Continuous DevOps IT Security Integration

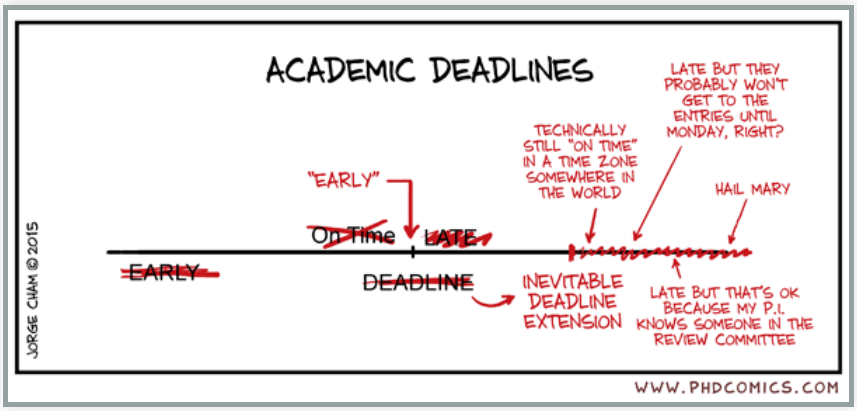
Geht das?

IT-SECX 2018

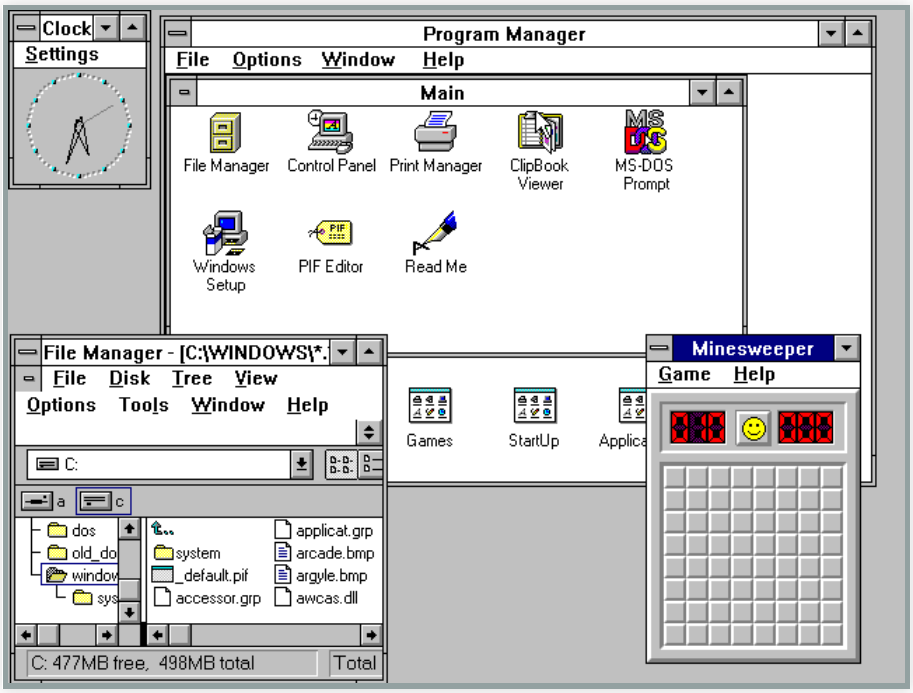
René Pfeiffer / @deepsec / DeepSec

IT-SECX 2018, FH St. Pölten

Software & Forschung



Software veraltet ständig



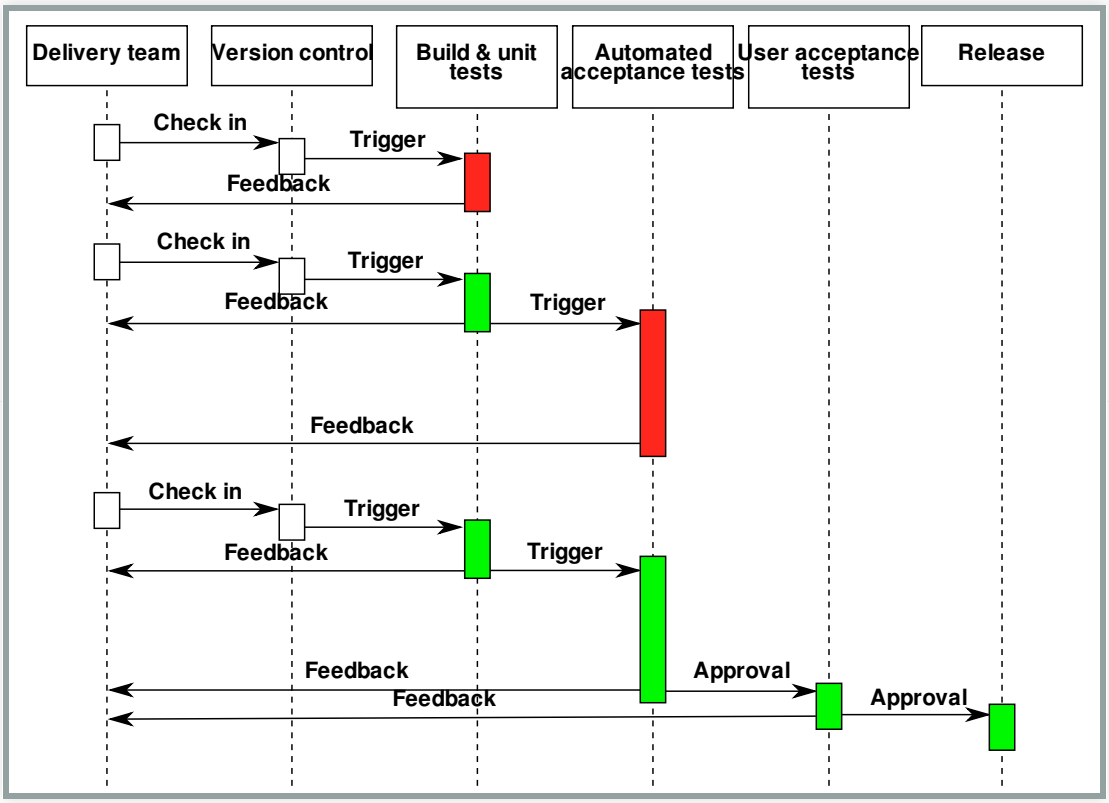
Brief History of CI

- Merge Code über alle Branches täglich (CI, 1991)
