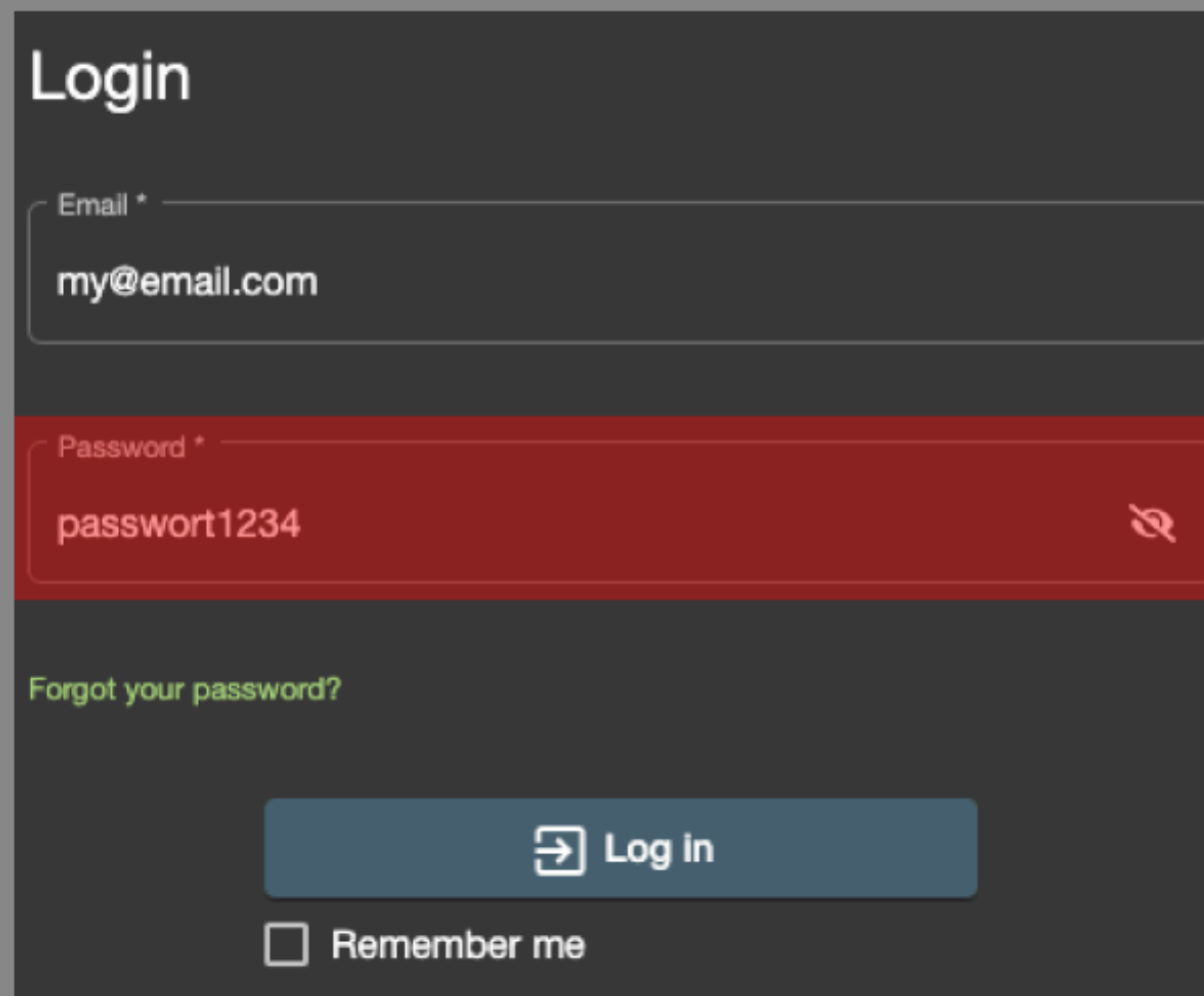


# Web Security

Authentication Fehler  
und wie man sie verhindert



**Login**

Email \*  
my@email.com

Password \*  
passwort1234

[Forgot your password?](#)

☐ Remember me

# Gliederung

---

## **Teil 1: Einführung (20 Minuten)**

- Ziel des WS
- OWASP top 10
- Was ist Authentication?
- Authentication Fehler und Lösungen
- OAuth
- Nicht behandelt

## **Teil 2: HandsOn: Kontext, Requisiten (10 Minuten)**

- Repo und der juice-shop
- Burp suite Community Edition

## **Teil 3: HandsOn: Aufgaben (70 Minuten + 10 Minuten Pause)**

- Aufgabe 1 (30 Minuten)
- Pause (10 Minuten)
- Aufgabe 2 (20 Minuten)
- Aufgabe 3 (20 Minuten)

