

APEX Security Grundlagen

Philipp Hartenfeller, Senior Consultant

APEX Connect 2022, Brühl

Zahlen und Fakten



Gründung
1994



inhabergeführt



> 44 Mio. Euro
Umsatz in 2021



> 125 Kunden



> 300
Beschäftigte



zertifizierter
Partner führender
Technologie-
hersteller

Ihr Partner für den digitalen Wandel
Individuelle IT-Lösungen aus einer Hand



herstellerneutral



branchen-
übergreifend



Ausbildungsbetrieb,
Partner im
dualen Studium



enabling the adaptive enterprise



Philipp Hartenfeller

Seit 2016 @ MT AG
APEX / DBs / Web / JavaScript
Aus Düsseldorf

Blog: <https://hartenfeller.dev/blog/>



[@phartenfeller](https://twitter.com/phartenfeller)

Agenda

1. Einleitung

2. Client vs. Server

3. Session State Protection / Checksummen

4. SQL Injection

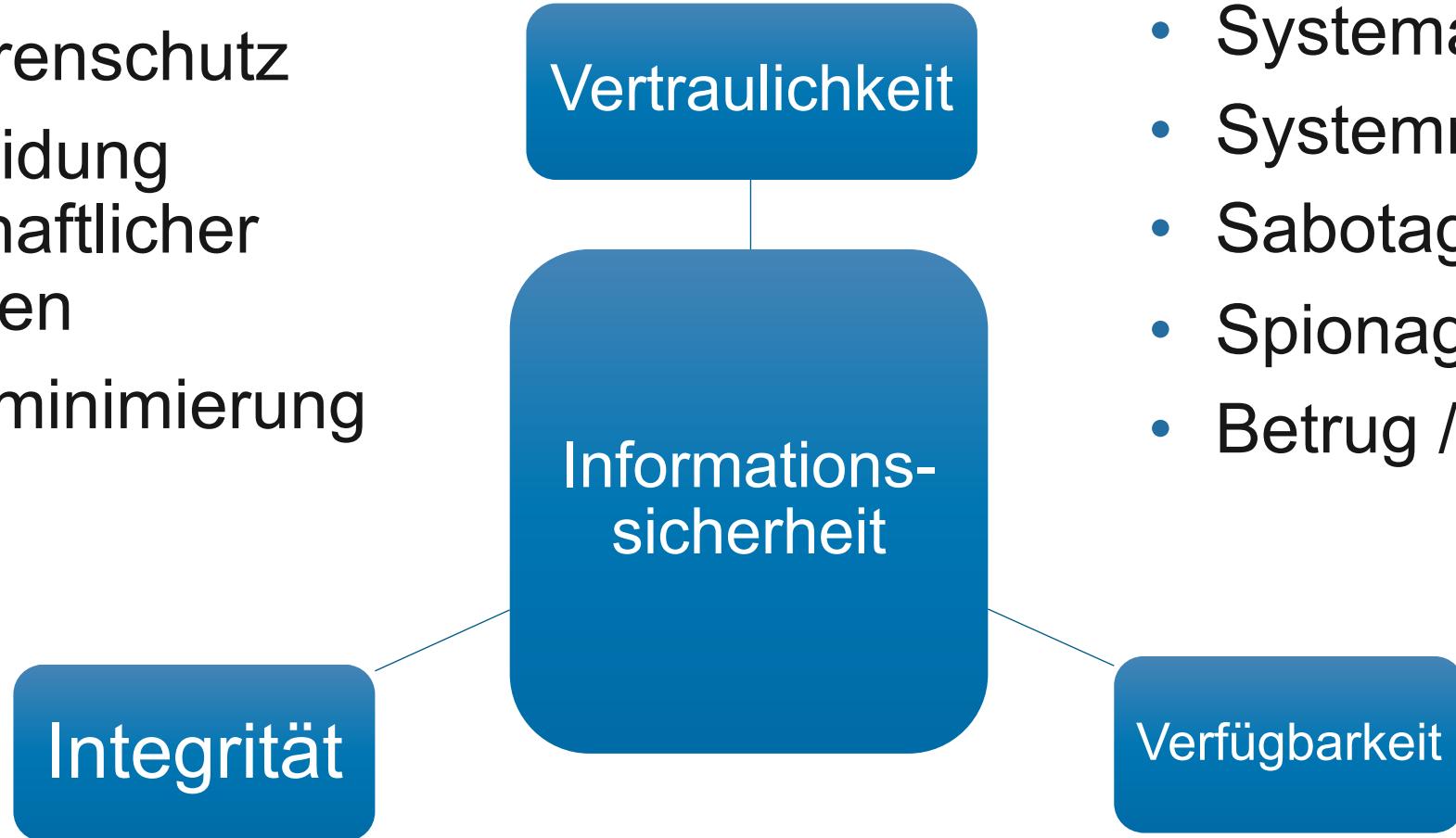
5. Cross Site Scripting (XSS)

