Thema

PHP-Sicherheits-Training-System

vorgelegt von Timo Pagel

Hamburg, den 29.04.2014

Agenda

Agenda

1. PHP-Sicherheit
   - Motivation
   - OWASP Top 10
   - Demonstration einer Lerneinheit
2. Erstellung
3. Evaluation
4. Fazit

Motivation

- 70% aller Sicherheitslücken: Anwendungsschicht [Gartner Group]
- PHP hat als Programmiersprache einen Anteil von 81,2%
- Maßnahmen: [SANS Institute]
  - Security Tools (WAF, Penetration Testing)
  - Dokumentation für Fehlerbehandlung
  - System-Härtung
  - Schulungen für Entwickler
  - ...

OWASP Top 10

Ref.	Bezeichnung	Arbeit	System
A1	Injection	✓	✓
A2	Fehler in Authentisierung und Session-Management	✓	✓

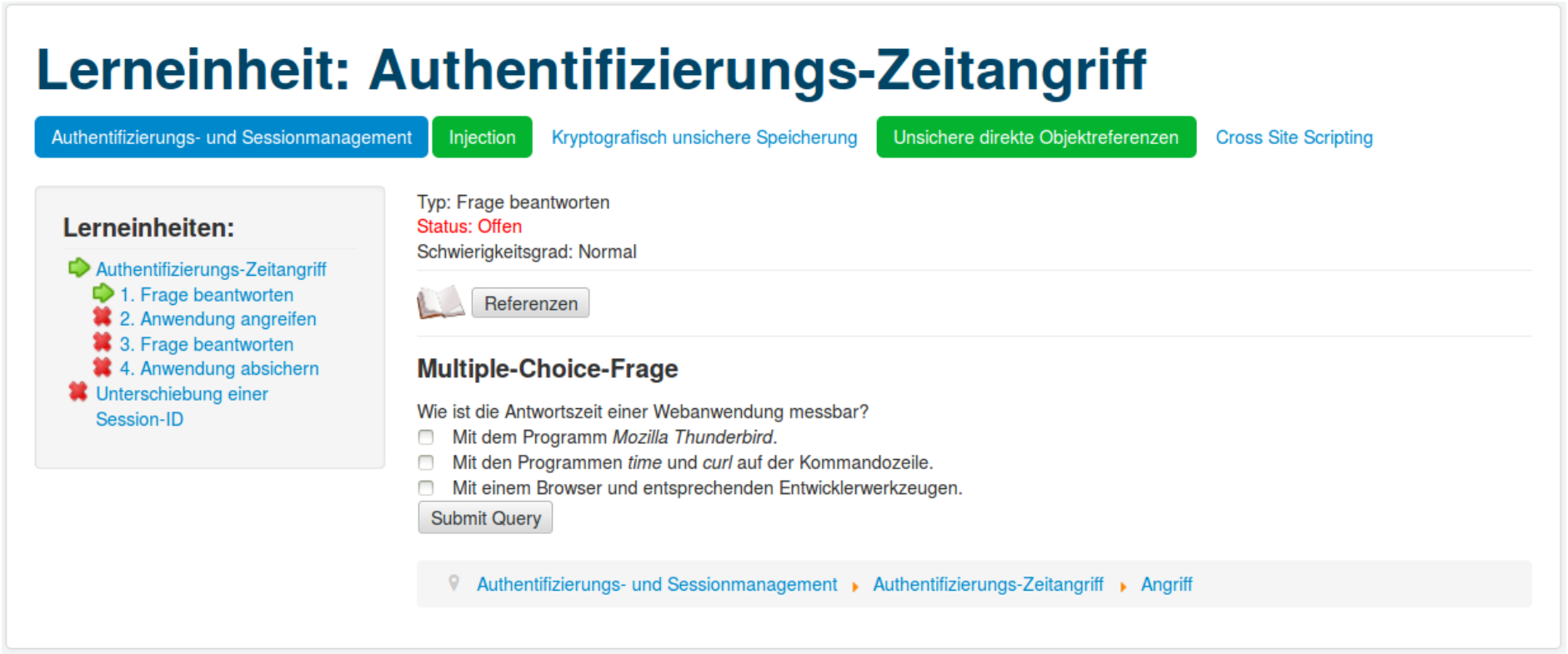
OWASP Top 10

Ref.	Bezeichnung	Arbeit	System
A1	Injection	✓	✓
A2	Fehler in Authentisierung und Session-Management	✓	✓
A3	Cross-Site Scripting (XSS)	✓	✓
A4	Unsichere direkte Objektreferenzen	✓	✓
A5	Sicherheitsrelevante Fehlkonfiguration	✓	✗
A6	Verlust der Vertraulichkeit sensibler Daten	✓	✓

