ETH Zürich

SRL
SOFTWARE RELIABILITY LAB
www.srl.inf.ethz.ch

DEGUARD Statistical De-obfuscation for Android

Petar Tsankov, ETH Zurich

Benjamin Bichsel

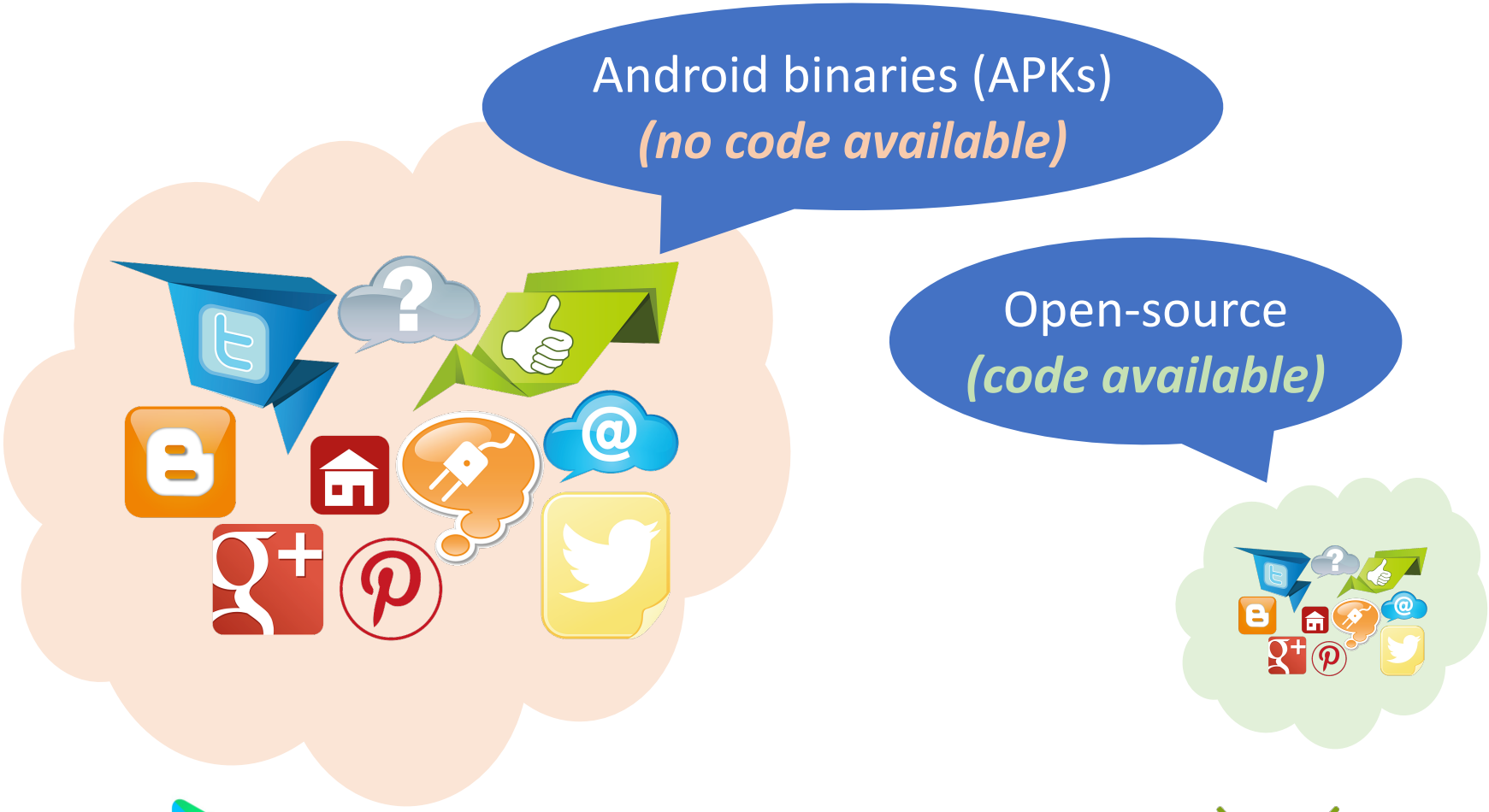
Veselin Raychev

Petar Tsankov

Martin Vechev

DeGuard Team

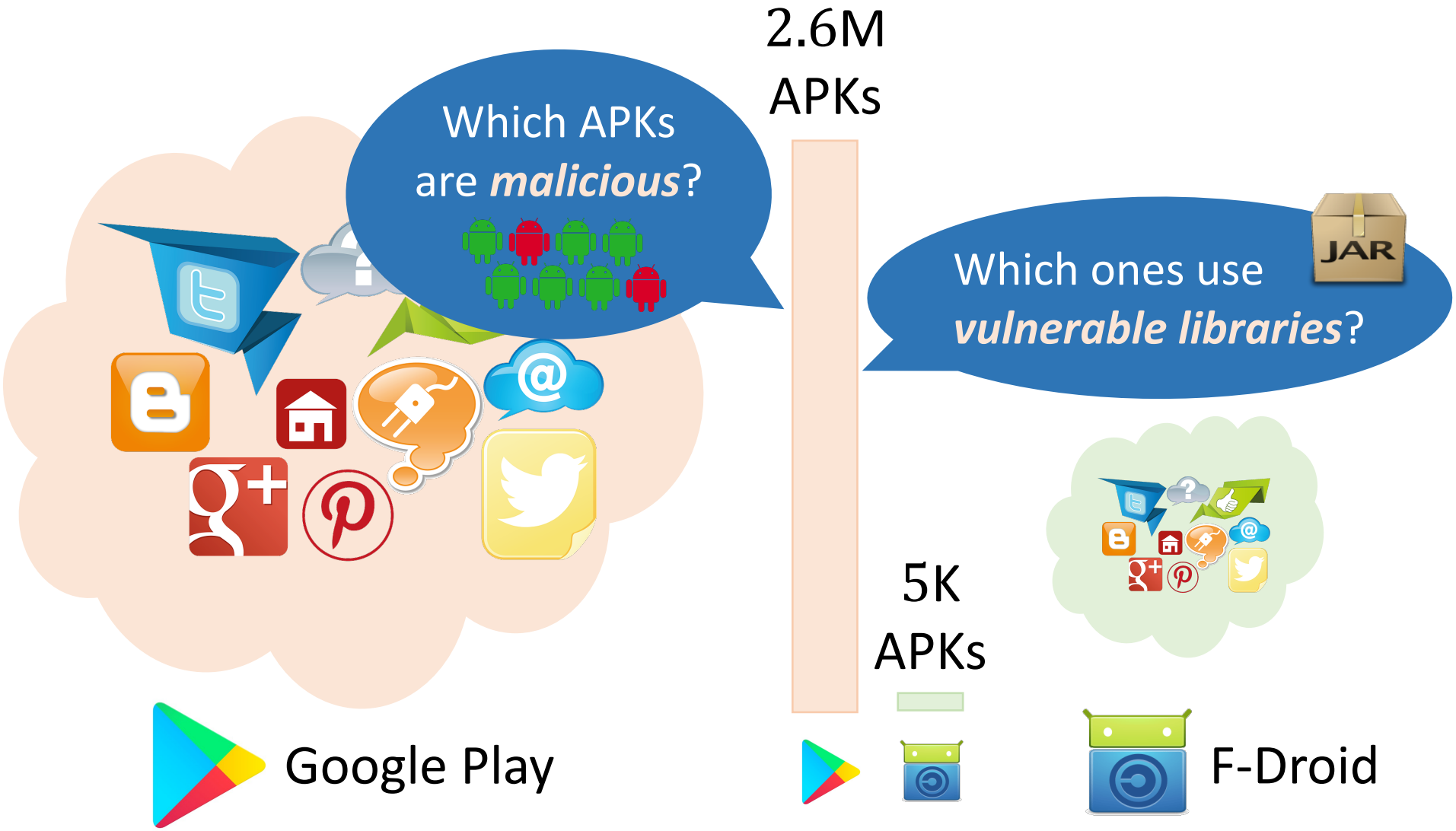
Why De-obfuscate Android Applications?



