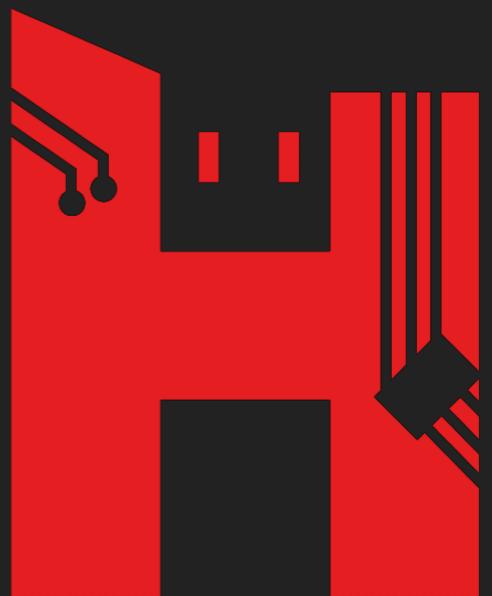


DEXCALIBUR

Hook it yourself !



WHO AM I ?

Georges-B. MICHEL // @FrenchYeti

- ▶ yeti@0xff.ninja
- ▶ Working as Software Security evaluator
- ▶ Author of Dexcalibur, CakkofonyJS, and more ...
- ▶ I ❤️ reverse obfuscated Android apps, HCE Payment applications, Trusted Applications, ARM binaries & develop security tools
- ▶ Frida addict



WHAT IS CODE INSTRUMENTATION ?

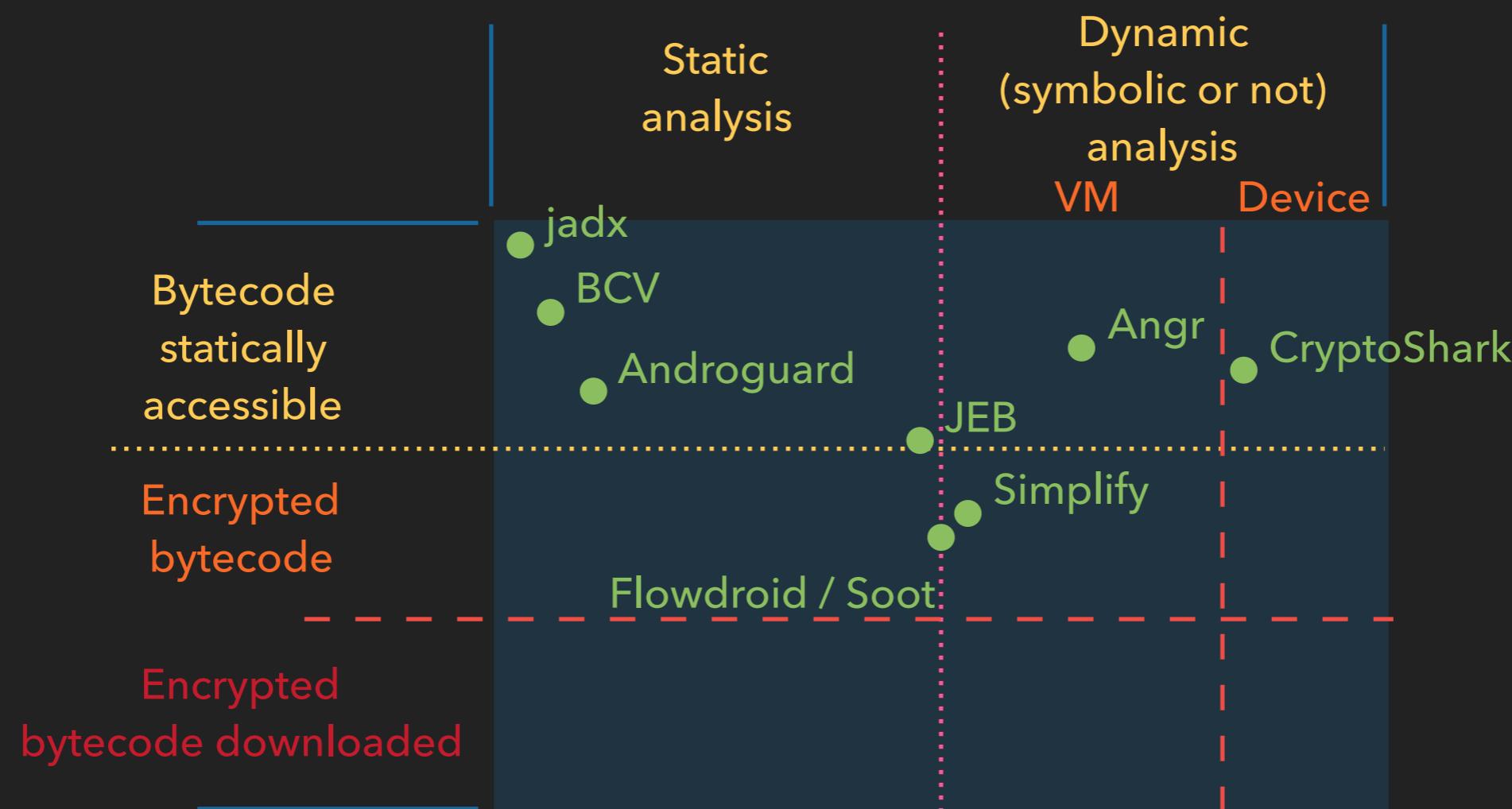
... injecting additional instructions into a computer program without requiring to modify original source code ...

Translated from wikipedia FR

WHAT ARE THE PURPOSES ?

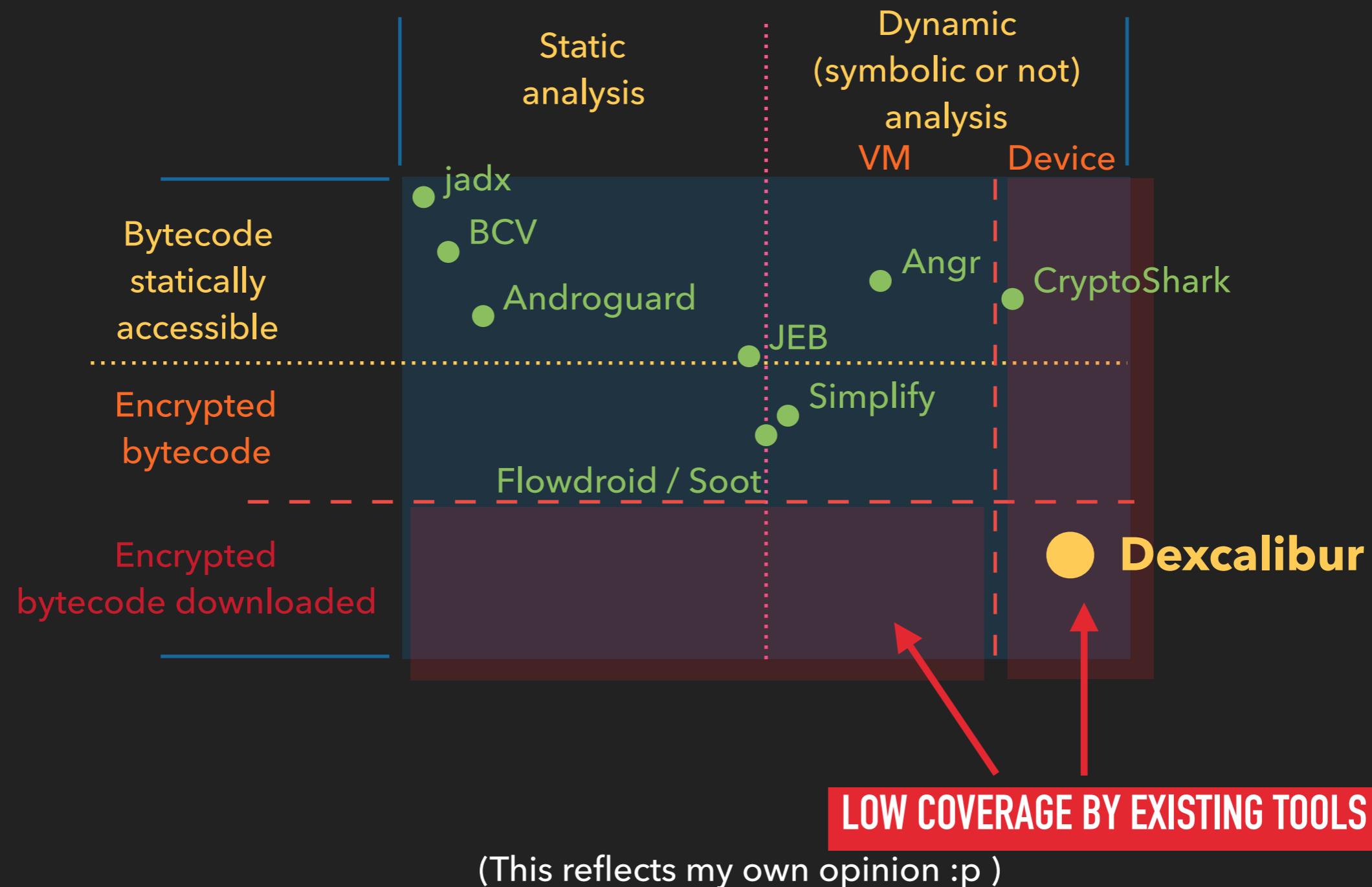
- ▶ Optimize unit test coverage
- ▶ Performance optimization through profiling
- ▶ Reverse : What happens when the application performs some actions ?
- ▶ More ...

ANDROID REVERSE ENGINEERING TOOLS



(This reflects my own opinion :p)

ANDROID REVERSE ENGINEERING TOOLS



STATIC OBFUSCATION BASICS

- ▶ Is "java.lang.StringSplitter" (for example) an internal class ?

```
:goto_1
invoke-static/range {v0 .. v0}, Lcom/eshard/crackme/act
const/4 v0, 0x1
move-result-object v0

goto/32 :goto_3

:goto_2
const-string v0, "3780b8133459f5a028d742efbcfc7d2d"

goto/32 :goto_1

:goto_3
invoke-static {v0}, Lcom/eshard/crackme/activities/Crac
```

WHAT HAPPENS ?

static	QSLrjaCtdwZMXxl(<java.lang.String><int>)<char>
static	RugleAJCMcOYiED(<java.lang.String>)<int>
static	UAozYOTEbCdtGyS(<java.lang.String><int>)<char>
static	VqxuLoJHbjvgTRW(<java.lang.String>)<javax.crypto.SecretKeyFactory>
static	XjYcgdUCozPhGQw(<java.lang.StringBuilder><char>)<java.lang.StringBuilder>
static	XvyipIDZgVArked(<java.lang.String>)<byte>[]
static	YIXqJEovWdtLHcU(<java.lang.String>)<int>
private static	a03674e6c(<java.lang.String>)<java.lang.String>
static	bVjSsqAehlQJpnT(<java.lang.String>)<int>
static	bXEVATfiSCtzpkh(<java.lang.String>)<javax.crypto.Cipher>
static	bwXLNAdkapcUmDB(<java.lang.StringBuilder>)<java.lang.String>
static	convertToString(<java.lang.String>)<java.lang.String>
static	dTxzoBgvOZreERy(<java.lang.StringBuilder><char>)<java.lang.StringBuilder>
static	deDnrvZgyKlqtmM(<java.lang.String><int>)<char>
private static	decode(<java.lang.String>)<byte>[]
private static	getStorageEncryption(<int><java.lang.String>)<javax.crypto.Cipher>
static	igDHotavmxKEbdh(<java.lang.String>)<int>
static	ihxCrtGAOVjZsgR(<java.lang.String><java.lang.String>)<byte>[]
static	klbUpHVXTuWmenL(<java.lang.StringBuilder>)<java.lang.String>
static	ldKjpFnmfkBtvxi(<java.lang.String>)<int>
static	lwLuoCTSFzaYIVf(<java.lang.String><int>)<char>
static	mlhOSnrEbkoFsXa(<java.lang.StringBuilder><char>)<java.lang.StringBuilder>
static	nKPabividtuJCyq(<java.lang.StringBuilder><char>)<java.lang.StringBuilder>
static	orteEvxHPWFaXNB(<java.lang.String><int>)<char>
static	quJicZulrtsfMTIH(<java.lang.String><int>)<char>
static	qoYcZGfwieahVQd(<java.lang.Object>)<java.lang.String>
static	rzmCKugynlUZGEh(<java.lang.String><int>)<char>
static	sJKaZhFGICTUuwz(<java.lang.String>)<int>

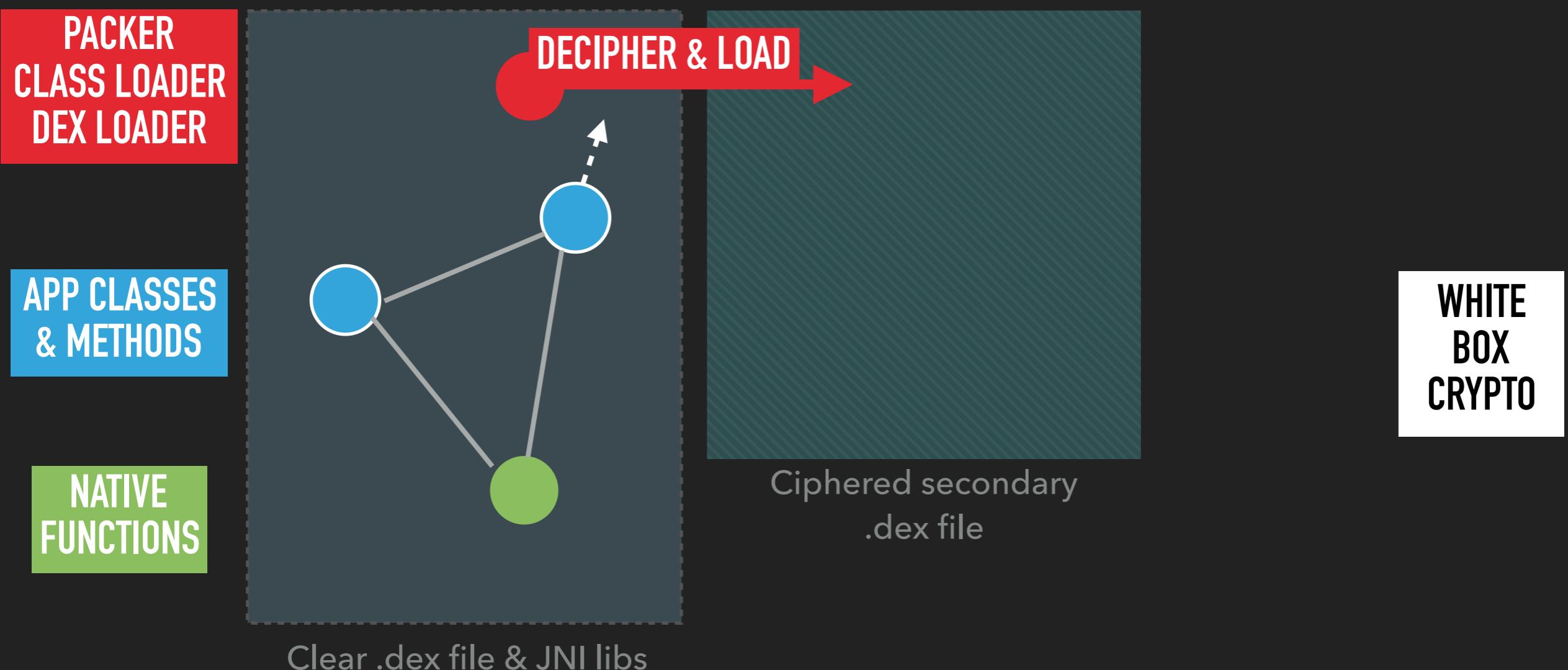
STATIC OBFUSCATION BASICS

```
2 goto/32 :goto_7b
3
4
5 :goto_0
6 goto/32 :goto_75
7
8
9 :goto_1
10 goto/16 :goto_27
11
12 :sswitch_0
13 .line 112
14 goto/32 :goto_79
15
16
17 :goto_2
18 invoke-static {v4, v0},
19 goto/32 :goto_33
20
21
22 :goto_3
23 if-ge v3, v8, :cond_0
24 goto/32 :goto_26
25
26
27
28 :cond_0
29 goto/32 :goto_5c
30
31
32 :goto_4
33 invoke-static {p0, v6},
34 move-result v6
35 goto/32 :goto_77
36
37
38 :cond_1
39 goto/5
40 const/16 v0, 0xd
41 .line 101
42 goto/32 :goto_23
43
44 :goto_6
45 if-lt v2, v6, :cond_1
46 goto/32 :goto_38
47
48
49
50 :cond_1
51 goto/32 :goto_3f
52
53
54 :goto_7
55 invoke-static {v6, v3}, Landroid/content/res/abltMZGC;->XjYcgdUCozPhGQw(Lj<
56 move-result-object v6
57 goto/32 :goto_53
58
59
60 :goto_8
61 if-lt v2, v6, :cond_2
62 goto/32 :goto_38
63
64
65 :cond_2
66 goto/32 :goto_20
67
68
69
70 :goto_9
71 add-int/lit8 v6, v2, 0x1
72 goto/32 :goto_5b
73
74
```

