UNIVERSITÄT
BERN

Detecting Potentially Malicious Behavior in Mobile Apps

with Static Code Analysis on Android OS

Pascal Gadient

AS2016 – Seminar Software Composition
University of Bern

Outline

1

Introduction

1.1 Introduction - **Problem Statement**

- Software gets more complex
  - Large Android OS fragmentation
  - Many security issues in media / web frameworks
  - Many security issues in HW drivers
  - Evolving «Freemium» apps

- «Malware» omnipresent on today's devices
  - User tracking
  - Collection of personal data
  - Advertisement SDKs
  - Exploitation of premium SMS / voice numbers
  - Phishing

1.1 Introduction - **Problem Statement**

u^b UNIVERSITÄT BERN

➔ We need to explore new approaches:

Static **C**ode **A**nalysis **F**eature **M**odel **E**valuation

(SCAFME)

1.2 Introduction - **Concept**

UNIVERSITÄT BERN

- Static Code Analysis
  - Applied on Android apps with FlowDroid[1]

- Feature Model
  - Methods / Classes
  - Permissions
  - ...

- Evaluation
  - SVM training and application in R

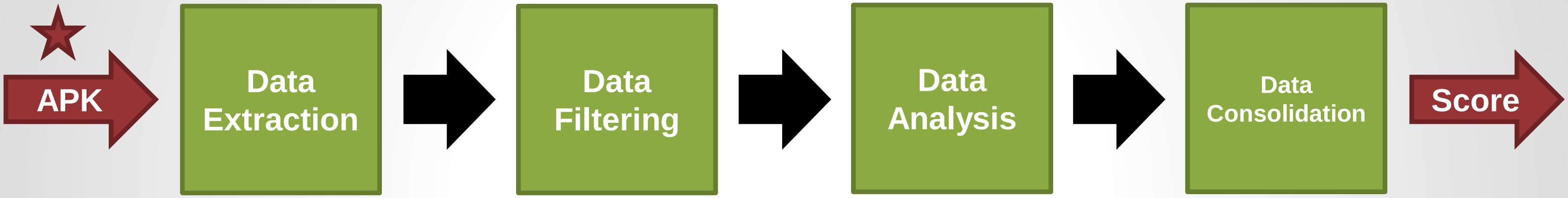
2

Workflows

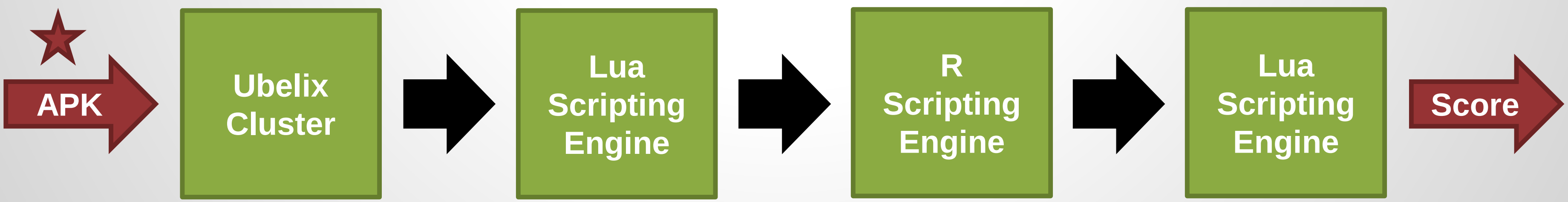
2.1 Workflows - **Overview**

UNIVERSITÄT BERN

- Conceptual view



- Technical view



2.2 Workflows - **Virus Archives** ★

- Encrypted, renamed and multi-platform

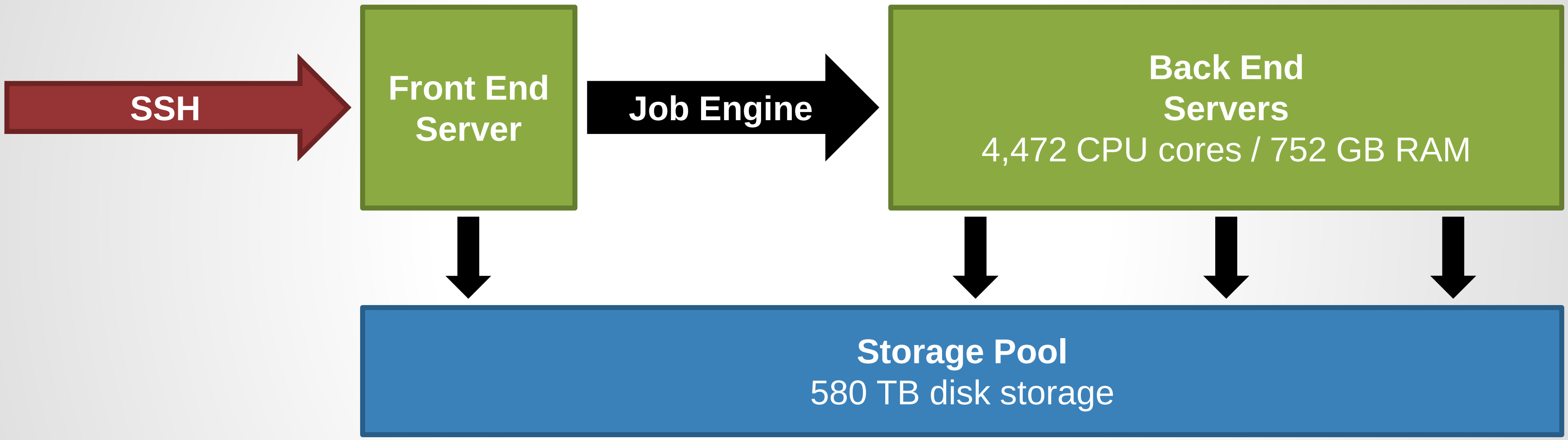
- We need pre-processing of:
  - ZIP headers (`80 75 03 04`)
  - ZIP integrity (TOC validity)
  - Folder structure (manifests)

- MALWARE IS DANGEROUS!