Google Play

F-Droid

Why De-obfuscate Android Applications?



Layout Obfuscation in Android

Names of API classes/methods

```
package com.example.dbhelper

class DBHelper extends SQLiteHelper {
  SQLiteDatabase db;

  public DBHelper(Context ctx) {
    db = getWritableDatabase();
  }

  Cursor execSQL(String str) {
    return db.rawQuery(str);
  }
}
```

Descriptive application-specific names

Obfuscate

Non-descriptive names

```
package a.b.c

class a extends SQLiteHelper {
  SQLiteDatabase b;

  public a(Context ctx) {
    b = getWritableDatabase();
  }

  Cursor c(String
    return b.rawQue
  }
}
```

API names remain

Layout Obfuscation in Android

Non-descriptive names

Names of API

```
package com.examp

class DBHelper ex
  SQLiteDatabase

  public DBHelper
    db = getWrita
  }

  Cursor execSQL(
    return db.raw
  }
}
```

```
teHelper {

tx) {
atabase();

}
```

Security Challenges

Code inspection

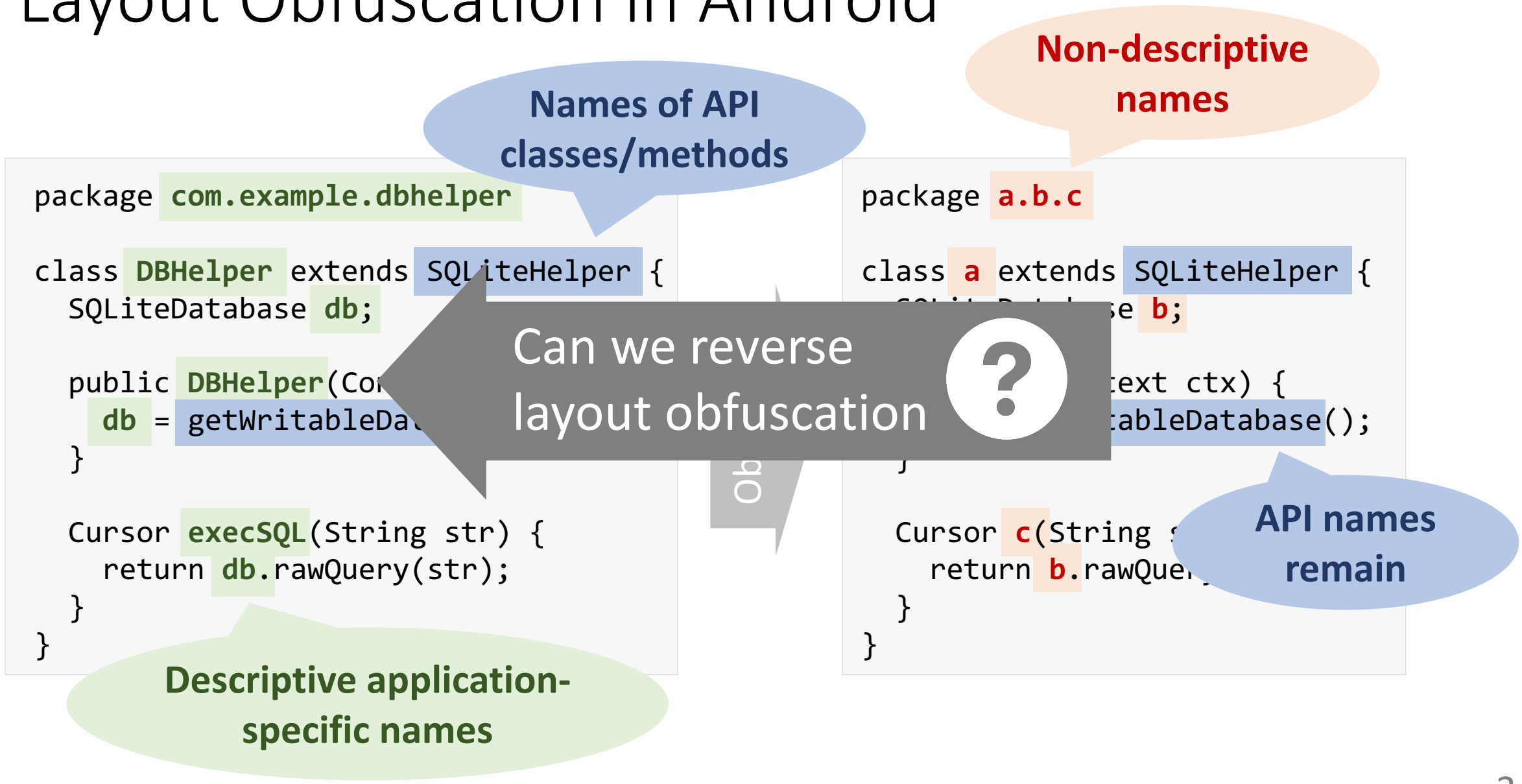
Third-party library detection

… many others

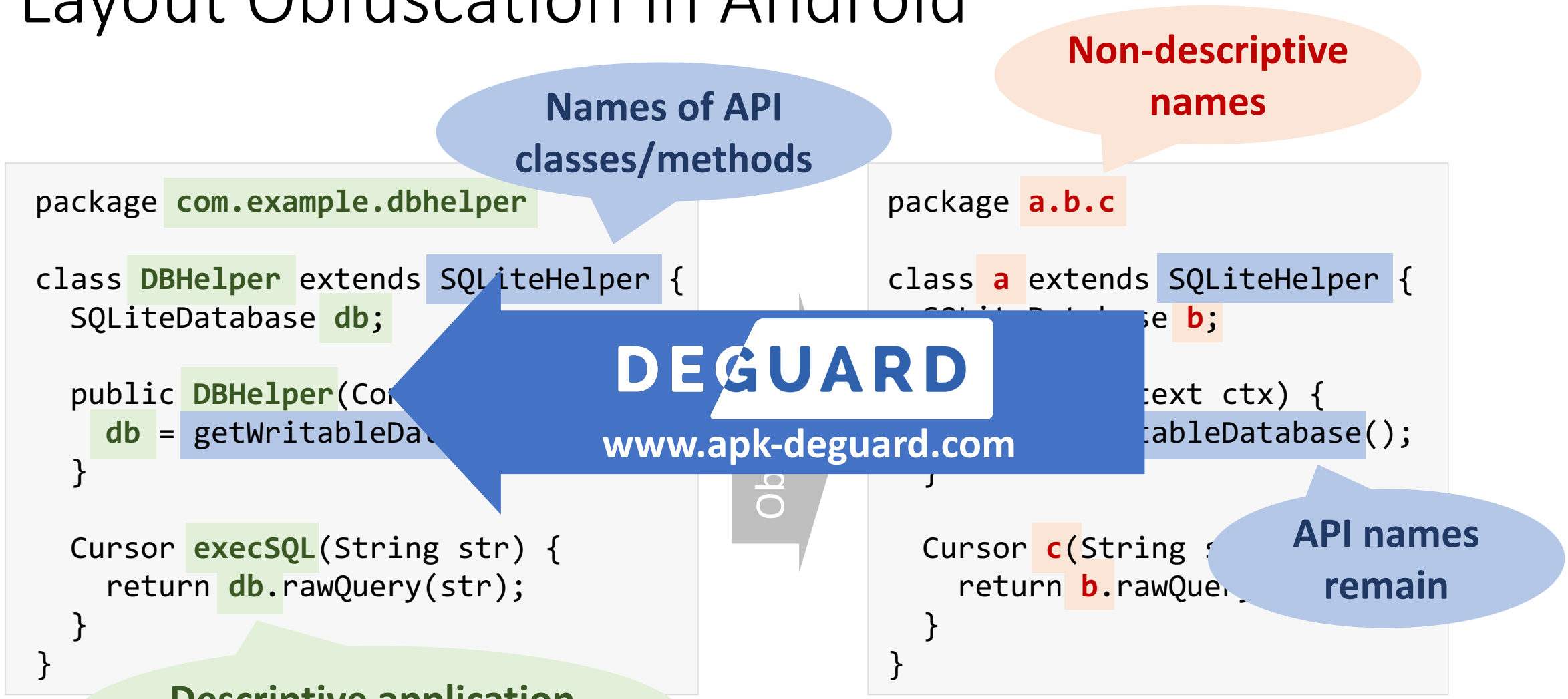
API names remain

Descriptive application-specific names

Layout Obfuscation in Android



Layout Obfuscation in Android



Yes, with roughly 80% accuracy!

Demo

DEGUARD

www.apk-deguard.com

Released in October 2016, so far: **> 100GB distinct APKs de-obfuscated**

Reddit posts/comments

[–] **evantatarka** WillowTree • **3 points** 2 days ago

Nice! This should help with debugging issues in play services or other libs that are obfuscated just to make my life harder a bit easier.

permalink embed pocket

[–] **oleeeEncantado** 2 points 4 days ago

Works quite well, I've tried on some small games.

permalink embed pocket

[–] **Tycon712** • **3 points** 2 days ago

Can someone tell me what the point of using Proguard is if there are tools out there like this?

permalink embed pocket

[–] **theheartbreakpug** • **6 points** 2 days ago

As far as I know, this is brand new. I asked the creator of ProGuard a week ago how hard it is to unobfuscate code after it's run through proguard. He said it strips all the names out of the code so it's essentially impossible. I'm super impressed by what they've done here.

Tweets

D|-|AR$|-|IN @dharshin · Oct 17

Deobfuscate #proguard 'ed APKs: apk-deguard.com. The paper on the inner workings: srl.inf.ethz.ch/papers/deguard… **#Android** #MobileSecurity

Brian Carpenter @geeknik · 9h

Android Deobfuscation with Machine Learning reverses the effects of ProGuard [PDF] srl.inf.ethz.ch/papers/deguard…

3 4

rvivek @vivek_310 · Oct 17

"**Android Deobfuscation** with Machine Learning reverses the effects of ProGuard" #security #feedly

Android Deobfuscation with Machine... • /r/Reverse...

8 points and 1 comments so far on reddit

How Does DeGuard Work?

DeGuard: System Overview

Open-source, unobfuscated APKs

Learning phase

Static analysis

Semantic representation

Training

Probabilistic model $P(\blacksquare|\blacksquare)$

Prediction phase

```
class a extends SQLiteHelper {
  SQLiteDatabase b;
  public a(Context ctx) {
    b = getWritableDB();
  }
}
```

Obfuscated code

Static analysis

MAP inference

Transform

```
class DBHelper extends SQLiteHelper{
  SQLiteDatabase db;
  public DBHelper(Context ctx) {
    db = getWritableDB();
  }
}
```

De-obfuscated code

Probabilistic Graphical Models

Probabilistic Graphical Models

	name1	name2	weight
f_1	SQLiteHelper	DBUtils	0.3
f_2	SQLiteHelper	DBHelper	0.2

SQLiteHelper — extends — a

	name1	name2	weight
f_5	DBUtils	instance	0.5
f_6	DBHelper	db	0.4
f_7	…	…	…

a — field-in — b — gets — getWritableDB

	name1	name2	weight
f_3	getWritableDB	db	0.7
f_4	getWritableDB	instance	0.4

```
class a extends SQLiteHelper {
  SQLiteDatabase b;
  public a(Context ctx) {
    b = getWritableDB();
  }
}
```

SQLiteHelper, getWritableDB — Known variables

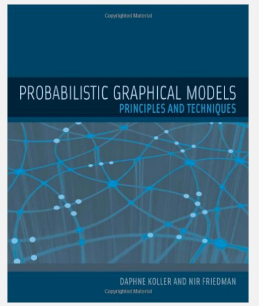
a, b — Unknown variables

$f_1, f_2, .., f_7$ — Feature functions

Graph + features define a **probabilistic graphical model**

$$P(a, b \mid SQLiteHelper, getWritableDB)$$

$$= \frac{1}{Z}\exp(0.3 \cdot f_1(SQLiteHelper, a) + 0.2 \cdot f_2(SQLiteHelper, a) + \cdots)$$