6. Weitere Tipps

1. Einleitung

Security?!

- Gefahrenschutz
- Vermeidung wirtschaftlicher Schäden
- Risikominimierung



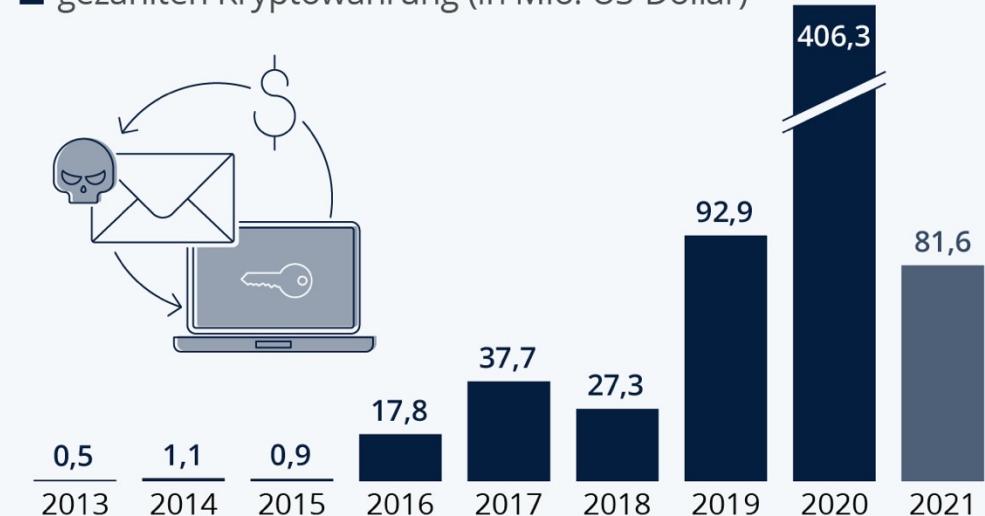
Quelle

Folgen fehlender Security

- Pegasus-Trojaner auf Telefonen vieler Politiker, Journalisten, Aktivisten etc. gefunden
- Einflussnahme auf Wahlen
 - Z. B. Frankreich, USA und Deutschland
- Ransomware legt Firmen, Universitäten und Krankenhäuser still

Das lukrative Geschäft mit dem Online-Lösegeld

Volumen der an Ransomware-Adressen gezahlten Kryptowährung (in Mio. US-Dollar)*



* Bitcoin Cash, Bitcoin, Ethereum, Tether
Stand: 10. Mai 2021
Quelle: chainalysis.com



statista

Abgrenzung des Vortags

Security findet fast überall statt

Nicht technisch:

- Zugangsschutz
- Social Engineering
- Standortwahl
- Enterprise-Architekturen / Personalrollen