- Extreme Programming (XP) – täglich mehrere Merges
- Vielzahl an Frameworks & Methoden
- Quintessenz
  - schnellere Reaktion
  - engere Feedbackschleifen
  - häufige(re) Tests der Änderungen

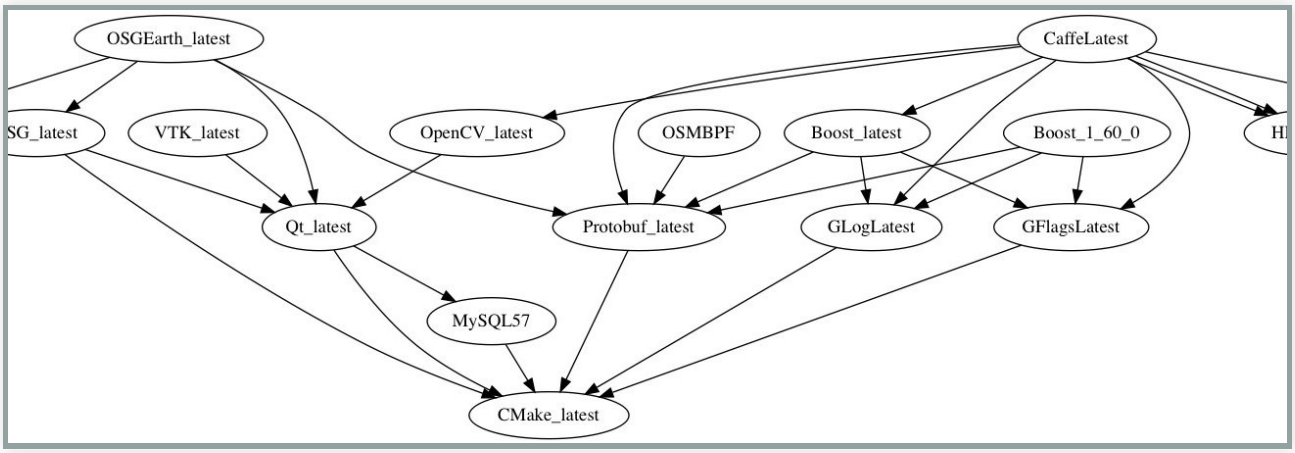
Continous Integration

- Code Repository
- automatisierter Build
- Build testet sich selbst
- tägliche Commits
- Tests in realistischer Umgebung
- Verteilung der Builds
- automatisiertes Deployment