For details see report on www.apk-deguard.com

Probabilistic Graphical Models

	name1	name2	weight
f_1	SQLiteHelper	DBUtils	0.3
f_2	SQLiteHelper	DBHelper	0.2

SQLiteHelper — extends — a

field-in

getWritableDB — gets — |

	name1	name
f_3	getWritableDB	db
f_4	getWritableDB	instan

```
class a extends SQLiteHelper {
  SQLiteDatabase b;
  public a(Context ctx) {
    b = getWritableDB();
  }
```

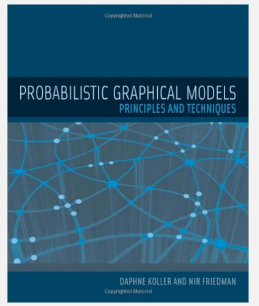
Next

How are the features and their weights learned?

a **probabilistic graphical model**

$$P(a, b \mid SQLiteHelper, getWritableDB)$$

$$= \frac{1}{Z}\exp(0.3 \cdot f_1(SQLiteHelper, a) + 0.2 \cdot f_2(SQLiteHelper, a) + \cdots)$$



SQLiteHelper, getWritableDB — Known variables

a, b — Unknown variables

$f_1, f_2, .., f_7$ — Feature functions

For details see report on www.apk-deguard.com

Learning

Learning

> 2,000

Unobfuscated APKs

Feature templates

28 templates

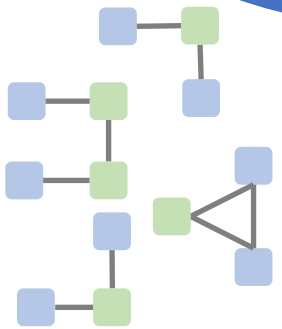
Static analysis

	name1	name2
f_1	SQLiteHelper	DBUtils
f_2	SQLiteHelper	DBHelper
f_3	getWritableDB	db
f_4	getWritableDB	instance
f_5	DBUtils	instance
f_6	DBHelper	db
f_7	…	…

Features (with candidate names)

> 100,000

Actual graphs have > 1,000 nodes



Dependency graphs

Train model

	name1	name2	weight
f_1	SQLiteHelper	DBUtils	0.3
f_2	SQLiteHelper	DBHelper	0.2
f_3	getWritableDB	db	0.7
f_4	getWritableDB	instance	0.4
f_5	DBUtils	instance	0.5
f_6	DBHelper	db	0.4
f_7	…	…	…

Compute ***weights*** that maximize $P(\vec{O} = \vec{o_i} | \vec{K} = \vec{k_i})$ for all training samples $(\vec{o_i}, \vec{k_i})$

DeGuard: System Overview

Open-source, unobfuscated APKs

Learning phase

Static analysis

Training

Probabilistic model $P(\blacksquare|\blacksquare)$

Prediction phase

```
class a extends SQLiteHelper {
  SQLiteDatabase b;
  public a(Context ctx) {
    b = getWritableDB();
  }
}
```

Obfuscated code

Static analysis

MAP inference

Transform

```
class DBHelper extends SQLiteHelper{
  SQLiteDatabase db;
  public DBHelper(Context ctx) {
    db = getWritableDB();
  }
}
```

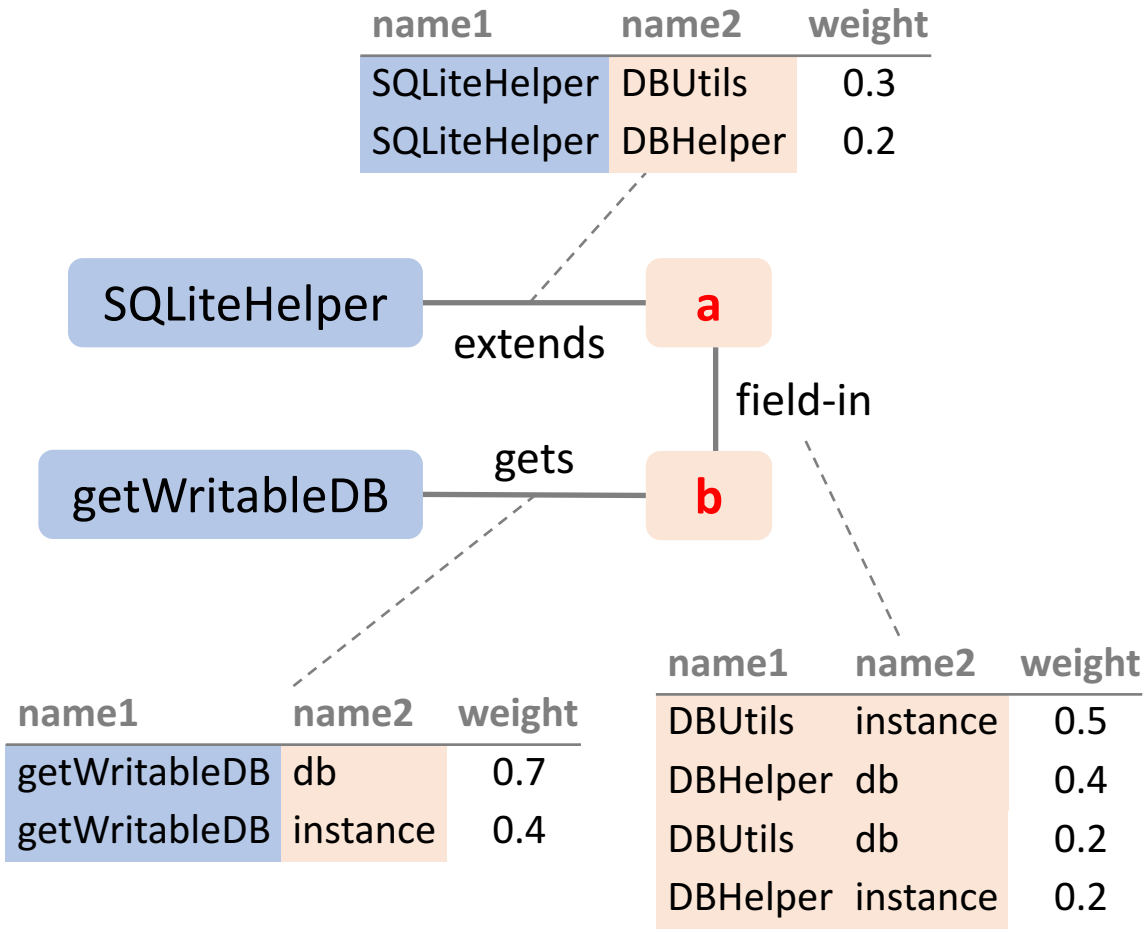
De-obfuscated code

Prediction Phase

```
class a extends SQLiteHelper {
  SQLiteDatabase b;
  public a(Context ctx) {
    b = getWritableDB();
  }
}
```

