ABB Technikerschule

Technik. Informatik. Wirtschaft. Management →

Dominik Meyer
Software Engineering

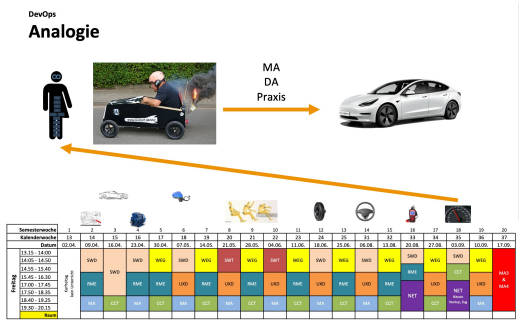
Security

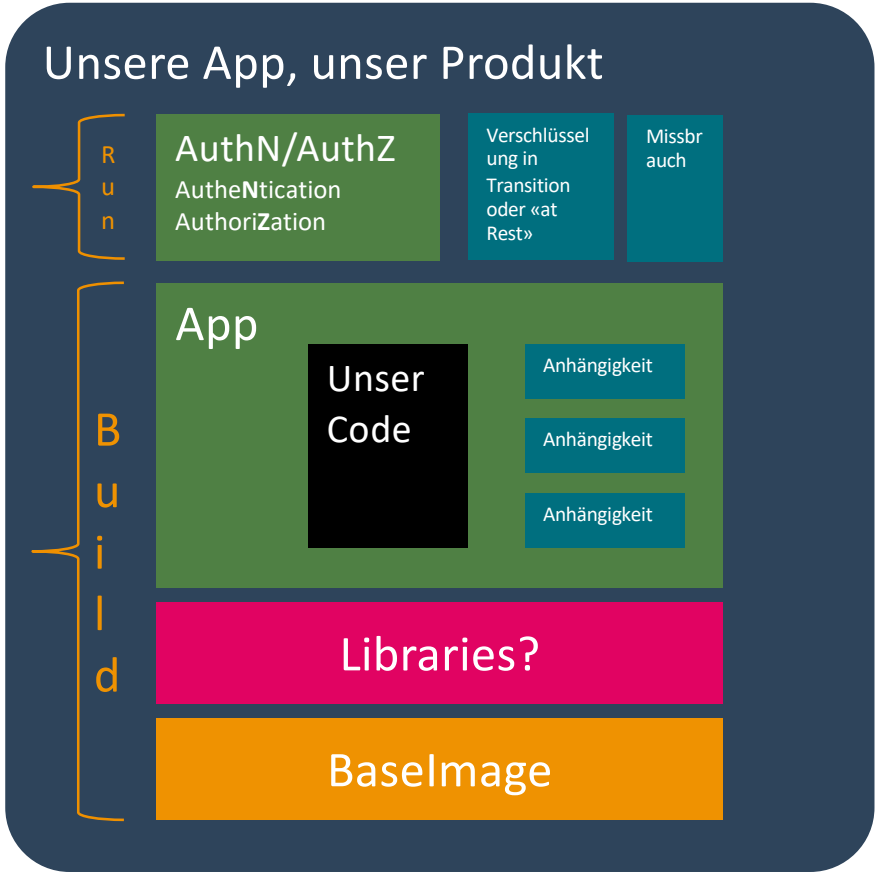
Ziel

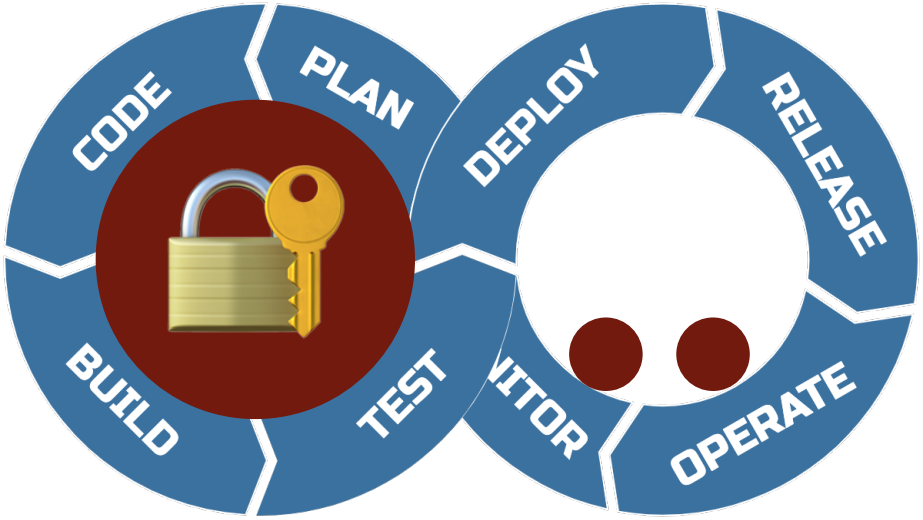
Nach der Lektion haben die Studierenden definiert, wie eine Applikation sicher entwickelt wird und wie die Daten der Kunden vor unerlaubtem Zugriff gesichert werden.

Security

– 1 Lektion wird dem Thema nicht gerecht
  – wir parallelisieren







Agenda

– Ein paar simple Grundregeln
– Abhängigkeiten
– Gruppenworkshop
– Zielkontrolle

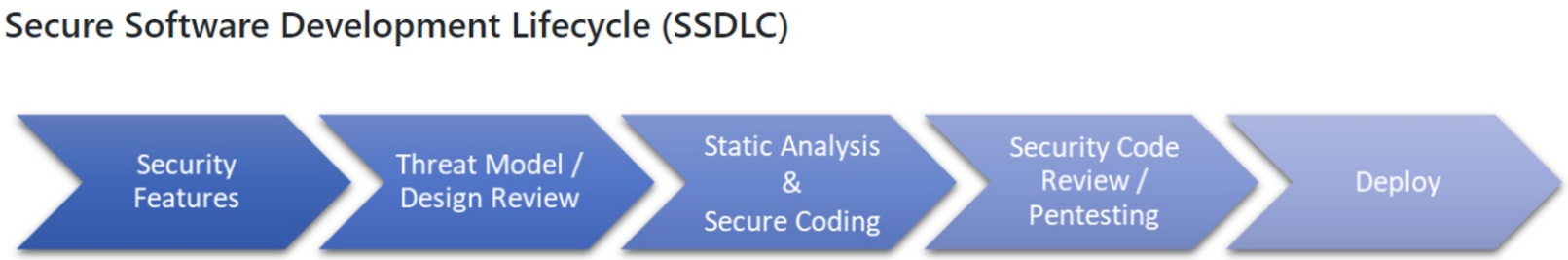
Die einfachste und effektivste Regel

- DR(Y)
  - Experten investieren ∞ Aufwand in sichere Abhängigkeiten → **Nutzen**
  - Lösungen gibt es für jede Sprache!
    - Java – Gradle, Maven (Lektion 8 und 9)
    - Node – NPM, Yarn
    - Go – go.mod
    - PHP – Composer, Packagist

- Gründe
  - Siehe /docs/theory/principles#do-not-repeat
  - Die Maintainer werden VULNAS schliessen, wir updaten
  - Alleine wird es bald ein 1-Personen-Projekt, dann 2, dann 3, dann 🪦

- **DO NOT REPEAT**

Natürlich ist DRY nicht der heilige Gral



– Wir müssen selber auch mitdenken
– Threat Model
– Design Review
– Static Analysis (Container und Applikation!)
– Secure Coding (u.a. DRY)
– Security Code Review
– Pen Testing
– Deployment (Simple Regel voran, HTTPS only!)

Security ist langweilig… 😴

- Wenn man es richtig macht ja, dann wird’s fast schon langweilig
- Security muss als Challenge und nicht «Stein im Weg» gesehen werden
  - Wir wachsen daran
  - Viele Security Tools und «Vertreter» sind im Jahre 1980 stehen geblieben, es gibt aber auch 2021 Lösungen!
- Moderne Tools machen das Leben einfacher, es macht sogar Spass!
  - «Culture eats strategy for breakfast», Developer Experience matters!