Continous Integration



Komponenten

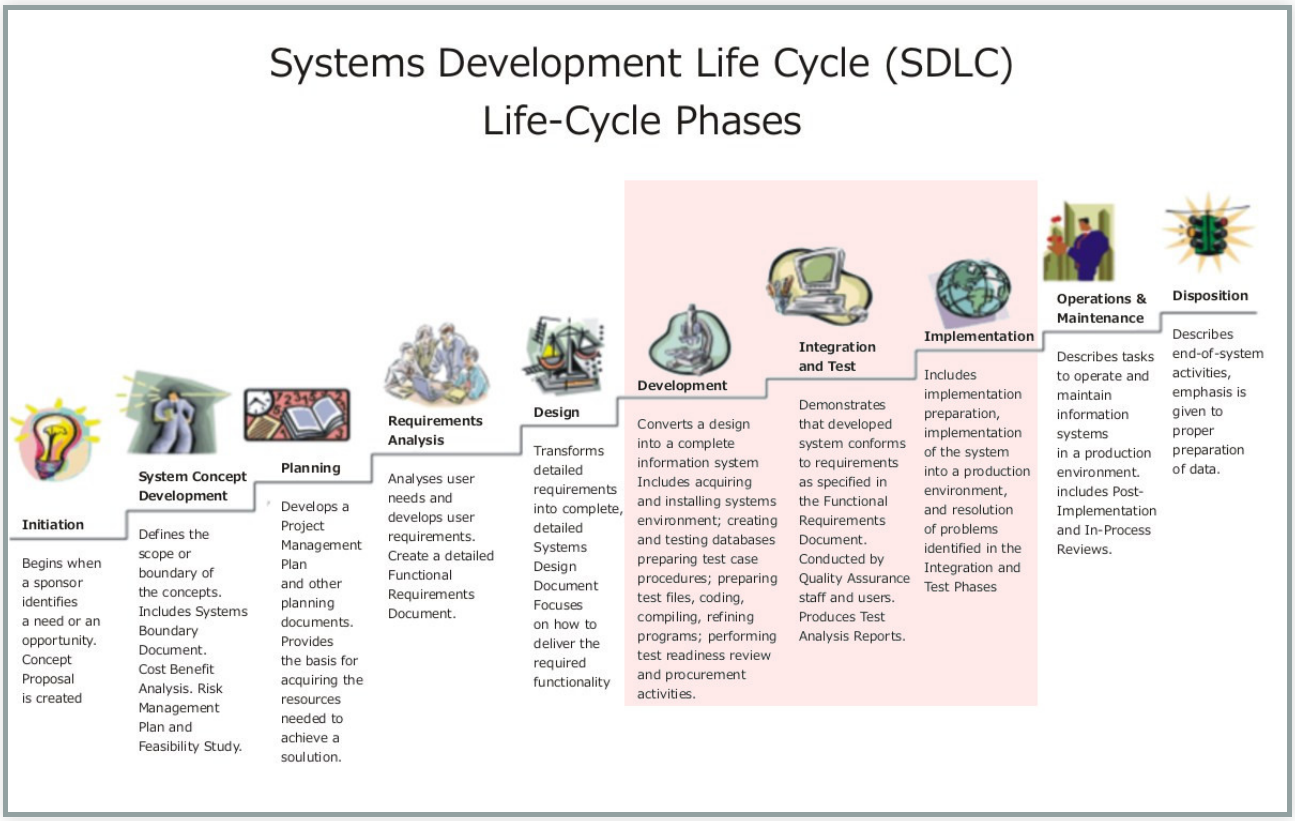


Umgang mit Fehlern

Wide Field Planetary Camera 1

Wide Field Planetary Camera 2

Security?


Systems Development Life Cycle (SDLC)
Life-Cycle Phases
Initiation
Begins when a sponsor identifies a need or an opportunity. Concept Proposal is created
System Concept Development
Defines the scope or boundary of the concepts. Includes Systems Boundary Document. Cost Benefit Analysis. Risk Management Plan and Feasibility Study.
Planning
Develops a Project Management Plan and other planning documents. Provides the basis for acquiring the resources needed to achieve a soulution.
Requirements Analysis
Analyses user needs and develops user requirements. Create a detailed Functional Requirements Document.
Design
Transforms detailed requirements into complete, detailed Systems Design Document Focuses on how to deliver the required functionality
Development
Converts a design into a complete information system Includes acquiring and installing systems environment; creating and testing databases preparing test case procedures; preparing test files, coding, compiling, refining programs; performing test readiness review and procurement activities.
Integration and Test
Demonstrates that developed system conforms to requirements as specified in the Functional Requirements Document. Conducted by Quality Assurance staff and users. Produces Test Analysis Reports.
Implementation
Includes implementation preparation, implementation of the system into a production environment, and resolution of problems identified in the Integration and Test Phases
Operations & Maintenance
Describes tasks to operate and maintain information systems in a production environment. includes Post-Implementation and In-Process Reviews.
Disposition
Describes end-of-system activities, emphasis is given to proper preparation of data.

