APPLIED HACKING

Ludwig-Maximilians-Universität München

Tobias Madl am 05.07.2019

Fraunhofer

AISEC

AGENDA

- Vorstellung
- Forschung am AISEC
- Einführung in das Thema Hacking & CTFs
- Hands-On
- Lerneffekt von CTFs

Vorstellung

- Tobias Madl
- Bachelor Informatik (Duales Studium)
- Master of Applied Research in Engineering Sciences (Automitive Security)
- Ehemalig angestellter im Eigenbetrieb IT@M bei der Landeshauptstadt München (Bereich externe Netze, DMZ & Security)
- Wissenschaftlicher Mitarbeiter und Ph.D Student am Fraunhofer AISEC

Vorstellung

ALLES!

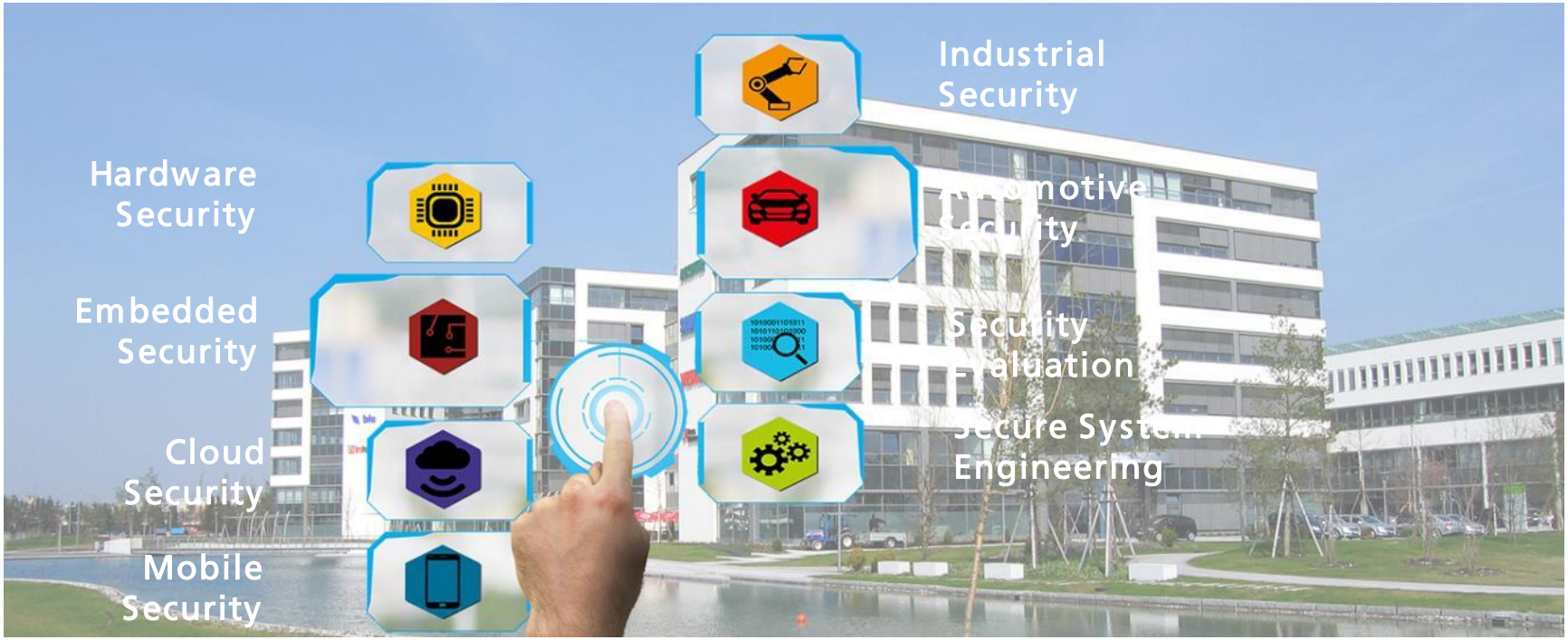
Cyber Security Challenge GERMANY

- Aktiver Capture The Flag (CTF) Spieler im Team ALLES
- Zwei jähriger Finalist der Cyber Security Challenge Germany
- Zwei jähriges Mitglied des deutschen Nationalteams
- Europameister der European Cyber Security Challenge 2018

Fraunhofer

AISEC

Angewandte und Integrierte SECurity



AISEC Partner *

* Ohne Universitäten und Forschungseinrichtungen

Neubau am TUM Campus in Garching

Fraunhofer AISEC

Meine aktuelle Forschung am AISEC

- Security Analysen für Fahrzeuge und Fahrzeugkomponenten namhafter Automobilhersteller
- Analyse von Virtualisierungssoftware und deren Schwachstellen
- Pentesting von Saugrobottern
- Beratung in Bezug auf Security beim Entwurf von neuen Software und Hardware Lösungen für globale Unternehmen

Was ist ein Hacker?