## **Teil 4: Schluss, Feedback (10 Minuten)**

## Teil 1: Einführung

# Gliederung

---

## **Teil 1: Einführung (20 Minuten)**

- Ziel des WS
- OWASP top 10
- Was ist Authentication?
- Authentication Fehler und Lösungen
- OAuth
- Nicht behandelt

## **Teil 2: HandsOn: Kontext, Requisiten (10 Minuten)**

- Repo und der juice-shop
- Burp suite Community Edition

## **Teil 3: HandsOn: Aufgaben (70 Minuten + 10 Minuten Pause)**

- Aufgabe 1 (30 Minuten)
- Pause (10 Minuten)
- Aufgabe 2 (20 Minuten)
- Aufgabe 3 (20 Minuten)

## **Teil 4: Schluss, Feedback (10 Minuten)**

## Wie sind meine Erfahrungen?

- Noch kein Experte
- Viele Themen, selbst in diesem einen übergeordneten Thema der Authentication
- SPV bei Prof. Dr. Karsch

# Ziel des Workshops

---

## **Verständnis von ein paar ausgewählten Authentication Fehlern**

- Schlechte Passwörter
- Schlechte Methoden für Logins und das “Vergessen” von Passwörtern
- Einfache Möglichkeiten der Fehlervermeidung

## **Ein wenig “hacking” Verständnis**

- Accounts übernehmen
- OSINT -> öffentlich zugängliche Daten sammeln

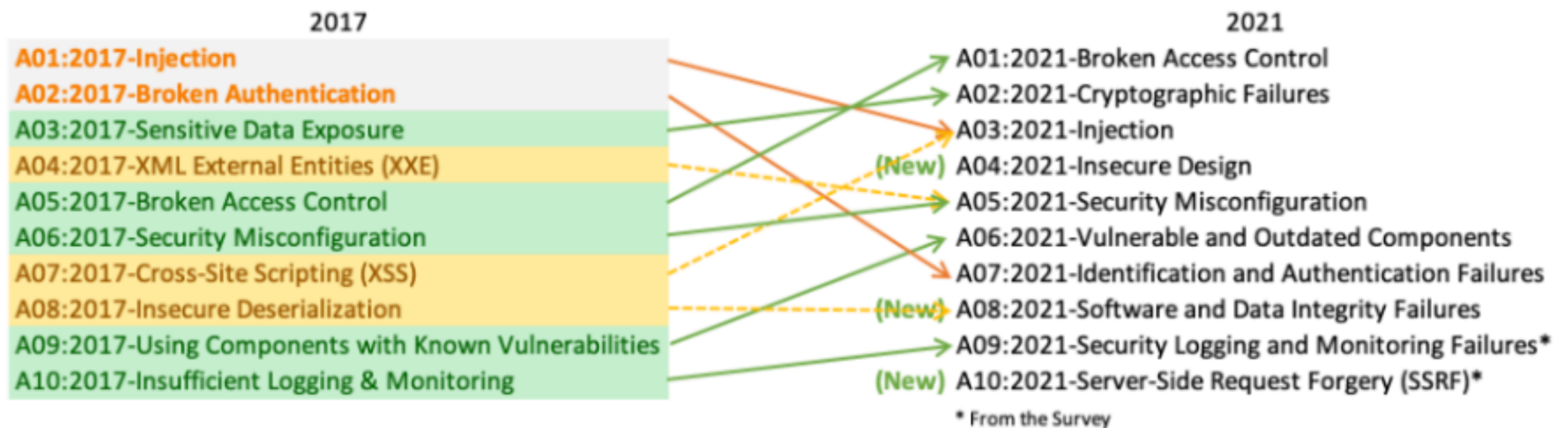
## **Aufmerksamer sein, was man auf Social-Media so postet**

## **Mehr Bock auf Security**

- =)

# OWASP top 10

---



<https://owasp.org/www-project-top-ten/>

# Was ist Authentication?

---

**Im Deutschen → Authentisieren, Authentifizieren -> Im Englischen → Authentication**

- Entität weißt sich aus, System überprüft und verifiziert
- Authentication != Authorisation

## **Drei Ansätze:**

- Wissen (Passwörter, Pins, Antworten auf Sicherheitsfragen, ...)
- Besitz (USB-Tokens, Authenticator auf dem Smartphone, Firmen/Uni-Ausweise)
- persönliches Merkmal (Fingerabdruck, Retina, bzw. Biometrie, Stimme, ...)

## **Was behandeln wir im Workshop?**

- Wissen -> Passwörter, Sicherheitsfragen, oauth



# Was ist Authentication?

---

## **Platform-Basiert:**

- Basic (.htaccess,..)
- Digest (nonce)
- Kerberos ( Auth Server)
- und weitere..