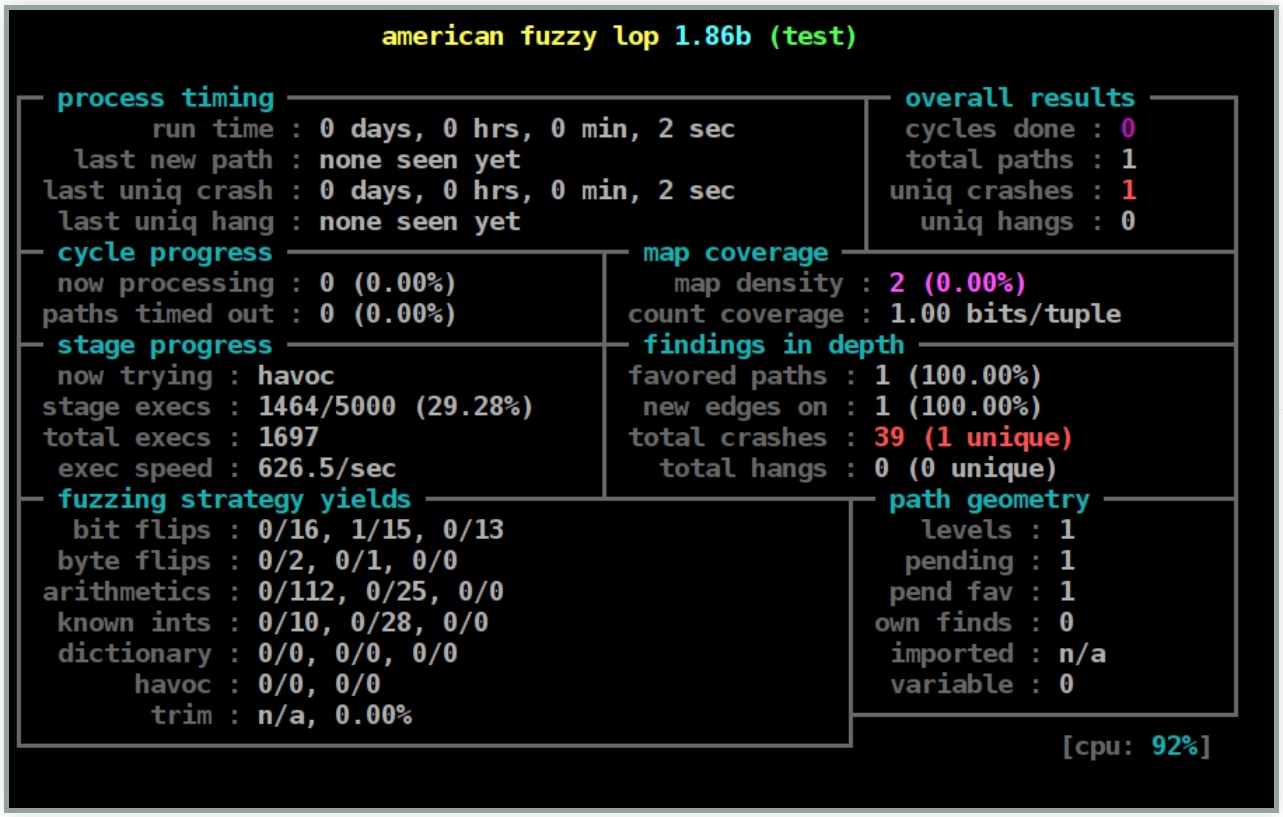
Security?

- Tests prüfen, ob Build funktioniert
- Testfälle orientiert an Use Cases / Fixed Bugs
- reine Sicherheitstests sind selten,
  - da „Angriffsdaten“ fehlen
  - weil Tests länger dauern
- unbekannte Schwachstellen sind unbekannt

