Security von Import-Diensten

Im Umfeld von Docker-Microservices

Thomas Jäger

Hochschule für Angewandte Wissenschaften Hamburg

November 8, 2016

Agenda

1. Motivation
2. Bisherige Arbeit
3. Laufende Arbeit
4. Risiken
5. Ausblick Masterarbeit

Motivation

Warum Security im Docker- und Microservice-Umfeld?

- Microservice-Anwendungen weit verbreitet, z.B.:
  - 34 % der Java-Anwendungen (2016)[Map]
  - Ideal für IoT-Anwendungen [Tay]
  - Spotify, Netflix, etc.
- Docker auf dem Vormarsch
  - Viermal so viele Image-Pulls wie im Vorjahr (Januar 2016) [New]
- Security-Bedarf
  - Massive DDoS-Attacke von IoT-Devices u.A. auf Spotify [Bei]

Bisherige Arbeit

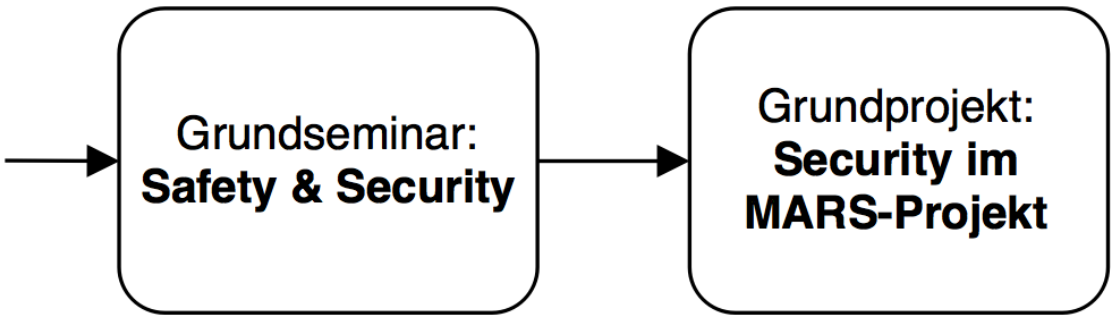
Abgrenzung

- Mögliche Risiken
- Bisherige Strategien zur Handhabung [MHTK]

Verschmelzung

- Verschmelzungspunkte - Beispiele [MHTK]
  - Automobile
  - Medizininformatik
  - Finanzsoftware
- Probleme und Herausforderungen [KZ10]
  - Zusammenführen verschiedener Kompetenzen
- Lösungsansätze
  - Erweitertes V-Modell [BLVW12]
  - Systemtheoretischer Ansatz [YL14]

Bisherige Arbeit



Grundprojekt: Security im MARS-Projekt

Ursprüngliches Ziel

- Durchführen einer Safety & Security Analyse

Vorbereitungen

- Suche nach geeignetem Untersuchungsobjekt
  - Möglichst bereits im HAW-Umfeld vorhanden

Probleme

- Safety-Fälle meist "konstruiert"
- Kein zugängliches, fertiges Untersuchungsobjekt vorhanden

$\Rightarrow$ Zunächst Konzentration auf Security

Grundprojekt: Security im MARS-Projekt

MARS als Untersuchungsobjekt [HWFTC14]
Ziele

- Überblick von Security-Aufgaben erstellen
  - Mögliche Schwachstellen identifizieren
  - Mögliche Gefährdungen der Schwachstellen zeigen
  - Mögliche Lösungsansätze zu Schwachstellen zeigen

