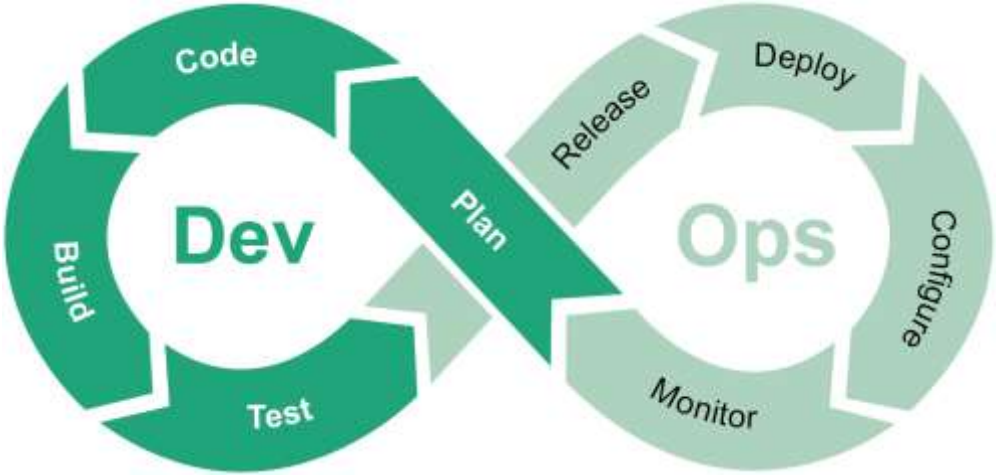
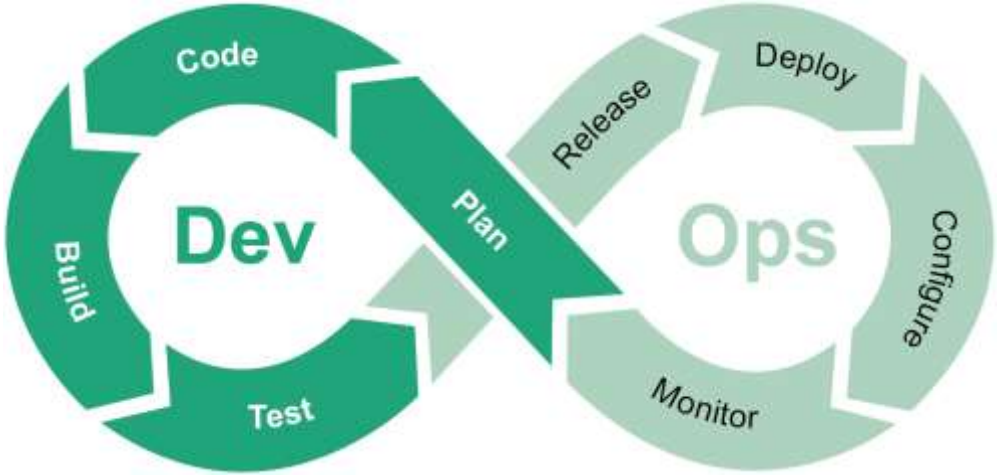
# ENABLING DEVSECOPS WITH THE NEW GENERATION OF STATIC ANALYSIS