Was ist ein Hacker?

Was ist ein Hacker?

Was ist ein Hacker?

Was ist ein Hacker?

- **Wikipedia:** *„findige Tüftler im Kontext einer verspielten selbstbezüglichen Hingabe im Umgang mit Technik und einem besonderen Sinn für Kreativität und Originalität“*
- Querdenker
- Technik affin
- Im Informatik Kontext:
  - Gut programmieren
  - Breit aufgestelltes Themeninteresse
  - Spaß am analysieren von Technologien und Systemen

Wie wird man Hacker?

- Software entwickeln, eigene Systeme aufbauen, Technologien ausprobieren
- Kontaktaufnahme mit gleichgesinnten (Chaos Computer Club, hackerspaces)
- Vertieftes Engagement im Bereich IT-Security z.B. durch Teilnahme an Capture The Flags (CTFs)
- Eigeninitiative ist das wichtigste!
- Letztendlich System oder Software besser verstanden haben, als Entwickler selbst
- Zuletzt aufgeschlossen und kreativ Probleme lösen

Was sind CTFs?

Attack – Defense Style

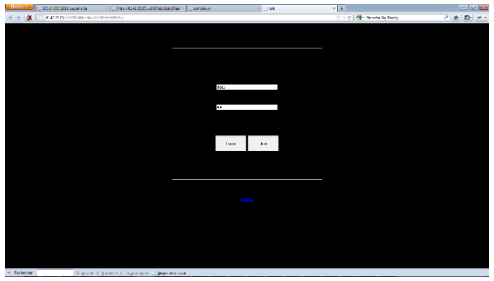
- Echtzeit Wettkampf
- Verteidige eigene Server/Services
- Attackiere andere Teams
- Punkte für
  - Angriff
  - Verteidigung
  - Verfügbarkeit
- Sehr realitätsnah

Jeopardy Style

- Probleme lösen
- Unterschiedliche Kategorien
- Punkte für Lösen der Aufgaben
- Wer mehr löst gewinnt
- Viel häufiger als A/D

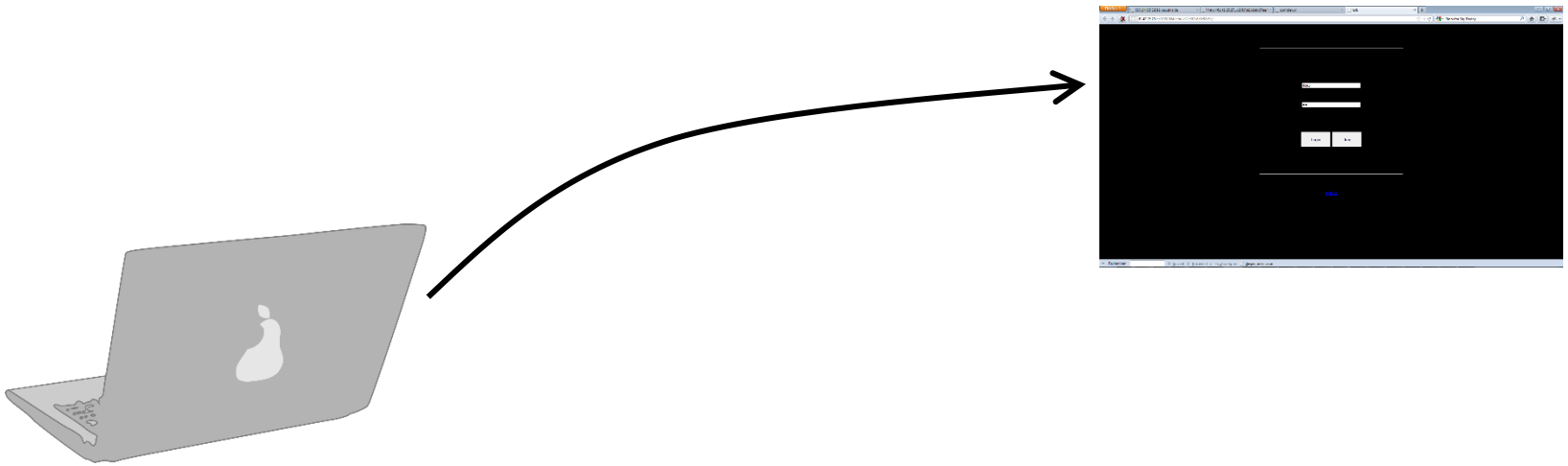
Was sind Jeopardy CTFs?