OWASP Top 10

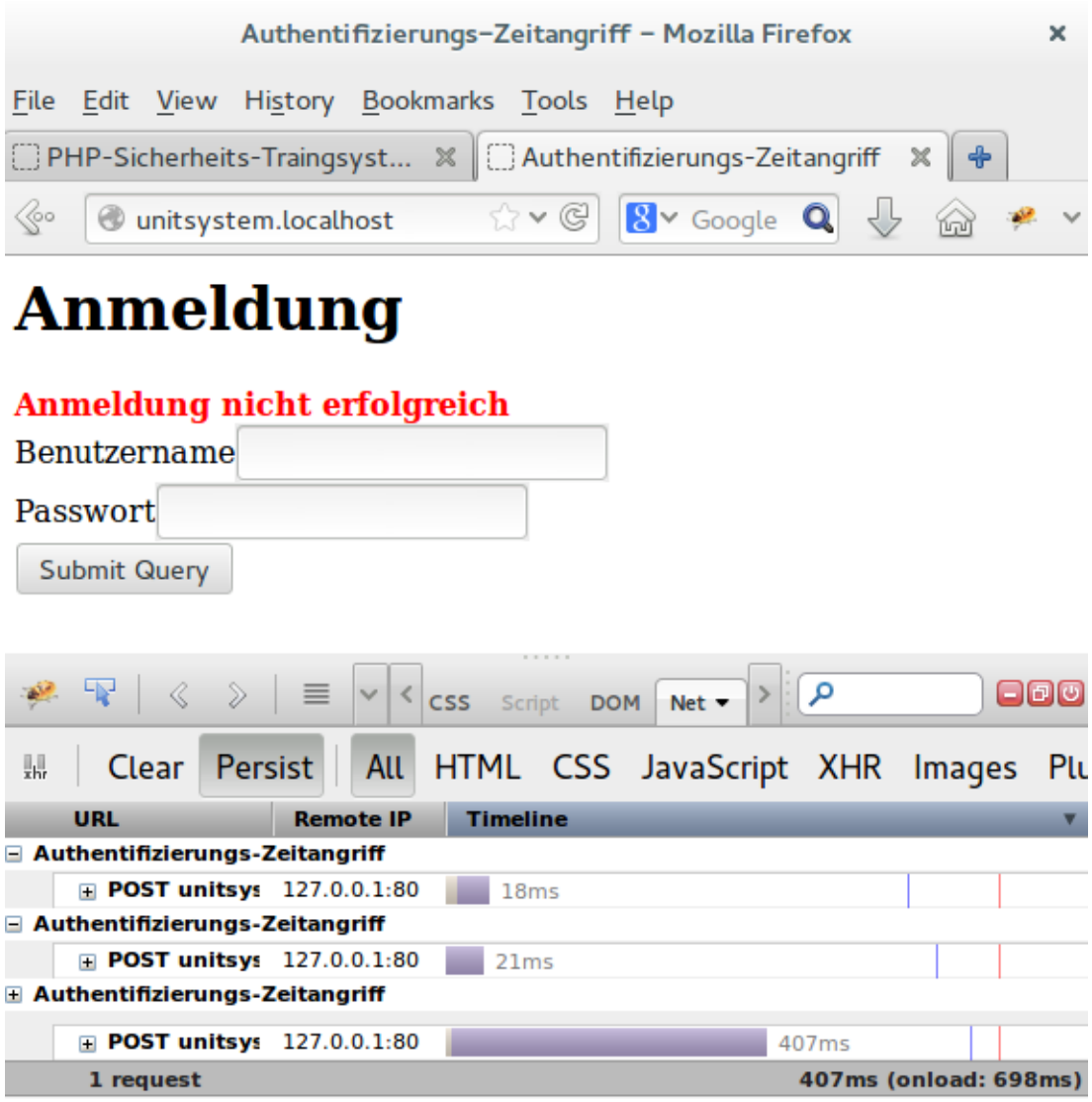
Ref.	Bezeichnung	Arbeit	System
A1	Injection	✓	✓
A2	Fehler in Authentisierung und Session-Management	✓	✓
A3	Cross-Site Scripting (XSS)	✓	✓
A4	Unsichere direkte Objektreferenzen	✓	✓
A5	Sicherheitsrelevante Fehlkonfiguration	✓	✗
A6	Verlust der Vertraulichkeit sensibler Daten	✓	✓
A7	Fehlerhafte Autorisierung auf Anwendungsebene	✓	✓
A8	Cross-Site Request Forgery (CSRF)	✓	✗
A9	Benutzen von Komponenten mit bekannten Schwachstellen	✓	✗
A10	Ungeprüfte Um- und Weiterleitungen	✗	✗