```
ubuntuNBK-VIRUSSHARE-EDU:~$ lua apkMalwareFilter.lua
Bus error (core dumped)
ubuntuNBK-VIRUSSHARE-EDU:~$
```

2.3 Workflows - **Data Extraction**

❖ System overview



❖ Quirks
- Job management
- Linux environment (Quotas / Software)

2.4 Workflows - **Data Filtering**

- Log parsing
  - Feature extraction
  - Feature selection
  - Data conversion into ORCA[2] format
  - ORCA outlier removal
- CSV file creation

2.5 Workflows - **Data Analysis**

- Analysis runs in R
  - Creation / execution of R scripts

- SVM algorithm from library "e1071"
  - Two configurations used[3]

2.6 Workflows - **Data Consolidation**

UNIVERSITÄT BERN

- Parsing of R results
  - Weighting of results
  - Calculation of final scores

3

Measures

3.1 Measures - **Android Package Files**

- AndroidManifest.xml
  - Activity classes
  - Receiver classes
  - Services classes
  - App permissions

- apktool.yml
  - Required SDK version
  - Targeted SDK version
  (needs backwards compatibility in code)

3.2 Measures - **Flows**

❖ Call flow sequence diagram

Source Class : Source Method

`PhoneManager : getDeviceId()`

...

...

...

Sink Class : Sink Method

`URLSocket : sendData()`

3.3 Measures - **SuSi Categories**

UNIVERSITÄT BERN

- SuSi[4]
  - Software developed by Steven Arzt et al.
  - Categorization of flows
  - Supervised machine learning approach
  - Currently 31 categories available

- Examples
  UNIQUE_IDENTIFIER, NETWORK_INFORMATION, SMS_MMS, EMAIL, BLUETOOTH_INFORMATION, NFC, ...