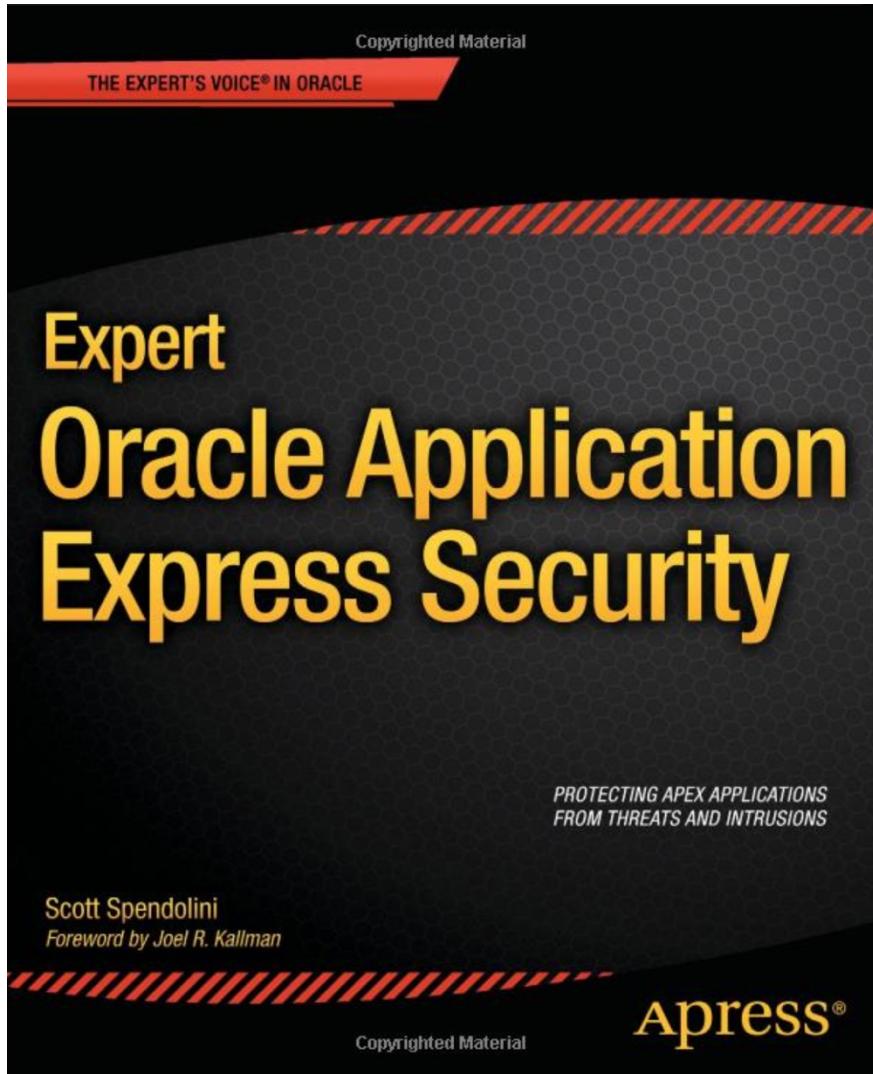
Technisch:

- Netzwerke
- Betriebssystem
- Verschlüsselung
- Backups
- Datenbankeinstellungen
- Webservereinstellungen

In diesem Vortrag: Was können APEX Devs tun?