## **Form-Basiert:**

- Von Entwicklern selbst gebaut, nicht standardisiert und z.B. in RFCs zu finden

# Authentication Fehler und Lösungen

---

## Schlechte Passwörter:

- 1234, meineliebblingsband, geburtstag, namedeshaustiers, passwort1234, pw == username, 1, \_ ,
- Kann man raten, anfällig für Brute Force, cracking-tools, Dictionary Attack (Burp-Suite)

## Gute Passwörter und Passwort-handling:

- Passwort-Policy: > 8 Zeichen, Groß- u- Kleinbuchstaben, Zahlen, Sonderzeichen
- Passwort am besten generieren (PW-Manager) “W)SLLViYrL+E#8b~\[Q\_O~G84\vrFn”
- fail2ban, um Fehlversuche einzuschränken (aber ALLE, nicht nur auf valide User)
- das S in HTTP muss mit

## User Enumeration:

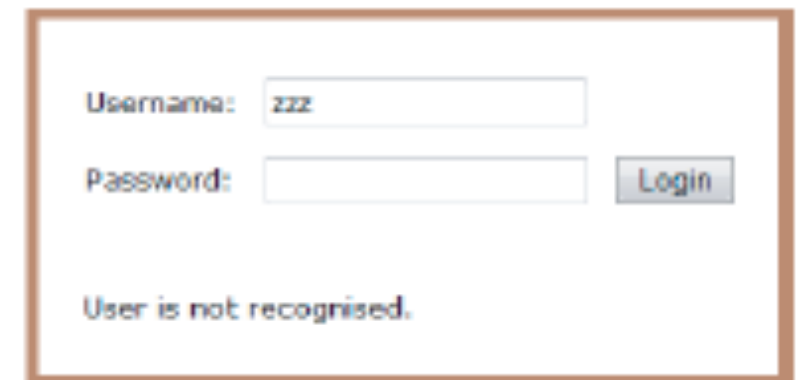
- Warum ist das falsch? —>



Username:

Password:

Password is incorrect.



Username:

Password:

User is not recognised.

# Authentication Fehler und Lösungen

---

## Password change:

- periodisch das PW ändern
- Verkürzt das Zeitfenster, in welchem das Passwort kompromittiert sein könnte
- Nutzer sollte bei Verdacht der Kompromittierung in der Lage sein, das PW schnell zu ändern

## Probleme:

- Im Login-Fenster alles richtig, im Change-PW-Fenster alles falsch

## Forgotten Password:

- Auch hier gilt das gleiche wie im Login und PW-Change: Error-handling! Eingabefelder!
- siehe gleich im HandsOn-Teil

## Default Password:

- Einfach mal mit in den Raum geworfen
- z.B. Router PW 0000
- Ändern!

## Initial Password:

- Sind generiert, Algorithmus könnte schlecht implementiert sein z.B.  $537 + n$
- Kann man raten

# Authentication Fehler und Lösungen

---

## Summary of Good-Practices:

- Starke Credentials = Eindeutiger Username, Password-Policy
- Credentials ordentlich ablegen = Hashes, SSL, Nicht in die URL als Parameter damit, nicht in die cookies, etc.)
- Ordentlich designte Login/ForgottenPassword/ChangePassword Fenster
- Change PW sollte nur über Authentifizierte Session erreichbar sein
- Passwort als ganzes validieren, nicht filtern
- Hin und wieder mal [haveibeenpwned.com](https://haveibeenpwned.com) checken :>
- IPs auf Zeit Sperren, die mehrere Login-Fehlversuche hatten, für alle Versuche!
- Account sperren bei zu vielen Fehlversuchen, muss dann Recovery nutzen über die E-Mail
- Nutzt MFA!
- Nutzt PW-Manager
- Email-Benachrichtigung über geändertes Passwort
- keine Passwort Hinweise im Login nutzen
- Password-Recovery am besten über gegebene E-Mail und nicht als Fenster hinter dem Login
- Monitoring, Intruder-Detection/Prevention

# OAuth

---

## Was ist OAuth:

- Open Authorization
- Protokoll
- Offener Standard
- API-Autorisierung
- Token statt Credentials werden übermittelt

## Warum:

- Third-Party Apps Zugang gewähren, ohne Zugangsdaten zu übermitteln
- BSP: Login in einen Web Shop mit dem Gmail-Account für bestimmte Informationen, die geteilt werden sollen