Follow the Rabbit!
(Web 200 pts)



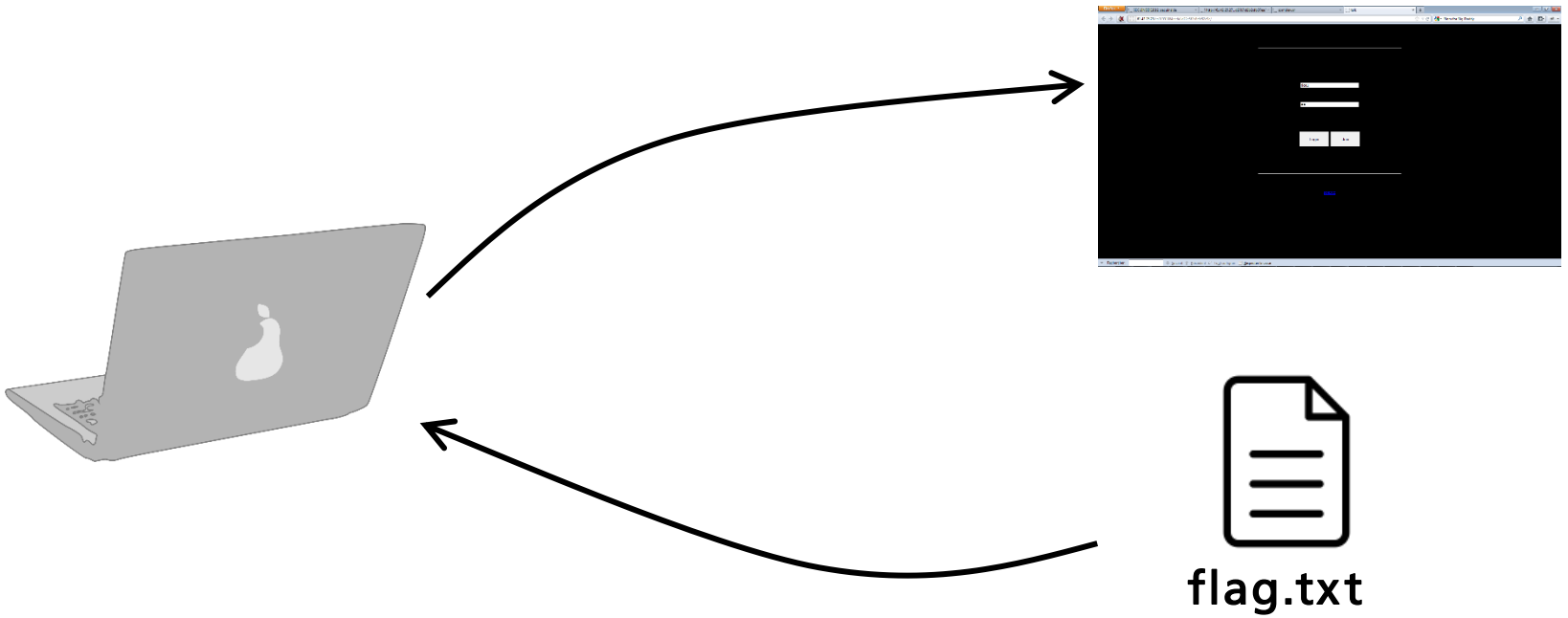
Was sind Jeopardy CTFs?

Follow the Rabbit!
(Web 200 pts)



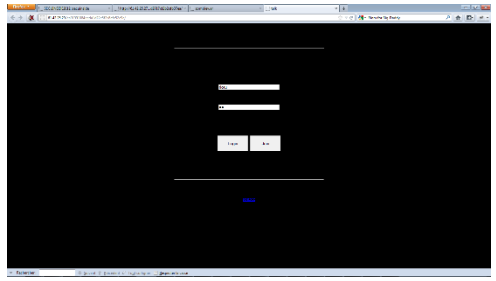
Was sind Jeopardy CTFs?