$\Rightarrow$ Ergebnis als Basis für zukünftige Arbeiten

Grundprojekt: Security im MARS-Projekt

Das MARS-Projekt

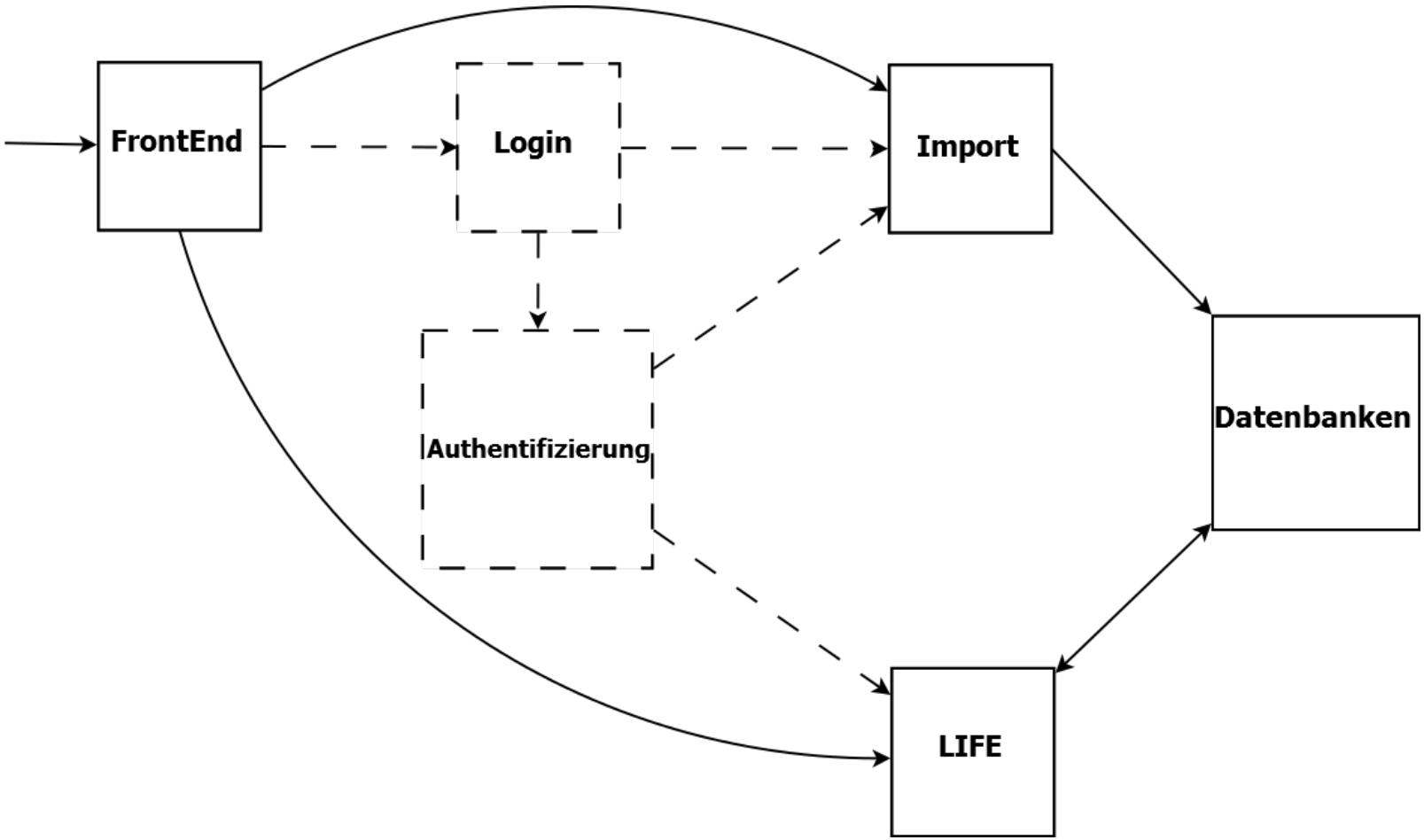
- Fachliche Funktionalität zweitrangig
- Microservice-Architektur
- Docker-Technologie
- Security nicht oder kaum betrachtet

Grundprojekt: Security im MARS-Projekt

Sicherheitsrisiken im MARS-Projekt

- Forschungsdaten geraten in falsche Hände
  - Diebstahl von Forschungsdaten
  - Benutzer erhalten unberechtigten Zugriff auf Fremddaten
- Daten werden manipuliert
  - Verfälschte, inkonsistente und unbrauchbare Simulationsergebnisse
- Daten gehen verloren
  - Durch Dritte gelöscht
  - Versagen von Backup-Routinen
  - Simulationsergebnisse gehen verloren