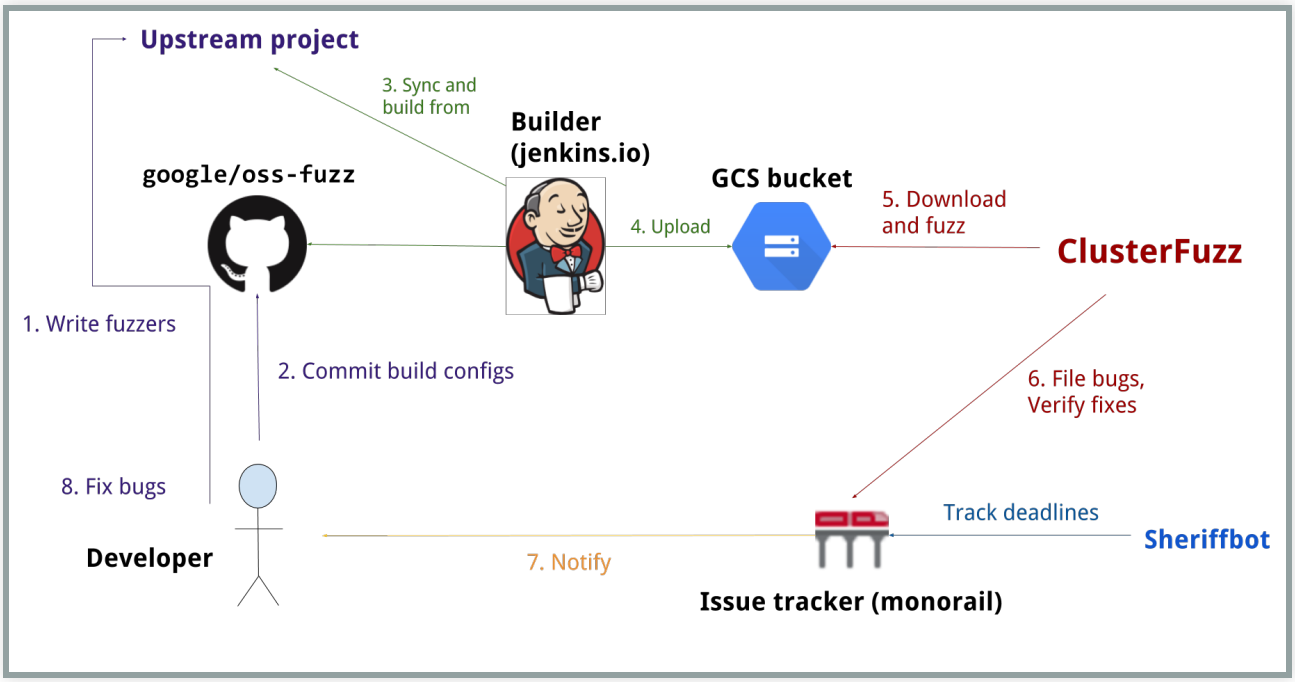
Sicherheitstests

- Testfälle müssen erweitert werden
- Einbau bekannter Sicherheitslücken
- automatische Stresstests über längeren Zeitraum
- automatische Generierung von Testdaten
- Ziel: maximale Abdeckung der Code Pfade
- Wichtig: Kriterien für Testergebnisse