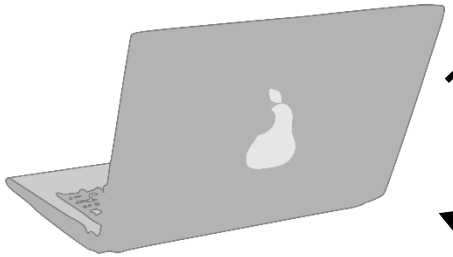
Follow the Rabbit!
(Web 200 pts)



Was sind Jeopardy CTFs?

Follow the Rabbit!
(Web 200 pts)





Congrats! +200 pts

flag.txt

CTF Aufgaben Beispiele

CTF Platformen

- Hacking-lab.com

- Ctftime.org

- Liveoverflow (Youtube)

Themengebiete im Hacking/IT-Security

- Kryptographie
- Web
- Exploiting
- Reverse Engineering
- Forensic
- Steganographie
- Mobile
- OSINT (Open Source Intelligence)

Warum CTFs?

- Durch CTFs umfasst allgemeine Informatik sowie IT Security Themen
- Selbstständiges und eigenverantwortliches lernen
- Problemlösen lernen (Suchen, bis man das Problem gelöst hat)
- Spielerisch und Interaktives lernen
- Einmalige Vorbereitung
- Zeit/Leistungsdruck kennenlernen/trainieren

⇨ Bisher sehr positive Rückmeldung von Schülern

Themengebiete im Hacking/IT-Security

- Kryptographie
- Web
- Exploiting
- Reverse Engineering
- Forensic
- Steganographie
- Mobile
- OSINT (Open Source Intelligence)

Kryptographie/Encoding

- Wissenschaft der Verschlüsselung/Verschleierung von Informationen
- **Ziel**: Entschlüsseln der Nachrichten
- Lerneffekt:
  - Sichere/Unsichere Verfahren
  - Konfigurationsfehler
  - Mathematik hinter kryptographischen Funktionen (z.B. durch leichtes abwandeln)
  - Gründe für Encodierung + Kryptographie

Web

- Webapplikationen welche einen bestimmten Service zur Verfügung stellen
- **Ziel:** Umgehen von Schutzmechanismen und Zugriff auf interne Bereiche oder sogar den Server dahinter
- Lerneffekt:
  - Kennenlernen verschiedener Technologien und ihrer Vor- & Nachteile
  - Konfigurationsfehler
  - Auch möglich einen Verbesserungsvorschlag zu fordern (Patch)
  - Erste kontakte zu Linux basierten Betriebssystemen

Reverse Engineering

- Umkehrung eines Herstellungsprozesses von z.B. Software oder Hardware
- **Ziel:** Rückgewinnung von z.B. Sourcecode aus Maschienencode
- Lerneffekt:
  - Grundverständnis von Maschinensprache (Assembler)
  - Auswirkungen programmierfehlern
  - Grundverständnis von Debuggern
  - Ggf. Sogar Hardwarereversing denkbar -> Elektrotechnik

Exploiting

- Verändern des Programmflusses zur Ausführung eigenen Codes oder eigener Funktionalität
- **Ziel:** Ausnutzen einer Softwareschwachstelle um erhöhe rechte zu erlangen oder auf gesperrte Daten/Bereiche zuzugreifen
- Lerneffekt:
  - Grundverständnis von Betriebssystemen und deren Speicherlayout
  - Kennenlernen von Betriebssystem seitigen Schutzmechanismen und deren Schwachstellen
  - Programmier- /Konfigurationsfehler

Mobile

- Mobile Applikationen wie z.B. Apps für Android oder iOS
- **Ziel:** Ähnlich RE, Programmcode wiederherstellen/verstehen um Zugriff beschränkte Informationen oder Funktionen zu erhalten
- Lerneffekt:
  - Kennenlernen verschiedener Technologien und ihrer Vor- & Nachteile
  - Reverse Engineering von Hochsprachen wie Java usw.
  - Grundverständnisse von mobilen Betriebssystemen und deren Schutzmechanismen

Vielen Dank für eure Aufmerksamkeit