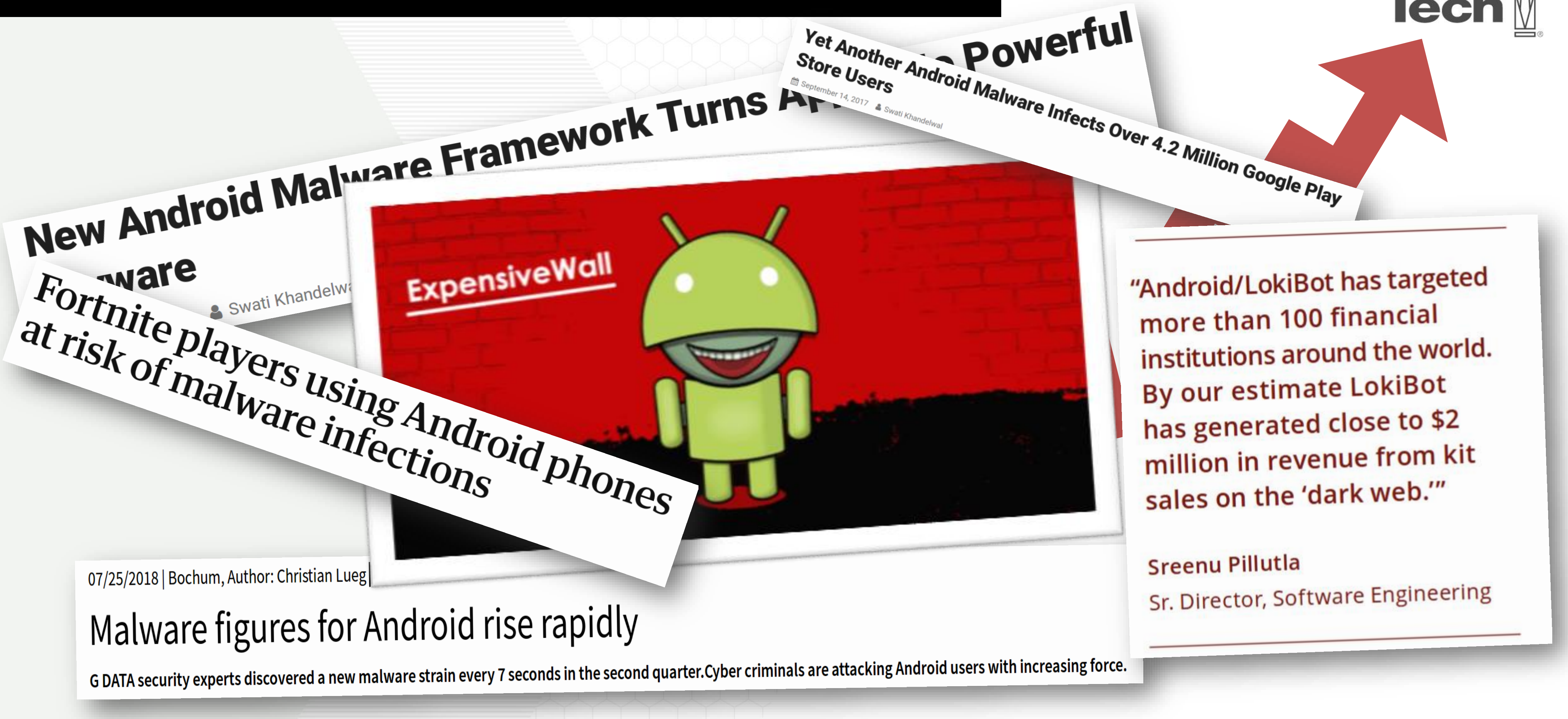
IMPROVING ACCURACY OF ANDROID MALWARE DETECTION WITH LIGHTWEIGHT CONTEXTUAL AWARENESS

JOEY ALLEN, MATTHEW LANDEN,
SANYA CHABA, YANG JI,
SIMON PAK HO CHUNG, WENKE LEE

ACSAC'18

STATE OF ANDROID ECOSYSTEM



New Android Malware Framework Turns Ap

Powerful

ware

Swati Khandelw

Yet Another Android Malware Infects Over 4.2 Million Google Play Store Users

September 14, 2017 Swati Khandelwal

ExpensiveWall

Fortnite players using Android phones at risk of malware infections

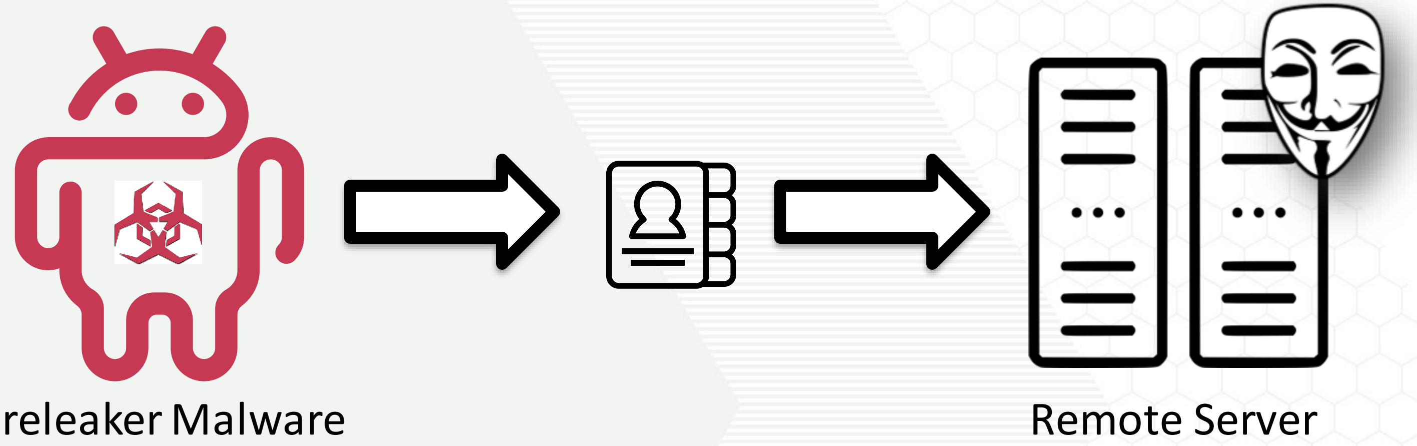
07/25/2018 | Bochum, Author: Christian Lueg|

Malware figures for Android rise rapidly

G DATA security experts discovered a new malware strain every 7 seconds in the second quarter.Cyber criminals are attacking Android users with increasing force.

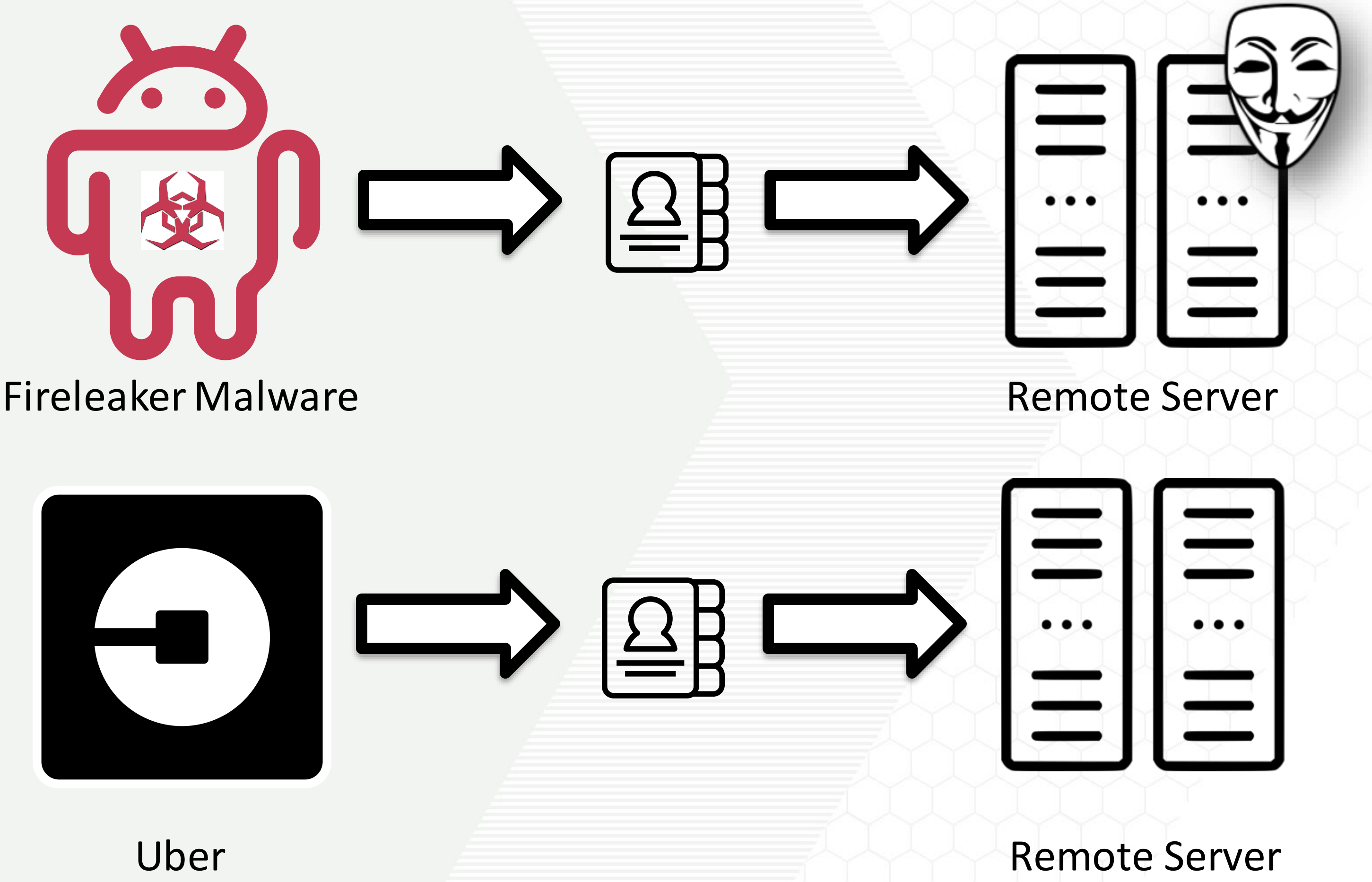
> "Android/LokiBot has targeted more than 100 financial institutions around the world. By our estimate LokiBot has generated close to $2 million in revenue from kit sales on the 'dark web.'"
>
> Sreenu Pillutla
> Sr. Director, Software Engineering

HOW TO INTERPRET AN APP'S INTENT?