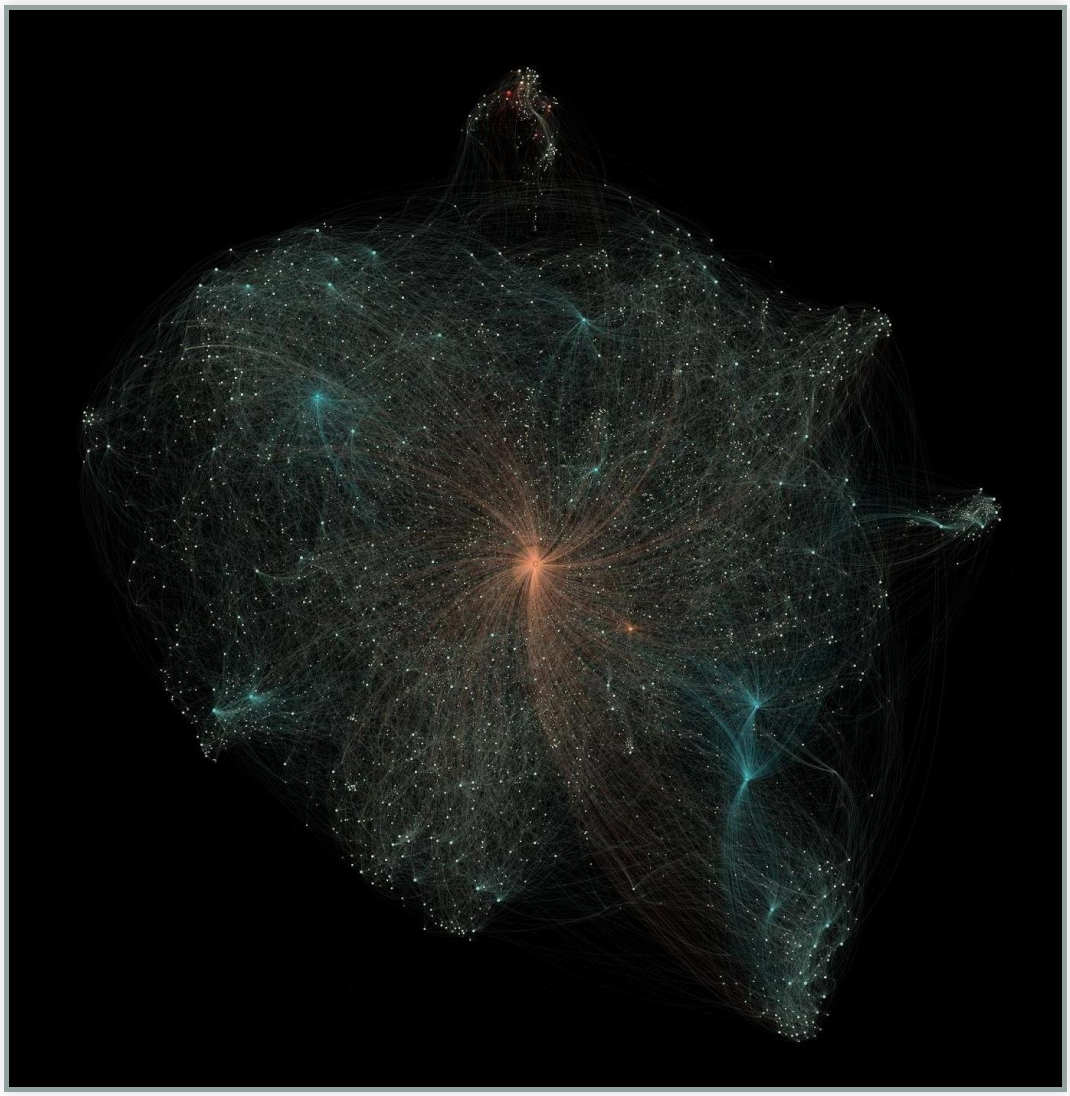
Fuzzing


american fuzzy lop 1.86b (test)
process timing
run time : 0 days, 0 hrs, 0 min, 2 sec
last new path : none seen yet
last uniq crash : 0 days, 0 hrs, 0 min, 2 sec
last uniq hang : none seen yet
overall results
cycles done : 0
total paths : 1
uniq crashes : 1
uniq hangs : 0
cycle progress
now processing : 0 (0.00%)
paths timed out : 0 (0.00%)
map coverage
map density : 2 (0.00%)
count coverage : 1.00 bits/tuple
stage progress
now trying : havoc
stage execs : 1464/5000 (29.28%)
total execs : 1697
exec speed : 626.5/sec
findings in depth
favored paths : 1 (100.00%)
new edges on : 1 (100.00%)
total crashes : 39 (1 unique)
total hangs : 0 (0 unique)
fuzzing strategy yields
bit flips : 0/16, 1/15, 0/13
byte flips : 0/2, 0/1, 0/0
arithmetics : 0/112, 0/25, 0/0
known ints : 0/10, 0/28, 0/0
dictionary : 0/0, 0/0, 0/0
havoc : 0/0, 0/0
trim : n/a, 0.00%
path geometry
levels : 1
pending : 1
pend fav : 1
own finds : 0
imported : n/a
variable : 0
[cpu: 92%]