Obfuscated Code

Static analysis



Prediction Phase

```
class a extends SQLitel
  SQLiteDatabase b;
  public a(Context ctx
    b = getWritableDB(
  }
}
```

Obfu

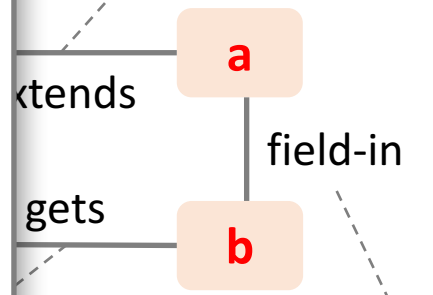
MAP Inference

$$\vec{o} = \underset{\vec{o}' \in \Omega}{argmax}\, P\left(\vec{O} = \vec{o}' \,\middle|\, \vec{K} = \vec{k}\right)$$

Candidate assignment $\vec{o}$		$P(\vec{o}\,\|\vec{k})$*
a = DBUtils	b = instance	1.2
a = DBHelper	b = db	1.3
a = DBUtils	b = db	0.8
a = DBHelper	b = instance	1.2

*Non-normalized

1	name2	weight
Helper	DBUtils	0.3
Helper	DBHelper	0.2



eight
0.7
0.4

name1	name2	weight
DBUtils	instance	0.5
DBHelper	db	0.4
DBUtils	db	0.2
DBHelper	instance	0.2

Prediction Phase

```
class a extends SQLitel
  SQLiteDatabase b;
  public a(Context ctx
    b = getWritableDB(
  }
}
```

Obfu

MAP Inference

$$\vec{o} = \underset{\vec{o}' \in \Omega}{argmax}\, P(\vec{O} = \vec{o}' \mid \vec{K} = \vec{k})$$

Candidate assignment $\vec{o}$		$P(\vec{o} \mid \vec{k})$*
a = DBUtils	b = instance	1.2
a = DBHelper	b = db	1.3
a = DBUtils	b = db	0.8
a = DBHelper	b = instance	1.2

*Non-normalized

1	name2	weight
Helper	DBUtils	0.3
Helper	DBHelper	0.2

xtends

gets

a

field-in

b

eight

0.7

0.4

name1	name2	weight
DBUtils	instance	0.5
DBHelper	db	0.4
DBUtils	db	0.2
DBHelper	instance	0.2

Prediction Phase

```
class a extends SQLiteHelper {
  SQLiteDatabase b;
  public a(Context ctx) {
    b = getWritableDB();
  }
}
```

Obfuscated Code

```
class DBHelper extends SQLiteHelper {
  SQLiteDatabase db;
  public DBHelper(Context ctx) {
    db = getWritableDB();
  }
}
```

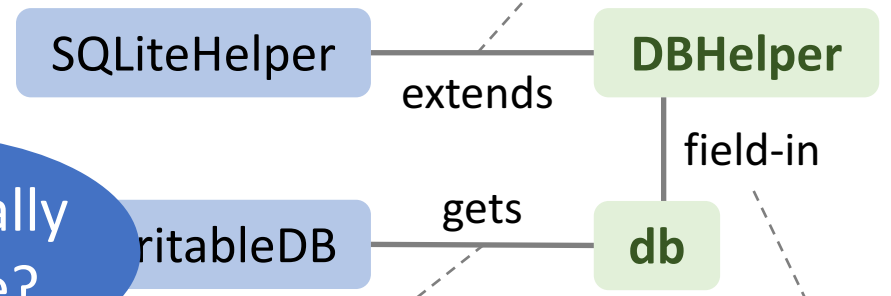
Deobfuscated Code

Static analysis

Semantically the same?

Transform

name1	name2	weight
SQLiteHelper	DBUtils	0.3
SQLiteHelper	DBHelper	0.2



name1	name2	weight
getWritableDB	db	0.7
getWritableDB	instance	0.4

name1	name2	weight
DBUtils	instance	0.5
DBHelper	db	0.4
DBUtils	db	0.2
DBHelper	instance	0.2

Semantics-Preserving De-obfuscation Constraints

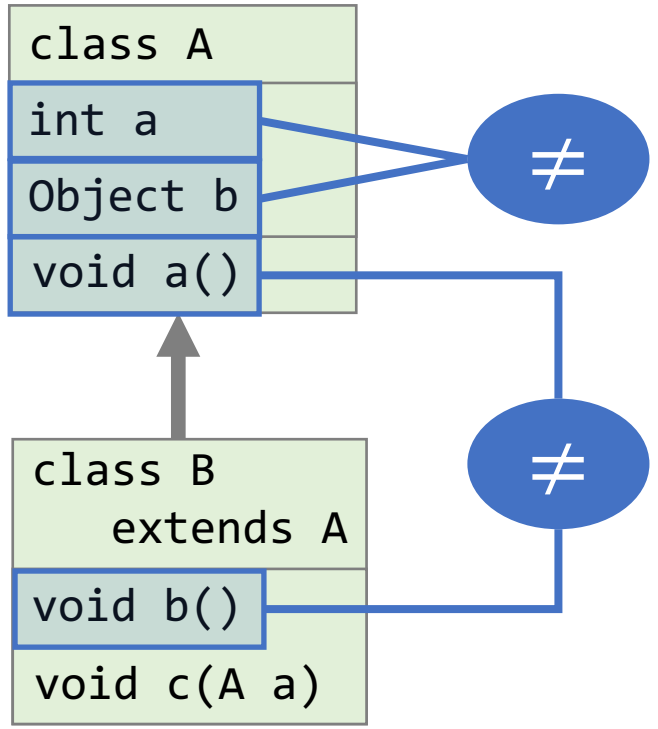
Freely renaming fields/variables/methods may **change** the application's **semantics**

Syntactic constraints

e.g. *"Fields within a class must have distinct names"*

Semantic constraints

e.g. *"Method overloads must be preserved"*



DeGuard: System Overview

Open-source, unobfuscated APKs

Learning phase

Static analysis

Training

Probabilistic model $P(\blacksquare|\blacksquare)$

Prediction phase

```
class a extends SQLiteHelper {
  SQLiteDatabase b;
  public a(Context ctx) {
    b = getWritableDB();
  }
}
```

Obfuscated code

Static analysis

MAP inference

Transform

```
class DBHelper extends SQLiteHelper{
  SQLiteDatabase db;
  public DBHelper(Context ctx) {
    db = getWritableDB();
  }
}
```