4

Evaluation

4.1 Evaluation - **Computational Complexity**

UNIVERSITÄT BERN

- Static analysis suffers combinatorial explosion
- Testbed configuration:
  - 8 CPU cores (x64)
  - 80 GB RAM
  - 3 hours runtime

- Still insufficient memory and CPU

```
# There is insufficient memory for the Java Runtime Environment to continue.
# Native memory allocation (malloc) failed to allocate 4088 bytes for AllocateHeap
# Possible reasons:
#   The system is out of physical RAM or swap space
#   In 32 bit mode, the process size limit was hit
```

4.2 Evaluation - **Data Set**

UNIVERSITÄT BERN

- Ubelix
  - Data generated ourselves
  - Includes 182 benign apps
  - Includes 1,131 malign apps

- MudFlow
  - Data used from existing data set
  - Includes 2,800 benign apps
  - Includes 15,097 malign apps

4.3 Evaluation - **Data Set Quality**

UNIVERSITÄT BERN

- Ubelix
  - High-quality settings
  - No speed hacks
  - Slow (> 3 hours per file)

- MudFlow
  - Low-quality settings
  - Application of speed hacks
  - Fast (~ 15 minutes per file)

4.4 Evaluation - **SVM Parameters**

UNIVERSITÄT BERN

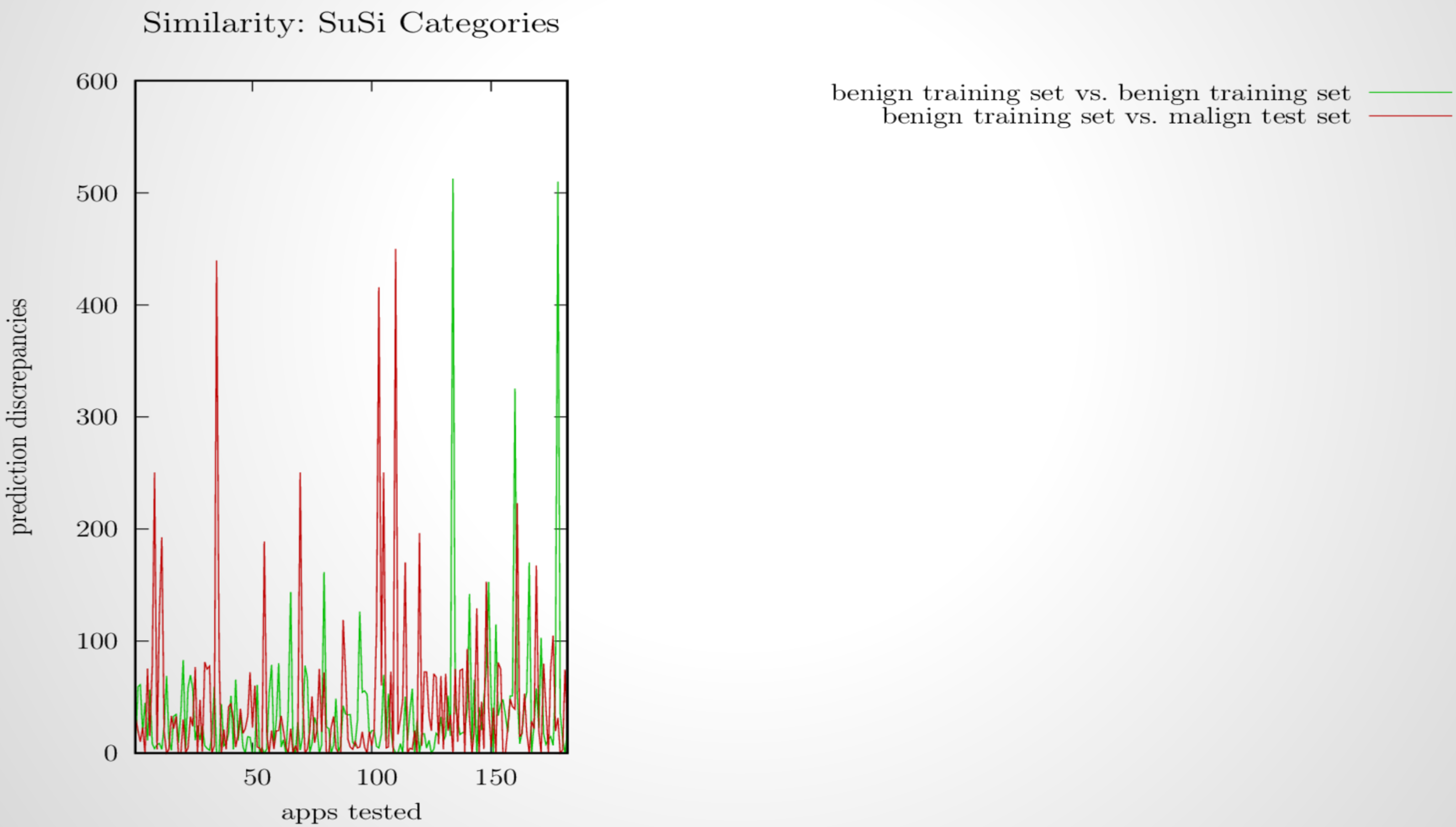
- Method "eps-regression"
  - Traditional regression based model
  - Determined best epsilon by 10-fold crossvalidation
  - Benign apps as training set (supervised)
  - Predicts each SuSi category count
  - Comparison of measured and predicted value