Screenshot Aufgabe 1; Einführung

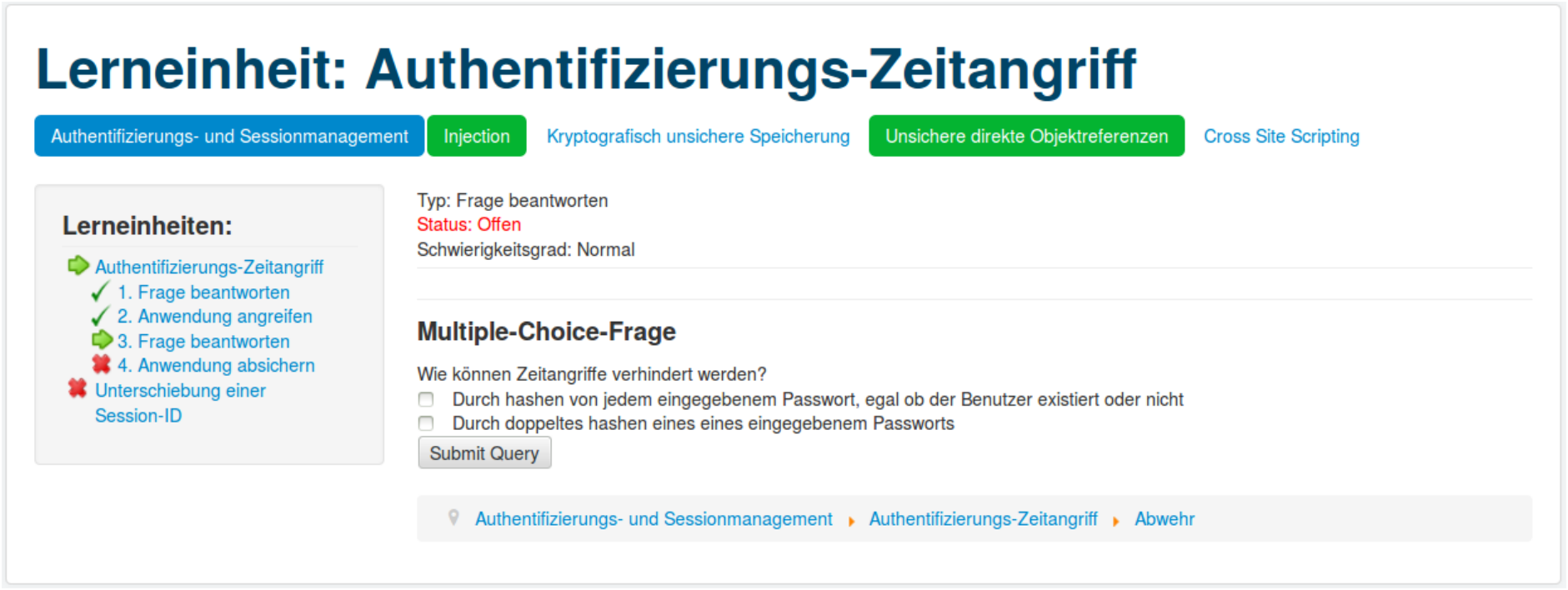


Screenshot Aufgabe 2; Angriff im Führungssystem

Screenshot Aufgabe 2; Angriff durchführen



Screenshot Aufgabe 3; Absicherungsmöglichkeiten



Screenshot Aufgabe 4; Abwehr - Aufgabe

Lerneinheit: Authentifizierungs-Zeitangriff

Authentifizierungs- und Sessionmanagement | Injection | Kryptografisch unsichere Speicherung | Unsichere direkte Objektreferenzen | Cross Site Scripting

Lerneinheiten:

- Authentifizierungs-Zeitangriff
  - ✓ 1. Frage beantworten
  - ✓ 2. Anwendung angreifen
  - ✓ 3. Frage beantworten
  - 4. Anwendung absichern
- ✗ Unterschiebung einer Session-ID

Typ: Anwendung absichern
Status: Offen
Schwierigkeitsgrad: Normal

Hinweis | Lösungsquelltext

Auftragsbeschreibung

Sichere den Quelltext gegen Zeit-Angriffe ab.