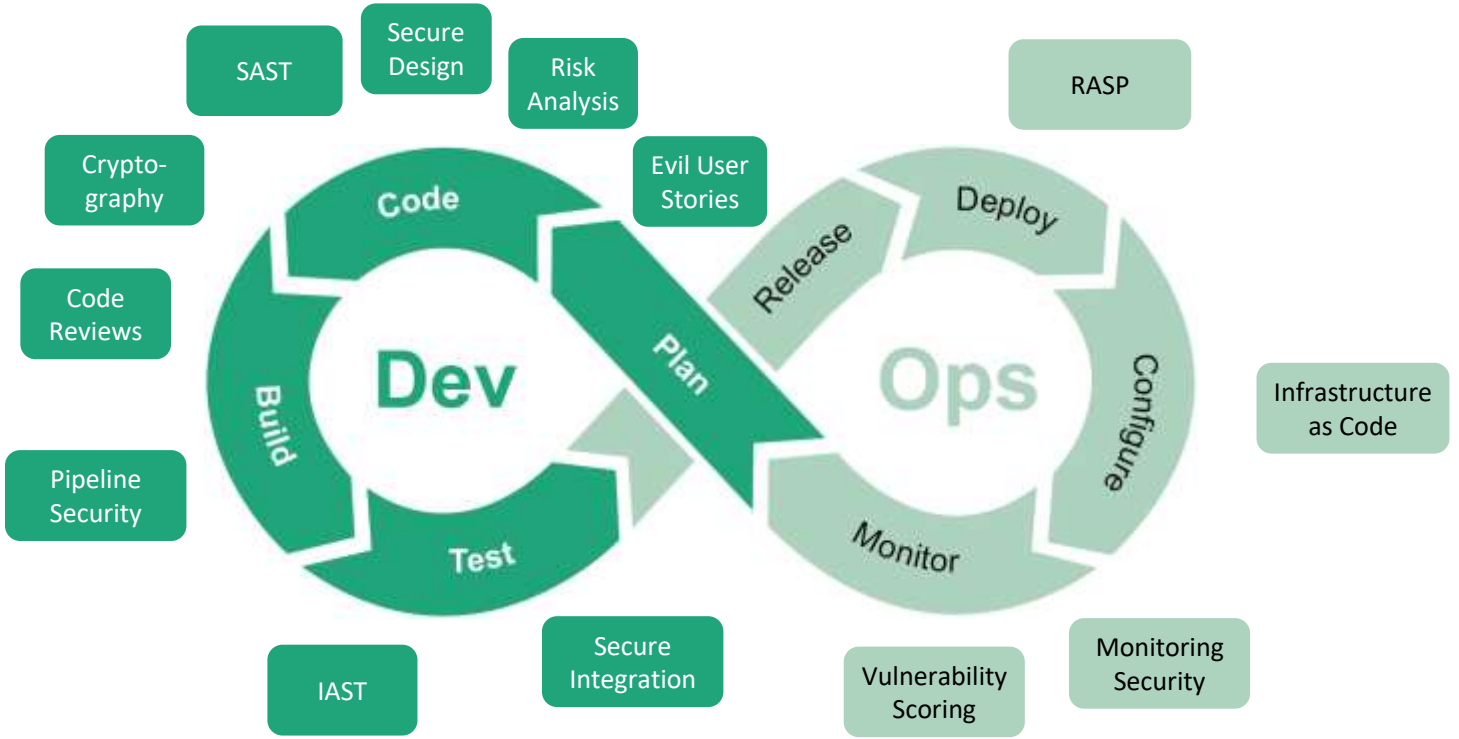
Prof. Dr. Eric Bodden



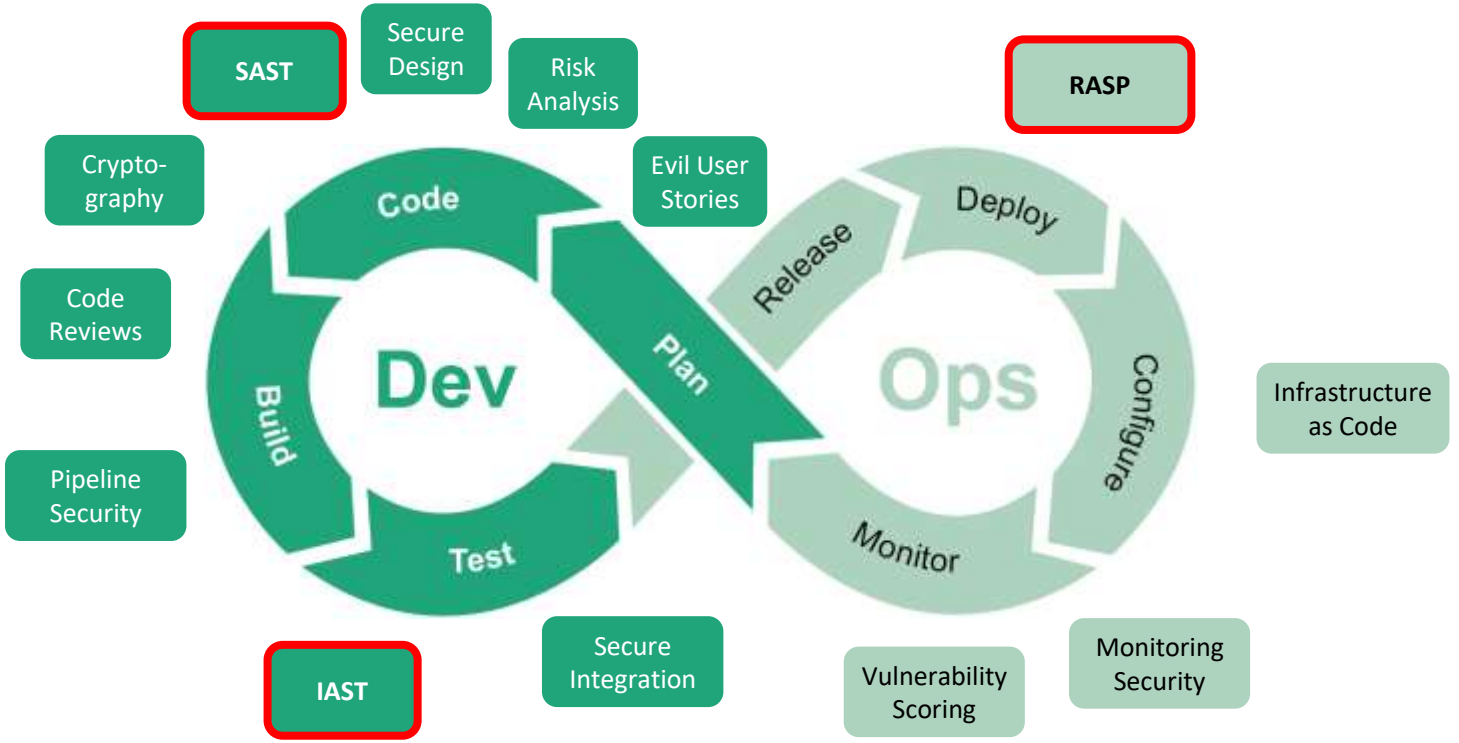
# DevOps


Code
Plan
Release
Deploy
Dev
Ops
Build
Test
Monitor
Configure

# Security touchpoints in DevSecOps



# Focus here: tool automation for security testing






HEINZ NIXDORF INSTITUT
UNIVERSITÄT PADERBORN
Fraunhofer IEM

# SAST vs. IAST vs. RASP

**Static Application Security Testing**

- Analyzes program code without executing it
- Can cover entire program, not just a single run
- Works on incomplete code
- White-box technique:
  - Can give immediate feedback to developer
  - Can instantly suggest sensible fixes
- But: traditionally many false warnings