MARS-Workflow



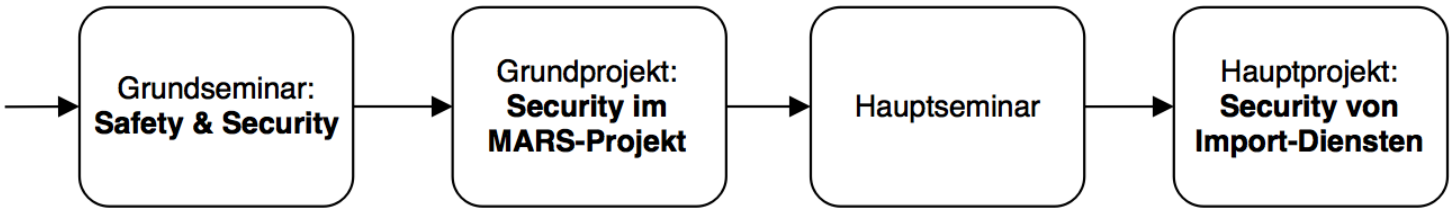
[Jä16]

Grundprojekt: Security im MARS-Projekt

Ergebnis [Jä16]

Sicherheitsaspekt	**Teilbereich**	**Priorität**
Microservice zur Authentifizierung	Zugriff auf Dienste	10/10
	Zugriff auf Daten	10/10
Import	File Upload Attacks	7/10
	Injections	4/10
Datenbanken	Zugriffsberechtigungen	8/10
	Verschlüsselung	6/10
	Monitoring & Forensik	3/10
Docker	Verschlüsselung	3/10
	Kommunikation	2/10

Laufende Arbeit



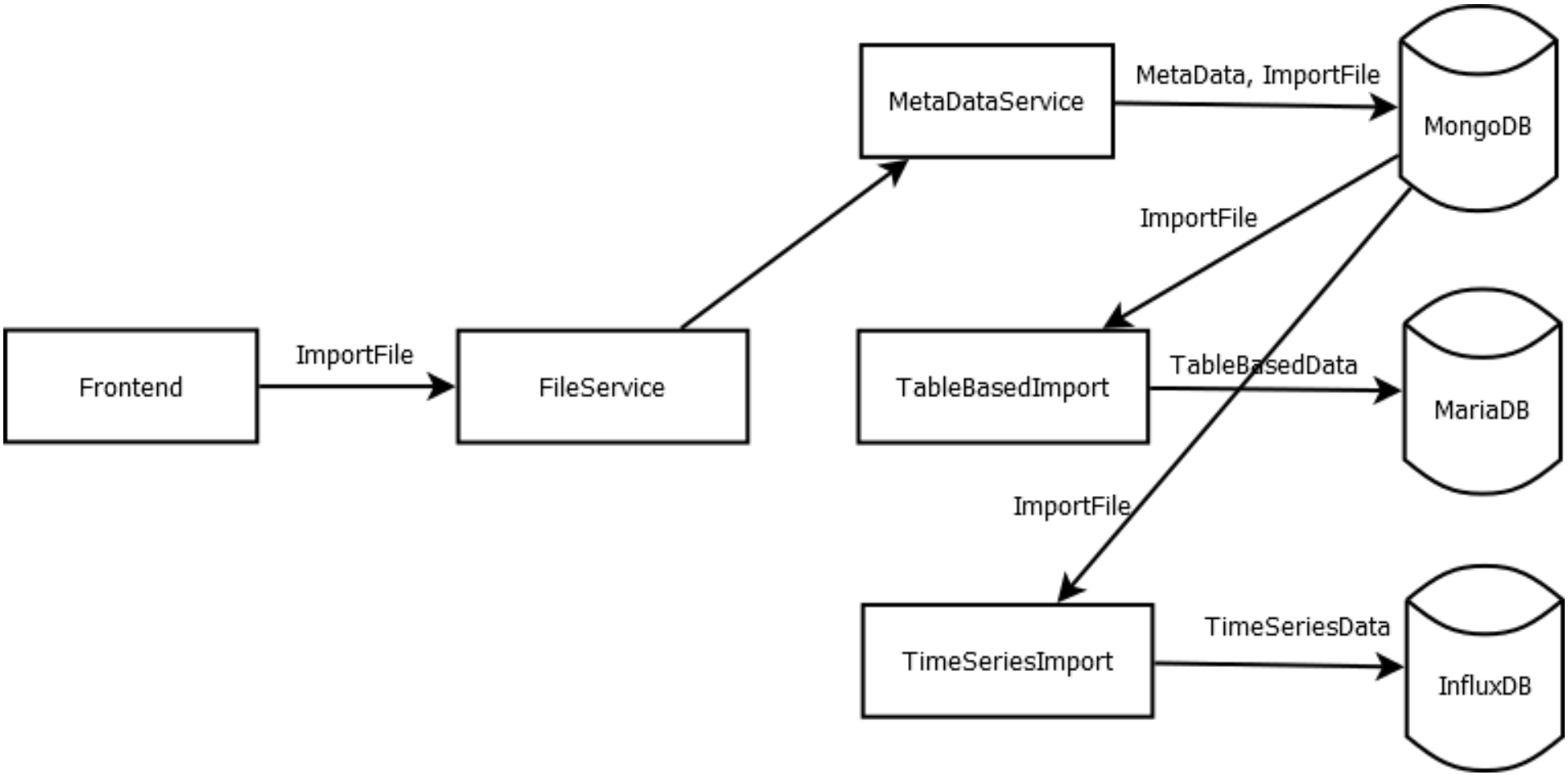
Hauptprojekt: Security von Import-Diensten in Docker-Microservices