Exfiltrates contacts to find new victims

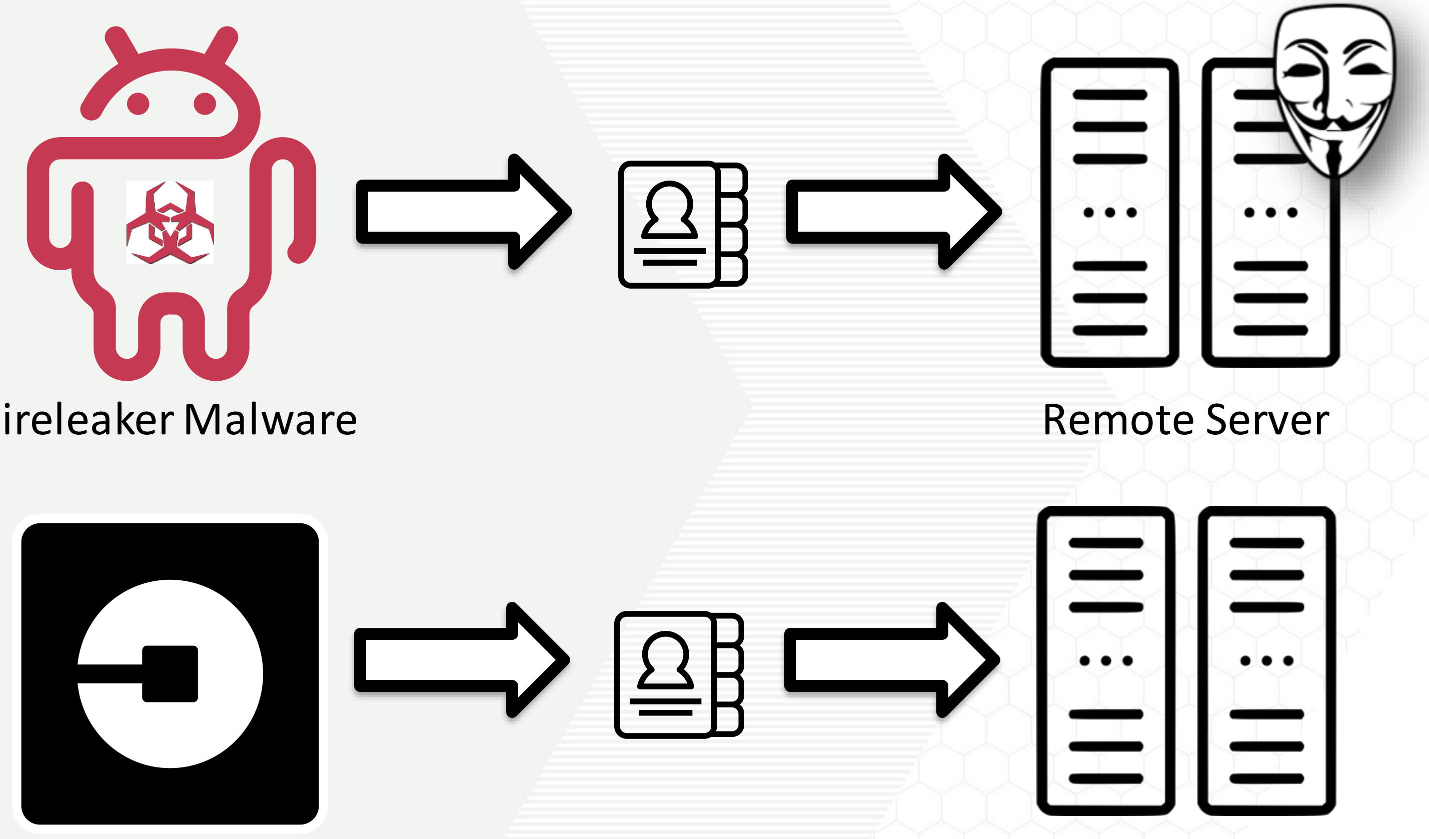
HOW TO INTERPRET AN APP'S INTENT?



Exfiltrates contacts to find new victims

Syncs contacts to refer friends

HOW TO DEFINE BEHAVIOR AS MALICIOUS?



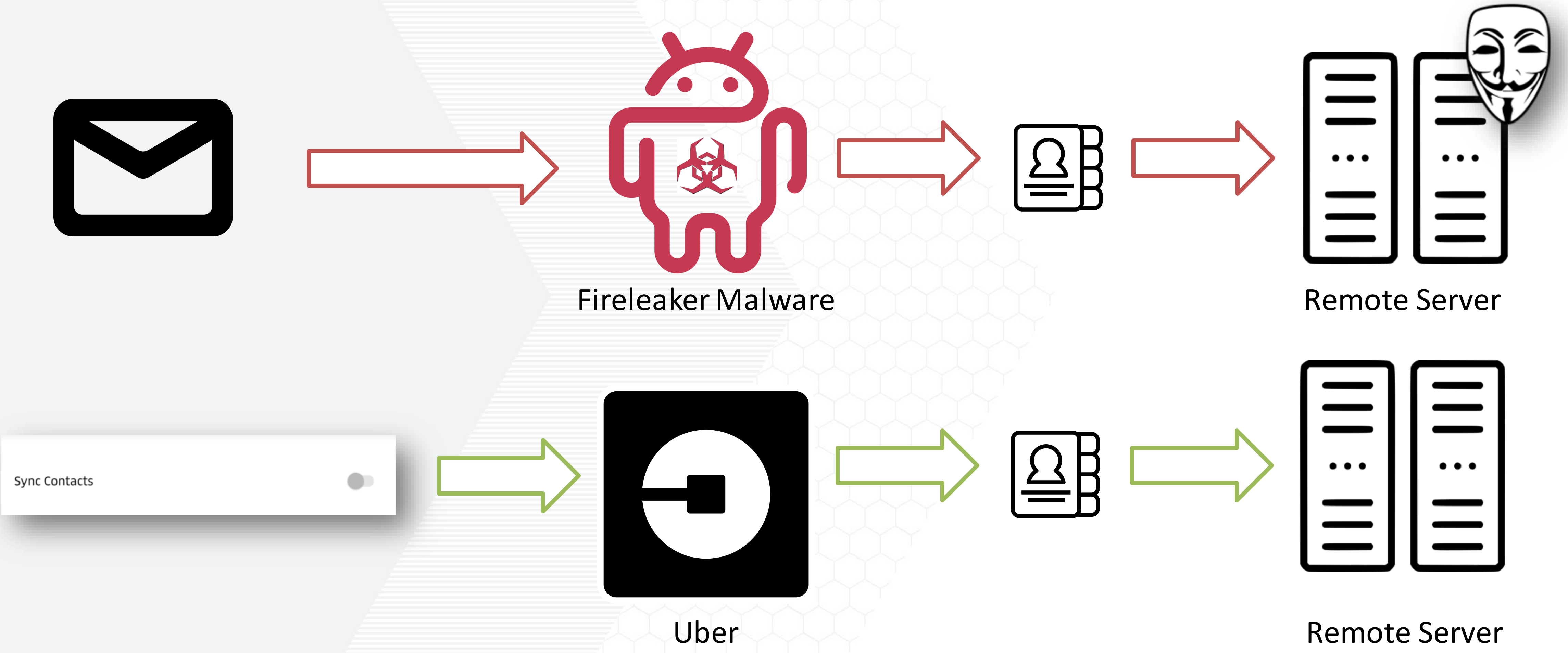
Fireleaker Malware

Remote Server

Uber

Remote Server

HOW WAS BEHAVIOR TRIGGERED?