**Interactive Application Security Testing**

- Analyzes program code during execution, alerting the developer about detected vulnerabilities
- Requires executable code
- Often too expensive for deployment / ops
- Good: Can be very precise, little to no false warnings

**Runtime Application Self-Protection**

- Analyzes program code during execution, automatically mitigating vulnerabilities
- Has to be efficient, thus often restricted to efficiently recognizable properties
- Sensible to use as an additional layer of defense

# At Fraunhofer IEM we focus on SAST, i.e., static analysis

**Static Application Security Testing**

- Analyzes program code without executing it
- Can cover entire program, not just a single run
- Works on incomplete code
- White-box technique:
  - Can give immediate feedback to developer
  - Can instantly suggest sensible fixes
- But: traditionally many false warnings

# At Fraunhofer IEM we focus on SAST, i.e., static analysis

**Static Application Security Testing**

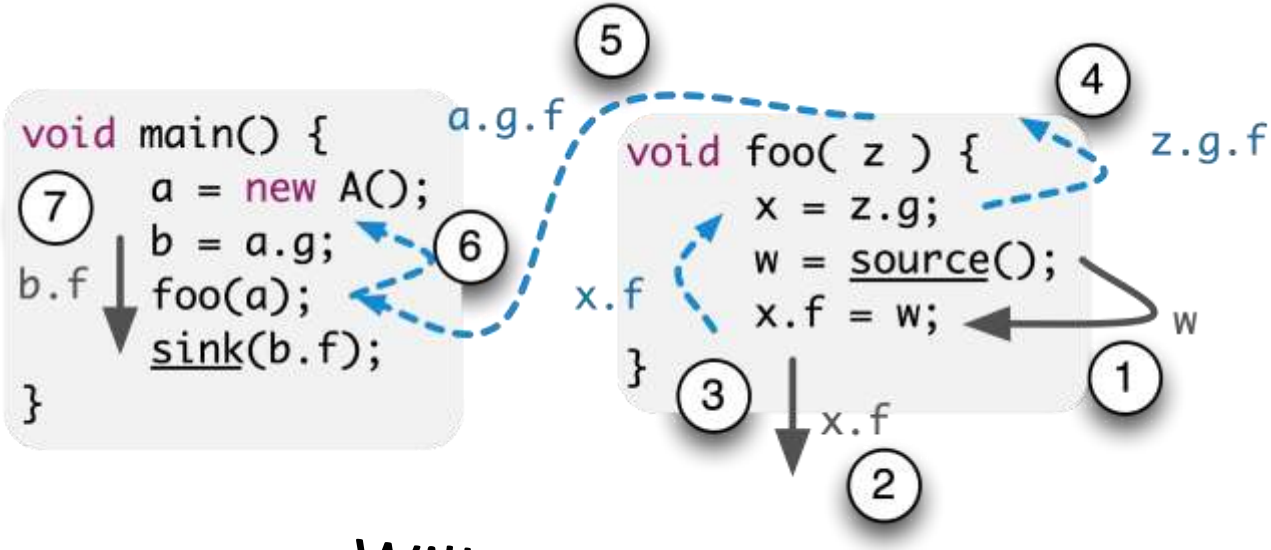
- Analyzes program code without executing it
- Can cover entire program, not just a single run
- Works on incomplete code
- White-box technique:
  - Can give immediate feedback to developer
  - Can instantly suggest sensible fixes
- **But: traditionally many false warnings**

Why is precise static analysis so challenging?

# 1st Major challenge: Heap assignments and “Aliasing”

```
String secret = secret();
```

```
print(output);
```


void main() {
a = new A();
b = a.g;
foo(a);
sink(b.f);
}
void foo( z ) {
x = z.g;
w = source();
x.f = w;
}
a.g.f
z.g.f
x.f
b.f
x.f
w
1
2
3
4
5
6
7

Will it leak?