Abzusichernder Quellcode:

```
1  <h1>Anmeldung</h1>
2  <?php
3  $isUserValid = false;
4  $validUser = file_get_contents("user.txt");
5  $passwordHash = '$2y$12$c2VjdXJlU2FsdHNTaG91b.NnkRM1xxMm6Ml0vt7BrEHtoODukVNSe';
6  if (isset($_POST['username']) && isset($_POST['password']) && $_POST['username'] == $validUs
7      $isPasswordValid = password_verify($_POST['password'], $passwordHash);
8
9      if ($isPasswordValid) {
10         $isUserValid = true;
11     } else {
12         $isUserValid = false;
13     }
14 }
15 if ($isUserValid) {
```

Screenshot Aufgabe 4; Abwehr - Lösungsweg

Lerneinheit: Authentifizierungs-Zeitangriff

Authentifizierungs- und Sessionmanagement | Injection | Kryptografisch unsichere Speicherung | Unsichere direkte Objektreferenzen | Cross Site Scripting

Lerneinheiten:

- Authentifizierungs-Zeitangriff
  - ✓ 1. Frage beantworten
  - ✓ 2. Anwendung angreifen
  - ✓ 3. Frage beantworten
  - 4. Anwendung absichern
- ✗ Unterschiebung einer Session-ID

Typ: Anwendung absichern
Status: Offen
Schwierigkeitsgrad: Normal

Hinweis | Lösungsquelltext

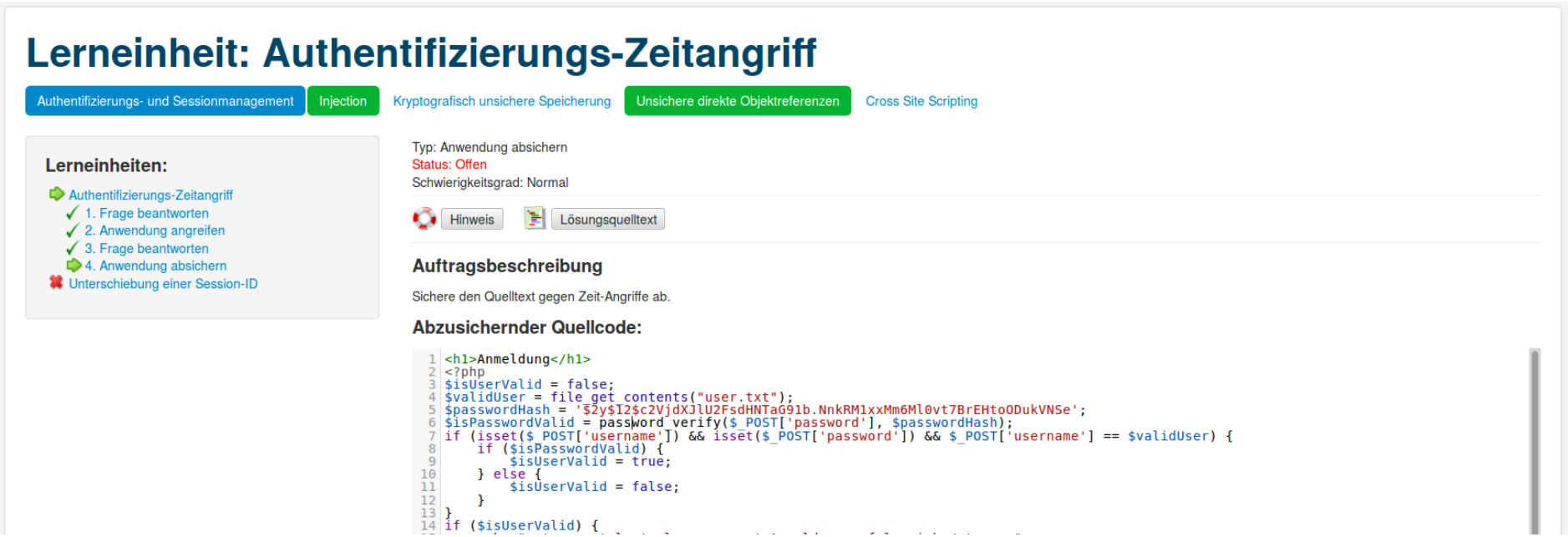
Auftragsbeschreibung

Sichere den Quelltext gegen Zeit-Angriffe ab.