Idee

- Vertiefung des Sicherheitsaspekts *Import* aus dem Grundprojekt
- Untersuchen der im Grundprojekt definierten Teilaspekte und Aufgaben
- Aus Ergebnis zukünftige Aufgaben identifizieren

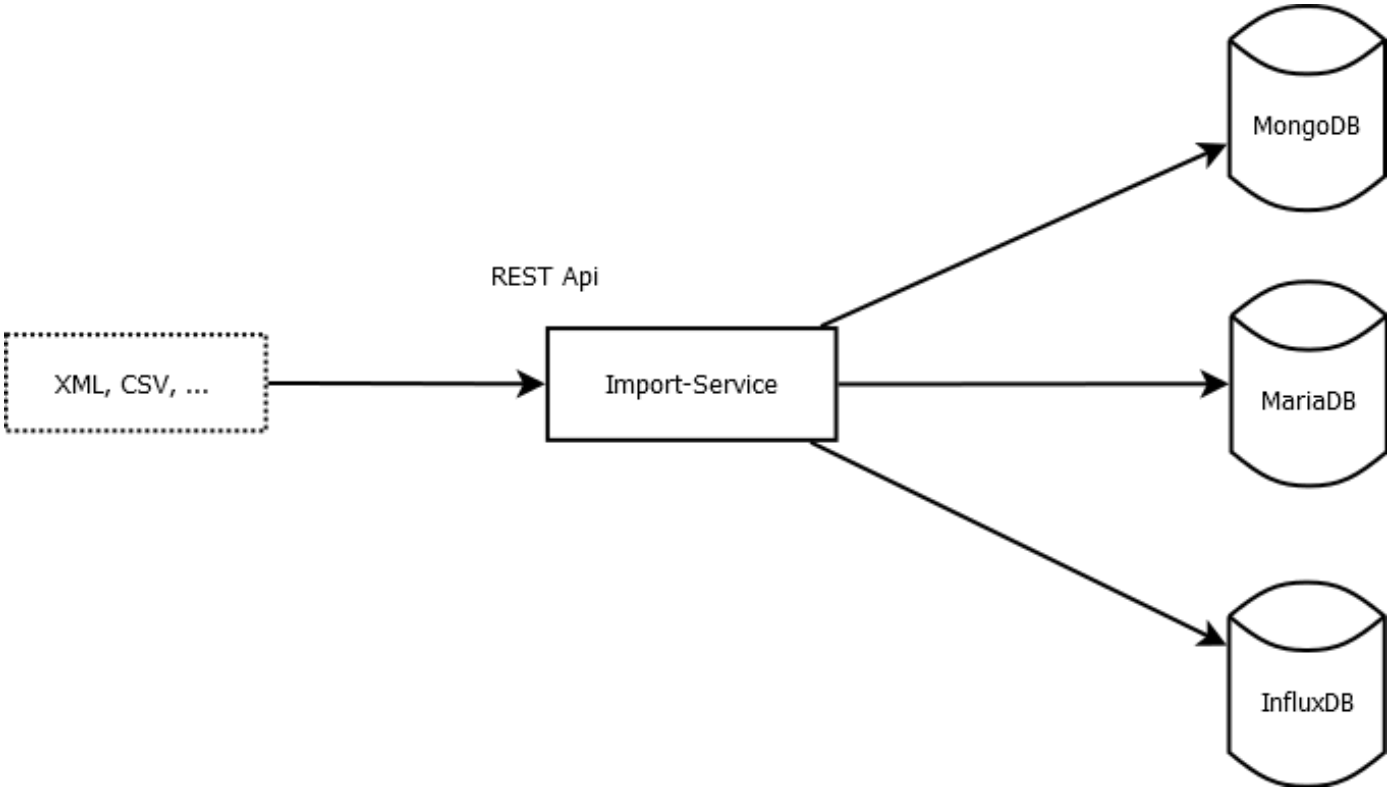
Hauptprojekt: Security von Import-Diensten in Docker-Microservices