- Method "one-classifier"
  - Traditional classification model for novelty detection
  - Settings from MudFlow
  - Benign apps as training set (supervised)
  - Classifies measured values into "similar" or "not similar"

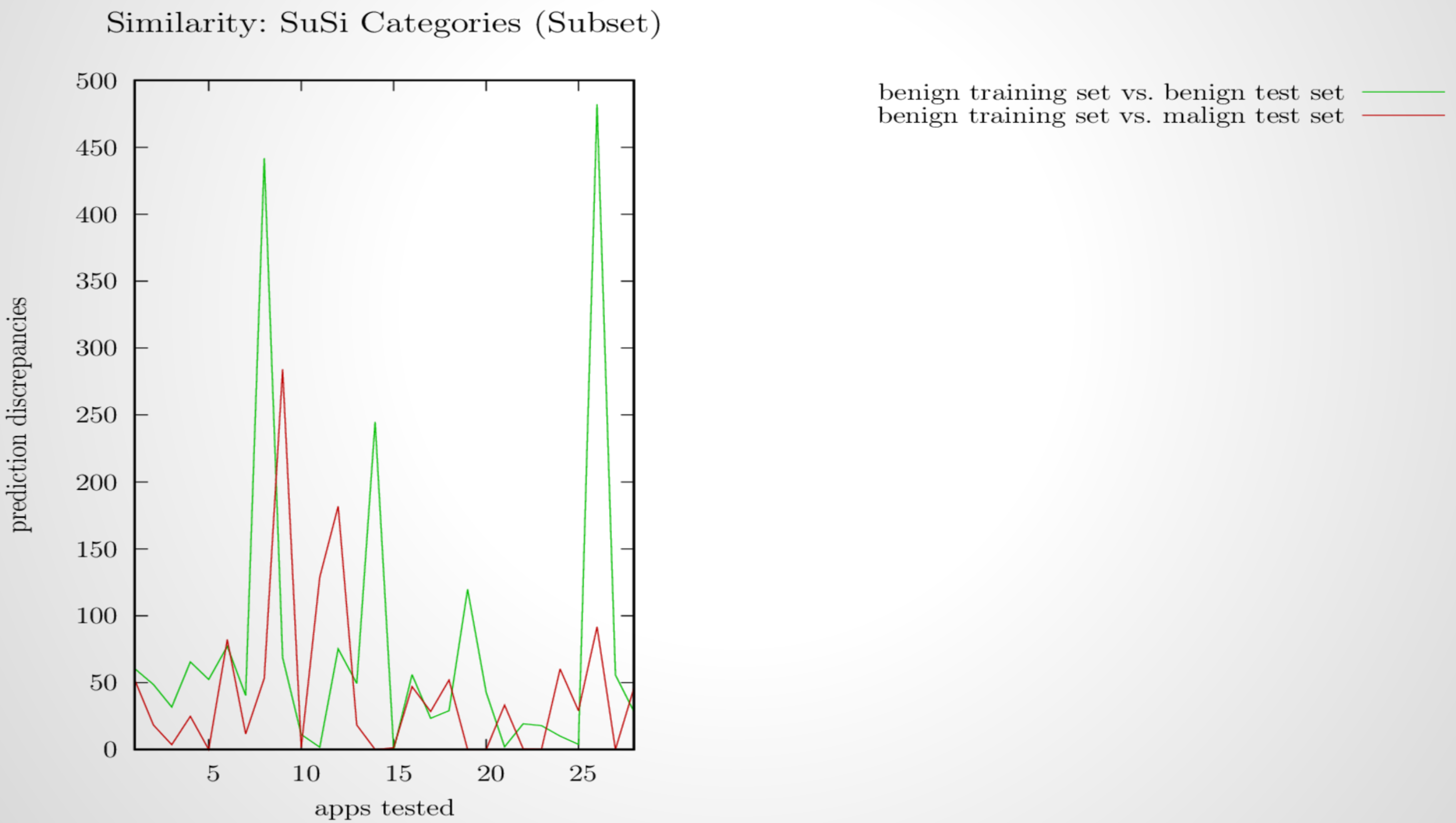
5

Results

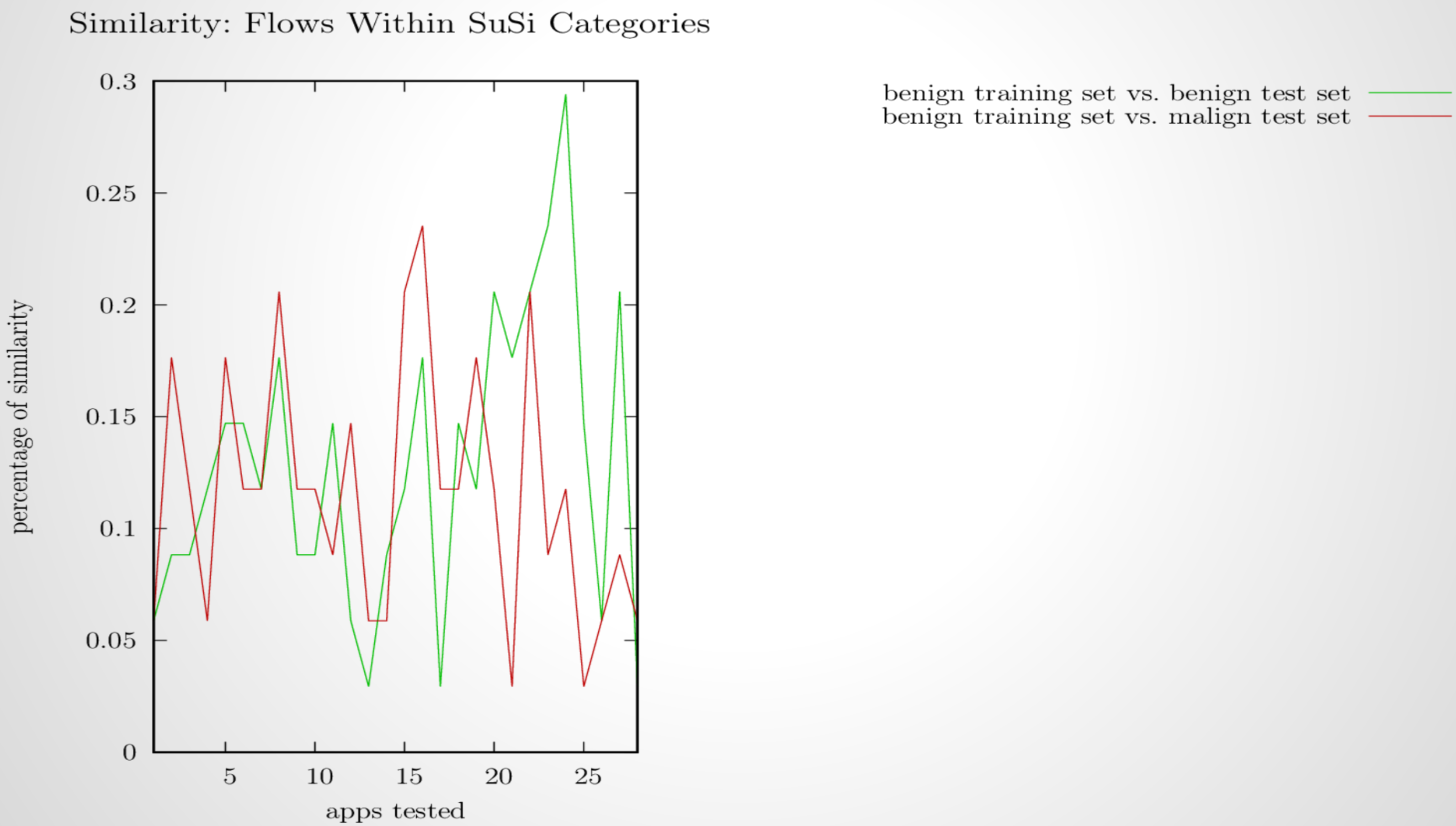
5.1 Results - **Potential malicious behavior #1**