De-obfuscated code

DeGuard Implementation

DeGuard Implementation

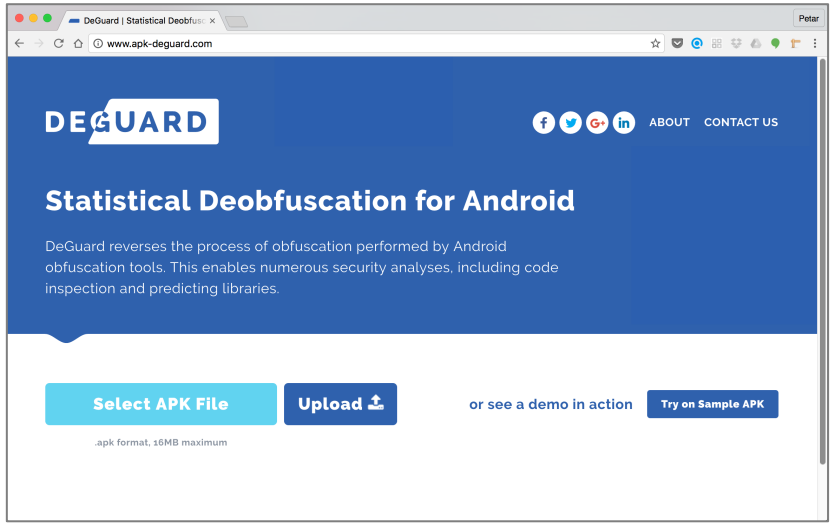
Static Analysis

- Static analysis framework for Java and Android

Learning and MAP Inference

- Scalable open-source framework for structured prediction
- Open-source: http://nice2predict.org