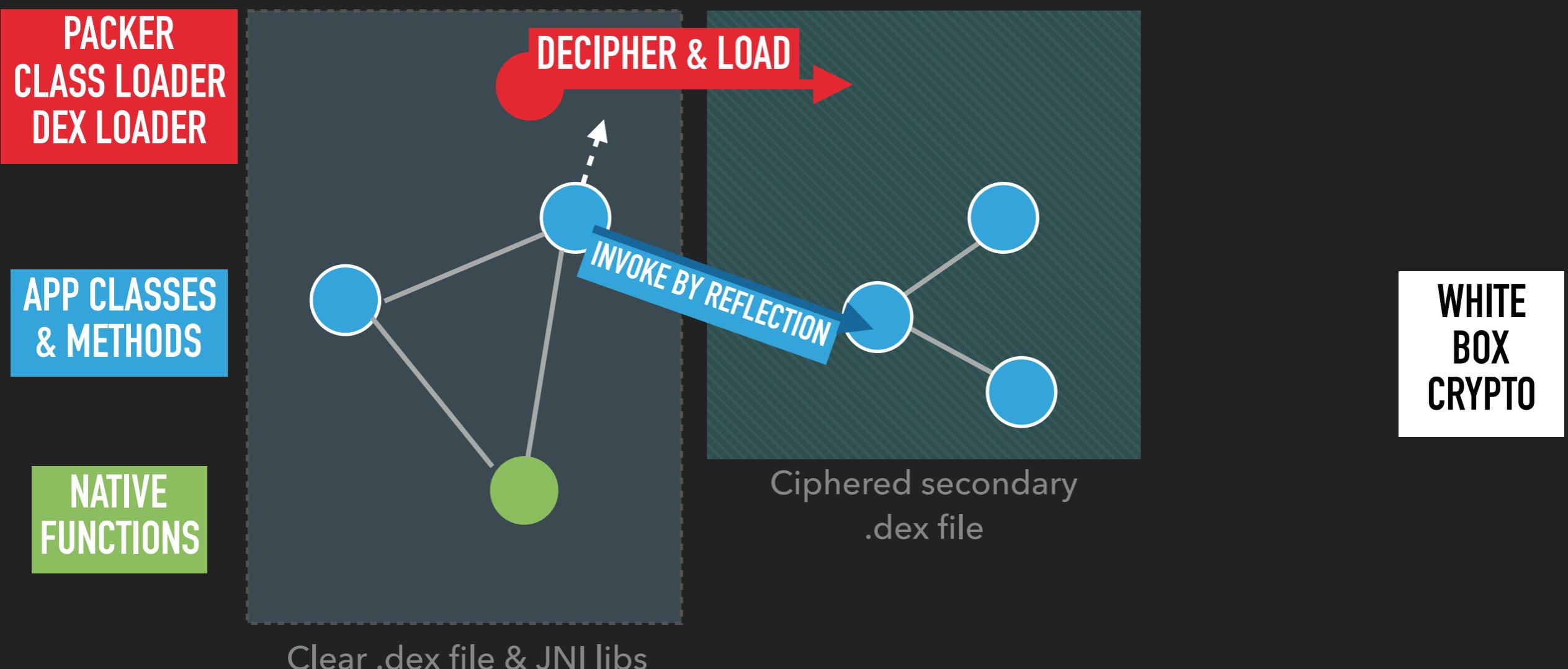
ETC ...

... ~ 124 goto(s)

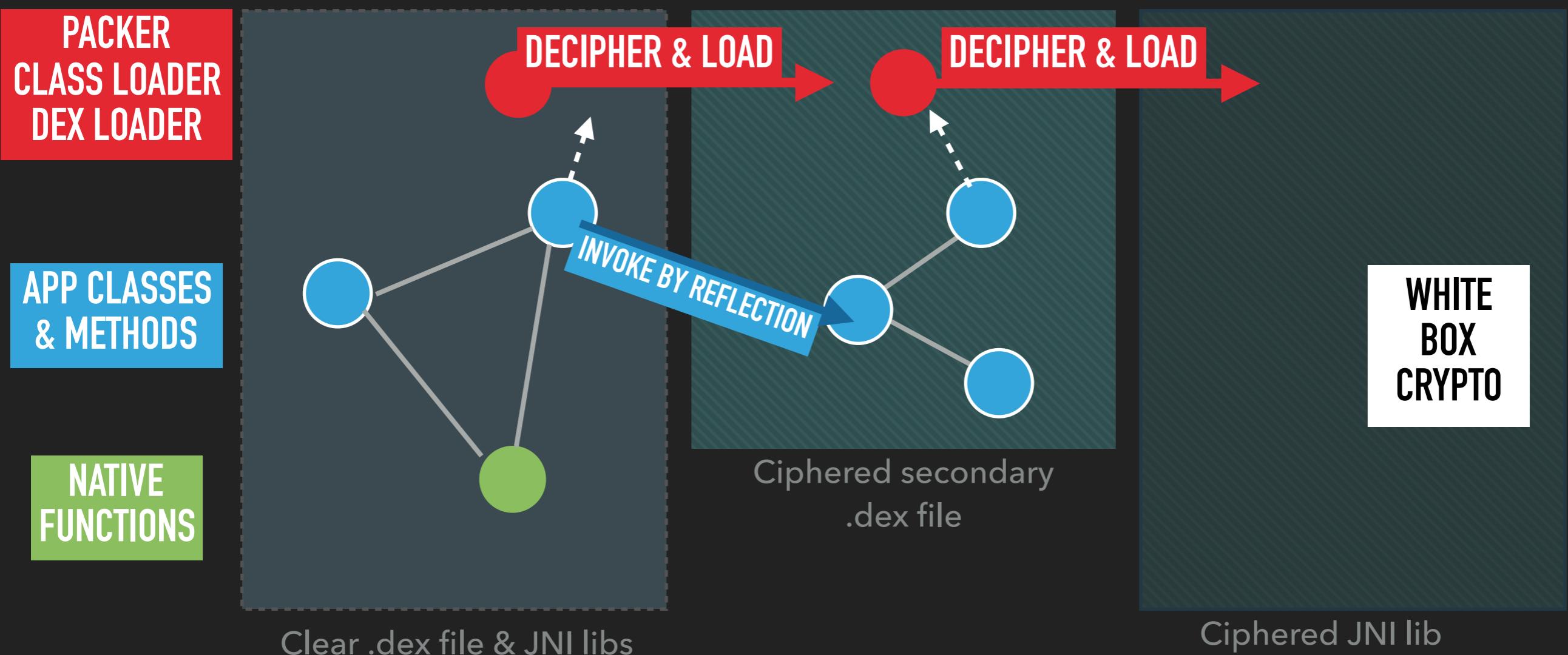
LET'S IMAGINE AN OBFUSCATED MULTI-DEX APPLICATION



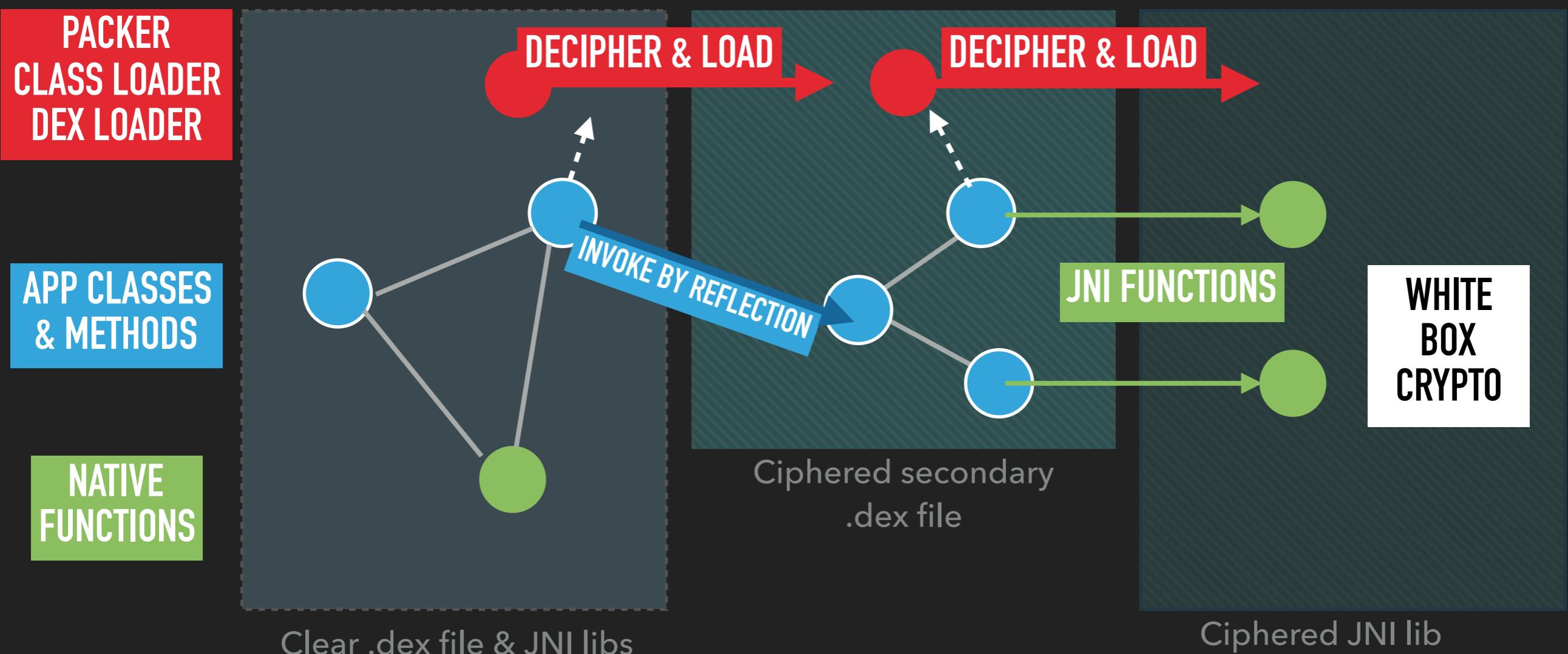
LET'S IMAGINE AN OBFUSCATED MULTI-DEX APPLICATION



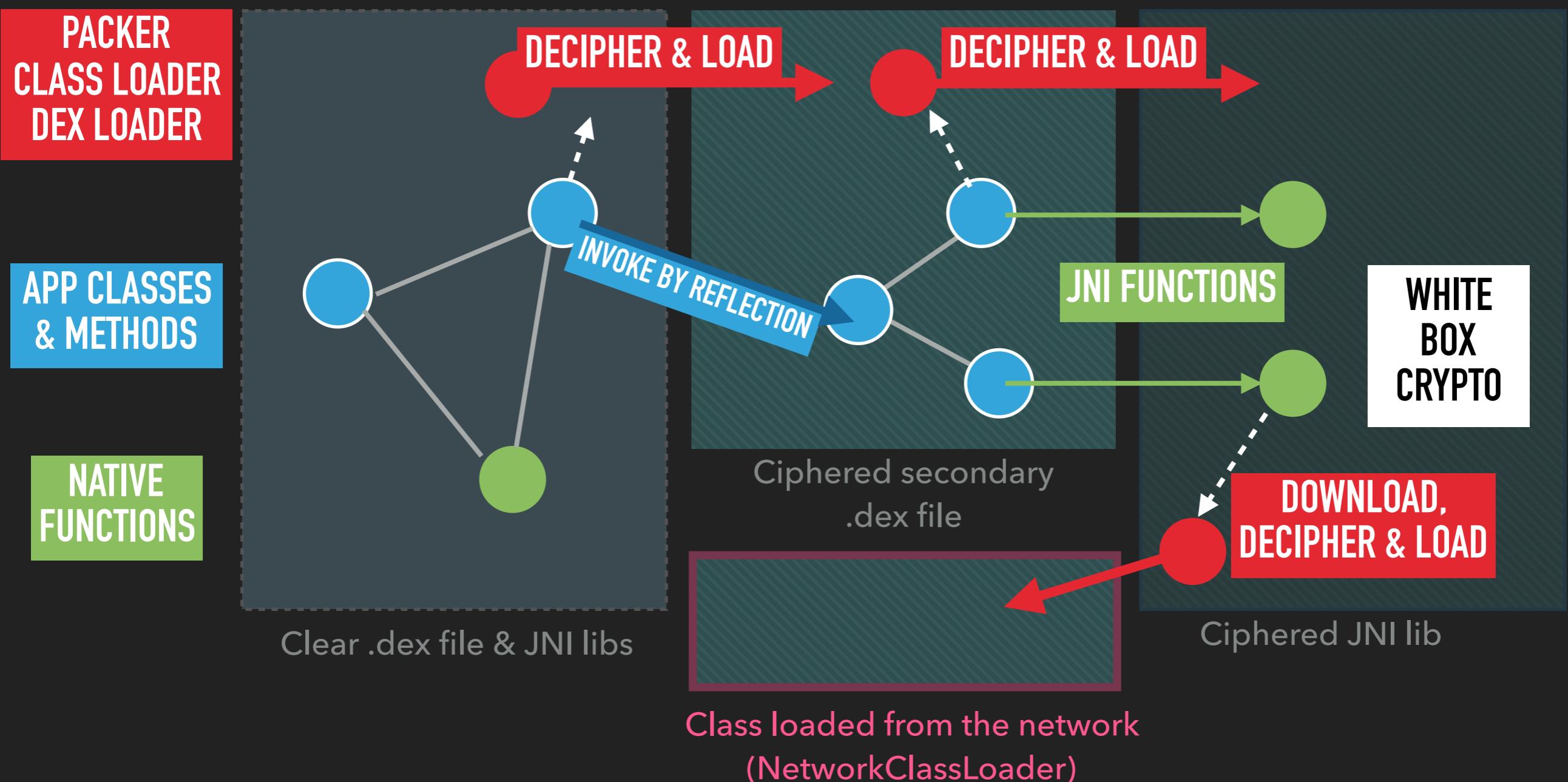
LET'S IMAGINE AN OBFUSCATED MULTI-DEX APPLICATION



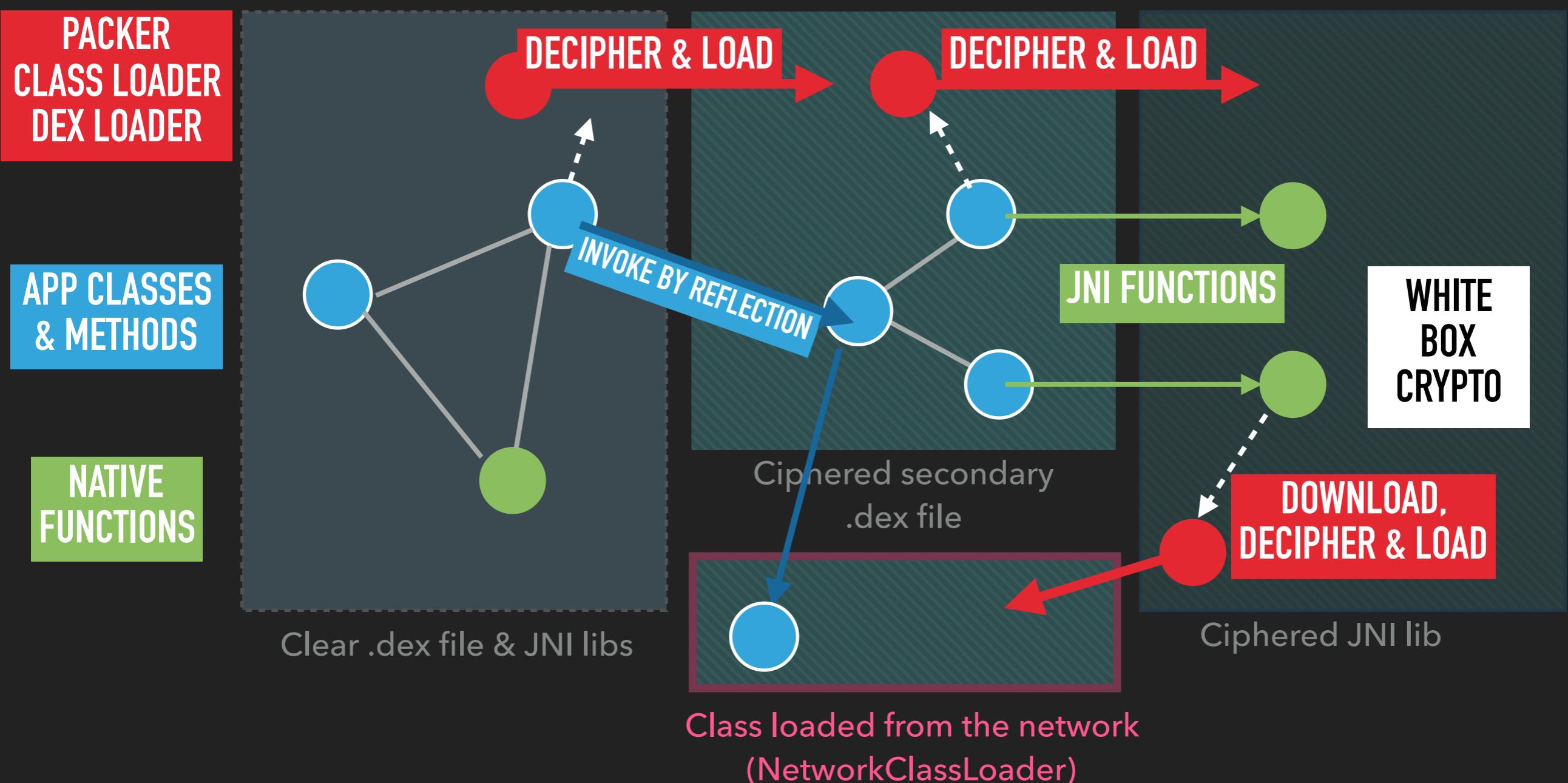
LET'S IMAGINE AN OBFUSCATED MULTI-DEX APPLICATION



