UNIVERSITÄT PADERBORN
Die Universität der Informationsgesellschaft

TECHNISCHE UNIVERSITÄT DARMSTADT

CROSSING

Funded by DFG

CamBench - Cryptographic API Misuse Detection Tool Benchmark

Michael Schlichtig, Anna-Katharina Wickert,
Stefan Krüger, Eric Bodden, Mira Mezini

Virtual MSR 2022

@akwickert

Zoom crypto misuse

63.7% applications ≥ 1 crypto misuse

Photo by Chris Montgomery on Unsplash

Photo by Maksim Goncharenok on Pexels

Comparison crypto misuse detectors difficult

Crypto Benchmarks Meets Analyses

Preliminary Study

Analysis capabilities | Fix complexity | Real-world usages | Exploits

Benchmark	A	AC	CC	CG	CO	DN	FD	FS	J	MA	MF	P	SB	SQ	V	X	Y	Z
CryptoAPI-Bench [6]	○	○	●	●	●	○	○	○	○	○	○	○	●	○	○	○	○	○
Braga Benchmark [13]	○	○	○	○	○	○	○	●	○	○	○	○	○	●	●	●	●	○
Parametric Crypto Misuse Benchmark [51]	○	○	○	○	○	○	○	●	○	○	○	○	○	○	○	○	○	○
MuBench [8]	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○
Ghera Benchmark [31]	●	●	○	○	○	●	●	○	○	●	●	○	○	○	○	○	○	○
OWASP Benchmark [49]	○	○	○	○	○	○	○	●	●	○	○	●	○	●	○	○	○	●

Crypto Benchmarks Meets Analyses

Preliminary Study

Benchmark	A	AC	CC	CG	CO	DN	FD	FS	J	MA	MF	P	SB	SQ	V	X	Y	Z
CryptoAPI-Bench [6]	○	○	●	●	●	○	○	○	○	○	○	○	●	○	○	○	○	○
Braga Benchmark [13]	○	○	○	○	○	○	○	●	○	○	○	○	○	●	●	●	●	○
Parametric Crypto Misuse Benchmark [51]	○	○	○	○	○	○	○	●	○	○	○	○	○	○	○	○	○	○
MuBench [8]	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○
Ghera Benchmark [31]	●	●	○	○	○	●	●	○	○	●	●	○	○	○	○	○	○	○
OWASP Benchmark [49]	○	○	○	○	○	○	○	●	●	○	○	●	○	●	○	○	○	●

Only 11 % of the analyses were evaluated on more than one benchmark

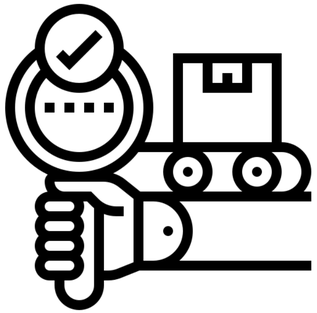
Design of CamBench

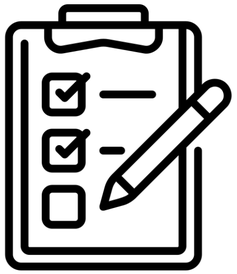
Real-world applications

- Mine Java applications on GitHub
- Compile projects with JCA
- Label usages manually

Analysis Capabilities

- Synthetic test cases
- Cover flow-, context-, field-, object-, and path-sensitivity





Heuristic for crypto API coverage

Harness both benchmarks

Application icons created by Eucalyp - Flaticon | Capability icons created by Eucalyp - Flaticon | Checklist icons created by Freepik - Flaticon

Comparison crypto misuse detectors difficult

Crypto Benchmarks Meets Analyses

Preliminary Study

Benchmark	A	AC	CC	CG	CO	DN	FD	FS	J	MA	MF	P	SB	SQ	V	X	Y	Z
CryptoAPI-Bench [6]	○	○	●	●	●	○	○	○	○	○	○	○	●	○	○	○	○	○
Braga Benchmark [13]	○	○	○	○	○	○	○	●	○	○	○	○	○	●	●	●	●	○
Parametric Crypto Misuse Benchmark [51]	○	○	○	○	○	○	○	●	○	○	○	○	○	○	○	○	○	○
MuBench [8]	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○
Ghera Benchmark [31]	●	●	○	○	○	●	●	○	○	●	●	○	○	○	○	○	○	○
OWASP Benchmark [49]	○	○	○	○	○	○	○	●	●	○	○	●	○	●	○	○	○	●

Only 11 % of the analyses were evaluated on more than one benchmark

Our talk in one slide :)

Design of CamBench

Real-world applications
- Mine Java applications on GitHub
- Compile projects with JCA
- Label usages manually

Analysis Capabilities
- Synthetic test cases
- Cover flow-, context-, field-, object-, and path-sensitivity

Heuristic for crypto API coverage
Harness both benchmarks

https://github.com/CROSSINGTUD/CamBench

michael.schlichtig@uni- paderborn.de
@M_Schlichtig

wickert@cs.tu-darmstadt.de
@akwickert