- Training data: 2K open-source, unobfuscated Android applications

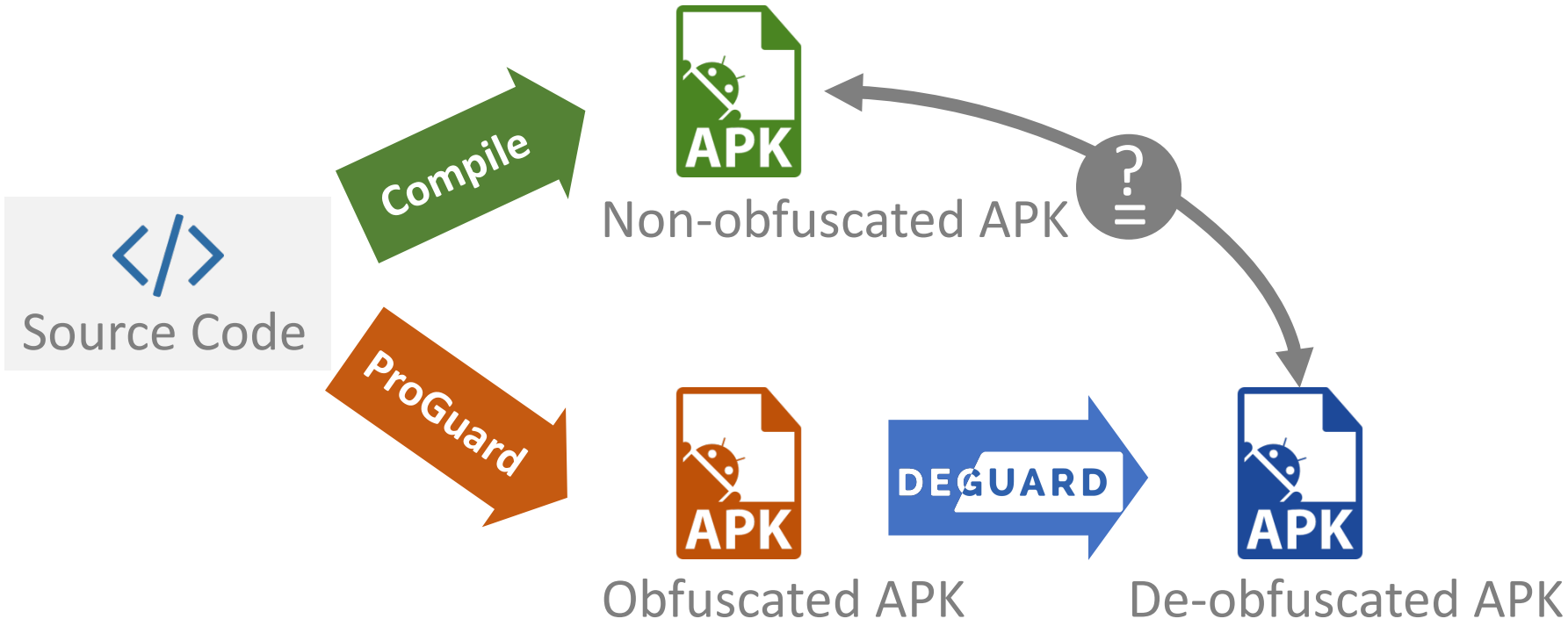


www.apk-deguard.com

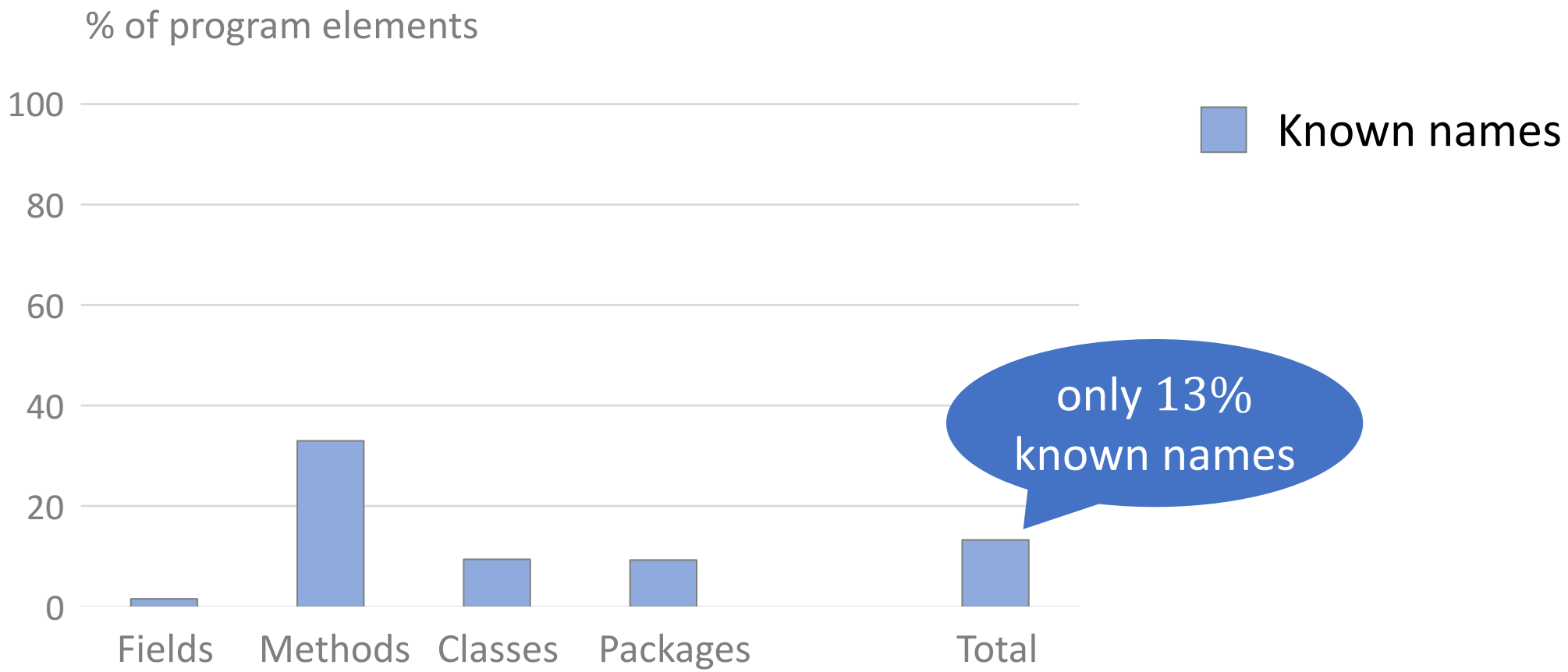
Evaluation

1. Can DeGuard reverse ProGuard?
2. Can DeGuard detect third-party libraries?
3. Is DeGuard useful for malware inspection?