Imports MARS [Jä16]



Hauptprojekt Vorgehen
I

Ausbau eines bereits vorhandenen Import-Dienstes zu Testobjekt



Hauptprojekt Vorgehen II

Untersuchung: Datenbank-Injections

- Relevanz
  - Häufigkeit SQL-Injections gleichbleibend hoch (2015: 31 % der Angriffsversuche auf Organisationen) [Ins]
- Ausgangspunkt
  - XML-Dateien und CSV-Dateien können SQL-Injections enthalten [Hea04] [Koc15]
- Zu untersuchen
  - Klassische SQL-Injection möglich?
    - MariaDB
  - SQL-Injection bei SQL-ähnlicher Abfragesprache möglich?
    - InfluxDB
  - Injection bei Nicht-SQL Abfragesprache möglich?
    - MongoDB [RSB15]

Hauptprojekt Vorgehen II

Untersuchung: Sonstige Injections

- Ausgangspunkt
  - XML-Dateien können ausführbaren Code einschleusen [Hea04]
- Zu untersuchen
  - Können CSV-Dateien ausführbaren Code einschleusen?
  - Verhalten im Zusammenhang mit Docker-Containern?

Hauptprojekt Vorgehen III

Untersuchung: Flooding, File Upload Attacks

- Relevanz
  - Flooding-Attacken nehmen weiter zu (2015) [Imp]
- Ausgangspunkt
  - Flooding-Attacken sind durch XML-Dateien möglich [Hea04]
  - Import-Dienste können ausfallen [Hea04]
  - Festplatten des Datenbankservers können volllaufen [EAK13]

Hauptprojekt Vorgehen III

Untersuchung: Flooding, File Upload Attacks

- Zu untersuchen
  - Auch für CSV-Dateien gültig?
  - Wie reagiert das Microservice / Docker-Umfeld?
  - Fallen Microservices aus?
    - Import-Service
    - Datenbank-Service
  - Geraten Daten in inkonsistente oder nicht rekonstruierbare Zustände?

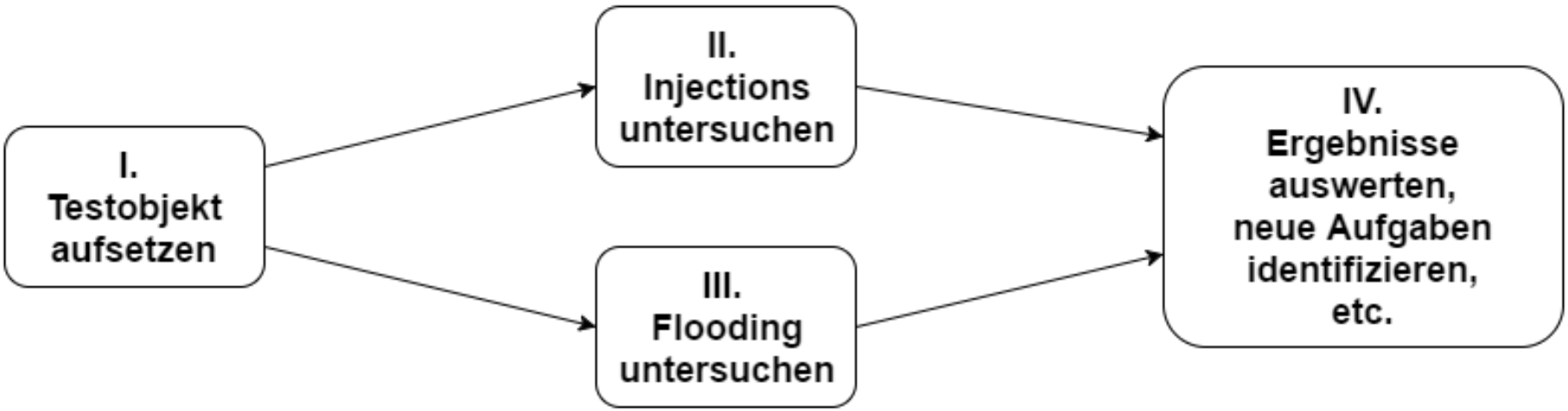
$\Rightarrow$ Lasttests, Penetrationtests einsetzen.