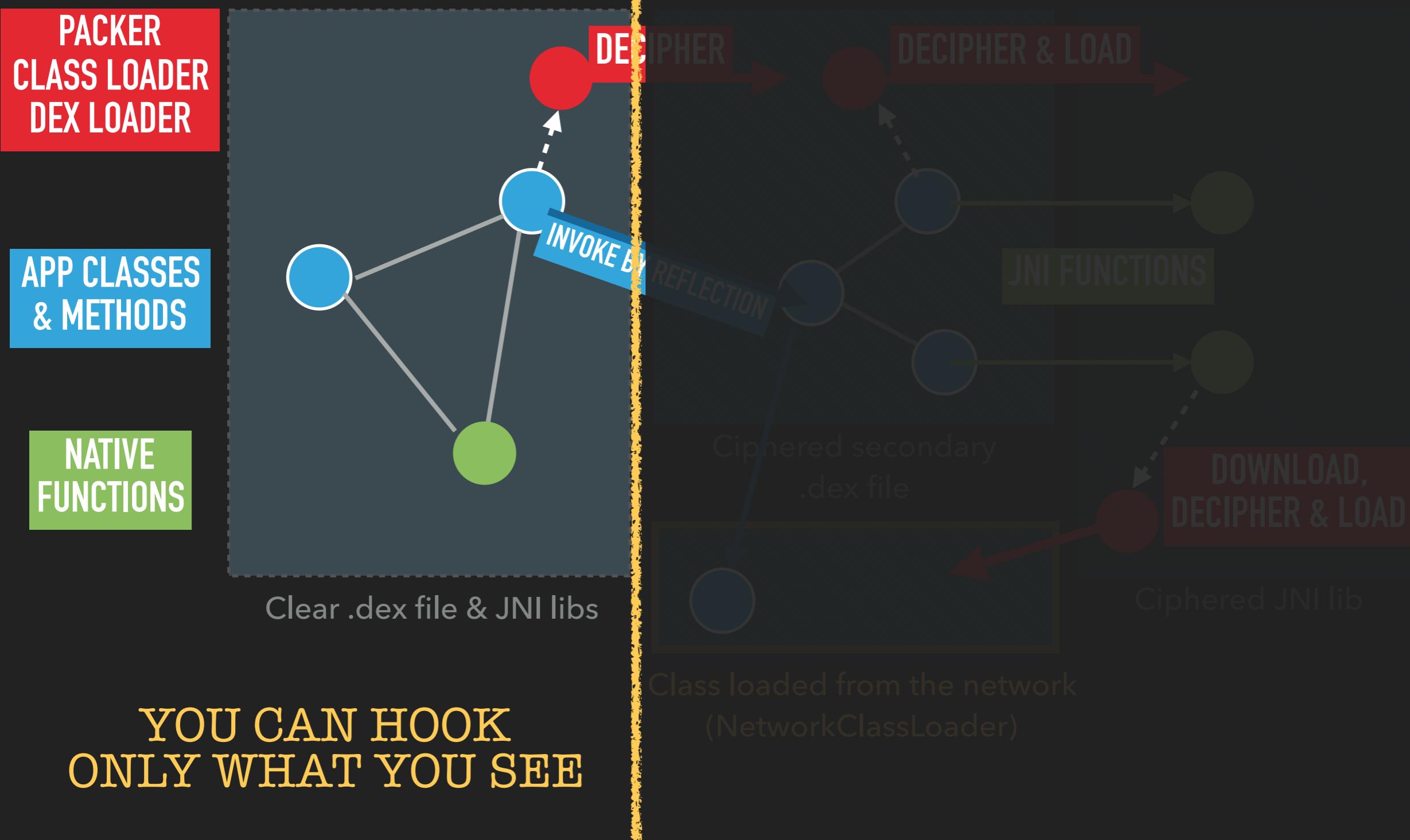
LET'S IMAGINE AN OBFUSCATED MULTI-DEX APPLICATION



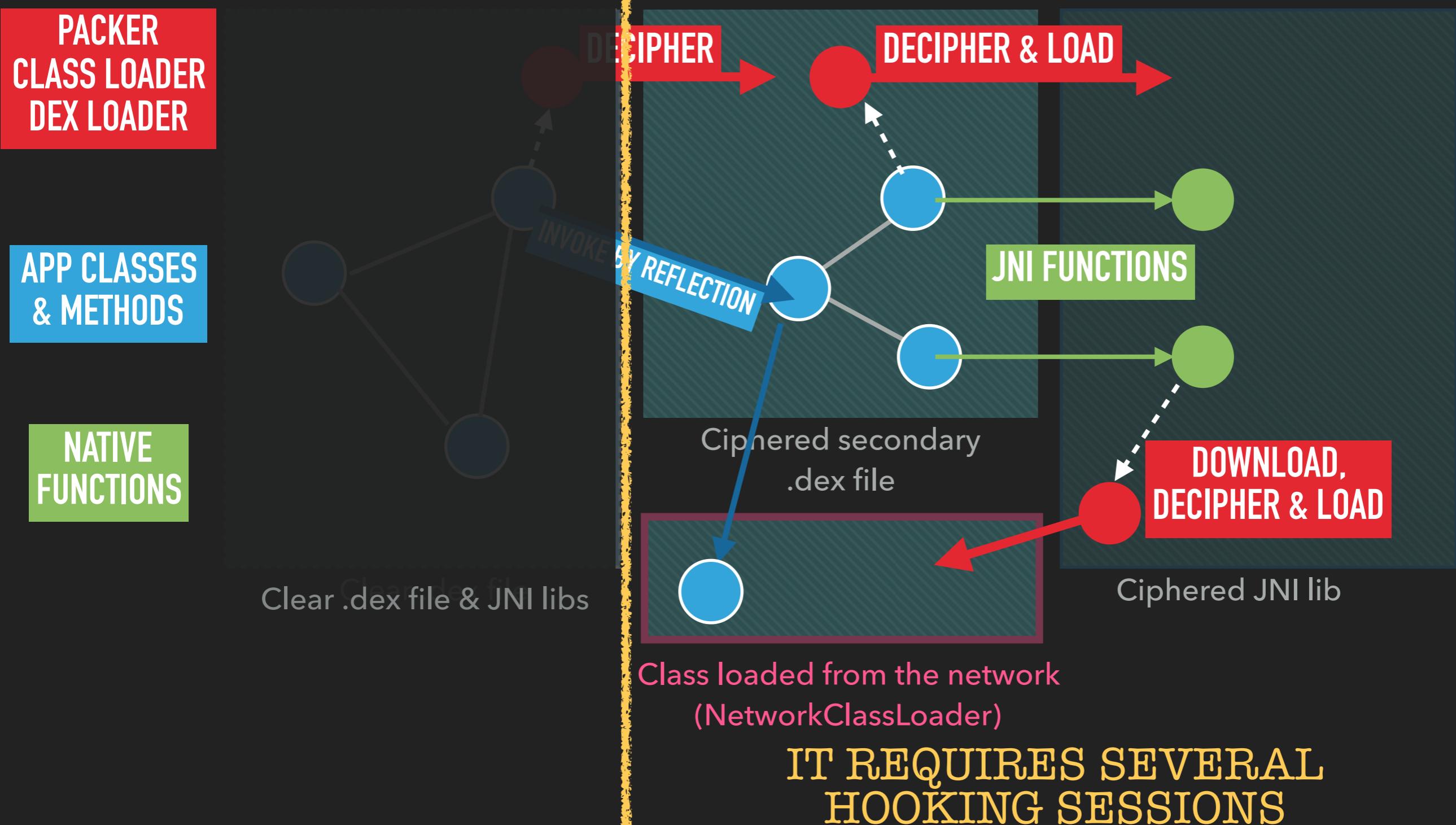
LET'S IMAGINE AN OBFUSCATED MULTI-DEX APPLICATION



WHAT CAN I HOOK ?



WHAT IS INTERESTING TO HOOK ?



APPLICATION SIZE IS ANOTHER PROBLEM

- ▶ YouTube app
 - ▶ + 42 000 classes
 - ▶ + 250 000 methods
 - ▶ + 650 000 calls
 - ▶ + 3M instructions



**WHAT IS
DEXCALIBUR ?**

WHAT IS DEXCALIBUR ?

WHAT IS DEXCALIBUR ?

- ▶ DBI helper / Dynamic Application Security Testing (DAST) tool
- ▶ Free and Open-Source Android RE platform
- ▶ Extensible, comprehensive, multi-user, GUI
- ▶ Personal project started in 2018
- ▶ Few contributors (3)
- ▶ Used by :
 - ▶ several security laboratories for evaluation (HCE, CSPN, ...)
 - ▶ private CERT/CSIRT for android app triage
 - ▶ AV editors, TI companies, Android app editors (debug ?)

WHAT IS DEXCALIBUR ?

NOT JUST A TOOLBOX

WHAT IS DEXCALIBUR ?

NOT JUST A TOOLBOX

DEX DISASSEMBLER *Baksmali*

WHAT IS DEXCALIBUR ?

NOT JUST A TOOLBOX

DEX DISASSEMBLER *Baksmali*

FILE IDENTIFIERS & PARSERS

WHAT IS DEXCALIBUR ?

NOT JUST A TOOLBOX

DEX DISASSEMBLER *Baksmali*

FILE IDENTIFIERS & PARSERS

MINIMALIST SMALI VM

WHAT IS DEXCALIBUR ?

NOT JUST A TOOLBOX

DEX DISASSEMBLER *Baksmali*

FILE IDENTIFIERS & PARSERS

MINIMALIST SMALI VM

STATIC BYTECODE ANALYZER

DYNAMIC BYTECODE ANALYZER

WHAT IS DEXCALIBUR ?

NOT JUST A TOOLBOX



DEX DISASSEMBLER *Baksmali*

FILE IDENTIFIERS & PARSERS

MINIMALIST SMALI VM

STATIC BYTECODE ANALYZER

DYNAMIC BYTECODE ANALYZER

WHAT IS DEXCALIBUR ?

NOT JUST A TOOLBOX



DEX DISASSEMBLER *Baksmali*

FILE IDENTIFIERS & PARSERS

MINIMALIST SMALI VM

STATIC BYTECODE ANALYZER

DYNAMIC BYTECODE ANALYZER

INSTRUMENTATION TOOL

FRIDA

WHAT IS DEXCALIBUR ?

NOT JUST A TOOLBOX



DEX DISASSEMBLER *Baksmali*

FILE IDENTIFIERS & PARSERS

MINIMALIST SMALI VM

STATIC BYTECODE ANALYZER

DYNAMIC BYTECODE ANALYZER

INSTRUMENTATION TOOL

FRIDA

MODULAR HEURISTIC & SEARCH ENGINE

WHAT IS DEXCALIBUR ?

NOT JUST A TOOLBOX



DEX DISASSEMBLER *Baksmali*

FILE IDENTIFIERS & PARSERS

MINIMALIST SMALI VM

STATIC BYTECODE ANALYZER

DYNAMIC BYTECODE ANALYZER

INSTRUMENTATION TOOL

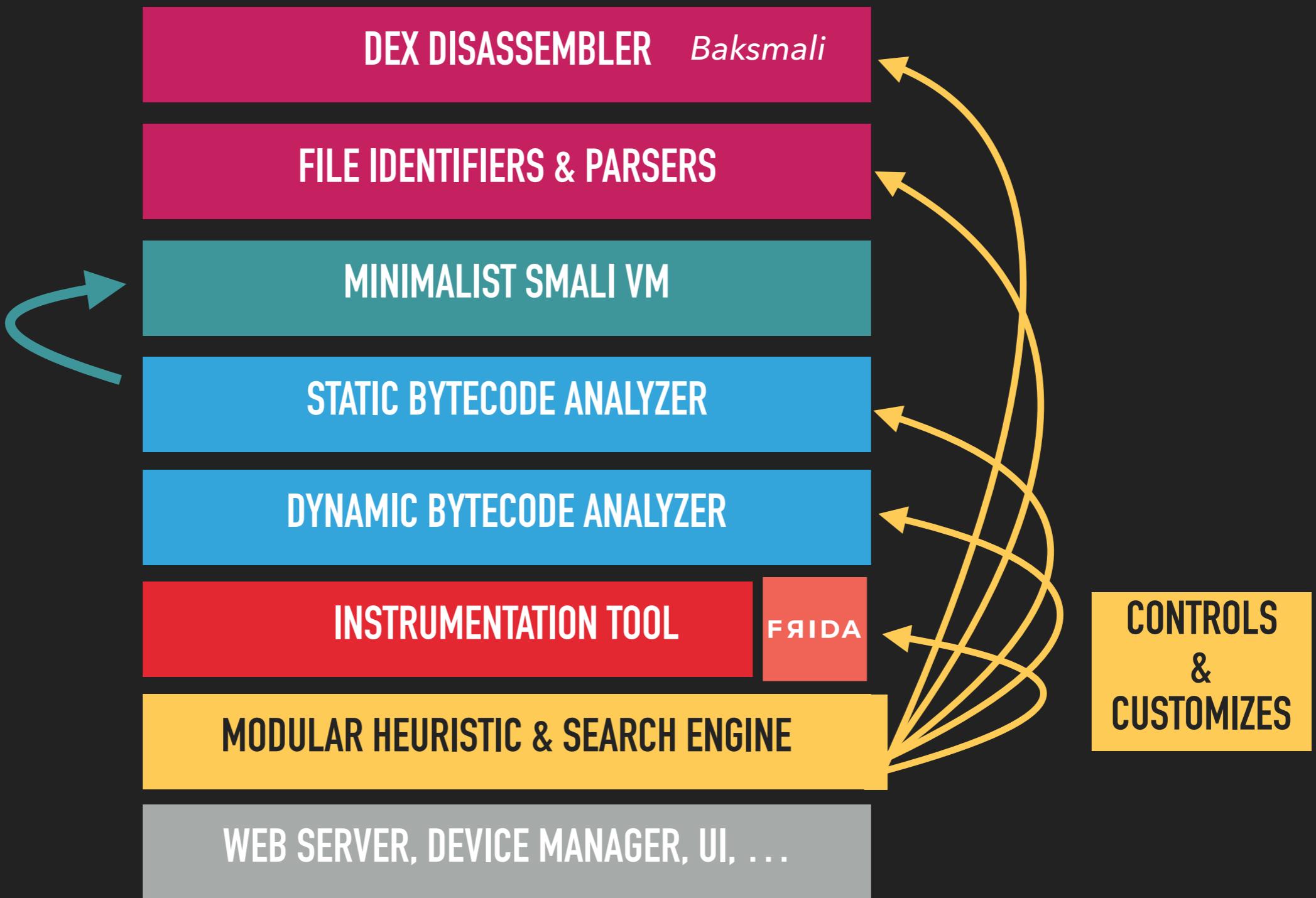
FRIDA

MODULAR HEURISTIC & SEARCH ENGINE

WEB SERVER, DEVICE MANAGER, UI, ...