## Nicht behandelt

---

**Session Management**

**MFA**

**Authorization**

**Backend: Configs, Algorithmen, Implementierungen ...**

## Teil 2: HandsOn: Kontext, Requisiten

# Gliederung

---

## **Teil 1: Einführung (20 Minuten)**

- Ziel des WS
- OWASP top 10
- Was ist Authentication?
- Authentication Fehler und Lösungen
- OAuth
- Nicht behandelt

## **Teil 2: HandsOn: Kontext, Requisiten (10 Minuten)**

- Repo und der juice-shop
- Burp suite Community Edition

## **Teil 3: HandsOn: Aufgaben (70 Minuten + 10 Minuten Pause)**

- Aufgabe 1 (30 Minuten)
- Pause (10 Minuten)
- Aufgabe 2 (20 Minuten)
- Aufgabe 3 (20 Minuten)

## **Teil 4: Schluss, Feedback (10 Minuten)**



# Kontext und Requisiten

---

## **Mein Repository klonen, so wie im README beschrieben**

- dort zu finden sind die Aufgaben, die Präsentationsfolien
- und der als Submodul eingebundene juice-shop

## **Juice-Shop von OWASP:**

- Hochvulnerable Web-App
- Sehr viele Challenges aus allen möglichen (Risiko) Bereichen
- verschiedene Schwierigkeitslevel

## **Burp-Suite:**

- Security-Framework für Webanwendungen, ähnlich wie Metasploit, etc.
- brauchen wir nur für eine Aufgabe

# Kontext und Requisiten

---

## Sonstiges:

- Browser, Dev-Tools im Browser (Debugger)
- bissle OSINT

## Was brauchen wir in der Burp-Suite?

- Die folgenden Tabs:
- Proxy, im Proxy-Tab dann: Intercept, HTTP history
- Intruder, Payloads

## Vorgehen in der Burp-Suite?

- Proxy -> Intercept -> Open Browser
- Im Browser "localhost:3000" eingeben -> Login -> richtige Email, irgendein Passwort
- HTTP history -> POST Eintrag suchen -> Rechtsklick -> Send to Intruder
- Intruder -> clear-Button -> add-Button -> Payloads-Tab -> Start attack

## Teil 3: HandsOn: Aufgaben

# Gliederung

---

## **Teil 1: Einführung (20 Minuten)**

- Ziel des WS
- OWASP top 10
- Was ist Authentication?
- Authentication Fehler und Lösungen
- OAuth
- Nicht behandelt

## **Teil 2: HandsOn: Kontext, Requisiten (10 Minuten)**

- Repo und der juice-shop
- Burp suite Community Edition

## **Teil 3: HandsOn: Aufgaben (70 Minuten + 10 Minuten Pause)**

- Aufgabe 1 (30 Minuten)
- Pause (10 Minuten)
- Aufgabe 2 (20 Minuten)
- Aufgabe 3 (20 Minuten)

## **Teil 4: Schluss, Feedback (10 Minuten)**

# Aufgaben

---

## **Zu den Aufgaben**

- Aufgaben sind beschrieben
- Es gibt Hinweise
- Ansonsten gerne Fragen, weil bewusst vage gehalten
- Breakout-Sessions

## Teil 4: Schluss

# Gliederung

---

## **Teil 1: Einführung (20 Minuten)**

- Ziel des WS
- OWASP top 10
- Was ist Authentication?
- Authentication Fehler und Lösungen
- OAuth
- Nicht behandelt

## **Teil 2: HandsOn: Kontext, Requisiten (10 Minuten)**

- Repo und der juice-shop
- Burp suite Community Edition

## **Teil 3: HandsOn: Aufgaben (70 Minuten + 10 Minuten Pause)**

- Aufgabe 1 (30 Minuten)
- Pause (10 Minuten)
- Aufgabe 2 (20 Minuten)
- Aufgabe 3 (20 Minuten)

## **Teil 4: Schluss, Feedback (10 Minuten)**

# Schluss

---

## **Authentication:**

- Starke Passwörter (generieren lassen)
- Passwort Manager
- Protokolle richtig implementieren
- MFA, auch wenns manchmal nervt
- SSO
- Fail2Ban
- u.v.m.

## **Weiter machen?**

- juice-shop
- OWASP Vulnerable Web Applications Directory
- Vulnhub
- PortSwigger Security Academy



# Links

---

## Linksammlung:

<https://owasp.org/www-project-top-ten/>

<https://owasp.org/www-project-vulnerable-web-applications-directory/>

<https://cheatsheetseries.owasp.org/index.html>

<https://pwning.owasp-juice.shop/>

<https://portswigger.net/web-security>

<https://attack.mitre.org/>

<https://cirt.net/passwords>

<https://ctfchallenge.com/>

# Web Technologien: Web Security

---

Authentication Fehler und wie man sie verhindert  
Christopher Toth