Hauptprojekt Vorgehen
IV

Auswertung

- Ergebnisse zusammentragen und auswerten
- Im Projektverlauf aufkommende Aspekte sammeln
- Zukünftige Aufgaben identifizieren
- Ergebnisse auf MARS-Umgebung beziehen

Zusammenfassung

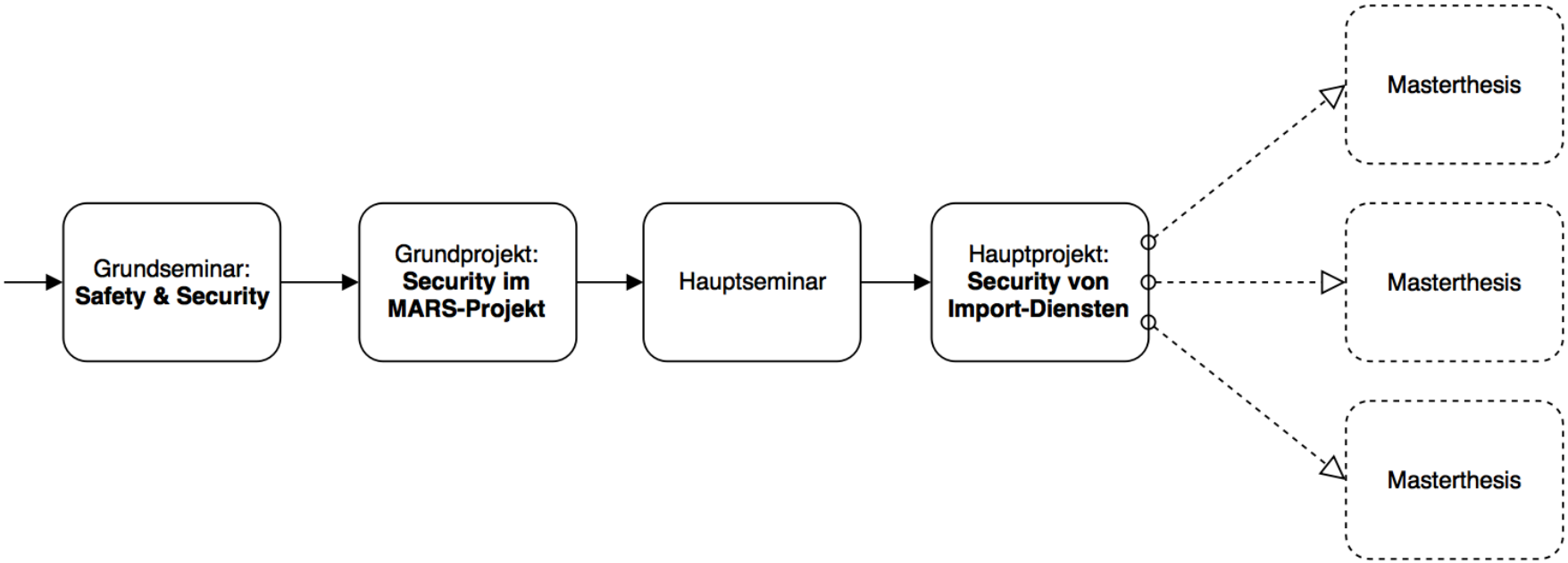


Projektrisiken

Projektrisiken

- Eigene Risiken
  - Praktische Durchführung von Injections wird zu zeitaufwendig
  - Praktische Durchführung von Last- und Penetrationtests wird zu zeitaufwendig
    $\Rightarrow$ Mangelnde Erfahrung
- Ergebnisse
  - Erwartete Risiken sind bereits abgedeckt
    $\Rightarrow$ Anderes Ergebnis als erwartet
- Äußere Risiken
  - Ergebnisse des Testobjekts lassen sich nicht auf MARS projizieren
  - Das MARS-System ändert sich / wird umgebaut
  - Die MARS-Umgebung beinhaltet noch nicht identifizierte Besonderheiten