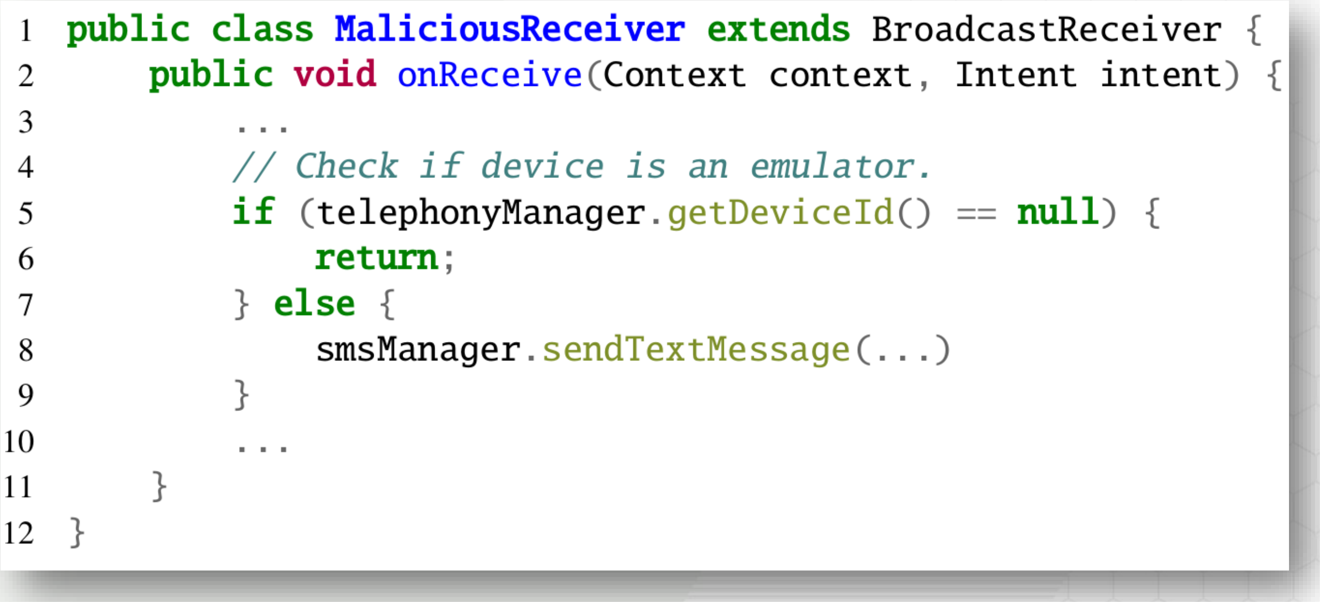
EXISTING CONTEXT-BASED SYSTEMS

- Detecting sensitive behavior is not enough...

- Embed *how* a behavior was invoked into infer *intent*.

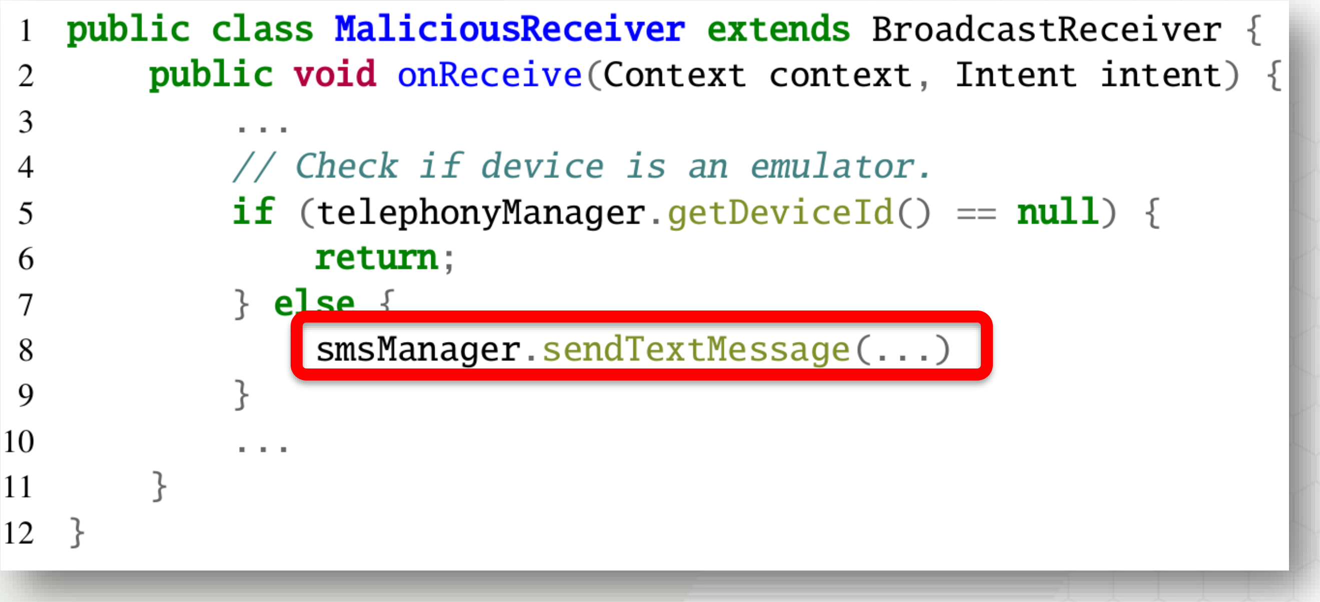
Android Context-Based Systems	
Framework	Context Factors
Whyper (USENIX'13)	Textual Description
DroidSift (CCS'14)	API-Dependencies, Entrypoints, Data Dependencies
AppContext (ICSE'15)	Triggering Events, System Information, Entrypoints
MudFlow (ICSE'15)	Information Flow
MaMaDroid (NDSS'17)	Abstracted call-sequences
EnMobile (ICSE'18)	Network Provenance

CONTEXT FACTORS: EXAMPLE



```java
public class MaliciousReceiver extends BroadcastReceiver {
    public void onReceive(Context context, Intent intent) {
        ...
        // Check if device is an emulator.
        if (telephonyManager.getDeviceId() == null) {
            return;
        } else {
            smsManager.sendTextMessage(...)
        }
        ...
    }
}
```

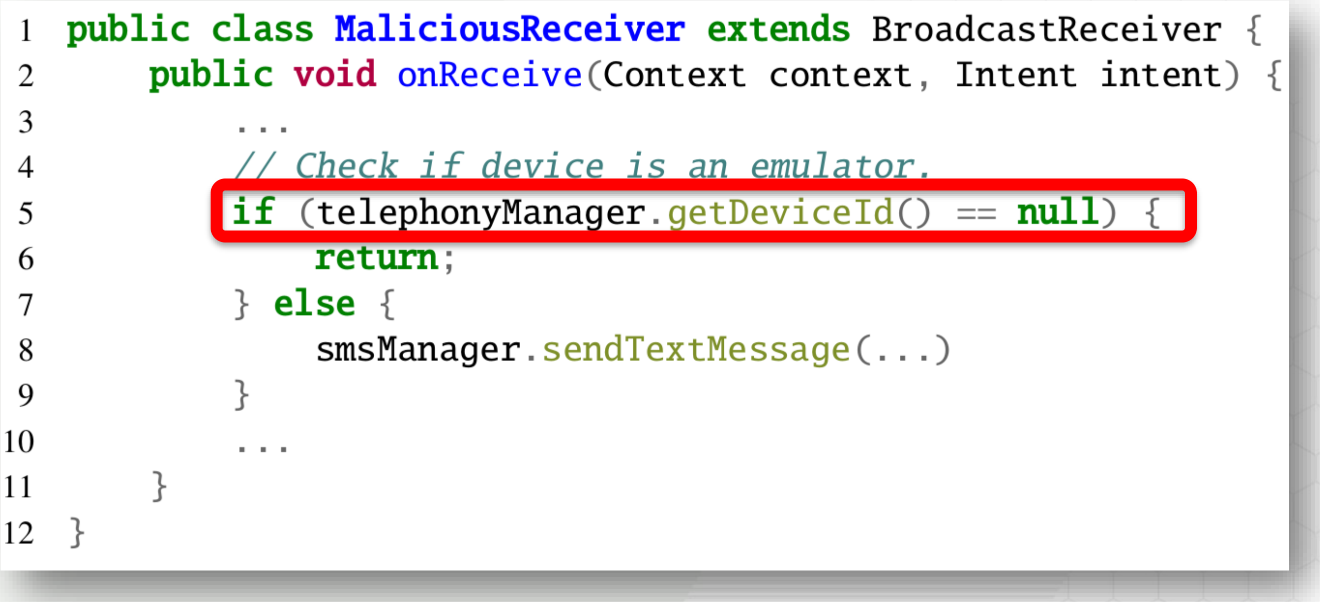
CONTEXT FACTORS: SENSITIVE API



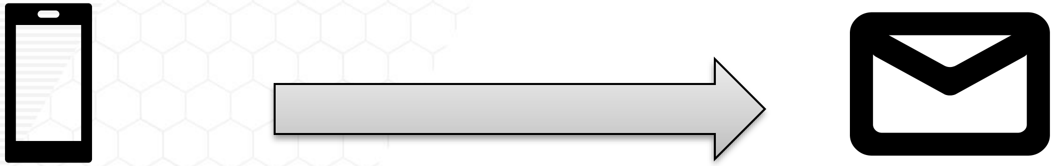
```
public class MaliciousReceiver extends BroadcastReceiver {
    public void onReceive(Context context, Intent intent) {
        ...
        // Check if device is an emulator.
        if (telephonyManager.getDeviceId() == null) {
            return;
        } else {
            smsManager.sendTextMessage(...)
        }
        ...
    }
}
```