# 2nd Major challenge: Field-sensitivity

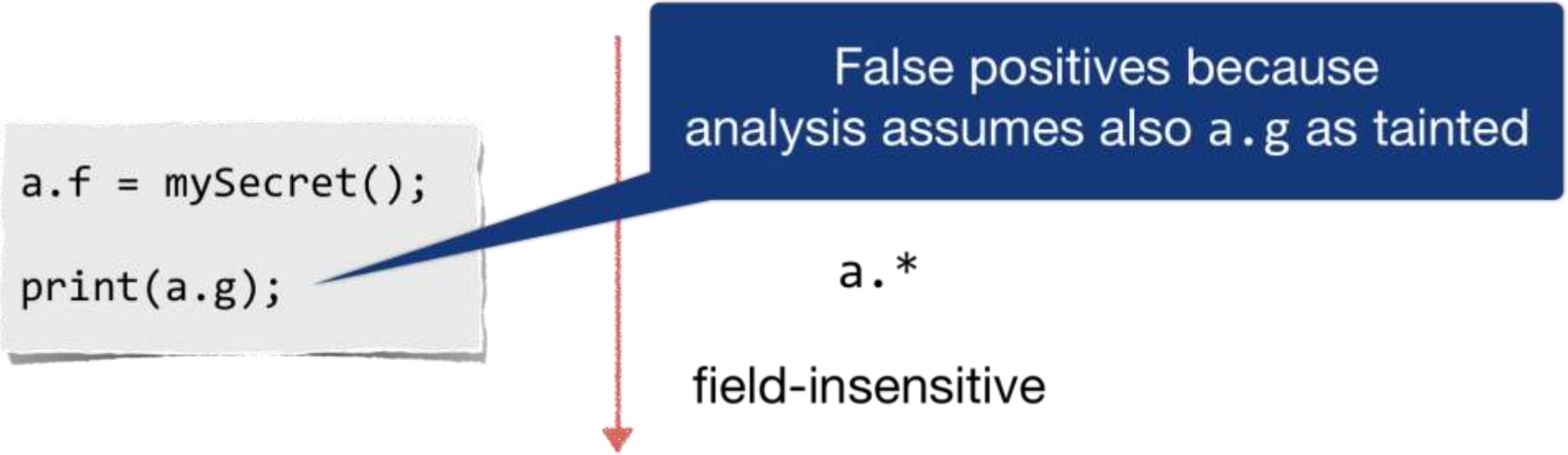
```
a.f = mySecret();

print(a.g);
```

can represent `a.f` as:

`a.*`

field-insensitive

# 2nd Major challenge: Field-sensitivity



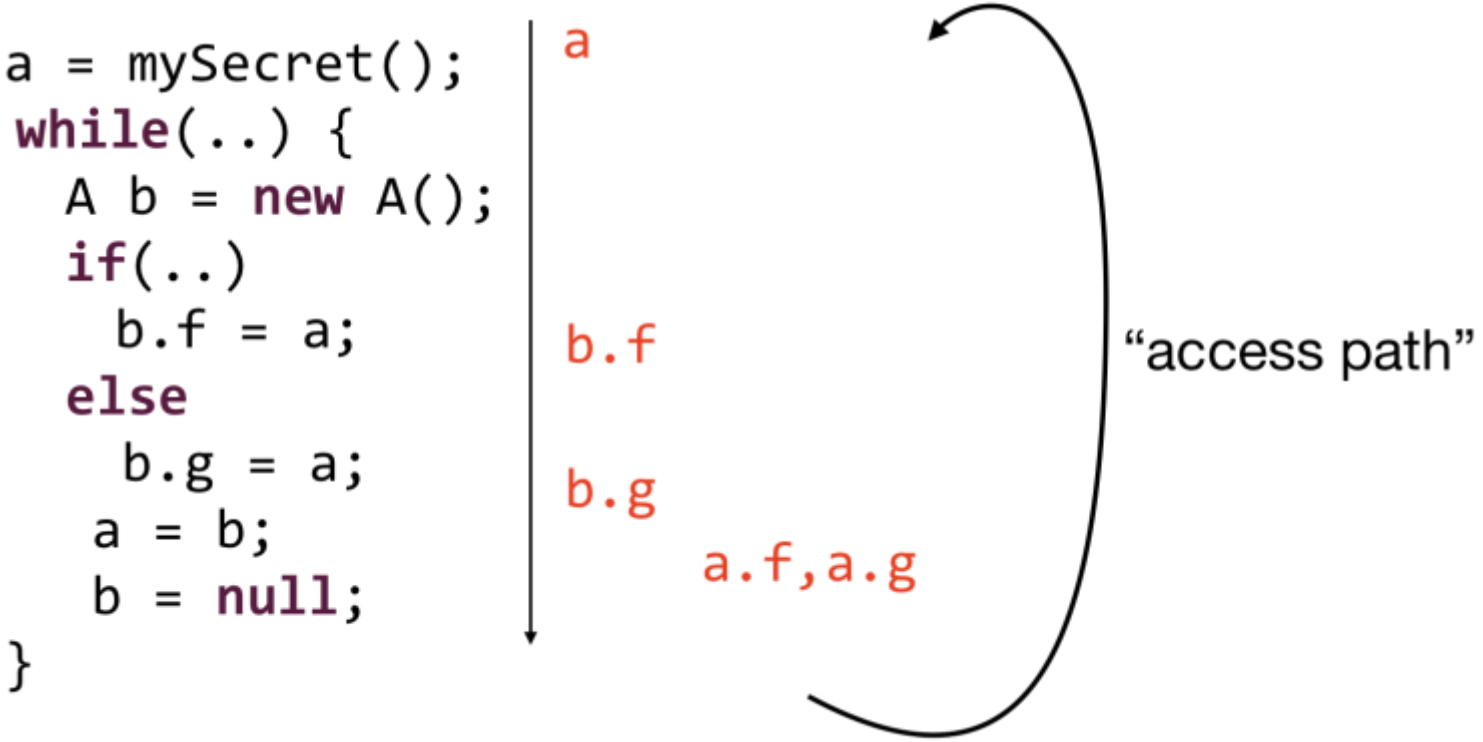
# 2nd Major challenge: Field-sensitivity

```
a.f = mySecret();

print(a.g);
```

can represent `a.f` as:

| `a.*` | `a.f` |
| --- | --- |
| field-insensitive | field-sensitive |

# 2nd Major challenge: Field-sensitivity



# 2nd Major challenge: Field-sensitivity

require finite representation!

```
a = mySecret();
while(..) {
  A b = new A();
  if(..)
    b.f = a;
  else
    b.g = a;
   a = b;
   b = null;
}
```