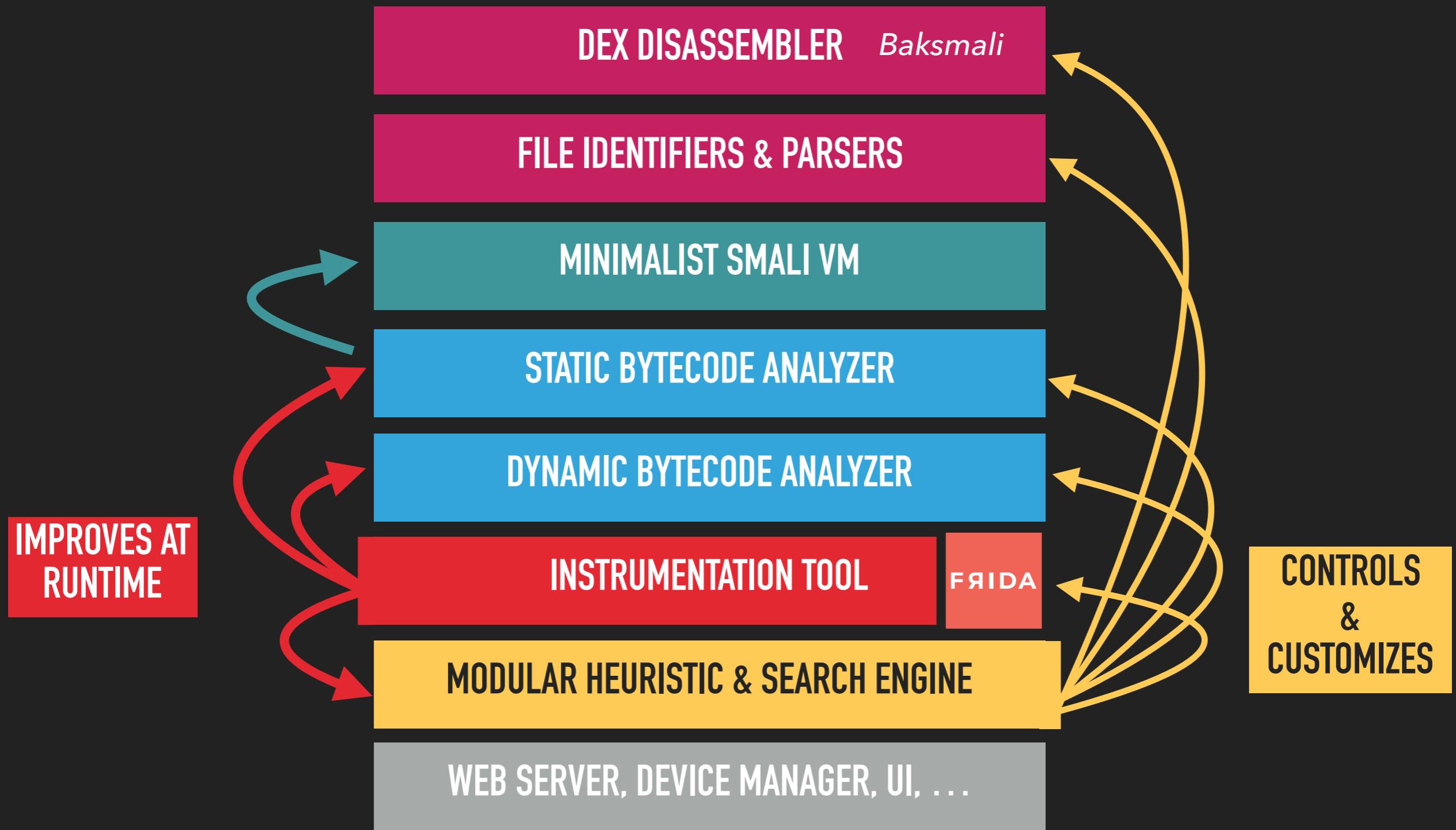
WHAT IS DEXCALIBUR ?

NOT JUST A TOOLBOX



WHAT IS DEXCALIBUR ?

NOT JUST A TOOLBOX



WHAT IS DEXCALIBUR ?

NOT JUST A TOOLBOX

DEXCALIBUR



WHAT IS DEXCALIBUR ?

HOW TO INSTALL DEXCALIBUR ?

FROM GIT

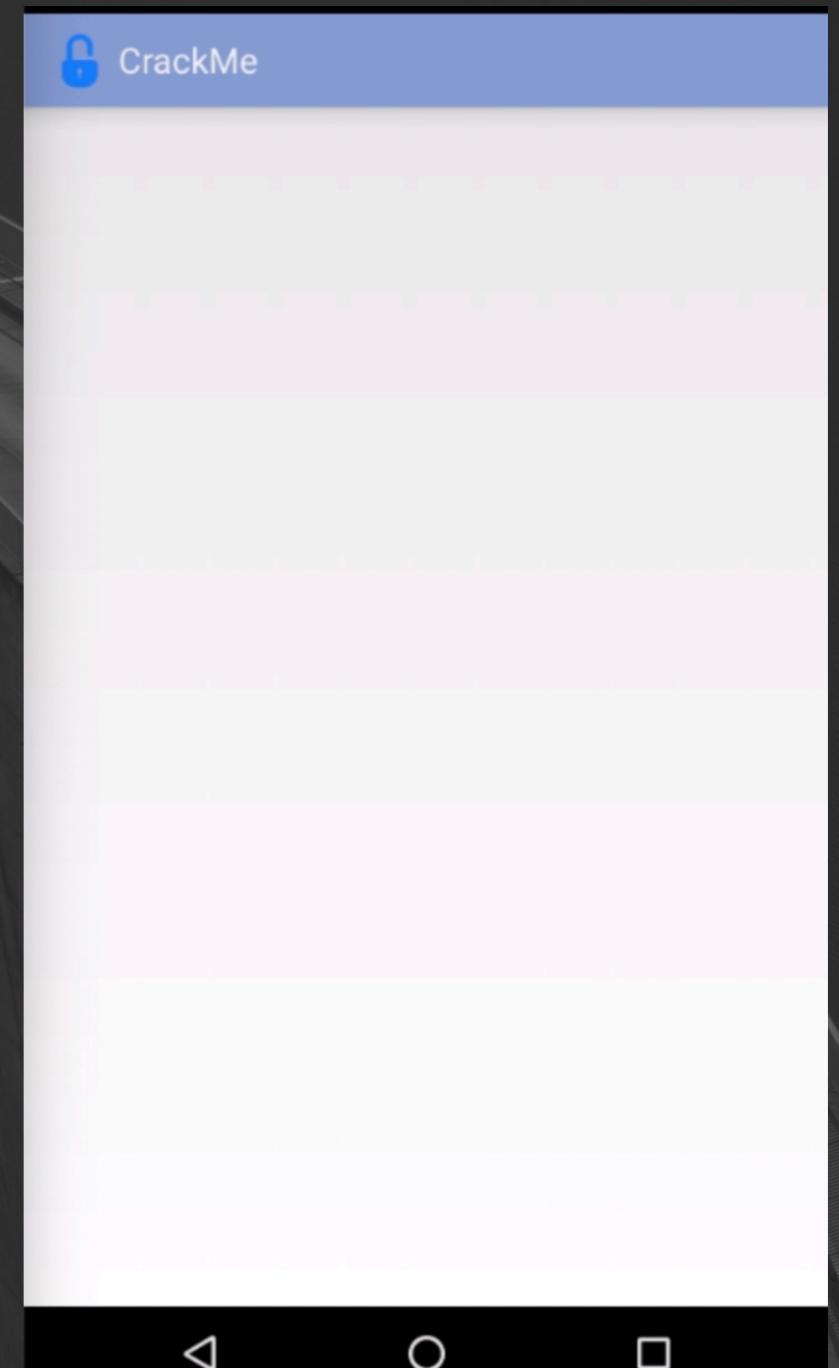
```
git clone https://github.com/FrenchYeti/dexcalibur.git
cd dexcalibur
npm install
```

FROM NPM (TO DO)

```
npm install dexcalibur
```

Why is the application not
starting correctly ?

DEMO



CASE #1

ANALYZING DEX FILE LOADED DYNAMICALLY

DETECT CLASS LOADERS STATICALLY AVAILABLE

- ▶ Before 1st run :

```
class('extends.name: ClassLoader')
```

APK min SDK version Android 7.1.0	DexFile ClassLoader	...	APP Specific ClassLoader
Device running Android 7.1.0	DexFile ClassLoader	...	Vendor specific ClassLoader
Device running Android 10	DexFile ClassLoader	...	InMemoryDex ClassLoader
			APP Specific ClassLoader

ANALYZING DEX FILE LOADED DYNAMICALLY

DETECT CLASS LOADERS STATICALLY AVAILABLE

The screenshot shows the DexCalibur interface with a search bar containing the query `class("extends.name:ClassLoader")`. Below the search bar is a row of filters: clean, xref, keep callers, keep calleds, filter by package (which is selected), filter by caller class, and Case insensitive. The main area displays a table of class loader entries:

Package	Name
dalvik.system	BaseDexClassLoader
dalvik.system	DexClassLoader
dalvik.system	PathClassLoader
java.net	URLClassLoader
java.security	SecureClassLoader

Base : Android API v24 - 7.0.0

InMemoryDexClassLoader introduced in API v26 (android 8.0)

ANALYZING DEX FILE LOADED DYNAMICALLY

GENERATE HOOK CLASS LOADERS

Hook manager

Disable all

Enable all

 Download script

 Run (spawn)

 Attach to app

 Attach to Gadget

Type	Method
 probe	android.os.Parcelable\$ClassLoaderCreator.createFromParcel(<android.os.Parcel><java.lang.ClassLoader>)<java.lang.Object>
 intercept	java.lang.Class.getMethod(<java.lang.String><java.lang.Class>[])<java.lang.reflect.Method>
 intercept	java.lang.Class.forName(<java.lang.String>)<java.lang.Class>
 intercept	dalvik.system.BaseDexClassLoader.findClass(<java.lang.String>)<java.lang.Class>
 intercept	dalvik.system.DexClassLoader.<init>(<java.lang.String><java.lang.String><java.lang.String><java.lang.ClassLoader>)<void>
 intercept	dalvik.system.DexFile.loadDex(<java.lang.String><java.lang.String><int>)<dalvik.system.DexFile>
 intercept	dalvik.system.DexFile.<init>(<java.io.File>)<void>
 intercept	dalvik.system.DexFile.<init>(<java.lang.String>)<void>

ANALYZING DEX FILE LOADED DYNAMICALLY

GENERATE HOOK CLASS LOADERS

intercept | dalvik.system.DexFile.loadDex(<java.lang.String><java.lang.String><int>)<dalvik.system.DexFile>

Hook UUID b09ce7e91589c1a71392545fb75427fb
Hooked method [dalvik.system.DexFile.loadDex\(<java.lang.String><java.lang.String><int>\)<dalvik.system.DexFile>](#)
Description empty

Hook code

Helpers: Java hook ▾ Native hook ▾

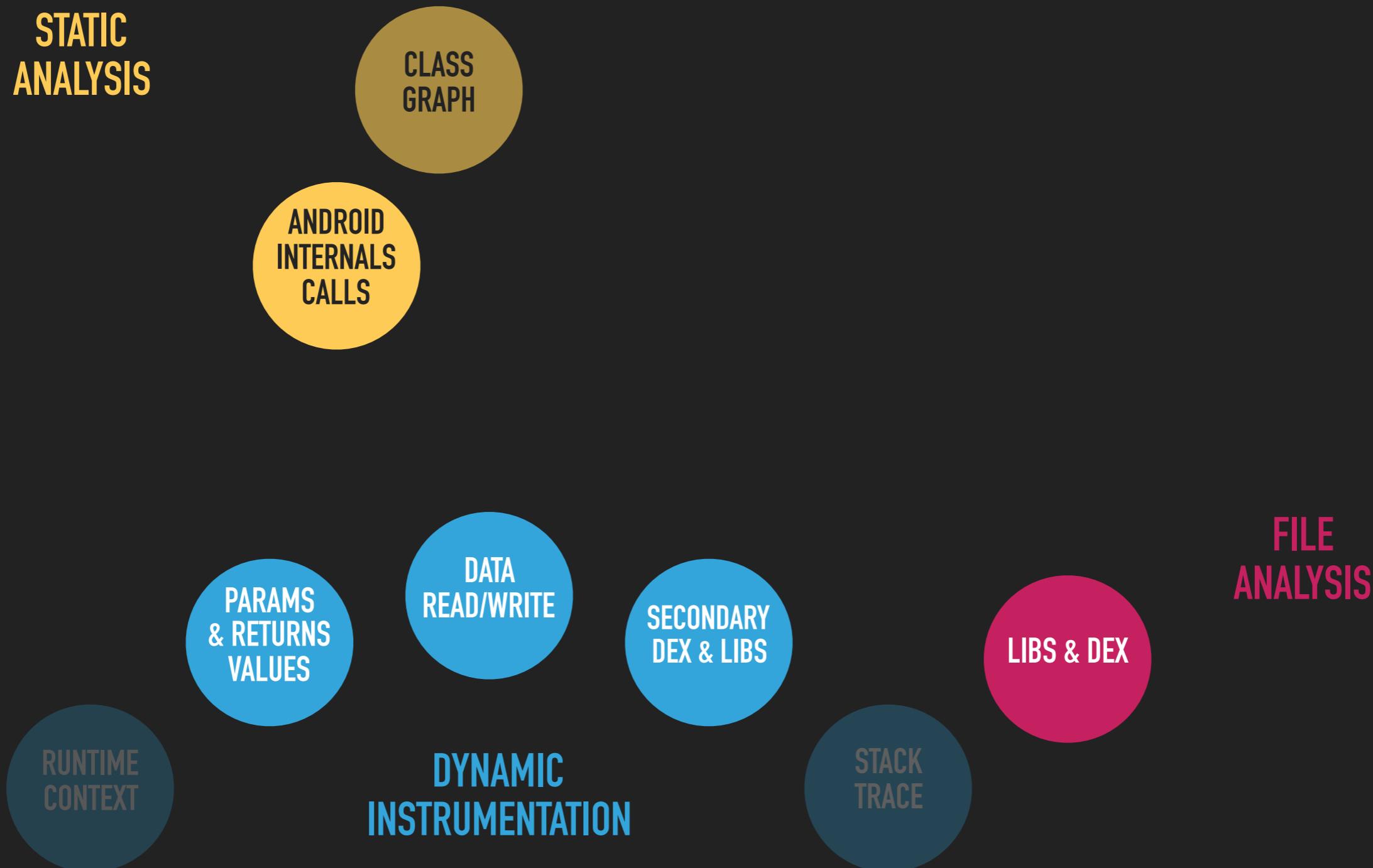
```
meth_49e17a6223b245a93ac8dff02c1c5bff.implementation = function(arg0,arg1,arg2) {
    var doCondition = true;
    if(b7b1ccb3e3d76b1557aa48d10e8c0688_VAR.names.indexOf(arguments[0])>-1)
        doCondition = false;
    if(doCondition){
        send({
            id:"YjA5Y2U3ZTkxNTg5YzFhNzEzOTI1NDVmYjc1NDI3ZmI=",
            match: true,
            data: {
                dex: arguments[0],
                odex: arguments[1],
                arg2: arguments[2],
                isNew: true,
                __hidden__data: DEXC_MODULE.common.readFile(arguments[0])
            },
            after: false,
            msg: "DexFile.loadDex()",  
tags: [{  
    "": ""  
}]});  
}
```

