacted]	100%

Antwort

Auf einer Shop-Website befindet sich unter der Adresse `http://www.shop.de/order` das folgende Formular, über das man eine beliebige Anzahl von Artikeln mit einem Stückpreis von 199,- Euro kaufen kann:

```
<form method="POST" action="submit_order">
  <input type="hidden" name="price" value="199">
  <input type="text" name="quantity" value="1">
  <input type="submit" value="Buy">
</form>
```

Wie kann ein Angreifer 10 Artikel zu einem Gesamtpreis von 10,- Euro kaufen?

```
Er kann den POST-Request, der zu http://www.shop.de/order geschickt wird
so manipulieren, dass price 1 ist und quantity gleich 10. Dies kann er
direkt im Browser über die Entwicklertools machen oder über andere
programme wie curl. Dabei muss der Cookie des eingeloggten Users
mitgesendet werden.
```

48 Wörter (max. {1})

Zurück zur Übersicht

Teil 3: Fragen zu Gruyere

Zurück zur Übersicht

3.1 Daten in Gruyere 1 von 1 Punkten (100%)

··· Data Exploit

Status	Beantwortet		
Erreichte Punktzahl	1 / 1		100%

Antwort

Wie lautet das geheime Snippet des Benutzers "Brie" in der Gruyere-Webanwendung?

I use the same password for all my accounts.

Zurück zur Übersicht

3.2 Angriffe auf Gruyere 0 von 1 Punkten (0%)

XSS über eingebettete Grafiken

Status	Beantwortet		
Erreichte Punktzahl	0 / 1		0%

Antwort

Über das Feld "Icon" kann der Benutzer die URL einer Grafik wählen, die neben seinen Snippets angezeigt wird.

Welche Benutzereingabe in dieses Feld sorgt dafür, dass beim Anzeigen von Seiten, die diese Grafik enthalten, eine JavaScript-Funktion `doEvilStuff()` ausgeführt wird?

```
url+<script>doEvilStuff()</script>
```

4 Wörter (max. {1})

Zurück zur Übersicht

Elevation of Privilege 1 von 1 Punkten (100%)

Elevation of Privilege

Status	Beantwortet		
Erreichte Punktzahl	1 / 1	[redacted]	100%

Antwort

Warum kann ein Angreifer nicht einfach in seinem GRUYERE-Cookie den Wert `admin` einfügen, um Administratorrechte zu erhalten?

```
Weil der Cookie beim Registrieren gesetzt wird und später dieser abgefragt
wird.
Wenn man angemeldet ist, kann der Cookie nicht geändert werden.
```

24 Wörter (max. {1})

Zurück zur Übersicht

3.3 Schwachstellen in Gruyere 2 von 2 Punkten (100%)

Benutzernamen

Status	Beantwortet		
Erreichte Punktzahl	2 / 2	[redacted]	100%

Antwort

Eine Reihe von Schwachstellen in Gruyere wird durch bestimmte Zeichen im Benutzernamen möglich.

In welcher Zeile in `gruyere.py` befindet sich die entscheidende Stelle, an der man Code ergänzen oder anpassen müsste, um zu verhindern, dass Accounts mit problematischen Benutzernamen angelegt werden?

Wie könnte diese Codestelle korrigiert werden?

```
Zeile 418 und folgende.
Die Variable uid müsste überprüft werden, ob sie einen gültigen Namen
entspricht. Also keine Sonderzeichen, etc.
```

21 Wörter (max. {1})

Zurück zur Übersicht

Teil 4: Abschlussaufgabe 0 von 0 Punkten (0%)

Aufgabenstellung

Status	Beantwortet		
Erreichte Punktzahl	0 / 0		0%

Antwort

Welche Aufgabenstellung wurde bearbeitet?

- ○ Aufgabenstellung 1: Stehlen von Cookies
- ◉ Aufgabenstellung 2: Phishing
- ○ Aufgabenstellung 3: Ausspähen von Daten
- ○ Aufgabenstellung 4: Cross-Site Script Inclusion
- ○ Aufgabenstellung 5: Account Hacking
- ○ Alle Aufgabenstellungen (Expertengruppe)

Dokumentation und Code

Status	Beantwortet		
Erreichte Punktzahl	0 / 0		0%

Kommentar / Beurteilung

Gut.

Antwort

Hier soll die Dokumentation der Lösung zur Abschlussaufgabe inkl. Quellcode und Screenshots hochgeladen werden.

Wenn mehrere Dateien hochgeladen werden sollen, können diese in einem ZIP-Archiv o. ä. zusammengepackt werden und dieses hier hochgeladen werden.

Hinweis: Diese Aufgabe ist notwendig zum Bestehen des Testats!

D_Abschlussaufgabe2_20210924T114217.pdf

Zurück zur Übersicht

Feedback 0 von 0 Punkten (0%)

Feedback

Status	Beantwortet		
Erreichte Punktzahl	0 / 0		0%

Antwort

Bitte gebt uns hier Feedback zu diesem Versuch und dem Online-Testat. Was gefiel gut, was ist verbesserungswürdig?

Die direkte Ausführung auf der gruyere WEbsite war hilfreich, die Angriffstechniken zu verstehen.

14 Wörter

Arbeitsaufwand

Status	Beantwortet		
Erreichte Punktzahl	0 / 0		0%

Antwort

Wie hoch war der Arbeitsaufwand für diesen Versuch (einschließlich Online-Testat) in Stunden?

24.0

Es stehen Ihnen keine weiteren Versuche zur Verfügung.