Um als Firma/Produkt zu beweisen, dass man alles tut für Security gibt es diverse Zertifikate, z.B. ISO 27001

Task

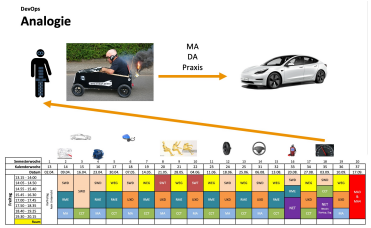
Workshop

AUFTRAG

Je Person ein Thema (nächster Slide)

Auftrag:

- Thema recherchieren
- Schlüsselfragen beantworten
  - Wann (im SSDLC) wird diese Methodik/Technik praktisch angewandt?
  - Was ist das Resultat und was machen wir damit?
  - Mit welchem minimalsten Aufwand oder Tool könnten wir damit anfangen?
- auf 1 Slide zusammenfassen
  - min. 1 Visualisierung



- Präsentieren in 90s, timeboxed
  - Aufgezeichnet und zusammen mit der Präsentation abgelegt

Task

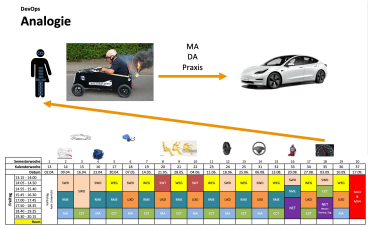
Workshop

AUFTRAG

Je Person ein Thema (nächster Slide)

Auftrag:

- Thema recherchieren
- Schlüsselfragen beantworten
  - Wann (im SSDLC) wird diese Methodik/Technik praktisch angewandt?
  - Was ist das Resultat und was machen wir damit?
  - Mit welchem minimalsten Aufwand oder Tool könnten wir damit anfangen?
- auf 1 Slide zusammenfassen
  - min. 1 Visualisierung



- Präsentieren in 90s, timeboxed
  - Aufgezeichnet und zusammen mit der Präsentation abgelegt

Task

Zuteilung

Stud	Thema	Präsentation Reihenfolge	Hilfe Start-Link
Angela Z.	Pen Testing	8	imperva.com/learn
Heinz L.	Authentication	2	what-authentication-means
Klaus C. B.	Authorization	3	definition/authorization
Adrian M.	Bug Bounty Program	4	wiki/Bug_bounty_program
Dominic B.	Secure Coding Practises	5	OWASP Secure Coding Reference
Simon B.	Encryption in Transit	6	cloud.google.com/security
Reto B.	Encryption at Rest	7	cloud.google.com/security
Peter H.	Incident Response	9	exabeam.com/incident-response
Bruno T.	Thread Modeling	1	owasp.org/.../Application_Threat_Modeling

Kommentar

Der Workshop gibt einen ersten Einblick
Wir treffen einige der Themen noch an!

Security alleine würde 2 Studiengänge füllen, wir tun das nötigste für eine «fahrbare Seifenkiste» (production-ready App).

Die meisten (99%) der Vulnerabilities können durch Updates und sicheres Engineering geschlossen werden.

Für den Rest müssen wir uns einen Plan zurechtlegen oder beweisen, dass die Lücke «not exploitable» ist.
– z.B. die Lücke taucht nur auf Windows auf, wir nutzen aber Linux

Zielkontrolle

Welches super simple Prinzip schafft einen grossen Anteil der Schwachstellen aus dem Weg?

Das Thema ist riesig. Was genau ist nochmal für die Modularbeit gefordert?

Welche Checks könnten wir in unsere CI einbauen um einen grossen Teil der Lücken – vor allem die der Abhängigkeiten - zu adressieren?

Hinweis: Das ging schnell, wir haben später 2 Lektionen (13, 31) wo wir vertiefen können.


inversions
for beginners

WEITER WISSEN.

Wir begleiten Sie!