Abzusichernder Quellcode:

```
1  <h1>Anmeldung</h1>
2  <?php
3  $isUserValid = false;
4  $validUser = file_get_contents("user.txt");
5  $passwordHash = '$2y$12$c2VjdXJlU2FsdHNTaG91b.NnkRM1xxMm6Ml0vt7BrEHtoODukVNSe';
6  if (isset($_POST['username']) && isset($_POST['password']) && $_POST['username'] == $validUs
7      $isPasswordValid = password_verify($_POST['password'], $passwordHash);
8
9      if ($isPasswordValid) {
10         $isUserValid = true;
11     } else {
12         $isUserValid = false;
13     }
14 }
15 if ($isUserValid) {
```

Screenshot Aufgabe 4; Abwehr - Lösung



Lerneinheit: Authentifizierungs-Zeitangriff

Authentifizierungs- und Sessionmanagement | Injection | Kryptografisch unsichere Speicherung | Unsichere direkte Objektreferenzen | Cross Site Scripting

Lerneinheiten:

- Authentifizierungs-Zeitangriff
  - ✓ 1. Frage beantworten
  - ✓ 2. Anwendung angreifen
  - ✓ 3. Frage beantworten
  - 4. Anwendung absichern
- ✖ Unterschiebung einer Session-ID

Typ: Anwendung absichern
Status: Offen
Schwierigkeitsgrad: Normal

Hinweis | Lösungsquelltext

Auftragsbeschreibung

Sichere den Quelltext gegen Zeit-Angriffe ab.

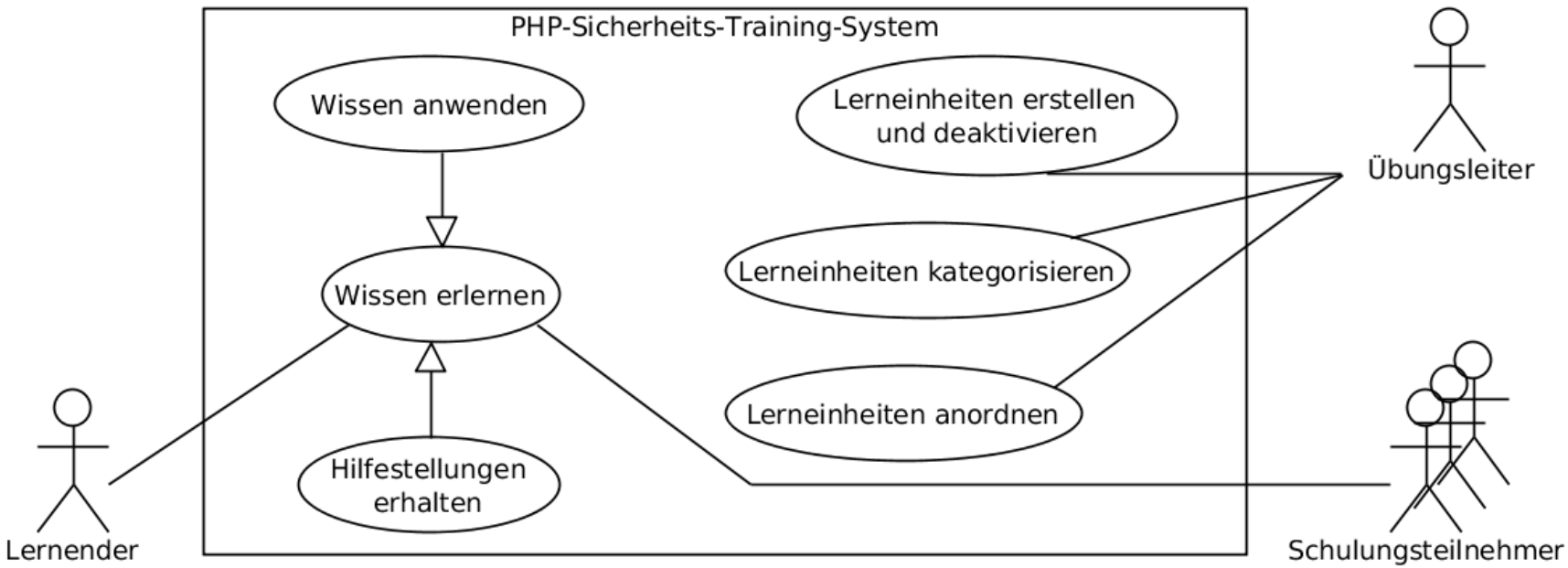
Abzusichernder Quellcode:

```
1  <h1>Anmeldung</h1>
2  <?php
3  $isUserValid = false;
4  $validUser = file_get_contents("user.txt");
5  $passwordHash = '$2y$12$c2VjdXJlU2FsdHNTaG91b.NnkRM1xxMm6Ml0vt7BrEHtoODukVNSe';
6  $isPasswordValid = password_verify($_POST['password'], $passwordHash);
7  if (isset($_POST['username']) && isset($_POST['password']) && $_POST['username'] == $validUser) {
8      if ($isPasswordValid) {
9          $isUserValid = true;
10     } else {
11         $isUserValid = false;
12     }
13 }
14 if ($isUserValid) {
```