Hook messages

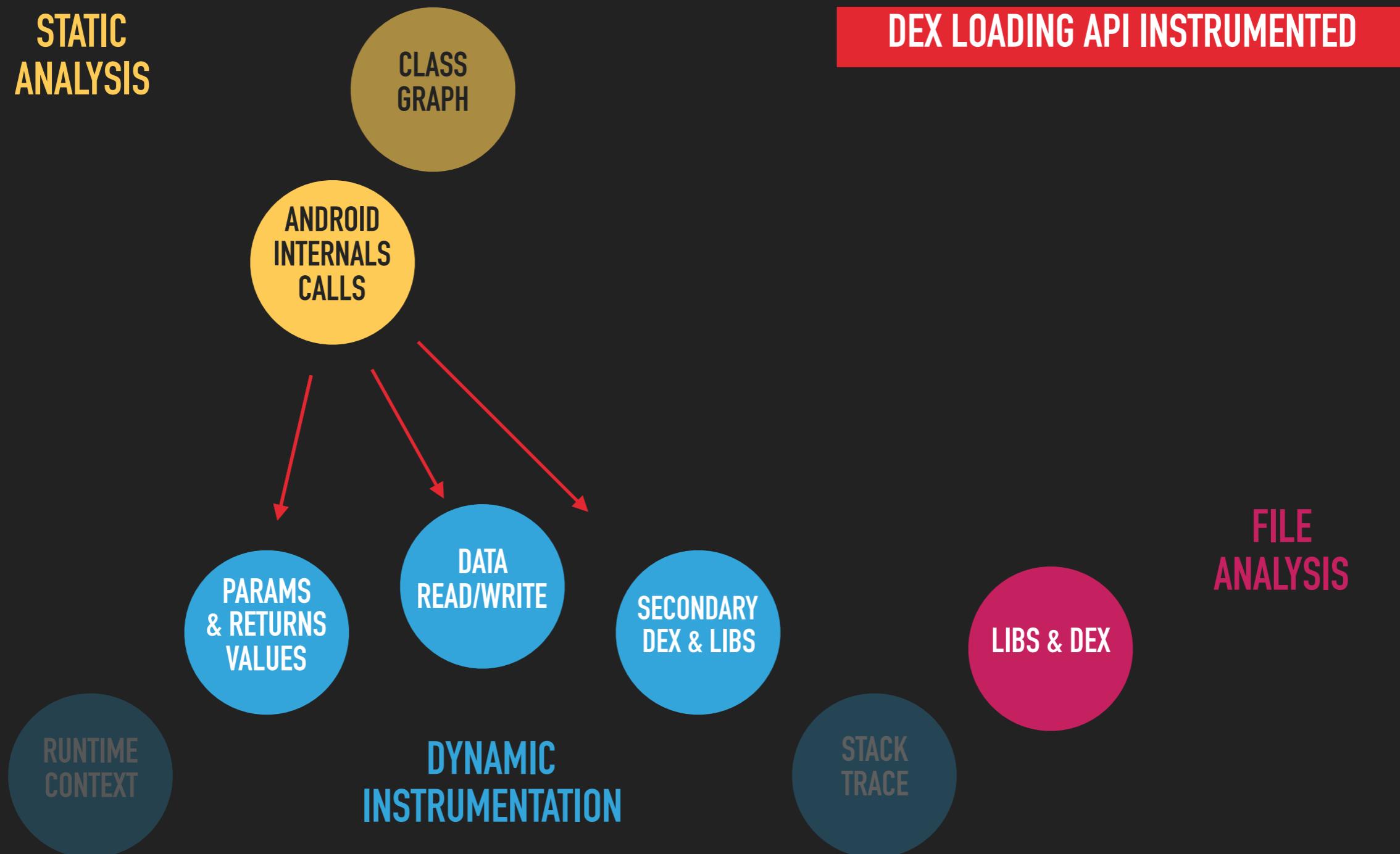
Hook data

Nothing to display

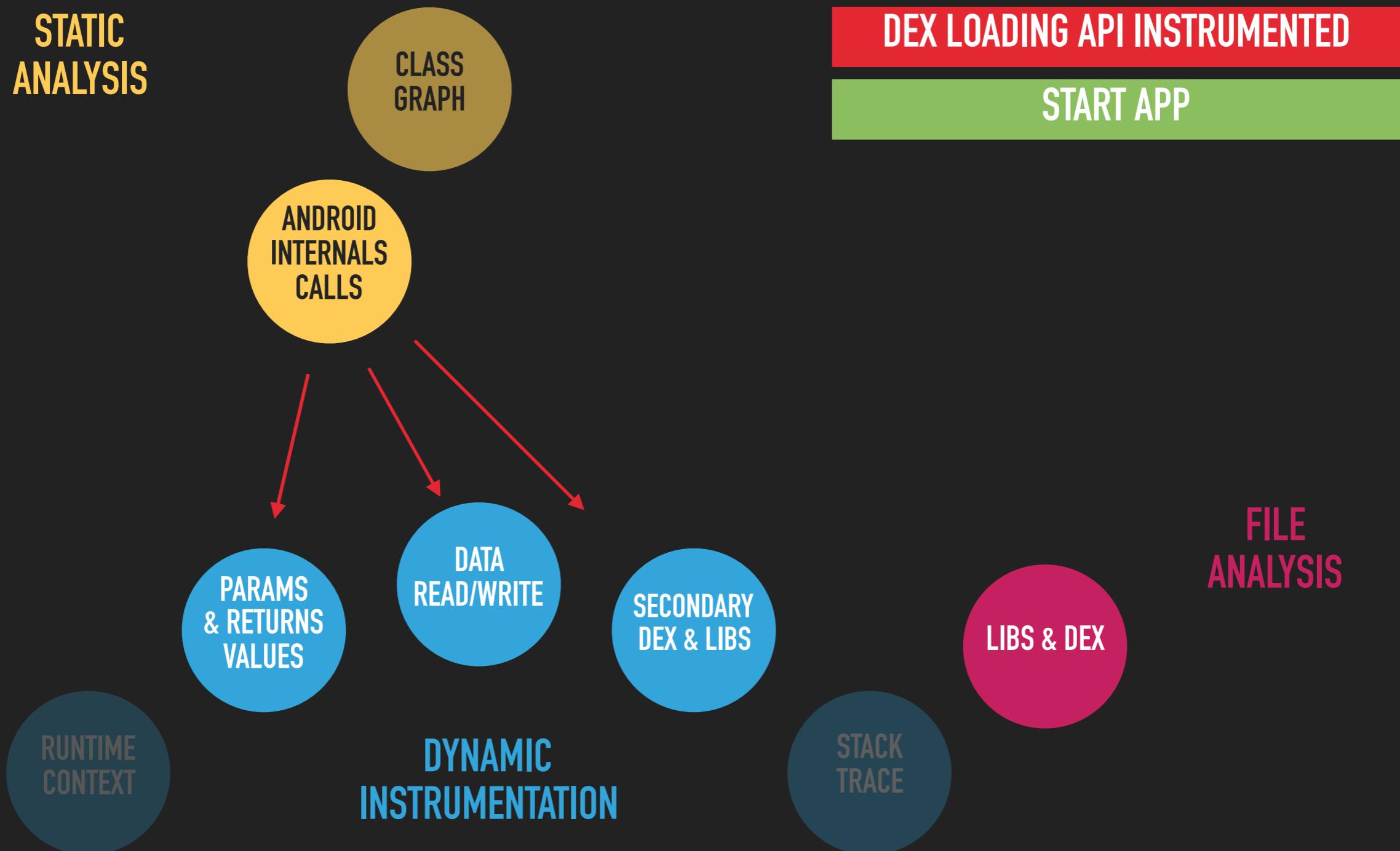
ANALYZING DEX FILE LOADED DYNAMICALLY



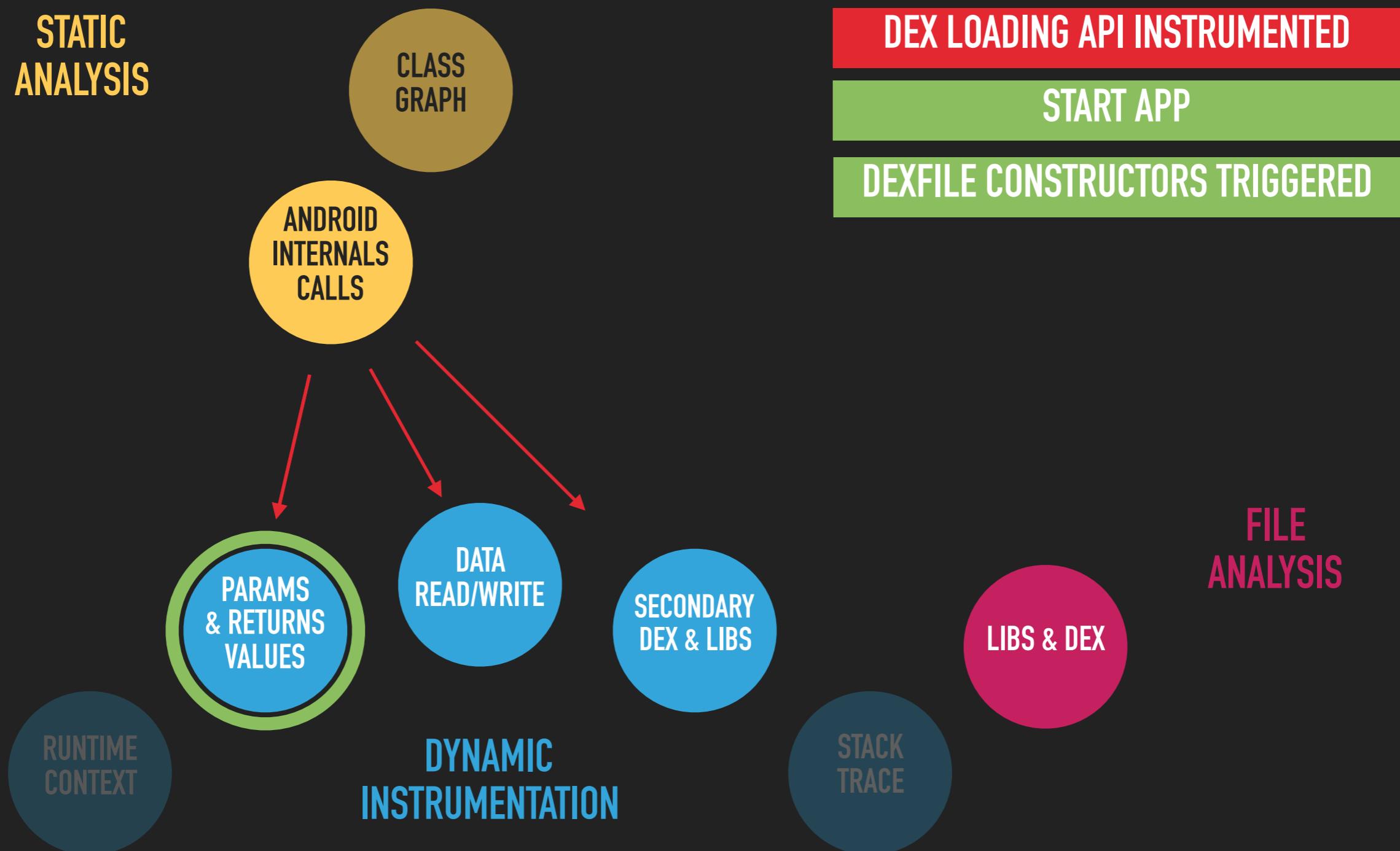
ANALYZING DEX FILE LOADED DYNAMICALLY



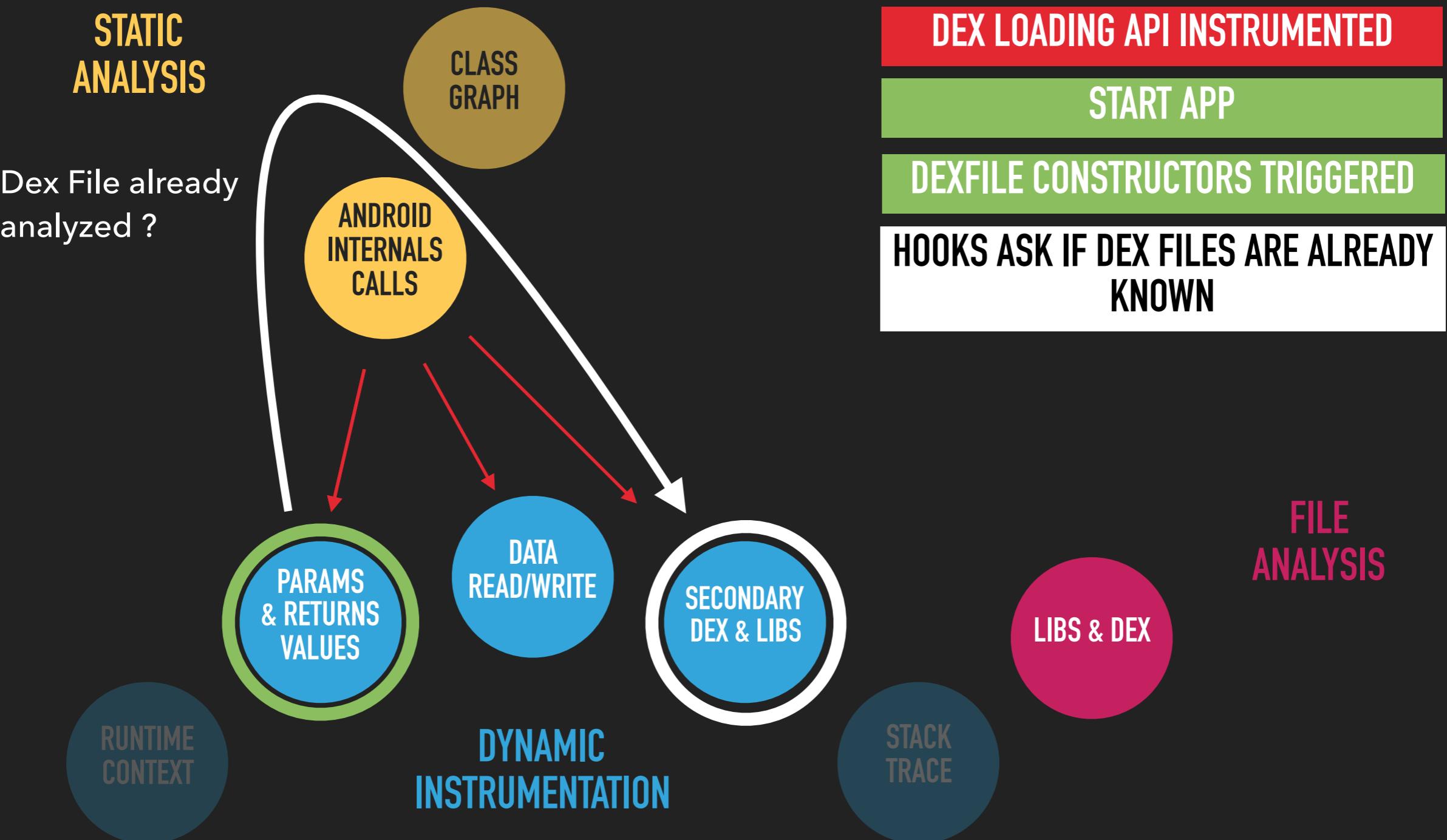
ANALYZING DEX FILE LOADED DYNAMICALLY



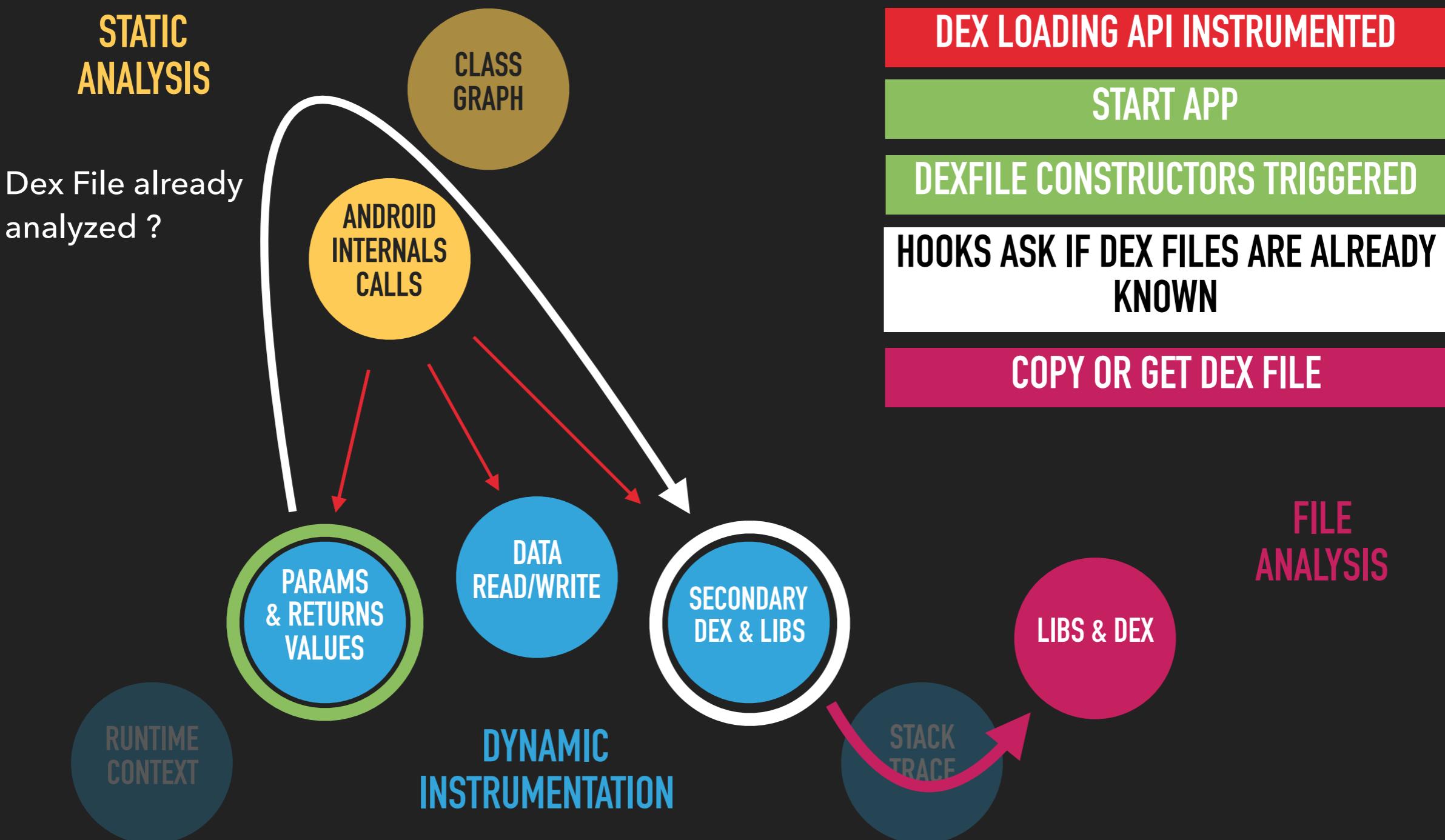
ANALYZING DEX FILE LOADED DYNAMICALLY



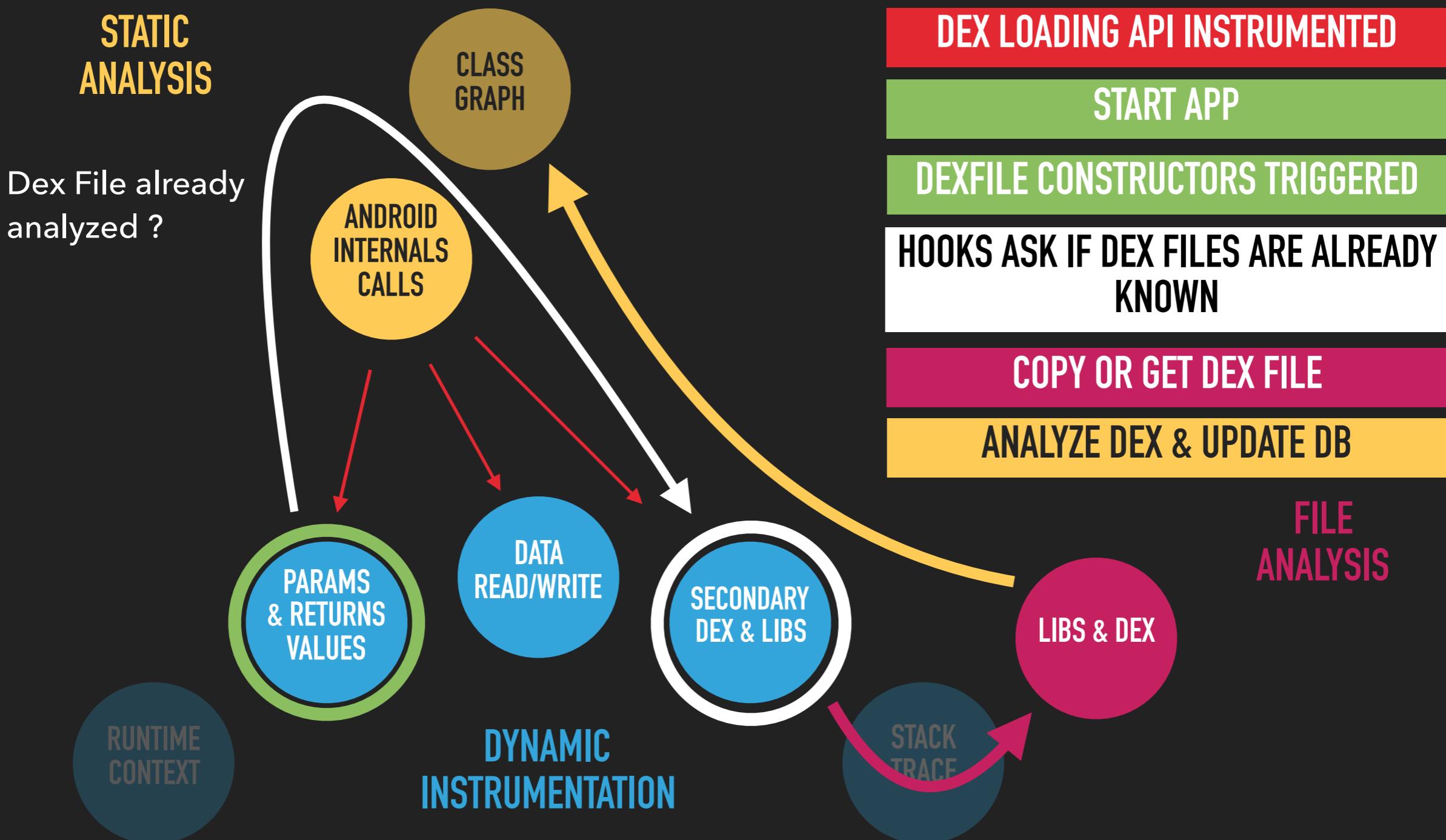
ANALYZING DEX FILE LOADED DYNAMICALLY



ANALYZING DEX FILE LOADED DYNAMICALLY



ANALYZING DEX FILE LOADED DYNAMICALLY



ANALYZING DEX FILE LOADED DYNAMICALLY

AFTER 1ST RUN

Dex file loaded dynamically Refresh

The table below lists all Dex files gathered at runtime and decompiled dynamically.

-	Name	Filepath
	unpacked-classes01.dex	/data/user/0/com.eshard.crackme.demo/files/unpacked-classes01.dex
	classes.dex	/data/user/0/com.eshard.crackme.demo/files/classes.dex

Showing 1 to 2 of 2 entries

Elements discovered Refresh

The table below lists all elements discovered (string, class, method, field, array, ...).

-	Type	Object
	Class	com.eshard.crackme.external.DynamicClass01
	Class	com.eshard.crackme.external.packed.ProtectedClass01

Showing 1 to 2 of 2 entries

ANALYZING DEX FILE LOADED DYNAMICALLY

AFTER 1ST RUN

-	com.eshard.crackme.external.packed	ProtectedClass01	D-dynamic
Package : com.eshard.crackme.external.packed (size: 1)			
Extends : java.lang.Object			
Implements : None			
Fields			
Action	Modifiers	Type	Name
Methods			
Action	Type	Name	
Probe OFF xref to xref from		<init>()<void>	
Probe OFF xref to xref from		getFlag()<java.lang.String>	

ANALYZING DEX FILE LOADED DYNAMICALLY

AFTER 1ST RUN

The screenshot shows the Dex2Smali interface after the first run. At the top, there is a table with the following data:

	com.eshard.crackme.external.packed.ProtectedClass01	getFlag()<java.lang.String>	D-dynamic
--	---	-----------------------------	-----------

Below the table, there are four tabs: "Modifiers" (public), "Class" (com.eshard.crackme.external.packed.ProtectedClass01), "Fullscreen" (com.eshard.crackme.external.packed.ProtectedClass01.getFlag), and "Return" (java.lang.String). The "Modifiers" tab is currently selected.

At the bottom, there are three buttons: "Smali" (selected), "Run smali (VM)" (new), and "Hook history" (new). The Smali button is highlighted with an orange bar.