**Using novel algorithms we are now able to analyze program code with almost no loss of precision**

**In large-scale empirical evaluations: false-positive rates below 5%!**

Context-, Flow-, and Field-Sensitive Data-Flow Analysis using Synchronized Pushdown Systems

JOHANNES SPÄTH, Fraunhofer IEM, Germany
KARIM ALI, University of Alberta, Canada
ERIC BODDEN, Heinz Nixdorf Institut, Universität Paderborn and Fraunhofer IEM, Germany

Precise static analyses are context-, field- and flow-sensitive. Context- and field-sensitivity are both expressible as context-free language (CFL)-reachability problems. Solving both CFL problems along the same data-flow path is undecidable, which is why most flow-sensitive data-flow analyses over-approximate *field*-sensitivity through *k*-limited access-path, or through access graphs. Unfortunately, as our experience and this paper show, both representations do not scale very well when used to analyze programs with recursive data structures.

Any single-CFL-reachability problem is efficiently solvable, by means of a pushdown system. This work thus introduces the concept of *synchronized pushdown systems* (SPDS). SPDS encode both procedure calls/returns and field stores/loads as separate but "synchronized" CFL-reachability problems. An SPDS solves both individual problems precisely, and approximation occurs only in corner cases that are apparently rare in practice: at statements where both problems are satisfied but not along the same data-flow path.

SPDS are also efficient: formal complexity analysis shows that SPDS shift the complexity from $|\mathbb{P}|^{k+1}$ under *k*-limiting to $|\mathbb{E}|\cdot|\mathbb{P}|^2$, where $\mathbb{P}$ is the set of fields and $\mathbb{E}$ the set of statements involved in a data-flow. Our evaluation using DaCapo shows this shift to pay off in practice: SPDS are almost as efficient as *k*-limiting with $k = 1$ although their precision equals $k = \infty$. For a typestate analysis SPDS accelerate the analysis up to 83× for data-flows of objects that involve many field accesses but span rather few methods.

We conclude that SPDS can provide high precision and further improve scalability, in particularly when used in analyses that expose rather local data flows.

CCS Concepts: • **Theory of computation** → **Program analysis**; *Object oriented constructs*; Grammars and context-free languages;

Additional Key Words and Phrases: static analysis, data-flow, aliasing, access paths, pushdown system

**ACM Reference Format:**
Johannes Späth, Karim Ali, and Eric Bodden. 2019. Context-, Flow-, and Field-Sensitive Data-Flow Analysis using Synchronized Pushdown Systems. *Proc. ACM Program. Lang.* 3, POPL, Article 48 (January 2019), 29 pages. https://doi.org/10.1145/3290361

1 INTRODUCTION

Static data-flow analysis helps detect bugs and security vulnerabilities early in the software development process, including semantic properties such as null-pointers [Nanda and Sinha 2009], data races [Kahlon et al. 2009; Yan et al. 2011], and misuses of application programming interfaces

Authors' addresses: Johannes Späth, Fraunhofer IEM, Germany, johannes.spaeth@iem.fraunhofer.de; Karim Ali, University of Alberta, Canada, karim.ali@ualberta.ca; Eric Bodden, Heinz Nixdorf Institut, Universität Paderborn and Fraunhofer IEM, Germany, eric.bodden@uni-paderborn.de.


2475-1421/2019/1-ART48
https://doi.org/10.1145/3290361



[POPL 2019] ACM Distinguished Paper



HEINZ NIXDORF INSTITUT
UNIVERSITÄT PADERBORN
Fraunhofer IEM





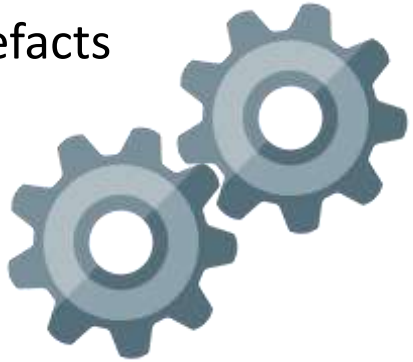

available at http://eclipse.org/cognicrypt/

# Analysis of MavenCentral

Maven™

The Central Repository

2.7+ Million Software Artefacts



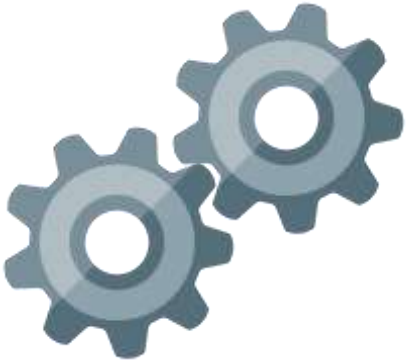
73% of all artefacts use crypto in an insecure manner!

[A Large-Scale Study of Non-Trivial Misuses of the Java Cryptography Architecture. Stefan Krüger Johannes Spaeth, Karim Ali, Eric Bodden, Mira Mezini.]