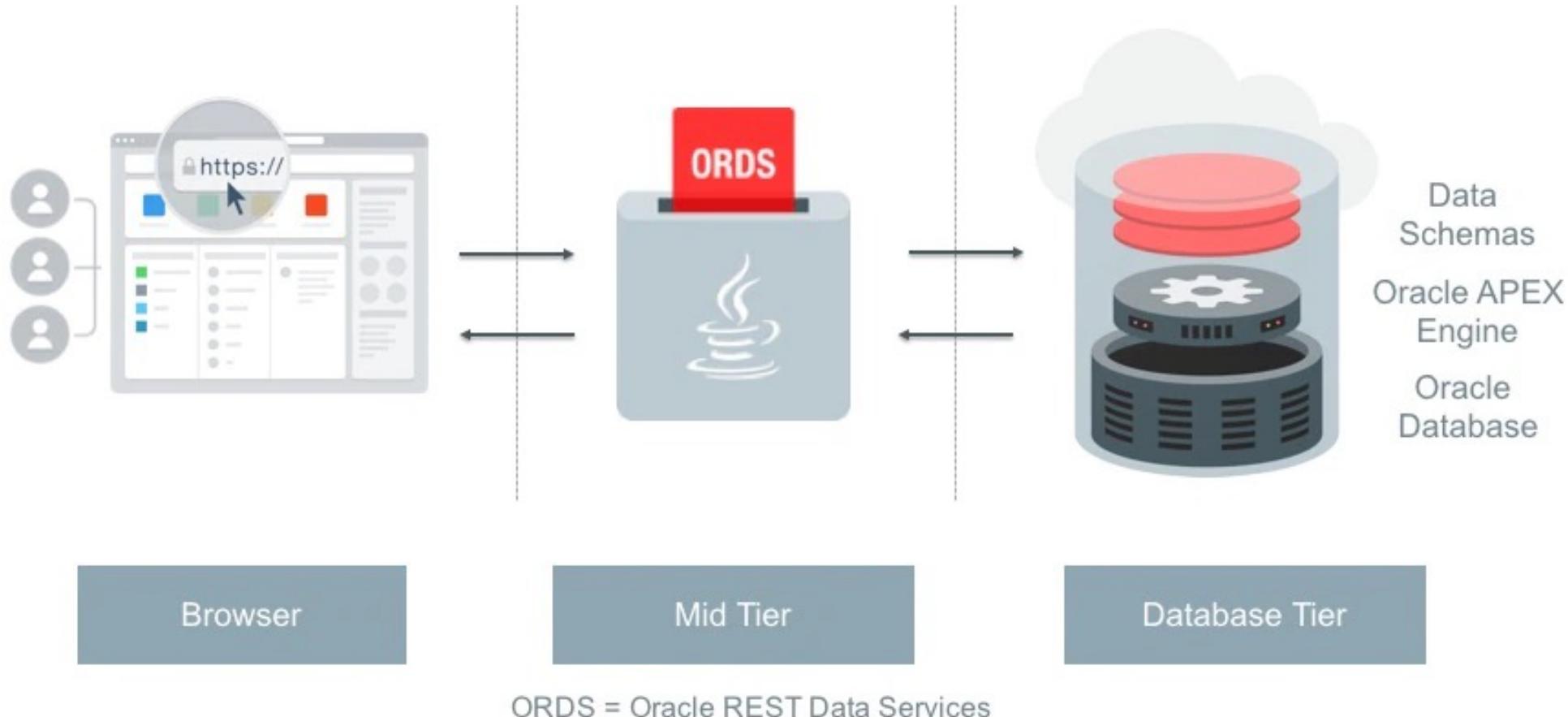
Also primär im Page Designer und in Datenbankcode
+ lediglich Grundlagen

Zum Weiterlesen



- Expert Oracle Application Express Security
- Autor: Scott Spendolini
- 22. April 2013
- ISBN: 978-1430247319

APEX Architektur



2. Client vs. Server

Was ist Client was ist Server?

Server	Client
<ul style="list-style-type: none">• SQL und PL/SQL• Server-side-conditions und Authorization-Schemes• → Auf Items, Spalten etc.	<ul style="list-style-type: none">• JavaScript + jQuery• Fast alle Dynamic Actions (z. B. hide)• Items vom Typ „hidden“

Wir haben keinen Einfluss auf den Client

- Sensible Daten müssen serverseitig gefiltert werden, sodass sie niemals den Client erreichen
 - Inspect Element
 - JS muss nicht ausgeführt werden (z. B. curl)
- Auch auf dem Server prüfen, ob ein Nutzer eine Aktion durchführen darf
 - Buttons sind lediglich Interaktionselemente, das nicht Vorhandensein verhindert nicht die Aktion



Demo: Client vs. Server

Sind wir kriminell?



@GovParsonMO/Twitter

Governor wants to prosecute journalist for right-clicking on a government website, thinks it's hacking

Apparently viewing a page source makes you a hacker, according to the Missouri governor.



Andrew Wyrich

Tech

Posted on Oct 14, 2021 Updated on Oct 15, 2021, 9:13 am CDT

<https://www.dailydot.com/debug/missouri-governor-reporter-hacker-mocked/>

3. Session State Protection / Checksummen

Session State Protection / Checksummen

Was ist das?

- Ziel: verhindern, dass Nutzer Parameter im Browser verändern
(wenn sie es nicht sollen)
- Funktionsweise wie bei Hashfunktionen → kryptischer Prüfwert den nur Server erzeugen kann
 - Wird mit eigentlichen Daten an den Client geliefert
- Beim Entgegennehmen / Submit:
 - Server prüft Input über dasselbe Verfahren auf Manipulationen

Beispiel Link: apex....p12_id=3...&cs=1lhBP1wVra-7EvIWFG....