Ausblick Masterarbeit



Ausblick Masterarbeit

Möglichkeiten

- Aufgreifen der in der Auswertung identifizierten Aufgaben
  - Flooding im Microservice-, Docker-, IoT-Umfeld?
  - Injections auf NoSQL-Datenbanken?
- Schließen von identifizierten Sicherheitslücken
  $\Rightarrow$ Rückführung auf das MARS-Projekt
- Erweiterter Bezug auf Datenbank-Security
  $\Rightarrow$ Bei Eintritt eines entsprechenden Risikofaktors

Quellen

Stefan Beirsmann.
Störungen bei spotify und twitter: Iot-botnet für massiven ddos-angriff benutzt.
http://www.zdnet.de/88281499/stoerungen-bei-spotify-und-twitter-iot-botnet-fuer-massive

Simon Burton, Jürgen Likkei, Priyamvadha Vembar, and Marko Wolf.
Automotive functional safety = safety + security.
In *Proceedings of the First International Conference on Security of Internet of Things*, SecurIT '12, pages 150–159, New York, NY, USA, 2012. ACM.

A. A. A. El-Aziz and A. Kannan.
A survey on xml security.
In *Recent Trends in Information Technology (ICRTIT), 2013 International Conference on*, pages 638–642, July 2013.

Jessica Heasley.
Securing xml data.

In *Proceedings of the 1st Annual Conference on Information Security Curriculum Development*, InfoSecCD '04, pages 112–114, New York, NY, USA, 2004. ACM.

Christian Hüning, Jason Wilmans, Nils Feyerabend, and Thomas Thiel-Clemen.
Mars - a next-gen multi-agent simulation framework.
In J. Wittmann and D. Maretis, editors, *Simulation in Umwelt- und Geowissenschaften, Workshop Osnabrück*. GI, Shaker, 2014.

Imperva.
2015 web application attack report (waar).
`https://www.imperva.com/docs/HII_Web_Application_Attack_Report_Ed6.pdf`.
eingesehen: 28.07.2016.

Ponemon Institut:.
2015 state of the endpoint report: User-centric risk.
`http://www.ponemon.org/blog/2015-state-of-the-endpoint-report-user-centric-risk`.
eingesehen: 28.07.2016.


Thomas Jäger.
Security im mars-projekt.
Hamburg, Germany, 2016. HAW Hamburg.


Matt Koch.
Web application file upload vulnerabilities.
In *InfoSec Reading Room*. The SANS Institute, 2015.


Andrew J. Kornecki and Janusz Zalewski.
Safety and security in industrial control.
In *Proceedings of the Sixth Annual Workshop on Cyber Security and Information Intelligence Research*, CSIIRW '10, pages 77:1–77:4, New York, NY, USA, 2010. ACM.


Simon Maple.
Java tools and technologies landscape report 2016.
`http://zeroturnaround.com/rebellabs/java-tools-and-technologies-landscape-2016/`.


Nadja Menz, Petra Hoepner, Jens Tiemann, and Frank Koußen.
S2: Safety und security aus dem blickwinkel der öffentlichen it.


Business Cloud News.
Exponential docker usage shows container popularity.
http://www.businesscloudnews.com/2016/02/11/exponential-docker-usage-shows-container-popularity/.


Aviv Ron, Alexandra Shulman-Peleg, and Emanuel Bronshtein.
No sql, no injection? examining nosql security.
CoRR, abs/1506.04082, 2015.


Manu Tayal.
Iot and microservices architecture.
http://www.happiestminds.com/blogs/iot-and-microservices-architecture/.


William Young and Nancy G. Leveson.
An integrated approach to safety and security based on systems theory.
Commun. ACM, 57(2):31–35, February 2014.