5.2 Results - Potential malicious behavior #2



5.3 Results - **Potential malicious behavior #3**



6

Conclusions

6.1 Conclusions - **Real-World Applications**

UNIVERSITÄT BERN

- App verification in app stores
- Behavorial anti-malware rankings
- Malware evolution analysis
- Malware trend prediction

6.2 Conclusions - **Future Work**

UNIVERSITÄT BERN

- Work on current analysis
  - Optimization of feature selection / SVM parameters
  - More comprehensive source / sink lists
  - Speed improvements
  - ***Much more*** training data

- Work on conceptual level
  - Evaluation of text description features
  - Evaluation of in-app-string features
  - Adaption to other platforms (Apple iOS, ...)
  - Integration of dynamic (hybrid) analysis models

7

References

7.1 References

[1] Steven Arzt, Siegfried Rasthofer, Christian Fritz, Eric Bodden
"FlowDroid: Precise Context, Flow, Field, Object-sensitive and Lifecycle-aware Taint Analysis for Android Apps", PLDI 2014, 2014
http://dx.doi.org/10.1145/2594291.2594299

[2] S. D. Bay, M. Schwabacher
"Mining Distance-Based Outliers in Near Linear Time with Randomization and a Simple Pruning Rule", Proceedings of The Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2003
http://stephenbay.net/papers/outliers.kdd03.pdf

[3] Steven Arzt, Siegfried Rasthofer, Eric Bodden, et al.
"Mining Apps for Abnormal Usage of Sensitive Data", 2015
https://www.st.cs.uni-saarland.de/appmining/mudflow/icse2015-mudflow.pdf

[4] Steven Arzt, Siegfried Rasthofer, and Eric Bodden
"SuSi: A tool for the fully automated classification and categorization of android sources and sinks", Technical Report TUD-CS-2013-0114, EC SPRIDE, 2013
https://www.informatik.tu-darmstadt.de/fileadmin/user_upload/Group_CASED/Publikationen/TUD-CS-2013-0114.pdf

Questions?