SMS

CONTEXT FACTORS: CONSTRAINT DEPENDENCY



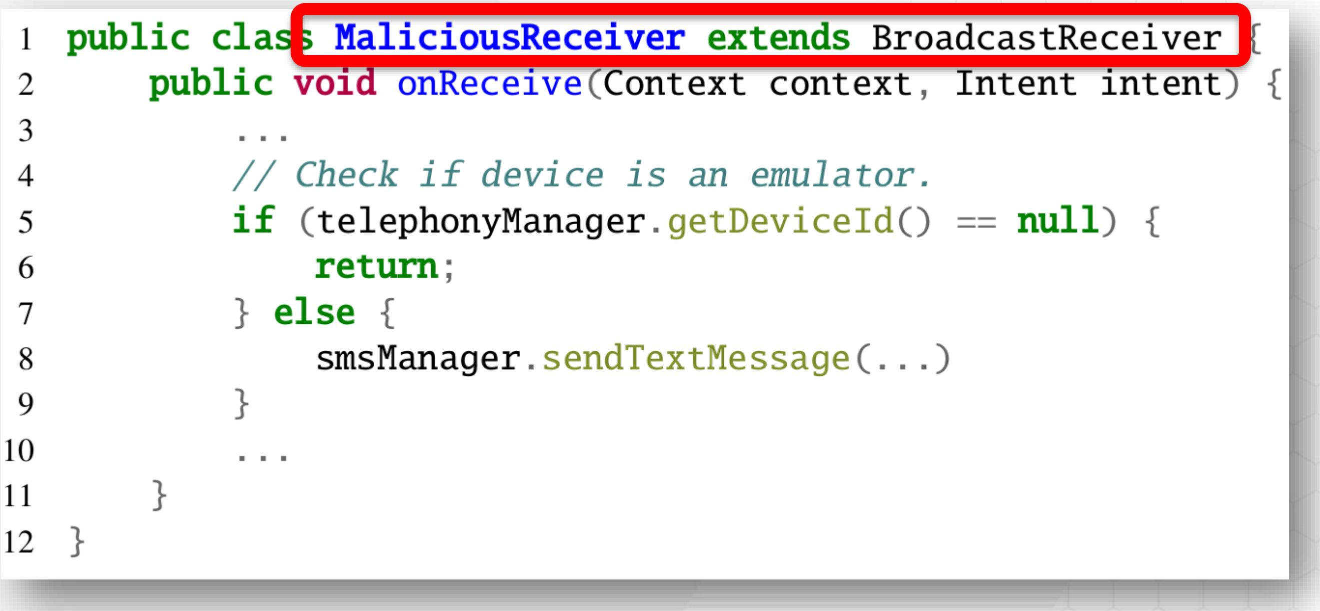
```
public class MaliciousReceiver extends BroadcastReceiver {
    public void onReceive(Context context, Intent intent) {
        ...
        // Check if device is an emulator.
        if (telephonyManager.getDeviceId() == null) {
            return;
        } else {
            smsManager.sendTextMessage(...)
        }
        ...
    }
}
```



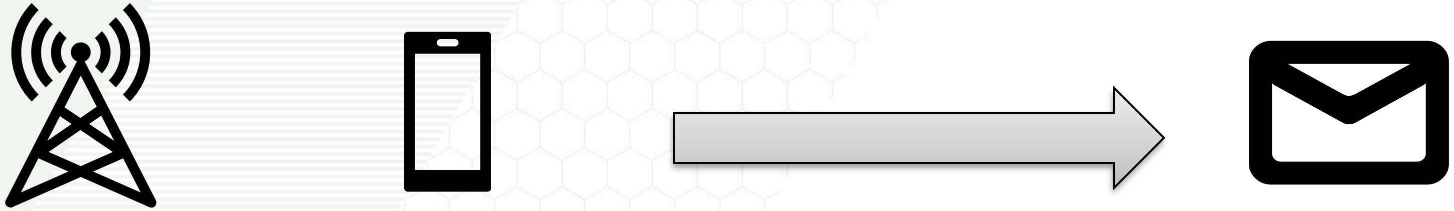
Device Info

SMS

CONTEXT FACTORS: ENTRYPOINT



```java
public class MaliciousReceiver extends BroadcastReceiver {
    public void onReceive(Context context, Intent intent) {
        ...
        // Check if device is an emulator.
        if (telephonyManager.getDeviceId() == null) {
            return;
        } else {
            smsManager.sendTextMessage(...)
        }
        ...
    }
}
```

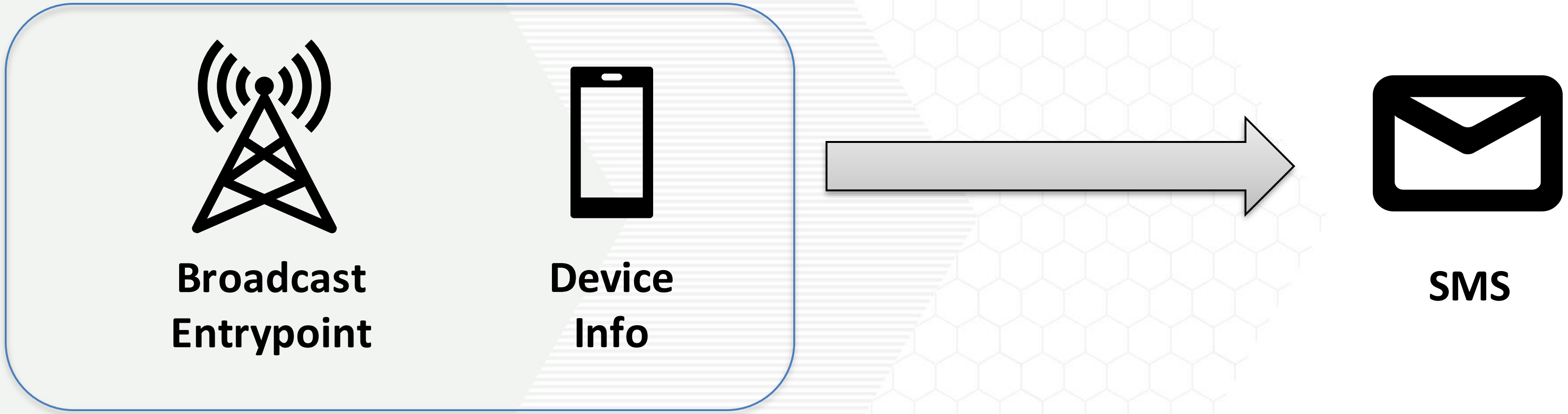


Broadcast Entrypoint

Device Info

SMS

CONTEXT FACTORS: ENTRYPOINT



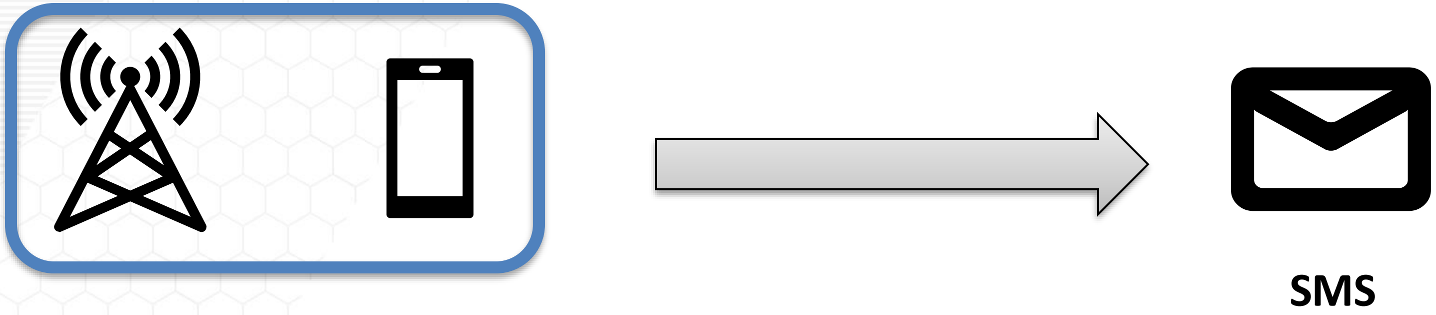
Context Factors

MOTIVATION: LIMITATIONS OF PRIOR WORK

- Classification is too tailored to samples in training set.

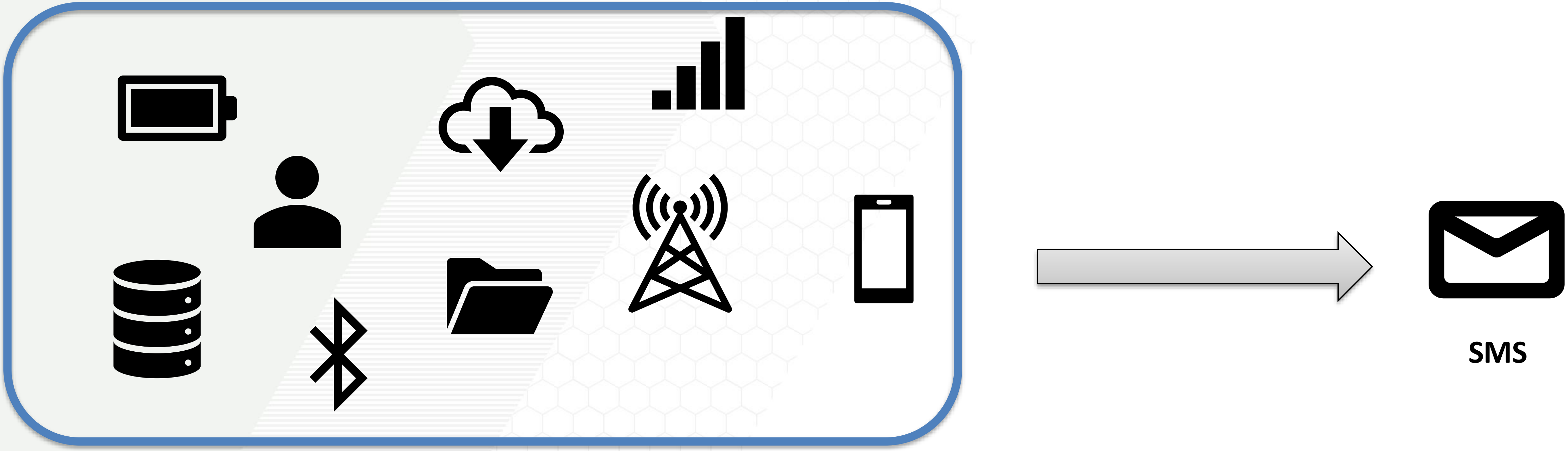
MOTIVATION: LIMITATIONS OF PRIOR WORK

- Classification is too tailored to samples in training set.