Agenda

1. PHP-Sicherheit
2. Erstellung
   - Analyse
   - Konzept
   - Durchführung der Quellcodeüberprüfung
3. Evaluation
4. Fazit

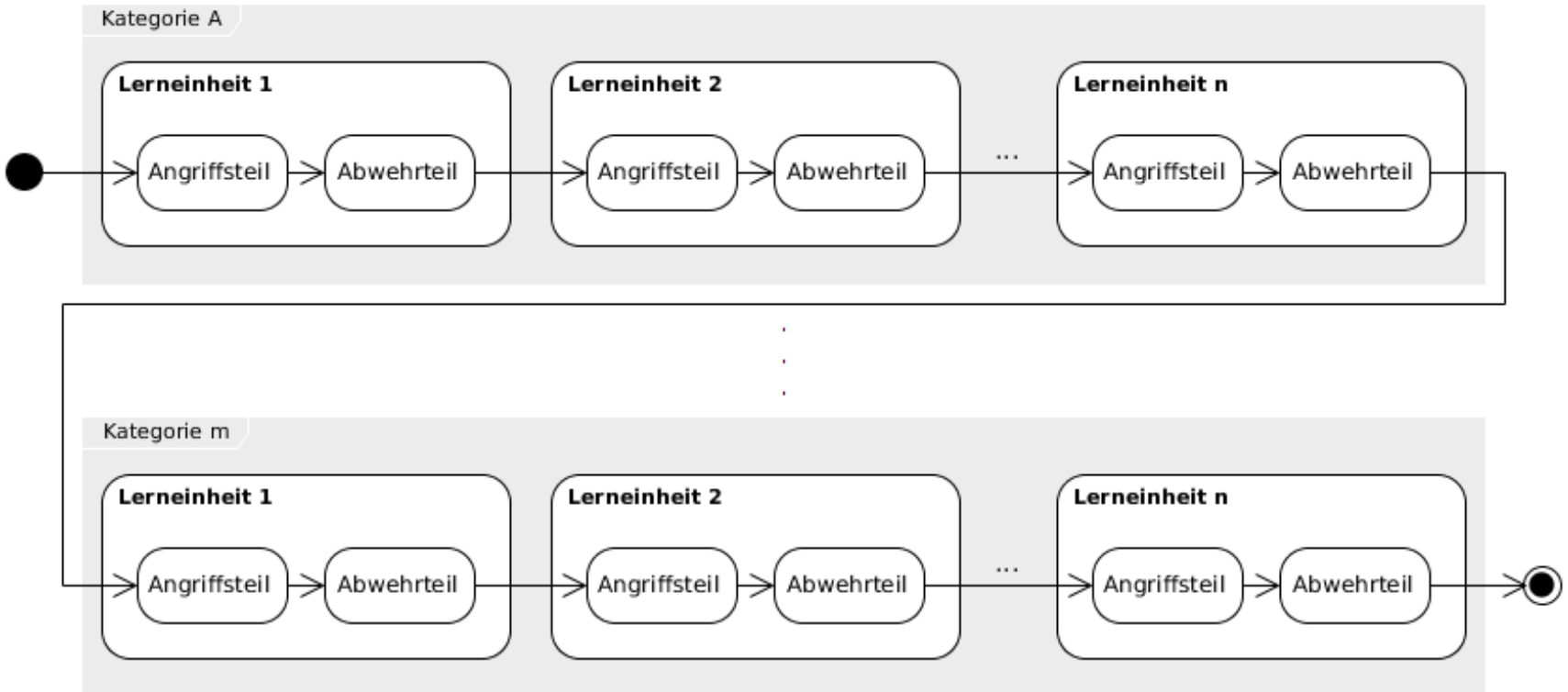
Anwendungsfall



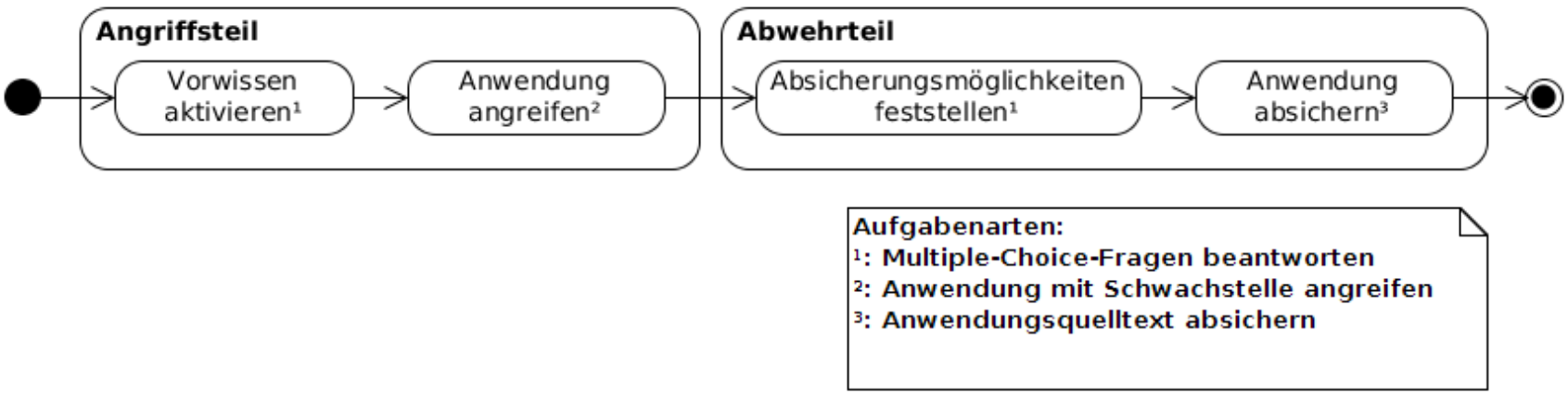
Anforderungen

- Sprache für Quellcode auf Englisch
- Beachtung lernunterstützender Maßnahmen diverser Mediendidaktiker
- Oberfläche entspricht der DIN EN ISO 9241 (Ergonomie der Mensch-System-Interaktion) Teil 110 (Grundsätze der Dialoggestaltung)