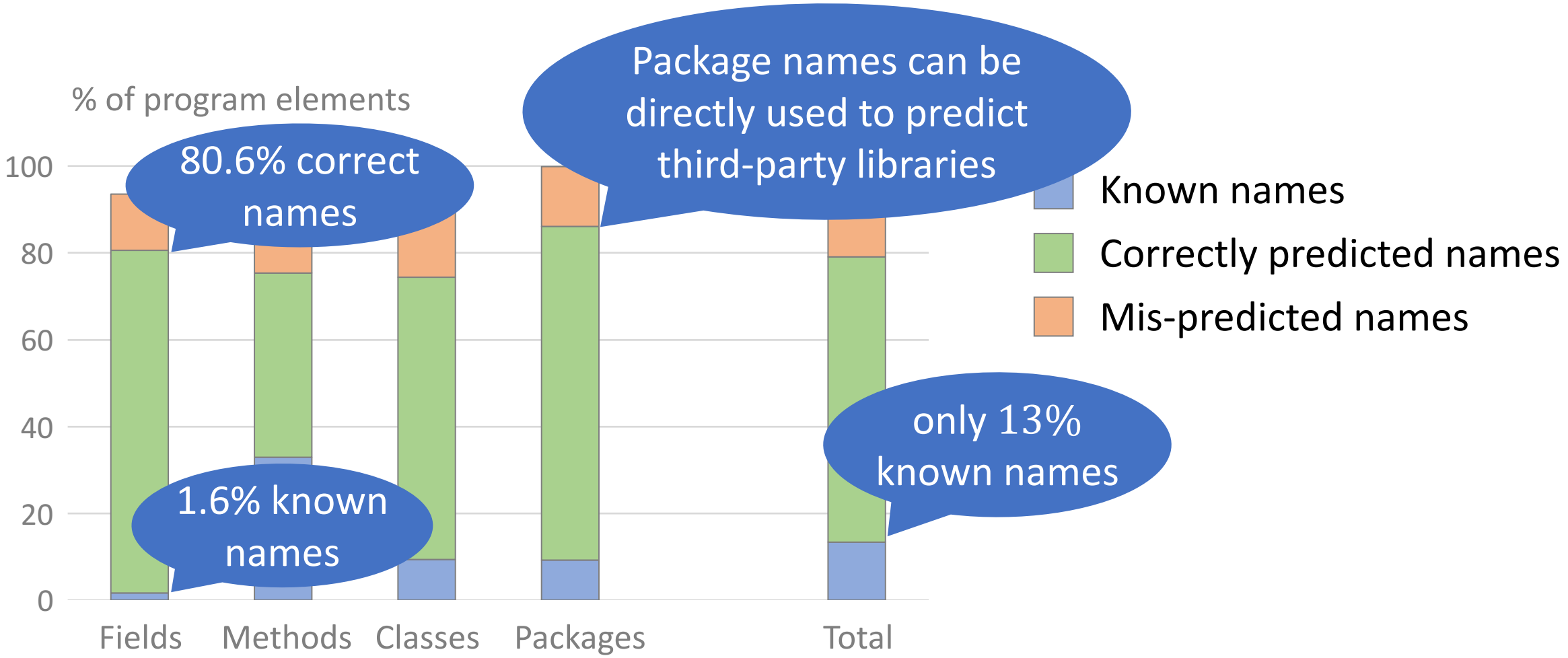
ProGuard Experiment



After Obfuscation

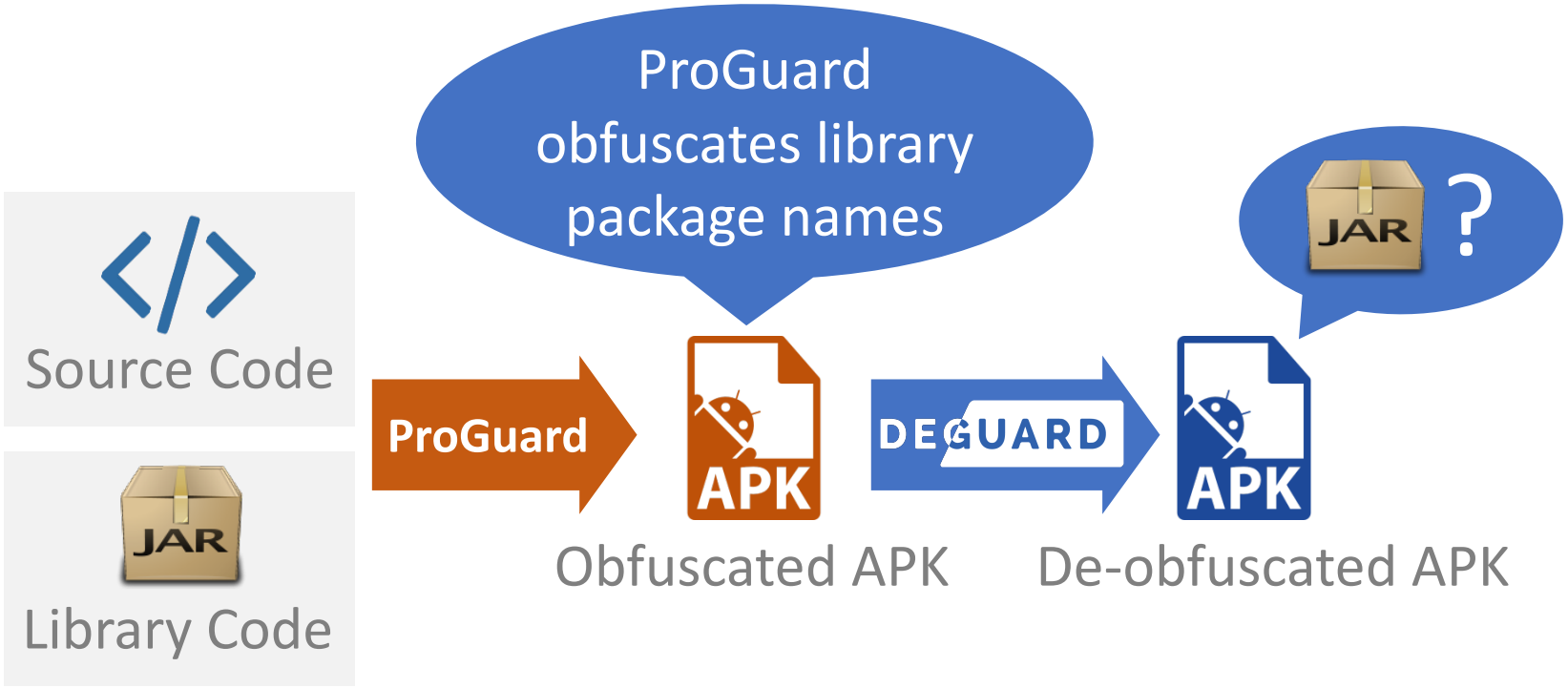


Can DeGuard Reverse ProGuard?



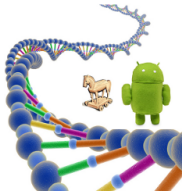
80% of the names are identical to the original ones

Can DeGuard Detect Third-Party Libraries?



Precision: 93.1% Recall: 91%

Is DeGuard Useful for Malware Inspection?



We de-obfuscated all samples from
the Android Malware Genome Project

```
class d {
  String a = System.getProperty(..)
  char[] b;
  byte [] c;
  byte[] a(String) {..}
}
```

Malware Sample

DEGUARD

```
class Base64 {
  String NL = S          perty(..)
  char[] ENC;
  byte [] DEC;
  byte[] decode(String) {..}
}
```

Base64 Decoder

De-obfuscated Malware Sample

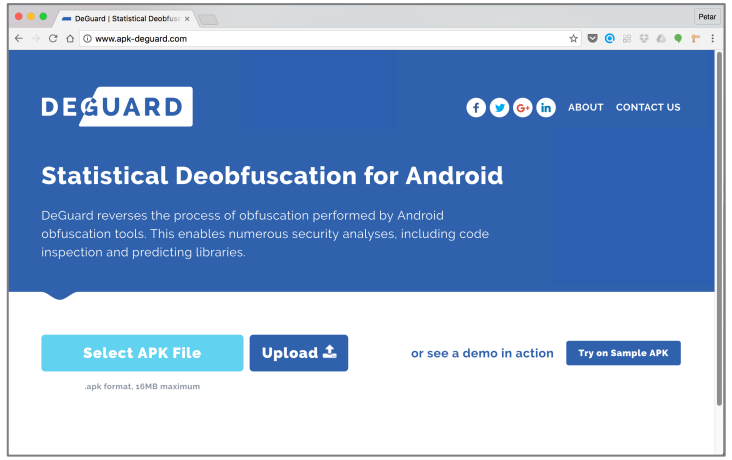
- ✓ Reveals string decoders
- ✓ Reveals classes that handle sensitive data (e.g. Location)