MOTIVATION: LIMITATIONS OF PRIOR WORK

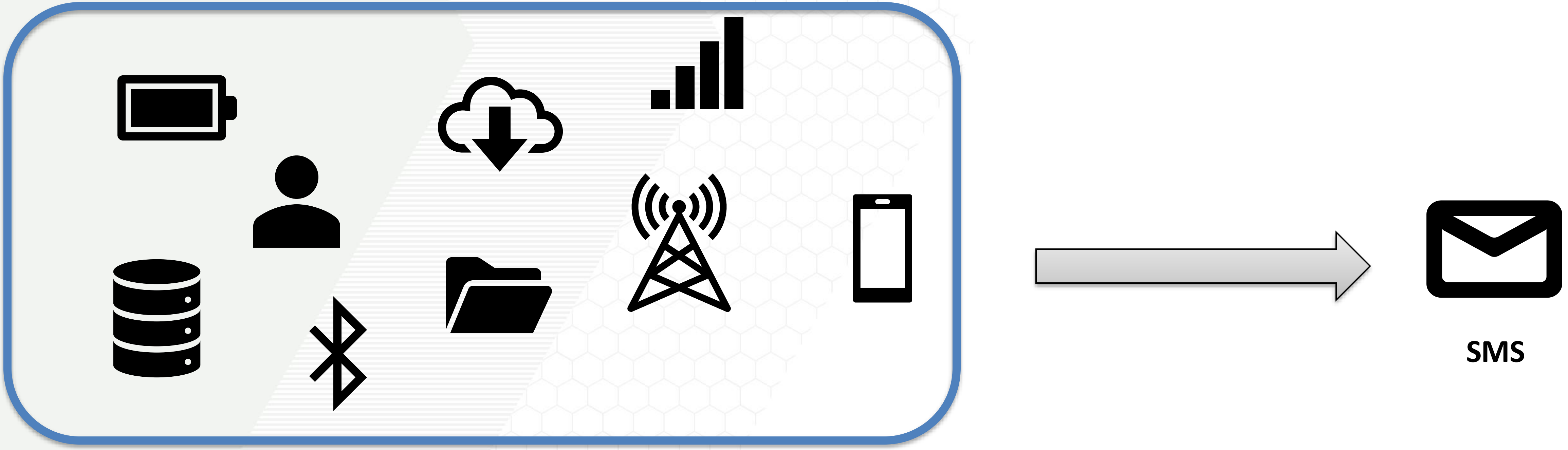
- Classification is too tailored to samples in training set.



MOTIVATION: LIMITATIONS OF PRIOR WORK

- Classification is too tailored to samples in training set.

Family-specific Signatures

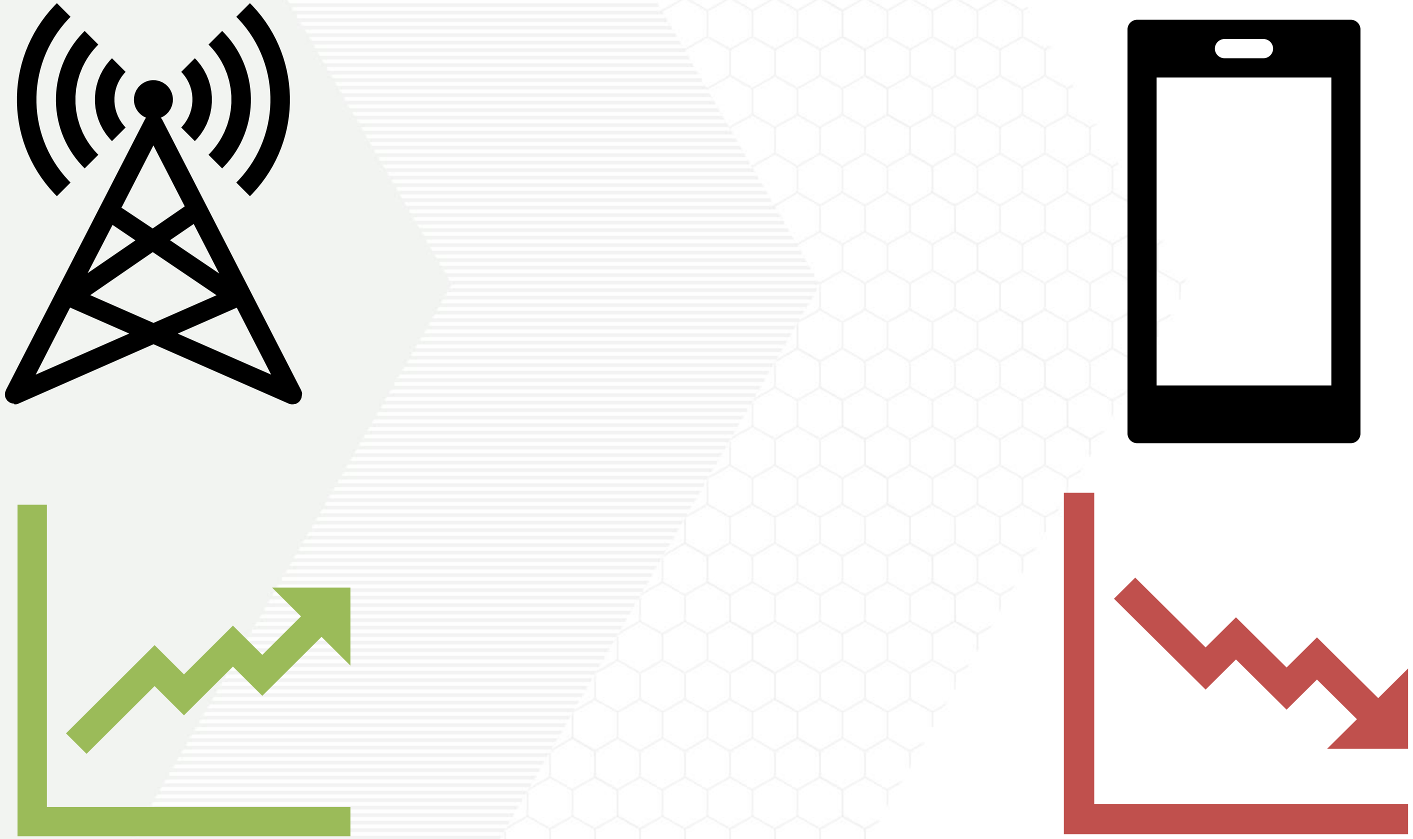


MOTIVATION: LIMITATIONS OF PRIOR WORK

- **Non-Informative Context Factors**

MOTIVATION: LIMITATIONS OF PRIOR WORK

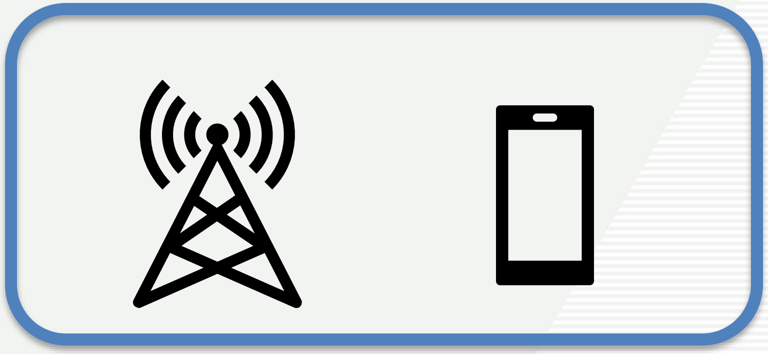
- **Non-Informative Context Factors**



INTUITION: LIGHTWEIGHT CONTEXT

Lightweight Context

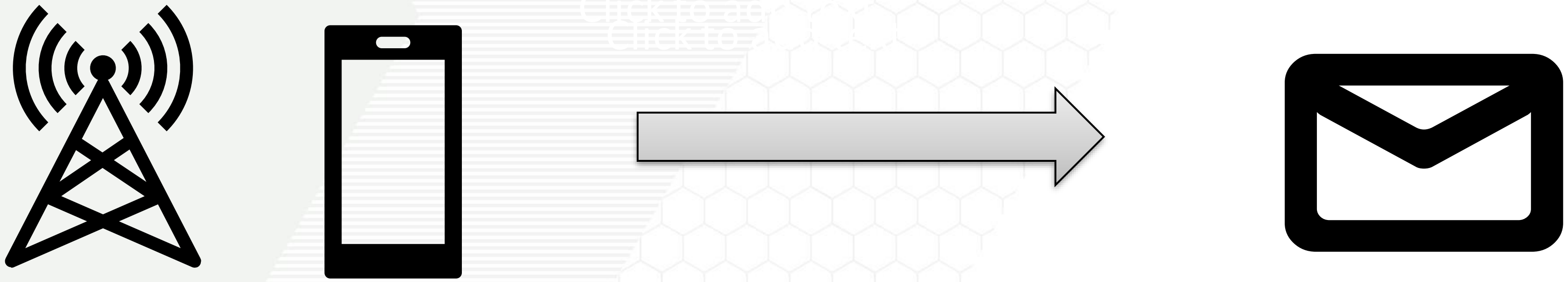
- Rely on **most-informative** contextual factors.
- Refine Context factors used



Refined Context Factors

Informative Context Factors

CASE STUDY



CASE STUDY: INFORMATIVE CONTEXT FACTORS

Purpose: Which Context Factors are most informative?