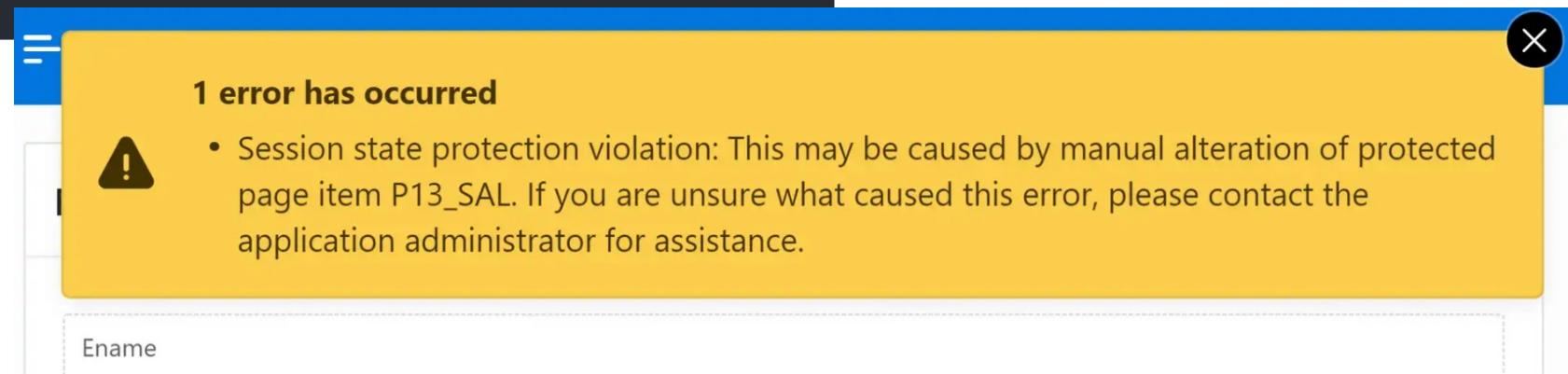
Demo: Checksummen in Links

Session State Protection / Checksummen

Read Only Items

- Können je nach Autorisierung editierbar sein
- Falls read-only -> Checksumme wird mitgeliefert

```
1 $s('P13_SAL', 123);
2 apex.submit({ request: 'SAVE' });
3
```



Session State Protection / Checksummen

Übersicht ob Items abgesichert sind

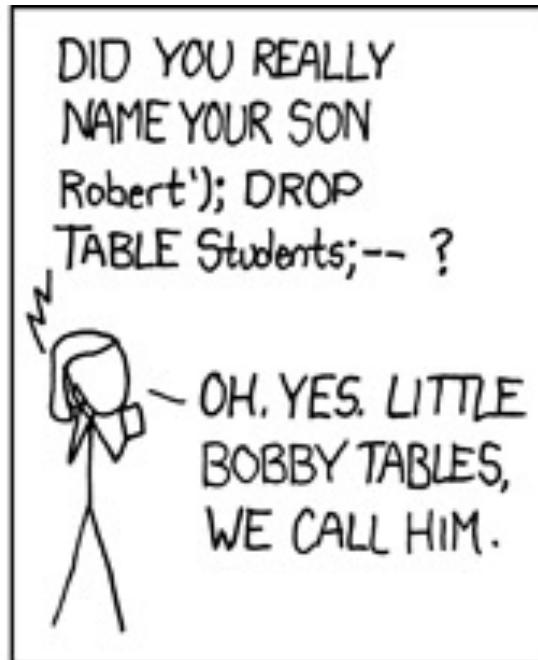
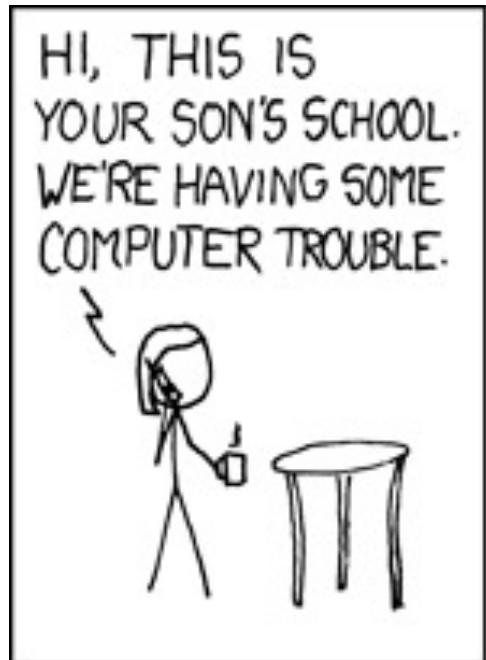
The screenshot shows the Oracle Application Builder interface for 'Application 109 \ Shared Components'. The 'Security' section is selected, specifically the 'Session State Protection' item, which is highlighted with a red border. Below this, the 'Existing Session State Protection Settings' table is displayed.