- ⊘ Hard to handle heavily-obfuscated code (e.g. reflection)

Summary

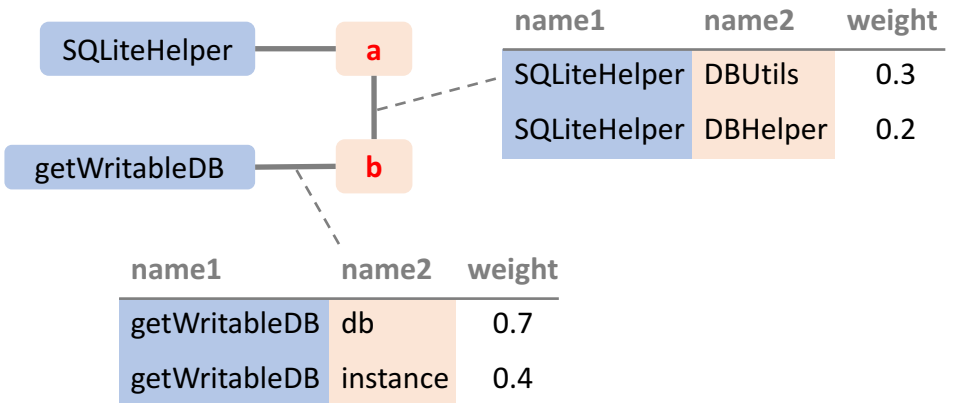
```
package com.example.dbhelper

class DBHelper extends
SQLiteHelper {
  SQLiteDatabase

  public DBHel
    db = getWri
  }

  Cursor execSQL(String str) {
    return db.rawQuery(str);
```

```
package a.b.c

class a extends
SQLiteHelper {
                e b;

                ext ctx) {
                ableDB();
  }

  Cursor c(String str) {
    return b.rawQuery(str);
```

DEGUARD



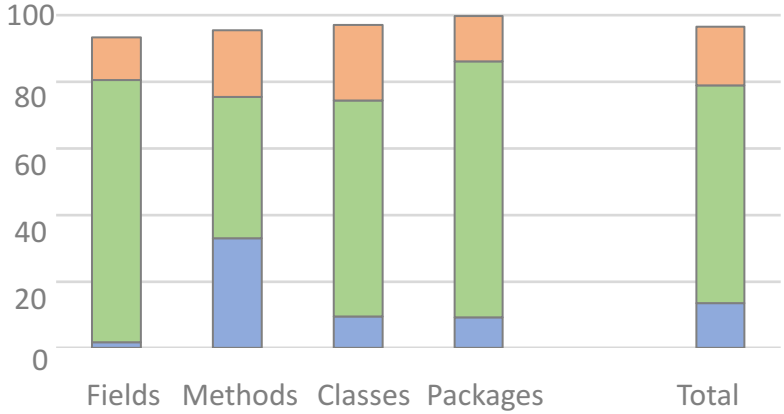
Try online: www.apk-deguard.com



name1	name2	weight
SQLiteHelper	DBUtils	0.3
SQLiteHelper	DBHelper	0.2

name1	name2	weight
getWritableDB	db	0.7
getWritableDB	instance	0.4

Probabilistic Models



High Prediction Accuracy

For more info: http://plml.ethz.ch / http://srl.inf.ethz.ch