Device Info

Network Info

Database

UI Behavior

Entrypoints

Context Factors

Network Communication

Email

Phone

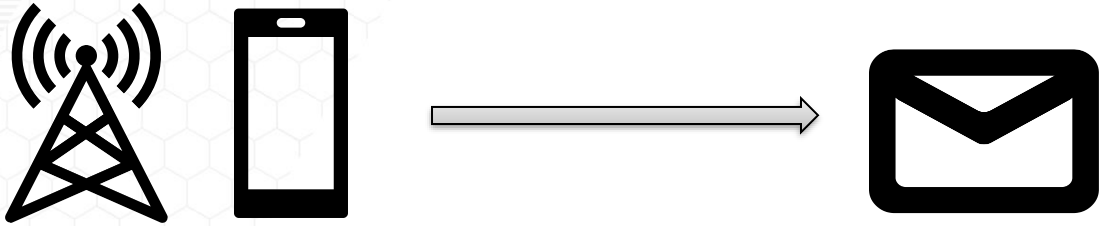
Media Access

SMS

Sensitive APIs

CASE STUDY: INFORMATIVE CONTEXT FACTORS

- Mapped Android APIs to Categories
  - 17 Behavior Categories
  - 8 Context Categories
- Dataset
  - 54,000 Contextual Dependency Graphs
- Feature Ranking
  - Ridge Regression



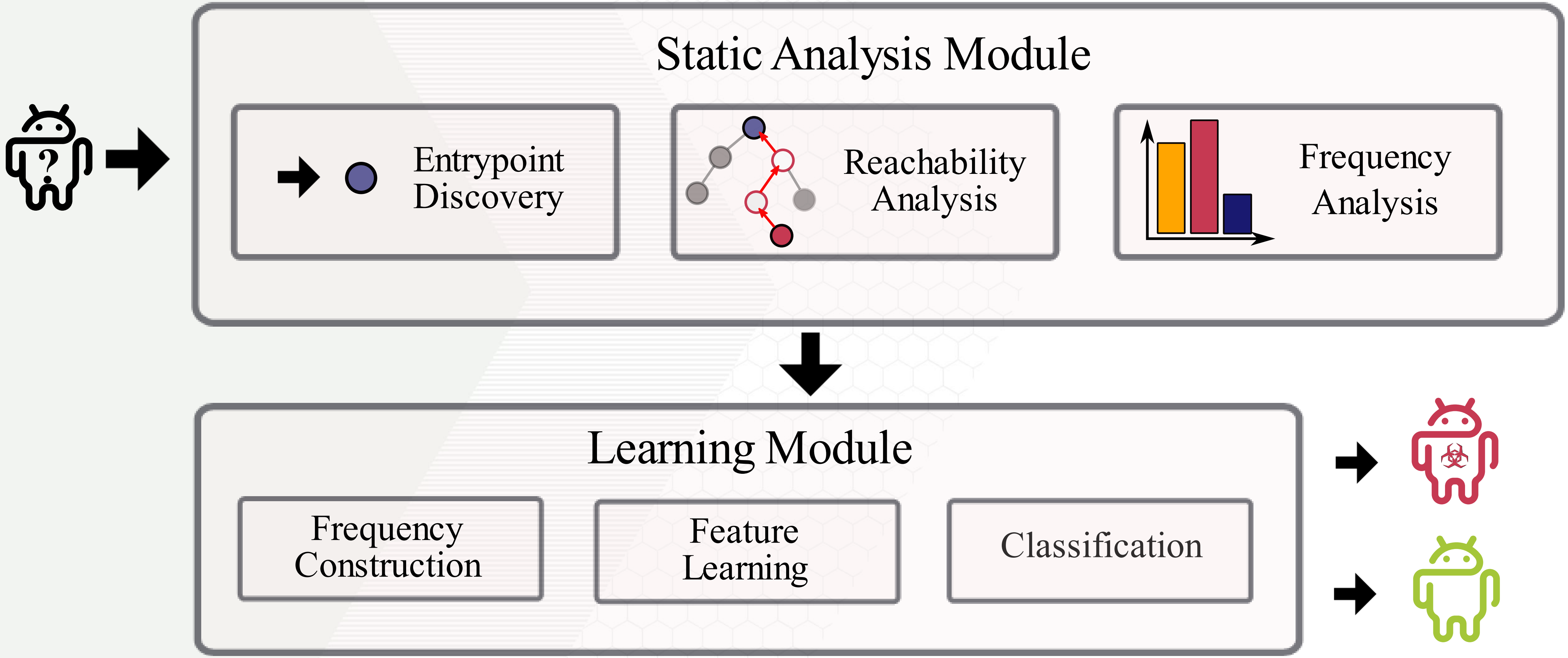
Contextual Dependency Graph

CASE STUDY: FEATURE IMPORTANCE

Behavior Category	1st	2nd	3rd	4th
Class Loading	Activity Entrypoints	Service Entrypoints	UI Entrypoints	Receiver Entrypoints
Account Information	Activity Entrypoints	UI Entrypoints	Service Entrypoints	Intent Information
Location Information	Activity Entrypoints	Intent Information	Device Information	Network Information
Phone State	Service Entrypoints	Activity Entrypoints	UI Entrypoints	Receiver Entrypoints

PikaDroid

PIKADROID



Static Analysis Module

Entrypoint Discovery

Reachability Analysis

Frequency Analysis

Learning Module

Frequency Construction

Feature Learning

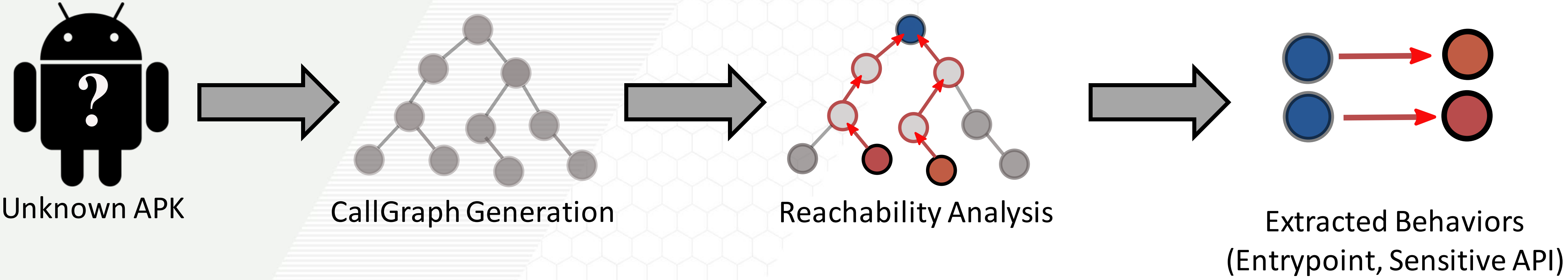
Classification

STATIC ANALYSIS MODULE

Purpose: Extract Sensitive Behavior and Context

Sensitive Behaviors: Android APIs in SUSI & PScout

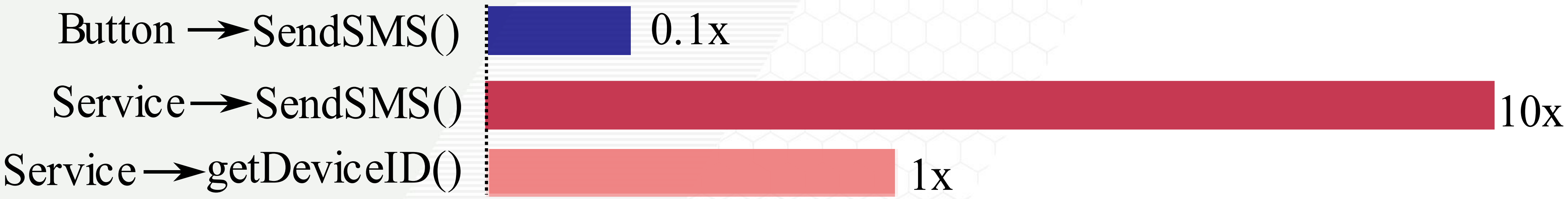
Context Factors: Entrypoints



Frequency Analysis

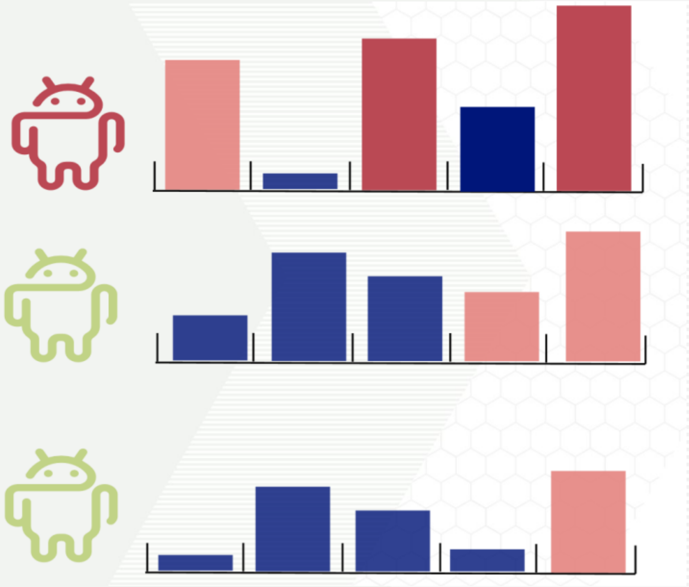
- **Input:** Training Set of Entrypoint-API *(e, s)* pairs
- **Output:** s -- Ratio of malicious to benign apps using *(E, S)*