Pages		Page Items		Application Items	
Page Access	Pages	Item Access Level	Items	Item Access Level	Items
Arguments Must Have Checksum	5	Unrestricted	11	Restricted - May not be set from browser	1
No URL Access	1				

4. SQL Injection

SQL Injection?!

- Nutzer können von außen Datenbankbefehle „einschleusen“, die auch verarbeitet werden



[Quelle](#)

- Kein Problem bei APEX SQL-Feldern
 - --> Sicher gebindet mit :P100_ITEM
- **Dynamisches SQL**
 - --> Inputs überprüfen
 - --> Werte mit :var binden
 - --> dbms.assert verwenden
 - --> dbms.sql verwenden

The screenshot shows a dark-themed interface of Oracle Database SQL Developer. In the top right corner, there is a small icon with an upward-pointing arrow. Below it, the title bar says "SQL Query". The main area contains the following SQL code:

```
select ID,
       FIRST_NAME,
       LAST_NAME,
       COUNTRY,
       CREDIT_CARD_NO,
       PHONE_NO
  from USERS
 where ID = :P991_ID
```

At the bottom of the window, there is a toolbar with several icons. To the left of the toolbar, the text "Page Items to Submit" is displayed, followed by a button labeled "P991_ID". On the far right of the toolbar, there is a small icon with three horizontal lines and a vertical ellipsis.



Demo: SQL Injection

5. Cross Site Scripting (XSS)

XSS!?

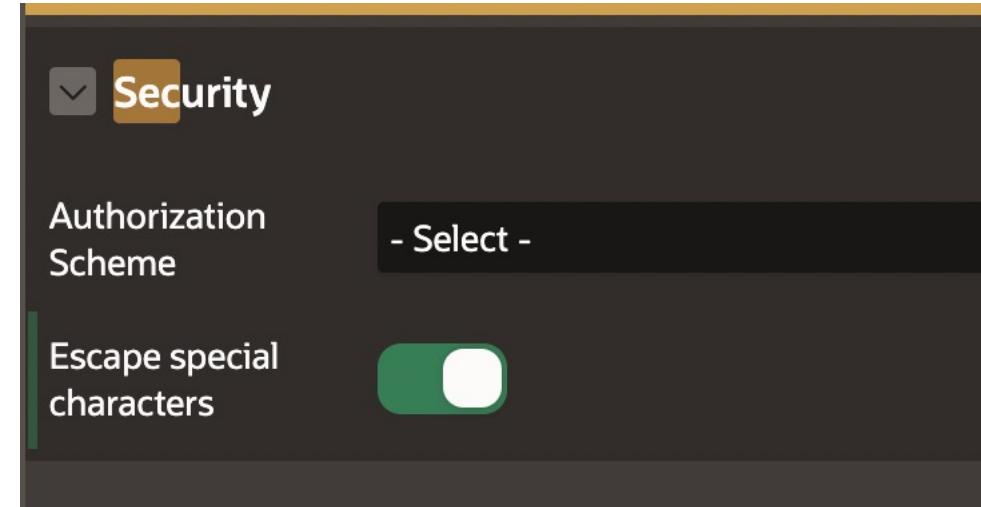
- Nutzer können von außen JavaScript-Befehle „einschleusen“, die auch verarbeitet werden
- Quasi SQL-Injection für JS
- Mögliche Folgen:
 - geheime Daten abgreifen
 - für andere Personen etwas ausführen
 - etc.