# Analysis of Google Play: top banking/health/password apps

250

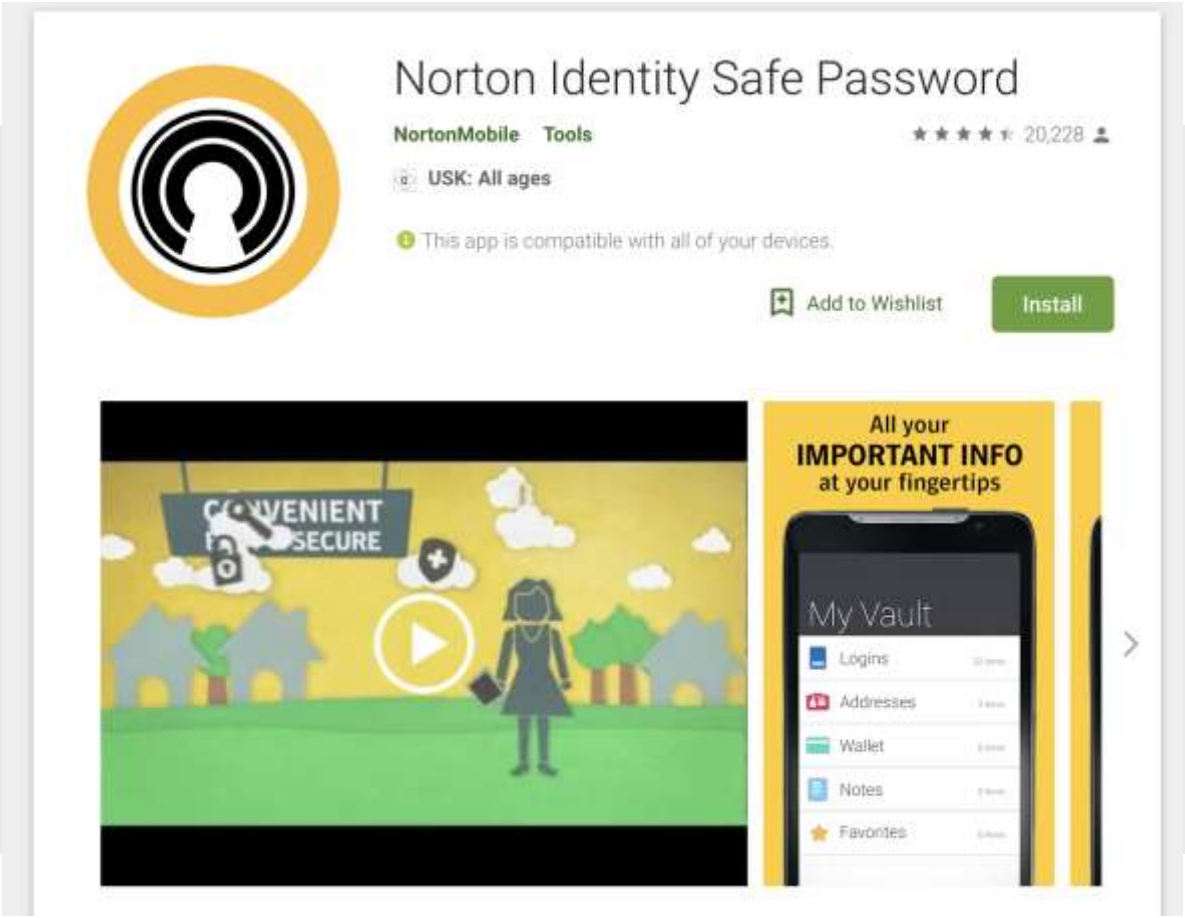
Apps



71%

of all apps use crypto in an insecure manner!