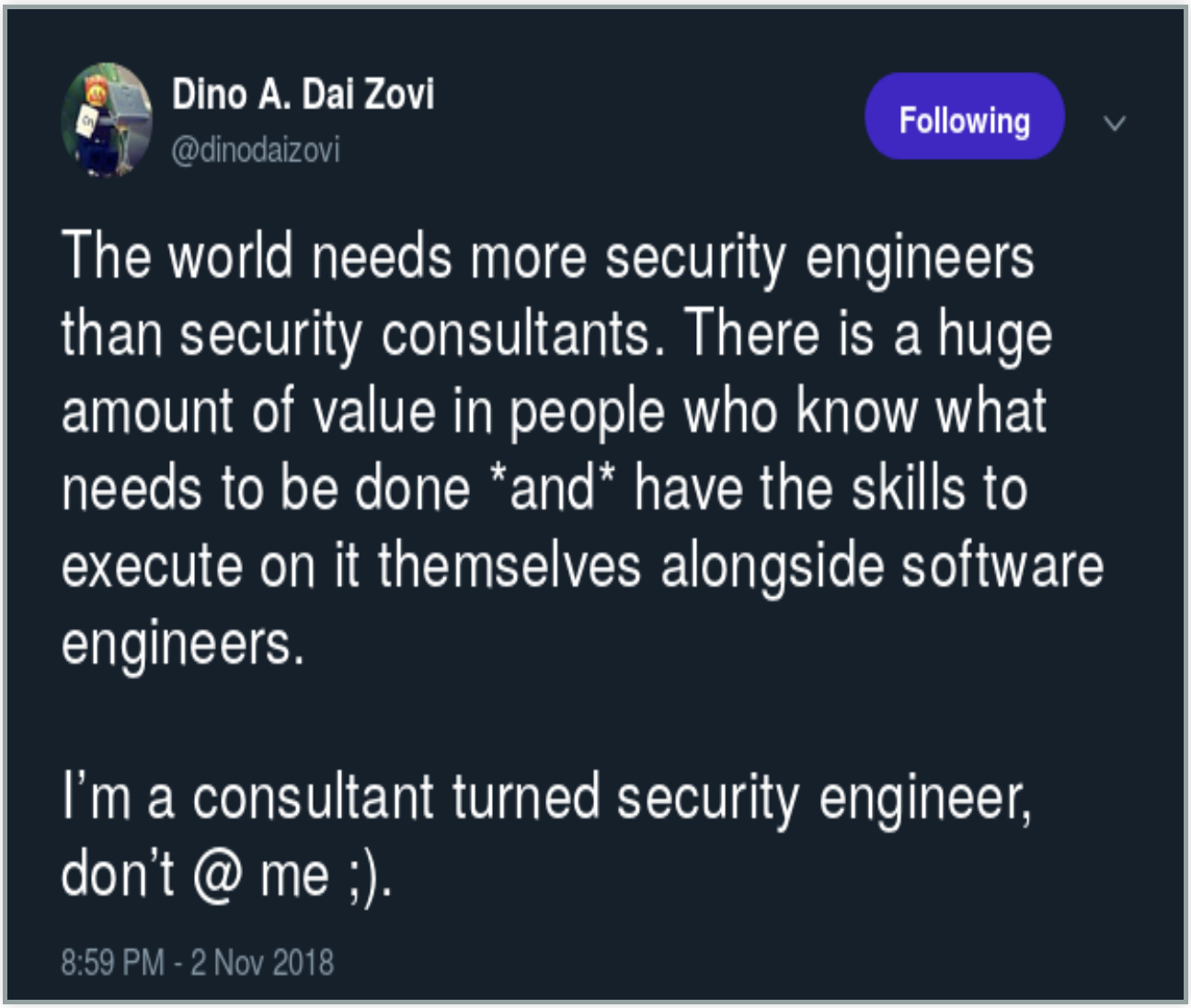
Cloud Fuzzing



Komponenten Teil 2



Engineers statt Consultants



Security Developer/Engineer

- Fuzzing benötig Security- & Development-Erfahrung
- (Interpreter/Compiler arbeiten auch nicht alleine)
- Rolle braucht Platz in Prozess / Team
- Prozess benötigt *Zeit*! – Gegensatz?
- Rolle kann Komponenten vor Integration testen
- Security Developer müssen Stimme haben!

DevOps ist keine Rolle!

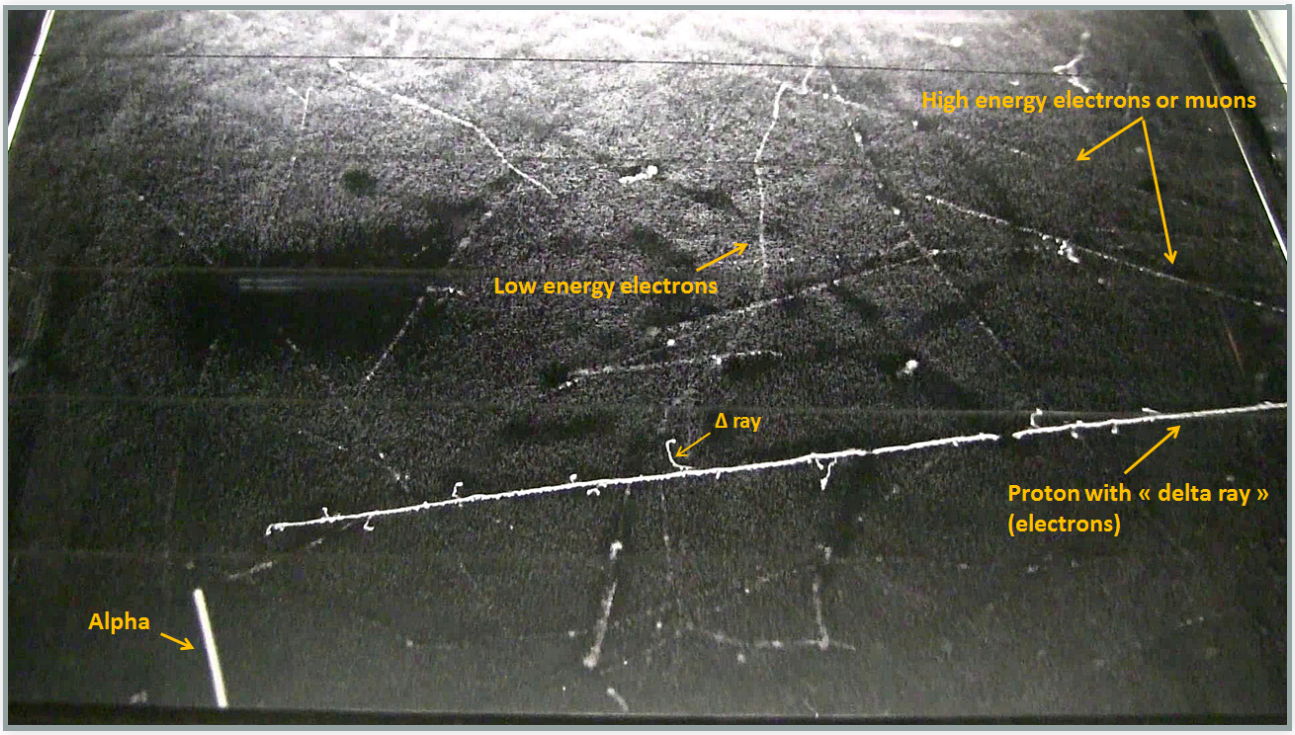
DevOps ist ein Konzept!