The Smali code editor displays the following assembly code:

```
1 |
2 .line 7
3 const-string v0, "NOT TH4T PR0TECT3D!"
4 return-object v0
5
```

The string "const-string v0, "NOT TH4T PR0TECT3D!" is highlighted with a red box.

ANALYZING DEX FILE LOADED DYNAMICALLY

AFTER 1ST RUN

com.eshard.crackme.external.packed.ProtectedClass01 getFlag()<java.lang.String> D-dynamic

Modifiers	public
Class	com.eshard.crackme.external.packed.ProtectedClass01
Fullscreen	com.eshard.crackme.external.packed.ProtectedClass01.getFlag
Return	java.lang.String

Smalli Run smali (VM) new Hook history new

```
1 |
2 .line 7
3 const-string v0, "NOT TH4T PR0TECT3D!"
4 return-object v0
5
```



CASE #2

DYNAMIC UPDATE OF XREF WITH INVOKED METHODS

METHOD INVOKED DYNAMICALLY

```
2 const v0, 0x1
3 new-array v1, v0, [Ljava/lang/Class;
4 new-array v2, v0, [Ljava/lang/Object;
5 const v0, 0x0
6 const-class v3, Ljava/lang/String;
7 aput-object v3, v1, v0
8 aput-object p0, v2, v0
9 const-string v0, "convertToString"
10 const-class v3, Landroid/content/res/abltMZGC;
11 invoke-virtual {v3, v0, v1}, Ljava/lang/Class; ->getMethod(Ljava/lang/String;[Ljava/lang/Class;)Ljava/lang/reflect/Method;
12     move-result-object v0
13 invoke-virtual {v0, v3, v2}, Ljava/lang/reflect/Method; ->invoke(Ljava/lang/Object;[Ljava/lang/Object;)Ljava/lang/Object;
14     move-result-object v0
15 check-cast v0, Ljava/lang/String;
16 return-object v0
```

Smali code

METHOD INVOKED DYNAMICALLY

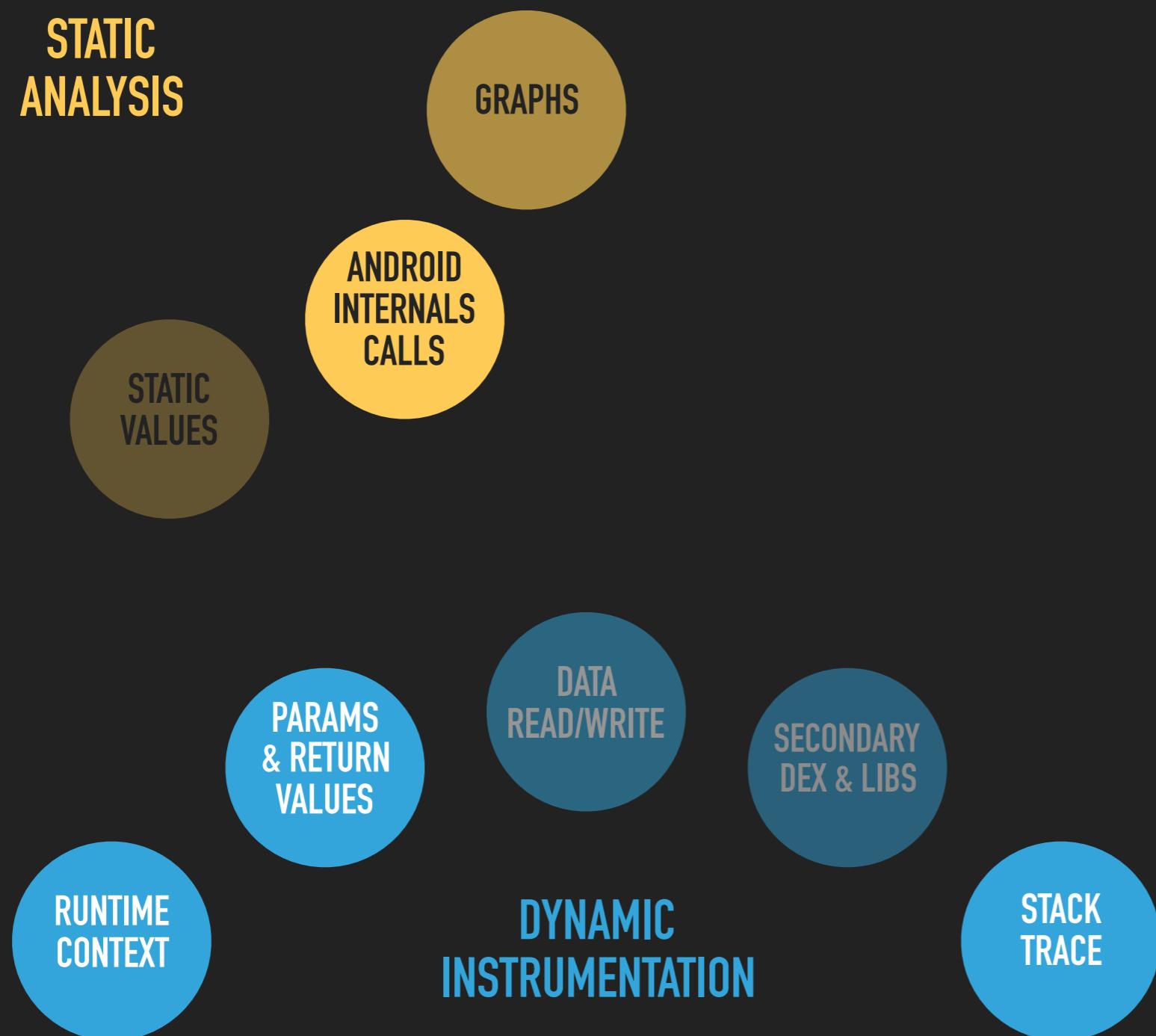
```
2 const v0, 0x1
3 new-array v1, v0, [Ljava/lang/Class;
4 new-array v2, v0, [Ljava/lang/Object;
5 const v0, 0x0
6 const-class v3, Ljava/lang/String;
7 aput-object v3, v1, v0
8 aput-object p0, v2, v0
9 const-string v0, "convertToString"
10 const-class v3, Landroid/content/res/abltMZGC;
11 invoke-virtual {v3, v0, v1}, Ljava/lang/Class; ->getMethod(Ljava/lang/String;[Ljava/lang/Class;)Ljava/lang/reflect/Method;
12     move-result-object v0
13 invoke-virtual {v0, v3, v2}, Ljava/lang/reflect/Method; ->invoke(Ljava/lang/Object;[Ljava/lang/Object;)Ljava/lang/Object;
14     move-result-object v0
15 check-cast v0, Ljava/lang/String;
16 return-object v0
```

Smali code

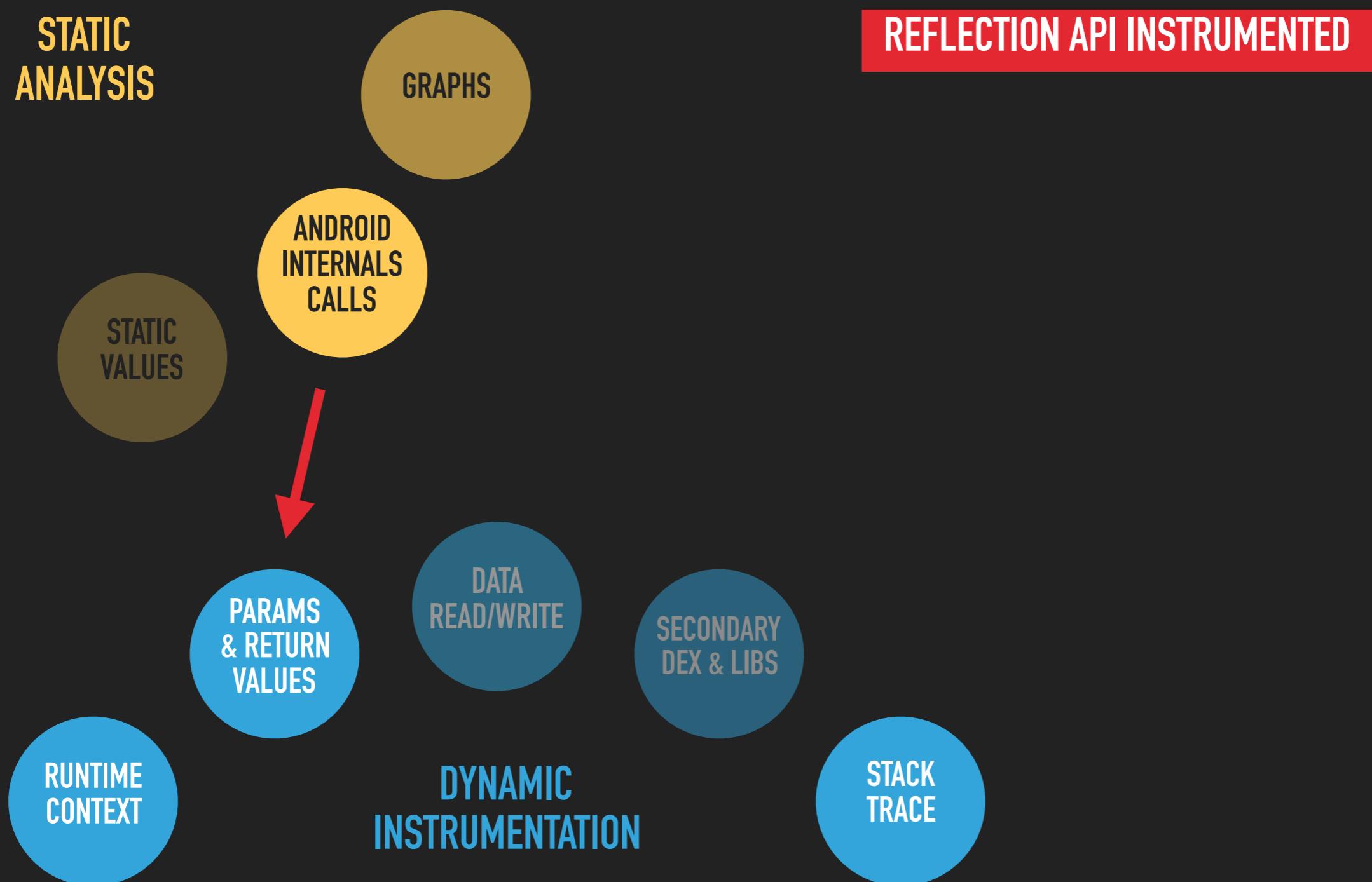
From a static point-of-view, only two methods are called :

- ▶ Class.getMethod()
- ▶ Method.invoke()