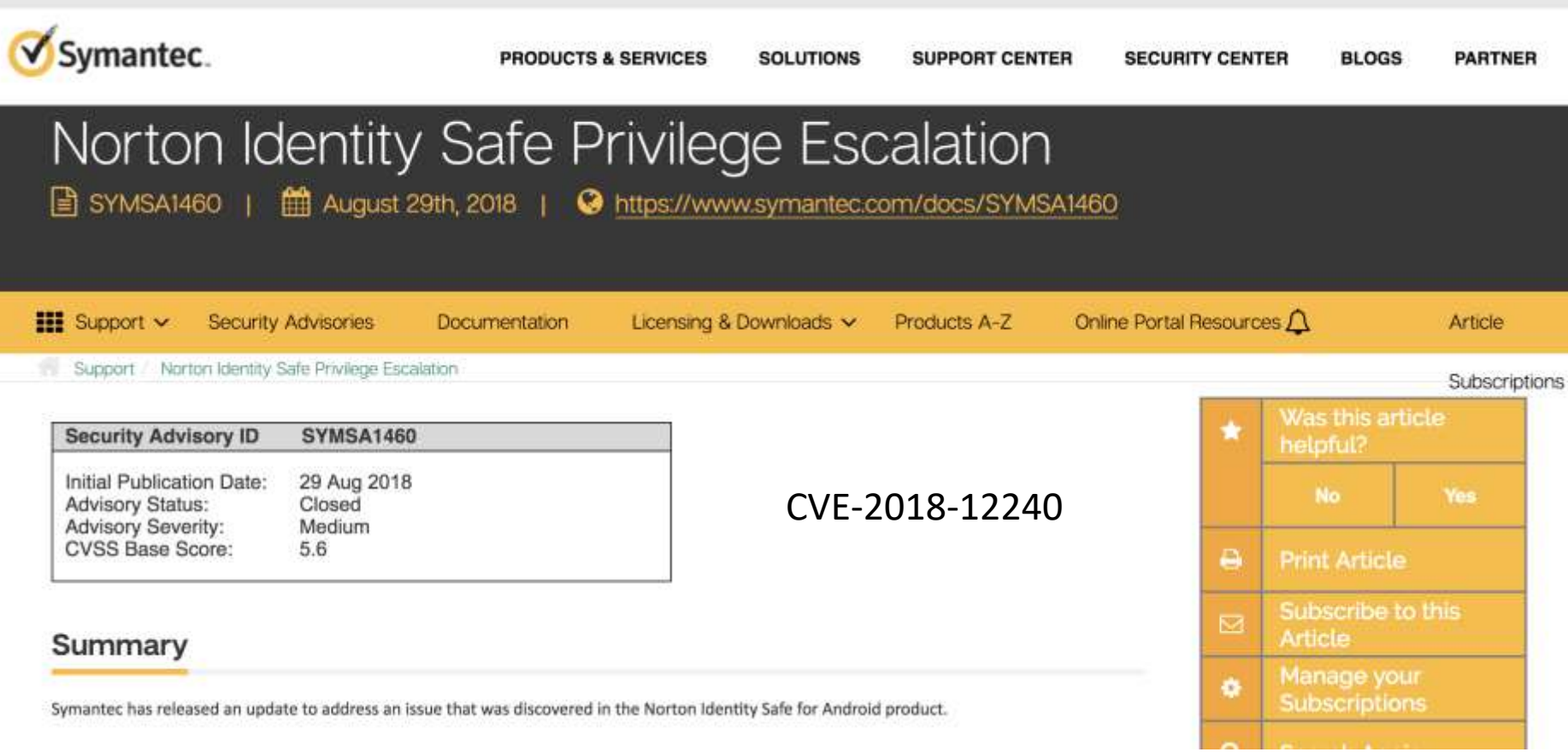
[A Large-Scale Study of Non-Trivial Misuses of the Java Cryptography Architecture. Stefan Krüger Johannes Spaeth, Karim Ali, Eric Bodden, Mira Mezini.]




One of
many examples:

Norton Identity Safe

> 500.000 Downloads in
Google Play


Symantec
PRODUCTS & SERVICES
SOLUTIONS
SUPPORT CENTER
SECURITY CENTER
BLOGS
PARTNER
Norton Identity Safe Privilege Escalation
SYMSA1460 | August 29th, 2018 | https://www.symantec.com/docs/SYMSA1460
Support
Security Advisories
Documentation
Licensing & Downloads
Products A-Z
Online Portal Resources
Article
Support / Norton Identity Safe Privilege Escalation
Subscriptions
Security Advisory ID SYMSA1460
Initial Publication Date: 29 Aug 2018
Advisory Status: Closed
Advisory Severity: Medium
CVSS Base Score: 5.6
CVE-2018-12240
Was this article helpful?
No
Yes
Print Article
Subscribe to this Article
Manage your Subscriptions
Summary
Symantec has released an update to address an issue that was discovered in the Norton Identity Safe for Android product.

# VR-Banking

**Fiducia & GAD IT AG** **Finanzen**

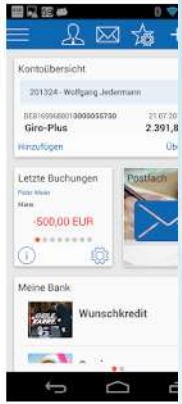
★★★½☆ 19.702

USK ab 0 Jahren

Diese App ist mit allen deinen Geräten kompatibel.

Zur Wunschliste hinzufüger






Ihre Finanzen im Blick – immer und überall mit der VR-Banking App der Volksbank
Raiffeisenbanken.

Sie möchten unterwegs mal eben den Kontostand abrufen, kurz die letzten Umsät
prüfen, eine dringende Überweisung erledigen, Börseninformationen einholen und
Kein Problem mit der VR-Banking App. Besonders praktisch: legen Sie Ihre beliebt

WEITERE INFORMATIONEN

## Zertifiziertes Produkt

Geprüfte Online-Applikation:
Zertifikatsinhaber: Fiducia & GAD IT AG
Prüfzeichennummer: 0000043889

TÜVRheinland ZERTIFIZIERT

Datenschutz/ Datensicherheit

www.tuv.com
ID 0000043889

Die Prüfung umfasst:
- Datenschutz/ Datensicherheit

FIDUCIA GAD ZUKUNFTSERFAHREN

### Informationen

**Beschreibung:**

Für die Online-Banking-Applikationen eBanking Private Edition und eBanking Business Edition sowie die VR-BankingApp und alle auf deren Basis individualisierten Banking-Apps und Applikationen (siehe Service Information) hat die Fiducia & GAD IT AG einen wirksamen Prozess zur Erreichung folgender Ziele etabliert:

- Vertraulichkeit und Integrität der verarbeiteten Informationen
- Wirksame Umsetzung der Aussagen der Datenschutzerklärung
- Wirksamer Schutz der personenbezogenen Daten gemäß anwendbarer, aktueller Datenschutzgesetzgebung
- Wirksame Absicherung der von außen zugänglichen technischen Systeme gegen unbefugte Nutzung

Zusätzlich wurde geprüft, ob die zur Autorisierung der Banking-Transaktionen genutzten Verfahren mobileTAN und Sm@rt-TAN nach Best-Practices implementiert wurden.