Aufbau – Kategorien



Aufbau einer Lerneinheit – Angriffs- und Abwehrteil



Komponenten

- Trainingsumgebung:
  - Ubuntu 13.10 als LiveDVD

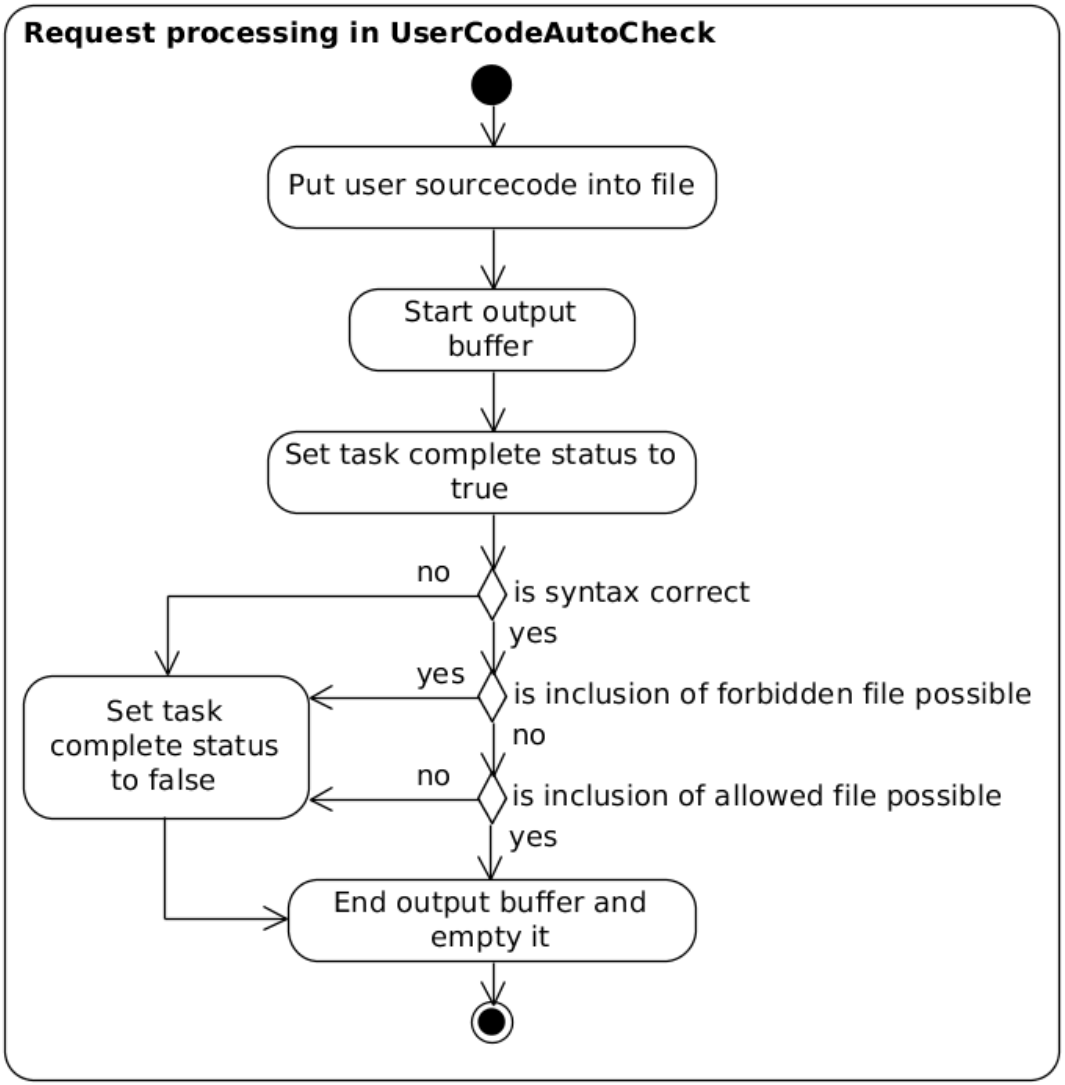
Komponenten

- Trainingsumgebung:
  - Ubuntu 13.10 als LiveDVD
- Führungssystem:
  - Aufgabe: Navigation (http://guidesystem.localhost)

Komponenten

- Trainingsumgebung:
  - Ubuntu 13.10 als LiveDVD
- Führungssystem:
  - Aufgabe: Navigation (http://guidesystem.localhost)
- Einheitensystem:
  - Aufgabe: Verwundbare Anwendung (http://unitsystem.localhost)
  - Beinhaltet: Lerneinheiten

Ablauf der dynamischen Quellcodeüberprüfung



Agenda

1. PHP-Sicherheit
2. Erstellung
3. Evaluation
   - Übersicht
   - Ergebnisse und Diskussion
4. Fazit

Evaluation

- Durchführung im Rahmen der IDW der Fachhochschule Kiel
- Ablauf:
  - Ausfüllen vom Pre-Umfragebogen
  - Absolvierung der Lerneinheiten
  - Ausfüllen vom Post-Umfragebogen
- Ausgefüllte Pre-Umfragebögen: 12
- Ausgefüllte Post-Umfragebögen: 9

Ergebnisse und Diskussion I

Kategorie	**Unterkategorie**	**Richtige Antwort**	
		Pre	**Post**
Eingabe-bereinigung	Fkt. `addslashes()`	6 (50%)	7 (78%)