- Entwicklerinnen und Sysadmins arbeiten eng zusammen
- Applikation und Infrastruktur agieren als Team
- Jede(r) kennt Komponenten der anderen
- Automatisierung benötigt beide Seiten
- Sysadmins haben Programmierkenntnisse (zumindest in der UNIX® Welt)

Fazit

- Security muss in Entwicklungsprozesse eingebaut werden
  – als Rolle mit echten Menschen
- Security bedarf interdisziplinärer Spezialistinnen
- Secure Coding betrifft trotzdem alle Bereiche
- Architektur des Projekts muss passen – Secure Design!
- Continuous Integration kann ein ständiger Sicherheitstest werden
  double sell – heißer Tip für Controlling

Fragen?


High energy electrons or muons
Low energy electrons
Δ ray
Proton with « delta ray »
(electrons)
Alpha

Über die DeepSec

Die DeepSec GmbH veranstaltet seit 2007 jährlich im November die *DeepSec In-Depth Security Conference* in Wien. Die DeepSec bringt als neutrale Plattform die Sicherheitsexperten aus allen Bereichen zum Gedanken- und Erfahrungsaustausch zusammen. Dort erhalten IT- und Security-Unternehmen, Anwender, Behördenvertreter, Forscher und die Hacker-Community in über 42 Vorträgen und Workshops die Chance, sich über die aktuellen und zukünftigen Sicherheitsthemen auszutauschen. Die Konferenz möchte insbesondere dem verbreiteten Vorurteil entgegen wirken, dass Hacker zwangsläufig Kriminelle sind.

Über den Autor

René Pfeiffer ist selbstständiger Systemadministrator und Vortragender an der Fachhochschule Technikum Wien im Bereich Computer- und Datensicherheit. Mit über 15 Jahren Berufs- und 30 Jahren Computererfahrung sowie einem akademischen Hintergrund in theoretischer Physik verbindet er in Schulungen und Projekten erfolgreich Theorie und Praxis.

Seine Themenschwerpunkte liegen im Bereich IT Administration, Aufbau sicherer Infrastrukturen (Host-/Netzwerkbereich), sichere Kommunikation in Organisationen und Infrastruktur (VPN Technologien, Nachrichtensysteme), Wireless Security, technische Dokumentation, Betreuung von Forschungsarbeiten in der IT Security, technischem Auditing und Evaluierung von Software.

Herr Pfeiffer hält seit 2000 jährlich Fachvorträge und Schulungen auf Tagungen und Seminaren (Institute for International Research (I.I.R.), Business Circle, Confare, Linuxwochen Österreich, SAE Institute Wien, DeepSec In-Depth Security Conference, Bundesverband Sicherheitspolitik an Hochschulen, Kundenveranstaltungen).

Kontaktmöglichkeiten

- E-Mail: *rpfeiffer@deepsec.net*
- PGP/GPG: `0x518A0576C3A9FF76` / `0x49CE0CBEC210A5CE`
- Mobil: `+43.676.5626390` (Signal verfügbar)
- Threema: `4WFYBWCJ`