First Name

```
<script>alert('XSS!')</script>
```

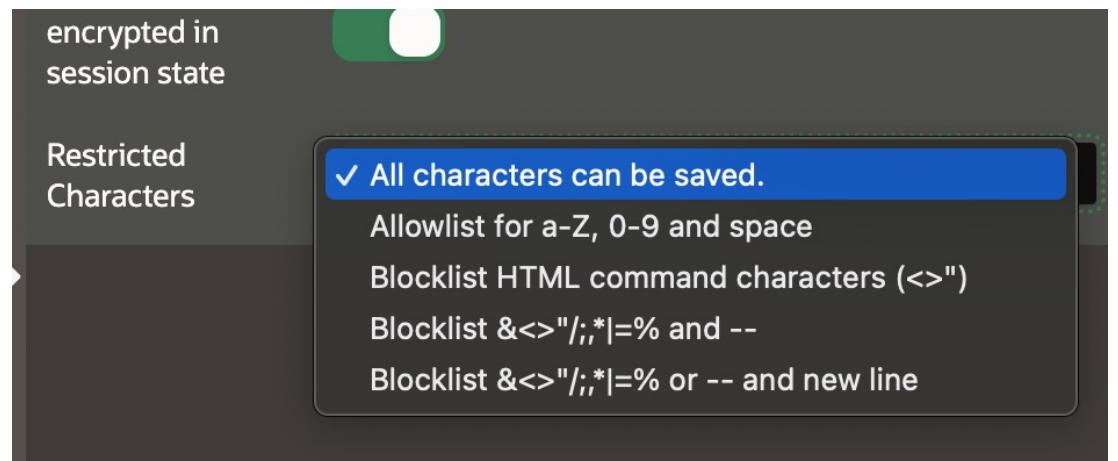
Gefahrenzonen

- htp.p
 - → APEX_ESCAPE Paket
- Report Spalten / Rich Text Editor...
 - → Escape special characters



Auch möglich:

- Input Items einschränken
- **Besser: Lücke schließen**





Demo: XSS

6. Weitere Tipps

Datenbankfeatures nutzen:

- Mehrere Schemen nutzen + nur nötige Rechte vergeben
- Virtual Private Databases

Scanner:

- APEX Advisor (integriert)
- ApexSec (kommerziell)
- APEX-SERT (OpenSource)

Single Sign-On

- Zentrale Authentifizierung
- Weniger Passwörter

Flexible und granulare Autorisierungskonzepte

- Definition Rechte (z. B. Nutzer lesen, bearbeiten, löschen...)
- Definition Rollen (z. B. PO, Analyst, Admin, Gast...)
- Zuordnung Rollen <-> Rechte
- Regelmäßig testen!!!

Logging

- Jegliche Operationen protokollieren
- Nachvollziehbarkeit + Debugging
- Missbrauch überhaupt feststellen
- Z. B. [OraOpenSource/Logger](#)

Infrastruktur

- Patches zeitlich einspielen
- [TLS](#) verwenden + [Security Header](#) einstellen

Weitere Tipps

- Keine Kopien von Anwendungen oder Seiten produktiv halten
 - Können gescannt → gefunden werden
 - Doppelter „Schatten“-Code
- KISS - Keep it simple, stupid
 - Mehr Komplexität → mehr Fehlerpotenzial

- APEX sind normale Webanwendungen
 - → Generell über Security für das Web informieren
 - Z. B. OWASP Top 10
- Selbstkritisch und aufmerksam programmieren
- Codereviews

„Testers don't break the code, they break your illusions about the code.“

Vielen Dank für Ihre Aufmerksamkeit!

Gibt es noch Fragen / Anmerkungen?



Philipp Hartenfeller

Senior Berater

Mail: philipp.hartenfeller@mt-ag.com

Twitter: [@phartenfeller](https://twitter.com/phartenfeller)

Blog: <https://hartenfeller.dev/blog>

MT AG
Balcke-Dürr-Allee 9
40882 Ratingen

www.mt-ag.com