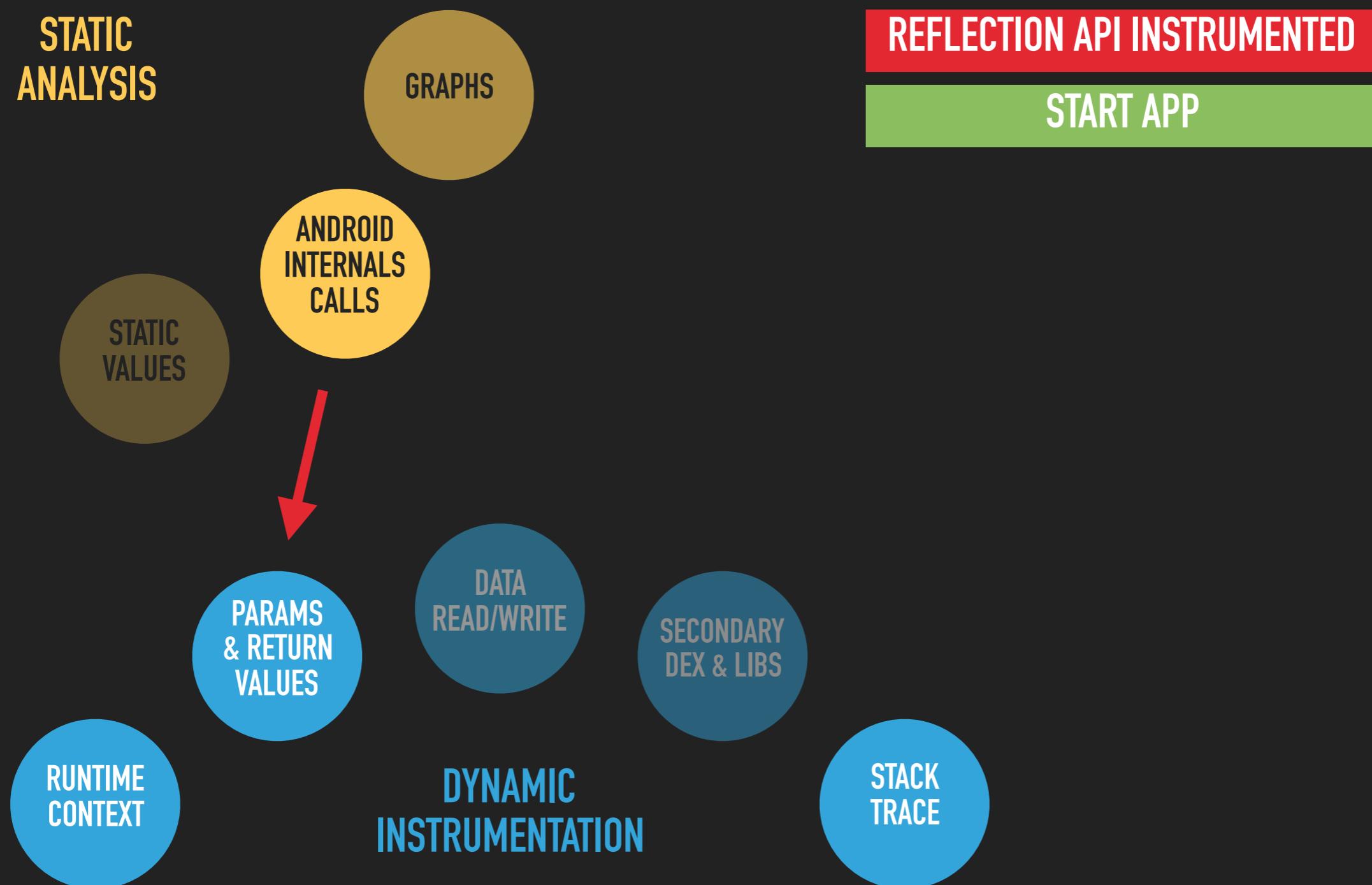
DYNAMIC UPDATE OF XREF WITH INVOKED METHODS



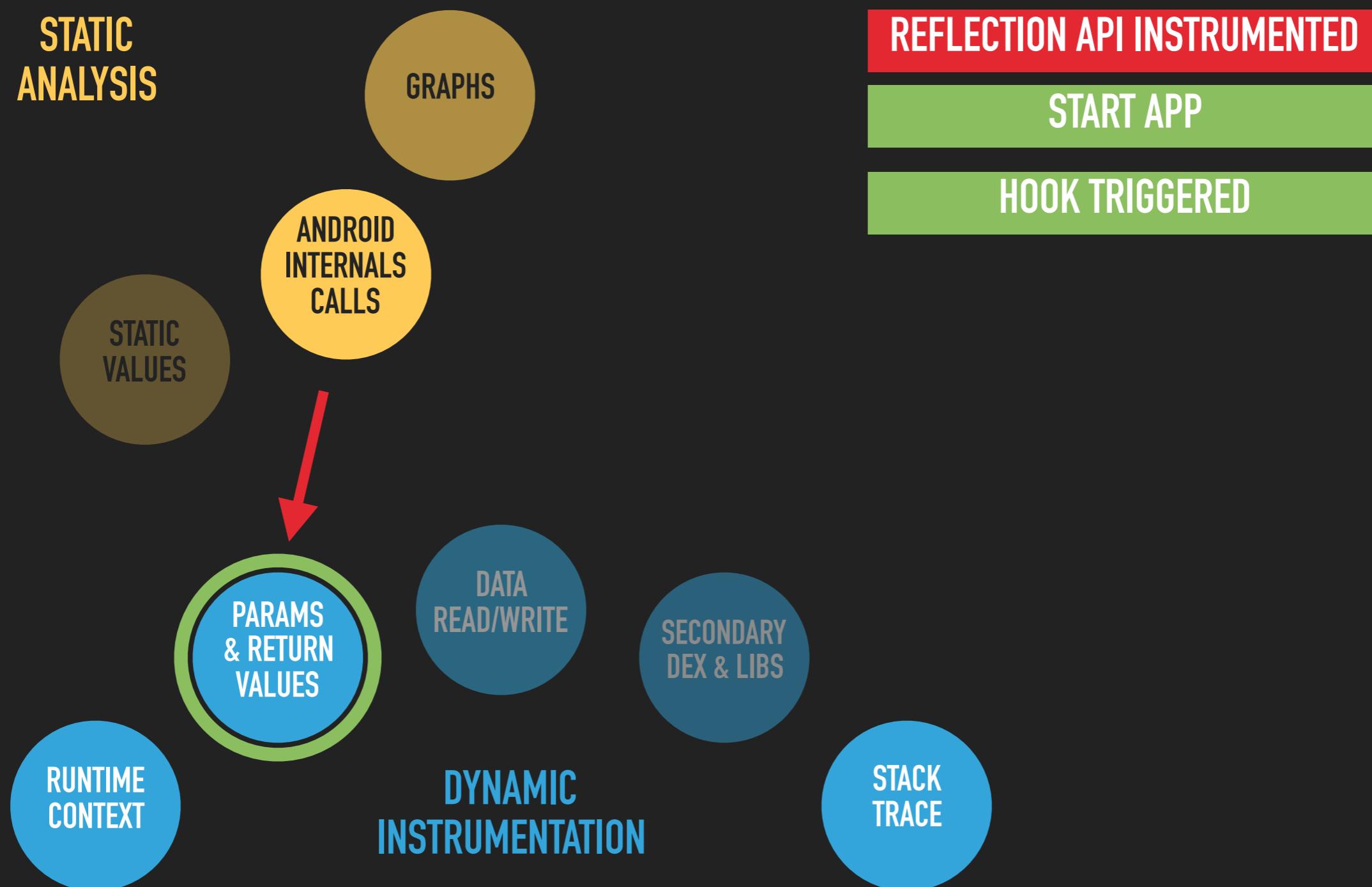
DYNAMIC UPDATE OF XREF WITH INVOKED METHODS



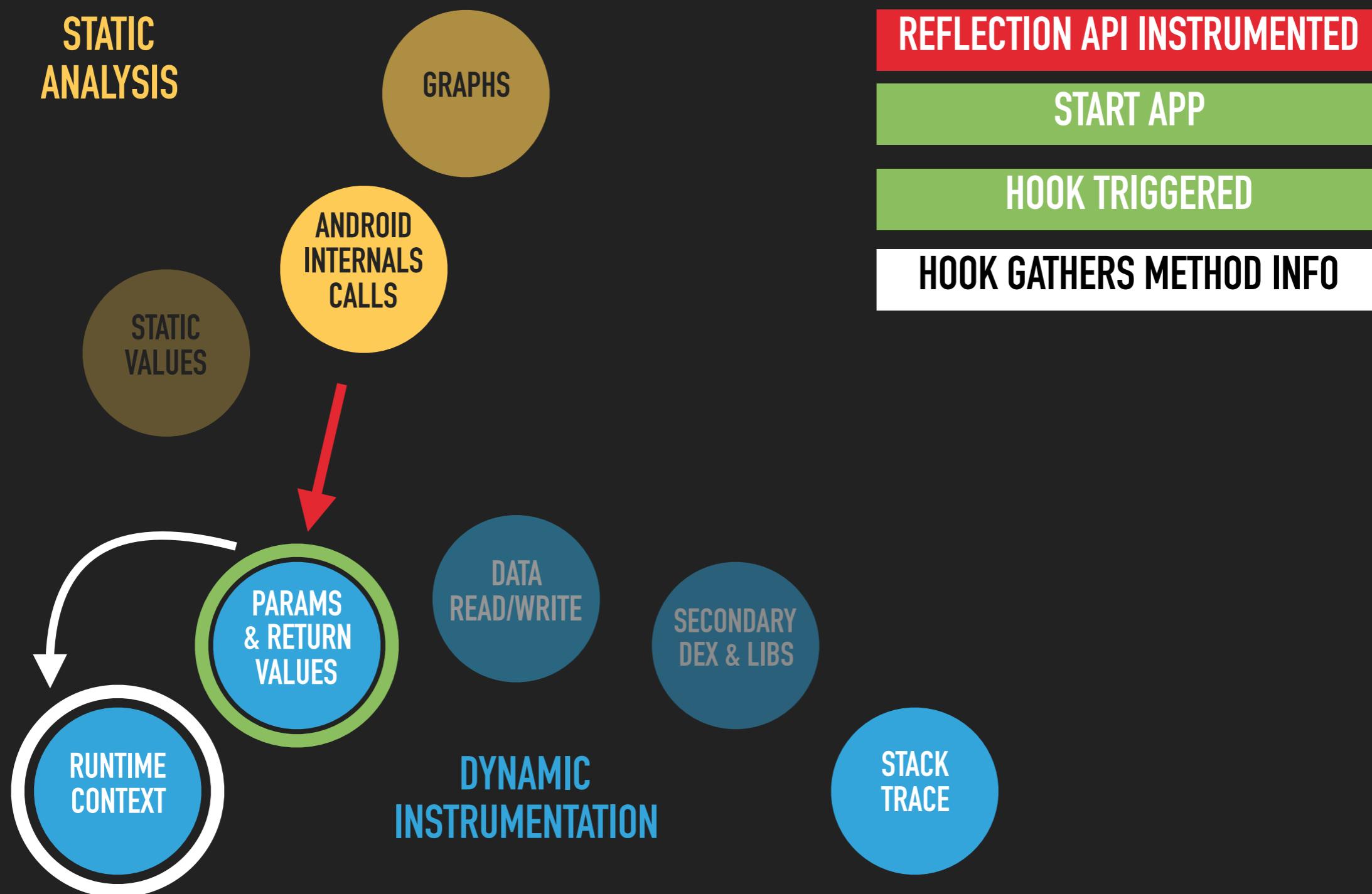
DYNAMIC UPDATE OF XREF WITH INVOKED METHODS



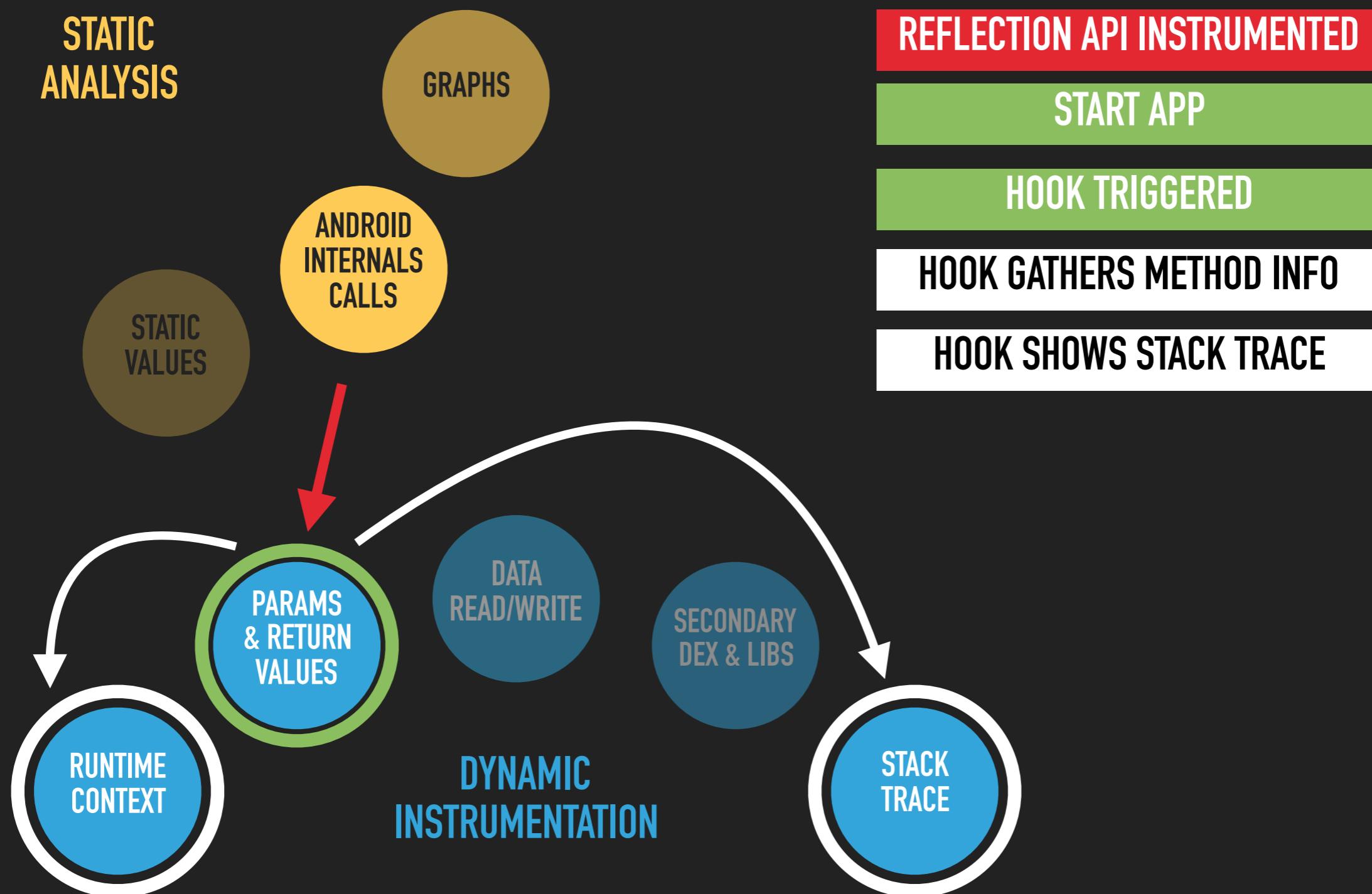
DYNAMIC UPDATE OF XREF WITH INVOKED METHODS