$$s_{e,\,t} = R(e,\ t)$$

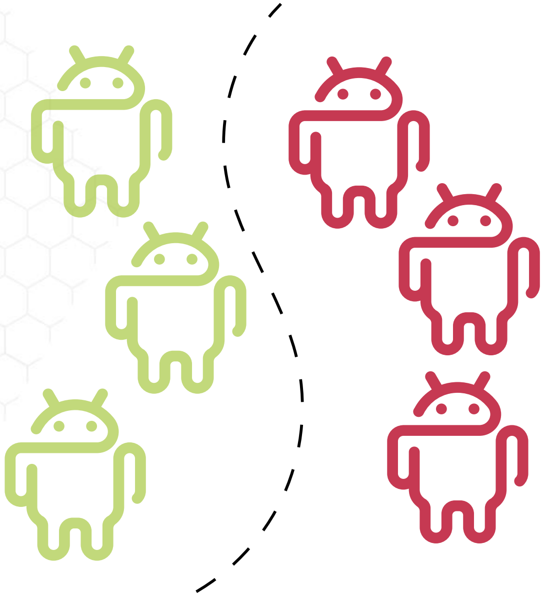


LEARNING MODULE: CLASSIFICATION

- **App Features:** $a_{e,\,t} = s_{e,t}$ if *(e, t)* in A else 0
- **Classification:** Random Forest



Feature Construction



Classification

Dataset Evaluation

DATASET STATISTICS

Dataset

- Apps from 2010 – 2018

Benign Apps

- Crawled from Google Play.

Malicious Apps

- Crawled from 16 different app markets.

Category	Time Period	# Samples
Malware	2010 – 2012	3,970
	2013 – 2015	2,158
	2016 – 2018	2,270
Adware	2010 - 2012	1,524
	2013 - 2015	1,325
Benign	2010 – 2012	3,788
	2013 – 2015	3,596
	2016 – 2018	5,000
Total	2010 – 2018	**23,631**

EVALUATION: PIKADROID VS MAMADROID

Benign vs Malware

- Both Systems perform well.
- PikaDroid outperforms MaMaDroid in 4/4 experiments.

F1-Score for Benign vs. Malware

Time Period	**PikaDroid**	MaMaDroid
2010 - 2012	**97.65%**	94.64%
2013 - 2015	**97.89%**	96.70%
2016 - 2018	**96.07%**	94.27%
2010 - 2018	**97.41%**	94.58%

EVALUATION: PIKADROID VS MAMADROID

Benign vs Adware

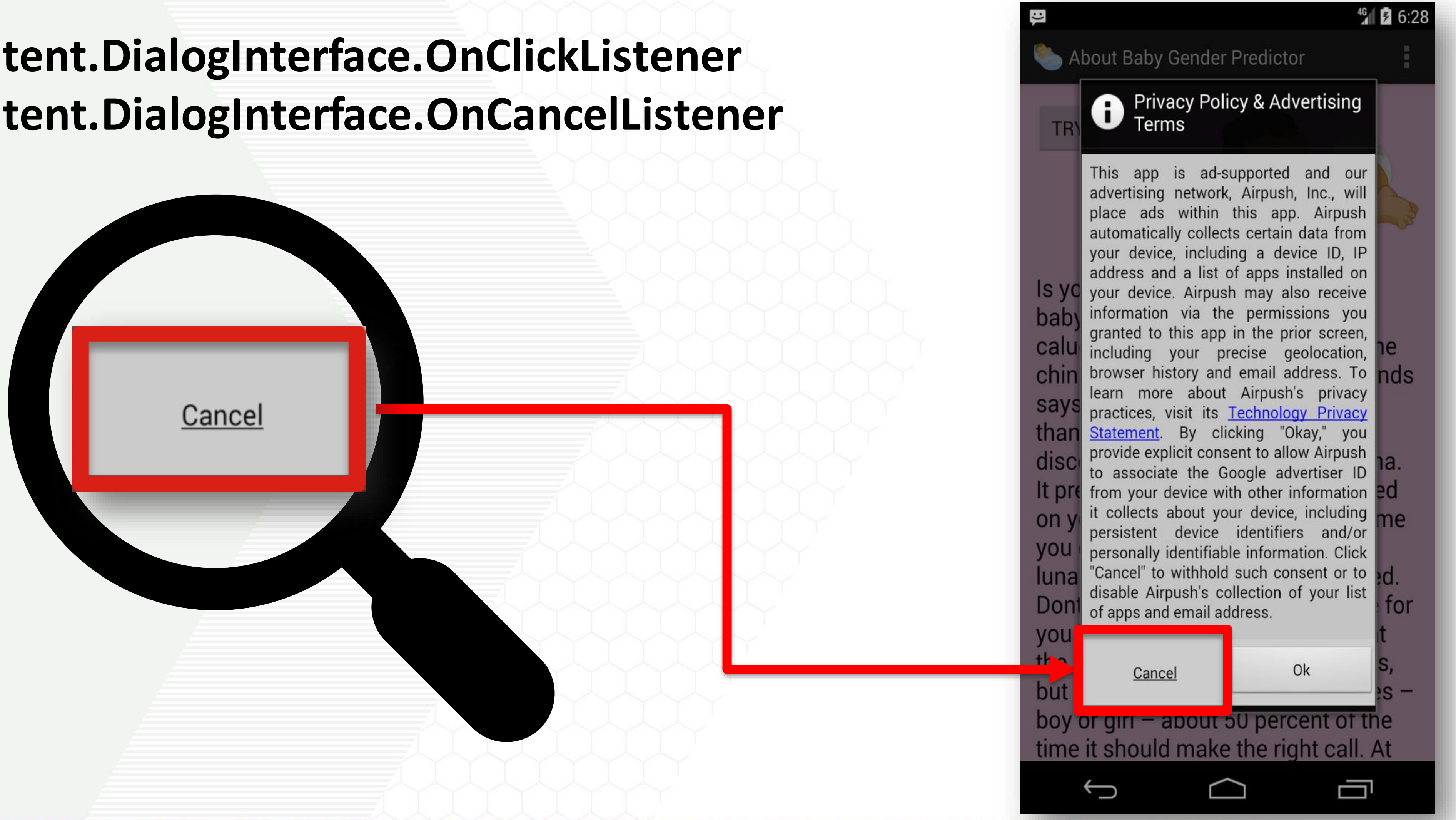
- PikaDroid has significantly higher F1-Score.
- PikaDroid outperforms MaMaDroid in 3/3 experiments.

F1-Score for Benign vs. Adware

Tie Period	**PikaDroid**	MaMaDroid
2010 - 2012	**96.74 %**	92.02 %
2013 - 2018	**94.04 %**	85.45 %
2010 - 2018	**94.15 %**	86.78 %

EVALUATION: PIKADROID VS MAMADROID

android.content.DialogInterface.OnClickListener
android.content.DialogInterface.OnCancelListener