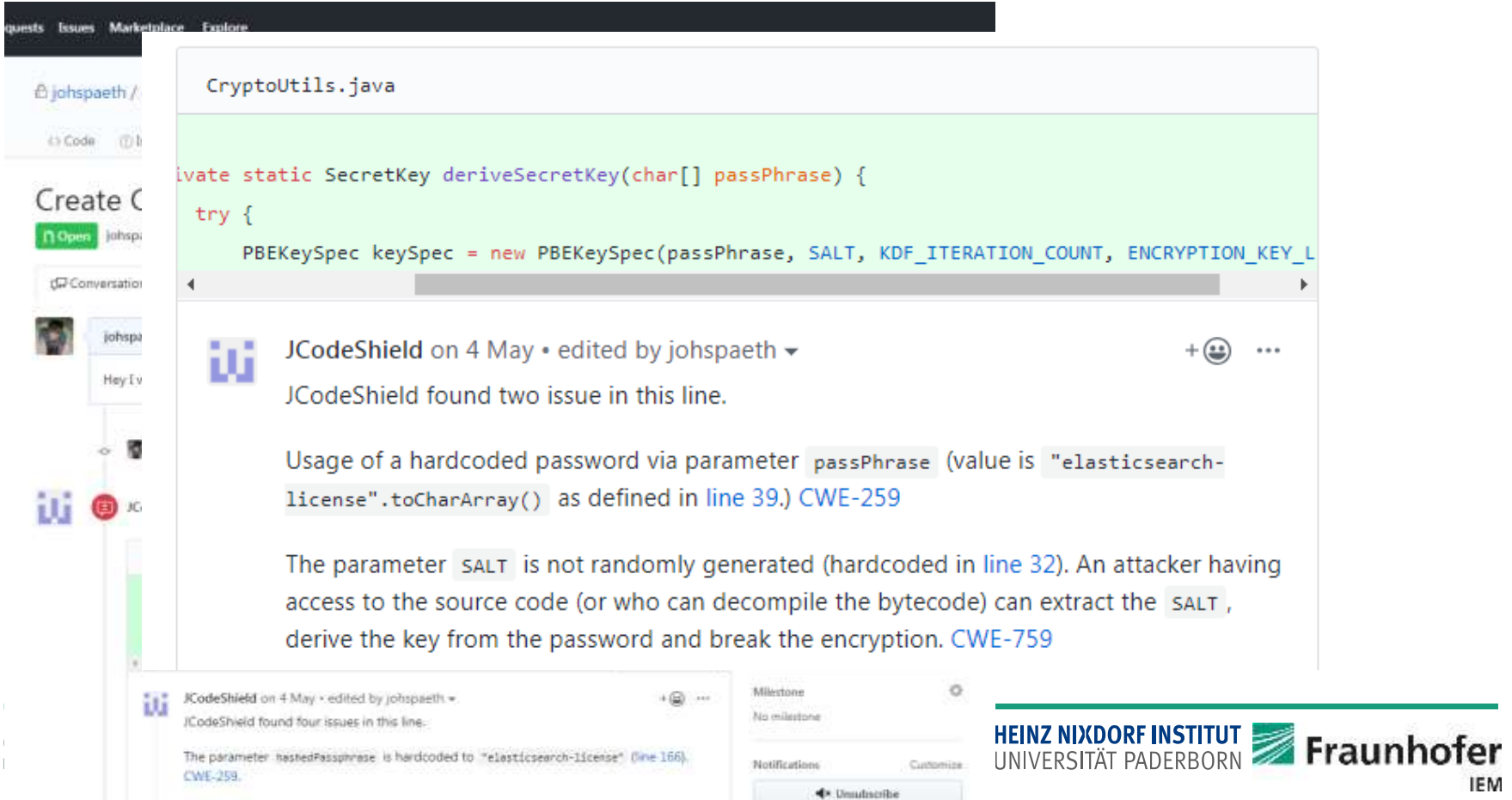
Der Nachweis wurde durch ein Datenschutzaudit sowie externe und interne Sicherheitsanalysen erbracht. Der Prüfbericht Nr. 63008709-01 in der aktuellen Version ist Bestandteil dieses Zertifikats.

Die Wirksamkeit des geprüften Prozesses wird durch die TÜV Rheinland i-sec GmbH regelmäßig überwacht.

Das Zertifikat basiert auf einem von der TÜV Rheinland i-sec GmbH entwickelten Anforderungskatalog und stellt kein akkreditiertes Zertifizierungsverfahren, Siegel oder Prüfzeichen im Sinne der Art. 42, 43 der Verordnung (EU) 2016/679 (Datenschutz-Grundverordnung) dar.

**Dieses Zertifikat ist gültig bis 13.12.2020.**

# Integrates with your favourite CI-environment




HEINZ NIXDORF INSTITUT
UNIVERSITÄT PADERBORN
Fraunhofer IEM

# Visit us at Hall 10, Booth 311

```
public class Msg {
    public byte[] sign(String data) throws InvalidKeyException, Exception {
        return null;
    }
}
```