	Fkt. `htmlspecialchars()`	12 (100%)	9 (100%)
	Fkt. `mysql_real_escape_string()`	2 (17%)	7 (78%)
Sichere Hashfunktion		1 (8%)	8 (89%)

Ergebnisse und Diskussion II

- Offener Teil: Wunsch nach Zurücksetzung von Quellcode
- Quellcode ist bei der Bearbeitung unterstützend: 1,33 im Mittel
- Die Lerneinheiten beinhalten praxistaugliches Wissen: 1,56 im Mittel
- Die Navigation ist strukturiert aufgebaut: 2,56 im Mittel

Ergebnisse und Diskussion II

- Offener Teil: Wunsch nach Zurücksetzung von Quellcode
- Quellcode ist bei der Bearbeitung unterstützend: 1,33 im Mittel
- Die Lerneinheiten beinhalten praxistaugliches Wissen: 1,56 im Mittel
- Die Navigation ist strukturiert aufgebaut: 2,56 im Mittel

Agenda

1. PHP-Sicherheit
2. Erstellung
3. Evaluation
4. Fazit
   - Allgemein
   - Links

Allgemein

- Erstellung LiveDVD
- Absicherung im Quellcode
  - Statische Quellcodeüberprüfung unterliegt Einschränkungen
  - Dynamische Quellcodeüberprüfung ist nicht trivial

Links

- PHP-Sicherheit: http://files.timo-pagel.de/studium/php-sicherheit.pdf
- System (Code): https://bitbucket.org/tpagel/php-security-training-system

Vielen Dank für Ihre Aufmerksamkeit!