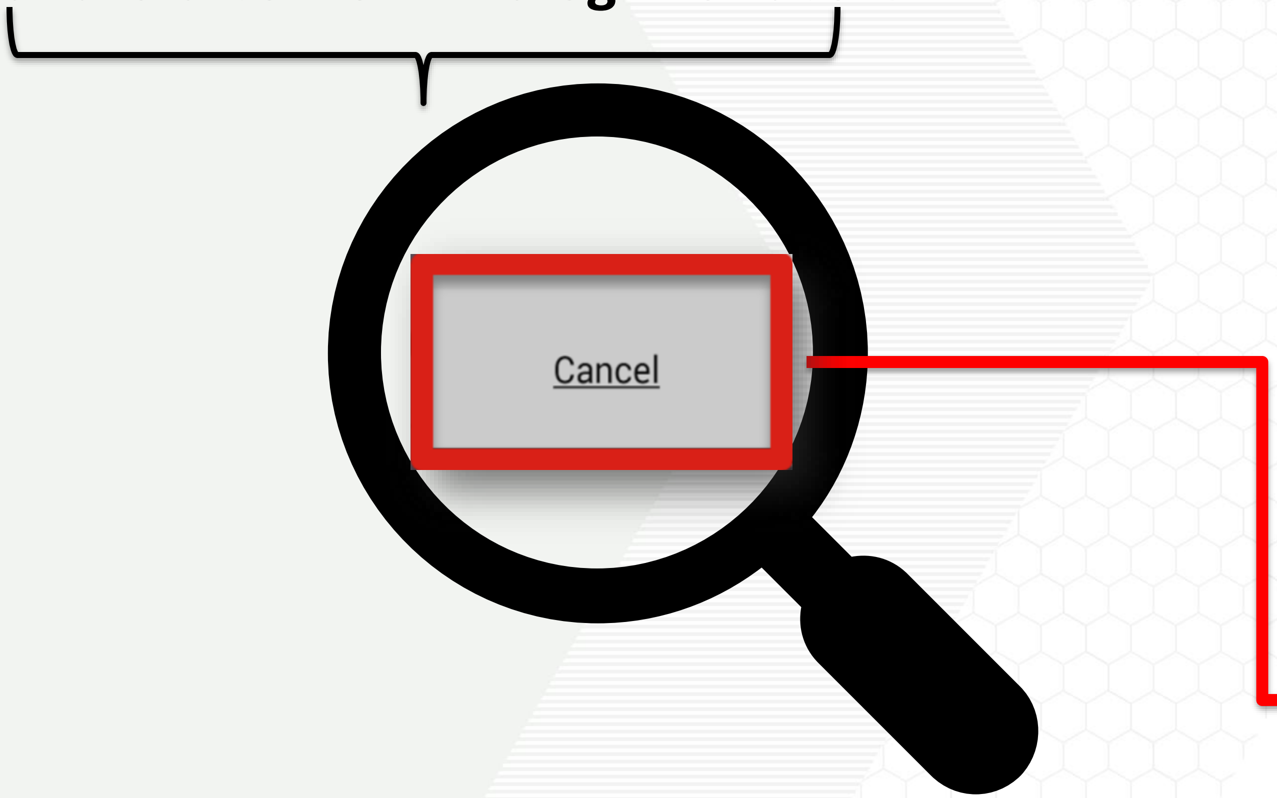


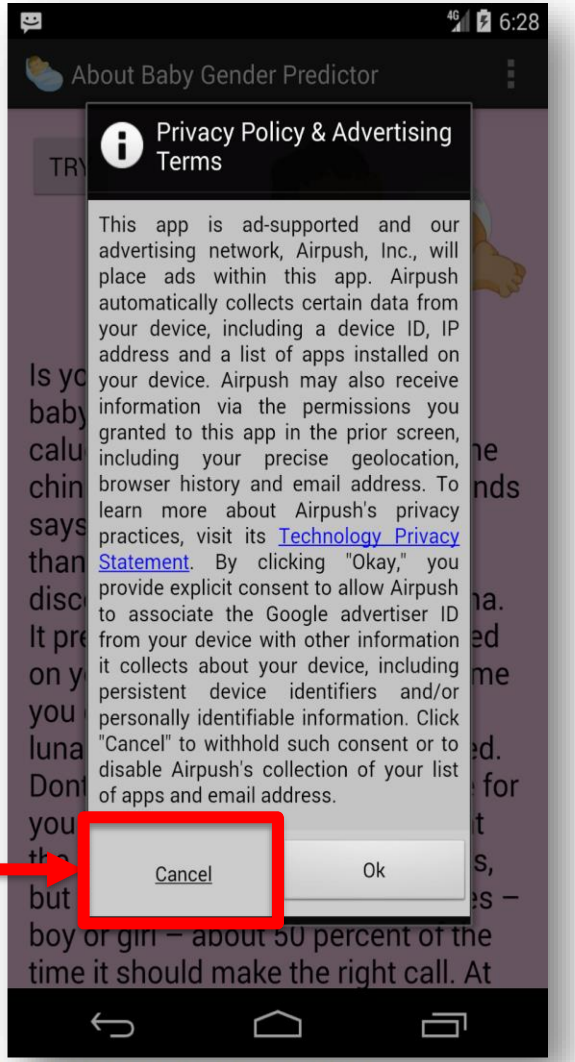
Privacy Policy & Advertising Terms

This app is ad-supported and our advertising network, Airpush, Inc., will place ads within this app. Airpush automatically collects certain data from your device, including a device ID, IP address and a list of apps installed on your device. Airpush may also receive information via the permissions you granted to this app in the prior screen, including your precise geolocation, browser history and email address. To learn more about Airpush's privacy practices, visit its Technology Privacy Statement. By clicking "Okay," you provide explicit consent to allow Airpush to associate the Google advertiser ID from your device with other information it collects about your device, including persistent device identifiers and/or personally identifiable information. Click "Cancel" to withhold such consent or to disable Airpush's collection of your list of apps and email address.

EVALUATION: PIKADROID VS MAMADROID

android.content.DialogInterface.OnClickListener
android.content.DialogInterface. OnCancelListener





Privacy Policy & Advertising Terms

This app is ad-supported and our advertising network, Airpush, Inc., will place ads within this app. Airpush automatically collects certain data from your device, including a device ID, IP address and a list of apps installed on your device. Airpush may also receive information via the permissions you granted to this app in the prior screen, including your precise geolocation, browser history and email address. To learn more about Airpush's privacy practices, visit its Technology Privacy Statement. By clicking "Okay," you provide explicit consent to allow Airpush to associate the Google advertiser ID from your device with other information it collects about your device, including persistent device identifiers and/or personally identifiable information. Click "Cancel" to withhold such consent or to disable Airpush's collection of your list of apps and email address.

boy or girl – about 50 percent of the time it should make the right call. At

HOW EFFECTIVE IS LIGHTWEIGHT CONTEXT?

Side-by-Side Evaluation of PikaDroid and APIMiner

1.45-3.02x less False-Positives during evaluation.

Entrypoint (E)	Targeted API(T)	Ratio (E,T)	Ratio (T)
Service.onStart	FileWriter.write	**3.06**	1.20
Service.onStart	DataOutputStream.writeBytes	**18.71**	0.256
Service.onCreate	TelephonyManager.getDeviceID	**11.05**	0.401

CONCEPT DRIFT

CONCEPT DRIFT

Malware Families Evolve

CONCEPT DRIFT

Malware Families Evolve

New Malware Families

CONCEPT DRIFT

Malware Families Evolve

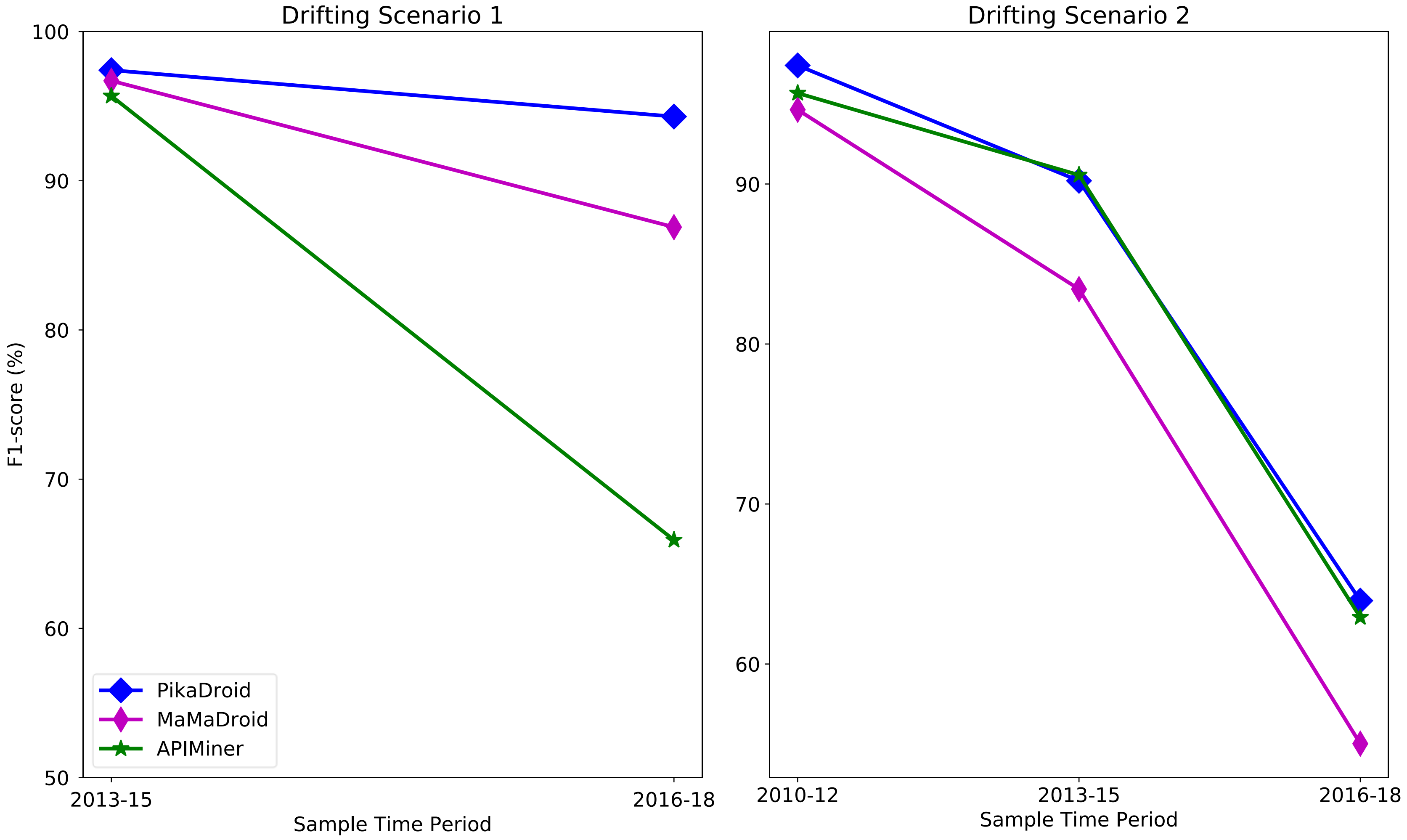
New Malware Families

Classification Model Becomes Outdated!!!!

CONCEPT DRIFT

Two drifting scenarios

Evaluation sensitive to undersampling



LIMITATIONS

Inherit limitations of Static Analysis

- Java Reflection, Dynamic-Code Loading, Native Code, Incomplete call graph, etc.

Entrypoint Manipulation

- Adversary leverages complex ICC chains to invoke sensitive behavior.

New APIs added to framework

- PikaDroid cannot handle new APIs like abstraction-based systems.

Questions?