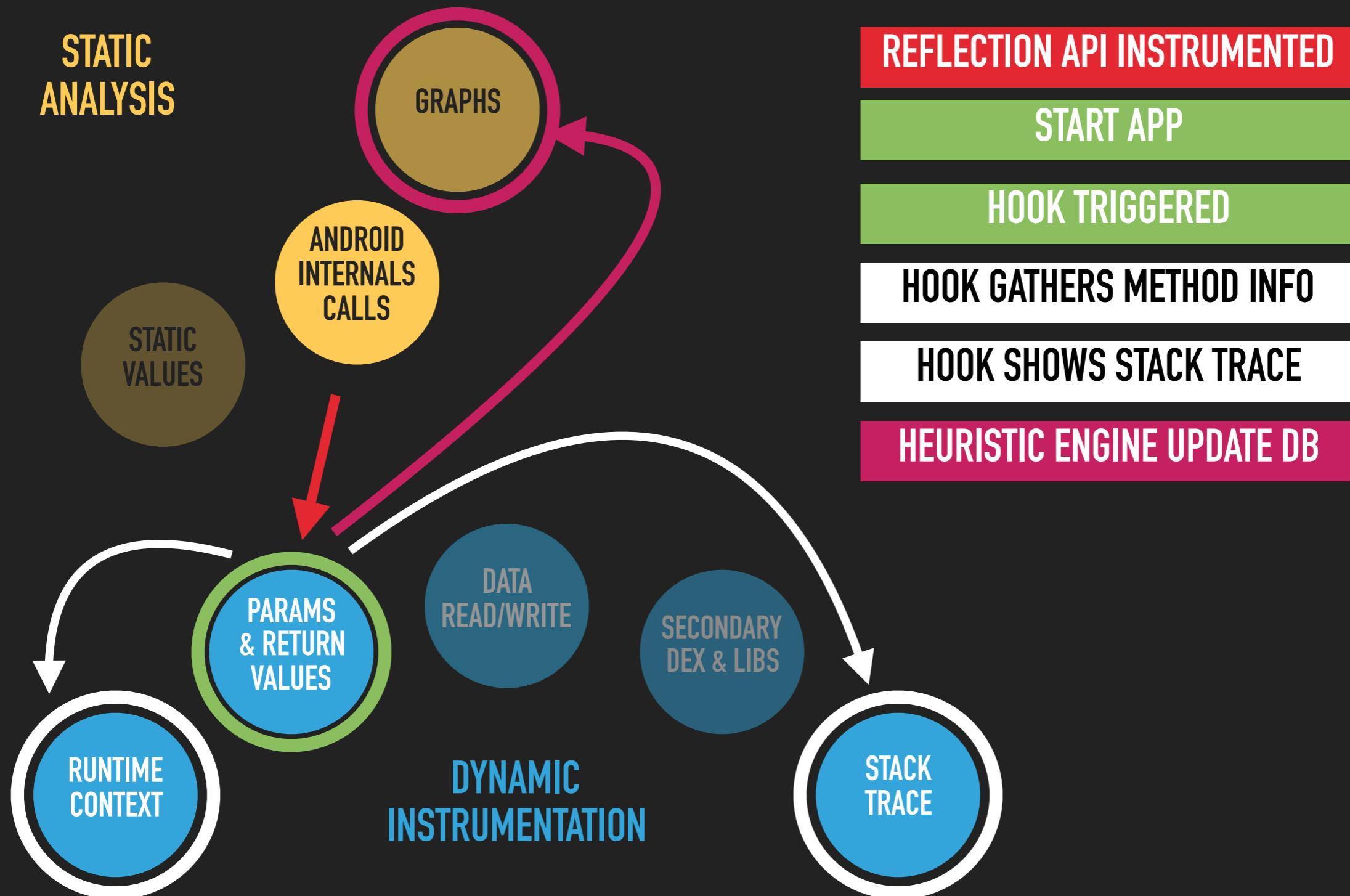
DYNAMIC UPDATE OF XREF WITH INVOKED METHODS



DYNAMIC UPDATE OF XREF WITH INVOKED METHODS



DYNAMIC UPDATE OF XREF WITH INVOKED METHODS



DYNAMIC UPDATE OF XREF FROM WITH INVOKED METHODS

METHOD INVOKED DYNAMICALLY

XRef from

com.eshard.a.a.CGSuroKqvFzdyAH(<java.lang.String>)<java.lang.String>

Method	Tags	Action
java.lang.Class.getMethod(<java.lang.String><java.lang.Class>[])<java.lang.reflect.Method>	internal	Probe OFF
java.lang.reflect.Method.invoke(<java.lang.Object><java.lang.Object>[])<java.lang.Object>	internal	Probe OFF

Showing 1 to 2 of 2 entries

BEFORE
RUNTIME

DYNAMIC UPDATE OF XREF FROM WITH INVOKED METHODS

METHOD INVOKED DYNAMICALLY

XRef from

com.eshard.a.a.CGSuroKqvFzdyAH(<java.lang.String>)<java.lang.String>

Method	Tags	Action
java.lang.Class.getMethod(<java.lang.String><java.lang.Class>[])<java.lang.reflect.Method>	internal	Probe OFF
java.lang.reflect.Method.invoke(<java.lang.Object><java.lang.Object>[])<java.lang.Object>	internal	Probe OFF

Showing 1 to 2 of 2 entries

BEFORE
RUNTIME

AFTER
RUNTIME

XRef from

com.eshard.a.a.CGSuroKqvFzdyAH(<java.lang.String>)<java.lang.String>

Method	Tags	Action
android.content.res.ablMZGC.convertToString(<java.lang.String>)<java.lang.String>	invoked	Probe OFF
java.lang.Class.getMethod(<java.lang.String><java.lang.Class>[])<java.lang.reflect.Method>	internal	Probe OFF
java.lang.reflect.Method.invoke(<java.lang.Object><java.lang.Object>[])<java.lang.Object>	internal	Probe OFF

DYNAMIC UPDATE OF XREF FROM WITH INVOKED METHODS

METHOD INVOKED DYNAMICALLY

XRef from

com.eshard.a.a.CGSuroKqvFzdyAH(<java.lang.String>)<java.lang.String>

Method	Tags	Action
java.lang.Class.getMethod(<java.lang.String><java.lang.Class>[])<java.lang.reflect.Method>	internal	Probe OFF
java.lang.reflect.Method.invoke(<java.lang.Object><java.lang.Object>[])<java.lang.Object>	internal	Probe OFF

Showing 1 to 2 of 2 entries

BEFORE
RUNTIME

AFTER
RUNTIME

XRef from

com.eshard.a.a.CGSuroKqvFzdyAH(<java.lang.String>)<java.lang.String>

Method	Tags	Action
android.content.res.ablMZGC.convertToString(<java.lang.String>)<java.lang.String>	invoked	Probe OFF
java.lang.Class.getMethod(<java.lang.String><java.lang.Class>[])<java.lang.reflect.Method>	internal	Probe OFF
java.lang.reflect.Method.invoke(<java.lang.Object><java.lang.Object>[])<java.lang.Object>	internal	Probe OFF

DYNAMIC UPDATE OF XREF FROM WITH INVOKED METHODS

METHOD INVOKED DYNAMICALLY

XRef from

com.eshard.a.a.CGSuroKqvFzdyAH(<java.lang.String>)<java.lang.String>

Method	Tags	Action
java.lang.Class.getMethod(<java.lang.String><java.lang.Class>[])<java.lang.reflect.Method>	internal	Probe OFF
java.lang.reflect.Method.invoke(<java.lang.Object><java.lang.Object>[])<java.lang.Object>	internal	Probe OFF

Showing 1 to 2 of 2 entries

BEFORE
RUNTIME

AFTER
RUNTIME

XRef from

com.eshard.a.a.CGSuroKqvFzdyAH(<java.lang.String>)<java.lang.String>

Method	Tags	Action
android.content.res.ablMZGC.convertToString(<java.lang.String>)<java.lang.String>	invoked	Probe OFF
java.lang.Class.getMethod(<java.lang.String><java.lang.Class>[])<java.lang.reflect.Method>	internal	Probe OFF
java.lang.reflect.Method.invoke(<java.lang.Object><java.lang.Object>[])<java.lang.Object>	internal	Probe OFF

CASE #3

BYTECODE SIMPLIFICATION BY PARTIAL EXECUTION

SOME COMMON PROBLEMS

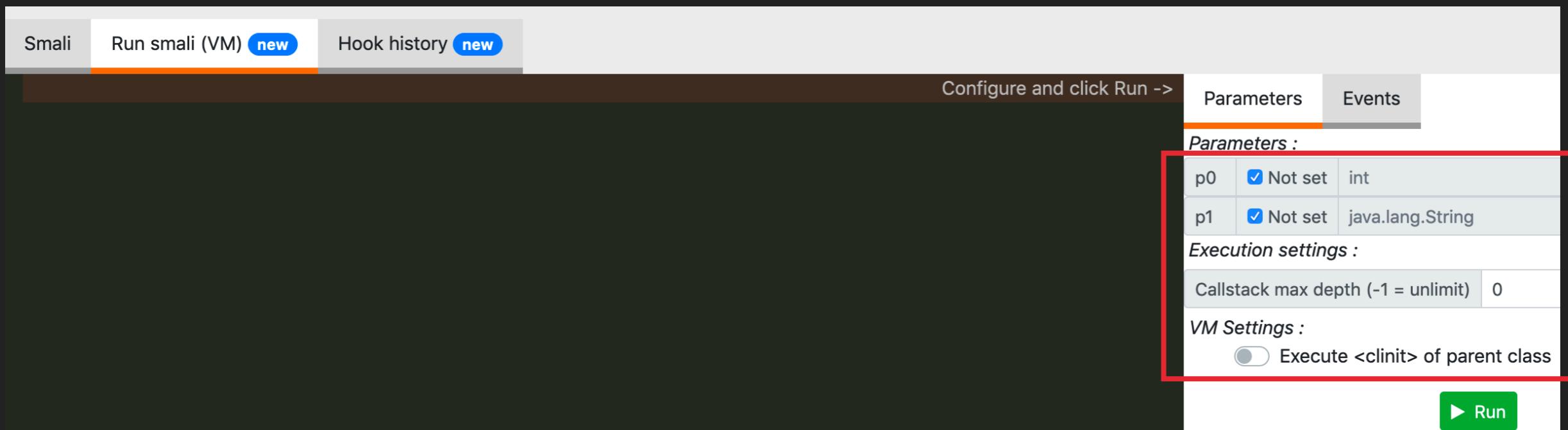
- ▶ Always TRUE / FALSE predicate
- ▶ Useless Goto(s)
- ▶ Implicit exceptions thrown (NPE, IOB, ..)
- ▶ Wrapped function
- ▶ ...

A- REMOVING USELESS GOTO(S) - BEFORE

Smali Run smali (VM) new Hook history new

```
1 goto/32 :goto_c
2
3
4 nop
5 nop
6
7 :goto_0
8 .line 47
9 goto/32 :goto_3
10
11 nop
12 nop
13 nop
14
15 :goto_1
16 invoke-static {v0, p0, v1}, Landroid/content/res/ablтMZGC;->AoUxThBEDrGiqSb(Ljavax/crypto/Cipher;ILjava/security/Key;)V
17 .line 58
18 goto/32 :goto_2
19
20 nop
21 nop
22
23 :goto_2
24 return-object v0
25 nop
26 nop
27
28 :goto_3
29 const-string v1, "DES/ECB/PKCS5Padding"
30 nop
31 goto/32 :goto_7
32
```

A- REMOVING USELESS GOTO(S) - BEFORE



BYTECODE SIMPLIFICATION BY PARTIAL EXECUTION

A- REMOVING USELESS GOTO(S) - AFTER

Smalli Run smali (VM) new Hook history new

Configure and click Run ->

```
1 v0 = android.content.res.abltMZGC.bXEVATfiSCtzpkh("DES/ECB/PKCS5Padding")
2 v1 = android.content.res.abltMZGC.VqxuLoJHbjvgTRW("DES")
3 v3 = android.content.res.abltMZGC.ihxCrtGAOVjZsgRC p1, "ASCII") // skipped, max depth reached
4 v2 = new javax.crypto.spec.DESKeySpec(v3) // skipped, max depth reached
5 v1 = android.content.res.abltMZGC.MqsIpHbmjWZv0QSC v1, v2) // skipped, max depth reached
6 android.content.res.abltMZGC.AoUxThBEDrGiqSb( v0, p0, v1) // skipped, max depth reached
7 return v0;
8
```

Parameters :
p0 Not set int
p1 Not set java.lang.String

Execution settings :
Callstack max depth (-1 = unlimit) 0

VM Settings :
 Execute <clinit> of parent class

▶ Run

B- HELP TO DETECT IMPLICIT EXCEPTION

```
:goto_1
invoke-static/range {v0 .. v0}, Lcom/eshard/crackme/activities/CrackMeChallenge; ->HVRQbvhaFlYdeGD(Ljava/lang/String;)Ljava/lang/String;
const/4 v0, 0x1
move-result-object v0
goto/32 :goto_3

:goto_2
const-string v0, "3780b8133459f5a028d742efbcfc7d2d"
goto/32 :goto_1

:goto_3
invoke-static {v0}, Lcom/eshard/crackme/activities/CrackMeChallenge; ->cVEqFjJdYlzfiuy(Ljava/lang/String;)V
goto/32 :goto_4
```



```
static {
    HVRQbvhaFlYdeGD("3780b8133459f5a028d742efbcfc7d2d");
    cVEqFjJdYlzfiuy(1);
}
```

BYTECODE SIMPLIFICATION BY PARTIAL EXECUTION

B- HELP TO DETECT IMPLICIT EXCEPTION

```
:goto_1
invoke-static/range {v0 .. v0}, Lcom/eshard/crackme/activities/CrackMeChallenge;→HVRQbvhaFlYdeGD(Ljava/lang/String;)Ljava/lang/String;
const/4 v0, 0x1
move-result-object v0
goto/32 :goto_3

:goto_2
const-string v0, "3780b8133459f5a028d742efbcfc7d2d"
goto/32 :goto_1

:goto_3
invoke-static {v0}, Lcom/eshard/crackme/activities/CrackMeChallenge;→cVEqFjJdYlzfiuy(Ljava/lang/String;)V
goto/32 :goto_4
```

Always throws AndroidVerifier exception at runtime

JADX



```
static {
    HVRQbvhaFlYdeGD("3780b8133459f5a028d742efbcfc7d2d");
    cVEqFjJdYlzfiuy(1);
}
```

WRONG

BYTECODE SIMPLIFICATION BY PARTIAL EXECUTION

B- HELP TO DETECT IMPLICIT EXCEPTION

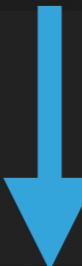
```
:goto_1
invoke-static/range {v0 .. v0}, Lcom/eshard/crackme/activities/CrackMeChallenge; -> HVRQbvhaFlYdeGD(Ljava/lang/String;)Ljava/lang/String;
const/4 v0, 0x1
move-result-object v0
goto/32 :goto_3

:goto_2
const-string v0, "3780b8133459f5a028d742efbcfc7d2d"
goto/32 :goto_1

:goto_3
invoke-static {v0}, Lcom/eshard/crackme/activities/CrackMeChallenge; -> cVEqFjJdYlzfiuy(Ljava/lang/String;)V
goto/32 :goto_4
```

Always throws AndroidVerifier exception at runtime

JADX



```
static {
    HVRQbvhaFlYdeGD("3780b8133459f5a028d742efbcfc7d2d");
    cVEqFjJdYlzfiuy(1);
}
```

WRONG

TRUE PATH



```
static {
    HVRQbvhaFlYdeGD("3780b8133459f5a028d742efbcfc7d2d");
    throw new AndroidVerifier();
}
```

Thanks



Q&A

EXTRA

SEARCH BYTE ARRAY CONTAINING HASHES OR STRINGS

	d.f.za.Yb.a()<int>[]::array_0		
	d.f.za.a.o.<clinit>()<void>::array_0		
	d.f.za.yb.<clinit>()<void>::array_0		
	d.f.za.yb.<clinit>()<void>::array_1		
	f.d.a.b.a.<clinit>()<void>::array_0		
	f.d.a.b.a.<clinit>()<void>::array_1		
	f.d.a.b.a.<clinit>()<void>::array_2		
Location		f.d.a.b.a.<clinit>()<void>	
Label		:array_2	
Size		288 bits	
Entry width		8 bits	
Tag	Data		Action
ascii	Noise_XXfallback_25519_AESGCM_SHA256		
raw	4e 6f 69 73 65 5f 58 58 66 61 6c 6c 62 61 63 6b 5f 32 35 35 31 39 5f 41 45 53 47 43 4d 5f 53 48 41 32 35 36 00 00 00 00 00 00 00 00 00 00 00 00		

SEARCH BYTE ARRAY CONTAINING HASHES OR STRINGS

	com.google.common.base.CharMatcher.showCharacter(<char>)<java.lang.String>::array_0 (12 bytes)	
	com.google.common.collect.ImmutableSortedMultiset.of(<java.lang.Comparable>) <com.google.common.collect.ImmutableSortedMultiset>::array_0 (16 bytes)	md5 key-128
	com.google.common.hash.Crc32cHashFunction\$Crc32cHasher.<clinit>()<void>::array_0 (1024 bytes)	
Location		com.google.common.hash.Crc32cHashFunction\$Crc32cHasher.<clinit>()<void>
Label		:array_0
Size		8192 bits
Entry width		32 bits
Tag	Data	
raw	00000000 -0d947cf0 -1ec48f09 1350f3f4 -386568e1 35f1141c 26a1e7e8 -2b359b15 -7526a731 78b2dbcc 6be22838 -6676540 105ec76f -1dcabb94 -0e9a4868 030e349b -283baf90 25afd373 36ff2087 -3b6b5c7c -65786060 68ec1ca3 7bbcef57 -762893a 20bd8ede -2d29f223 -3e7901d7 33ed7d2a -18d8e63f 154c9ac2 061c6936 -0b8815cb -559b29ef 580f5512 4b5fa6e6 -46cbda1 30e349b1 -3d77354e -2e27c6ba 23b3ba45 -08862152 05125dad 1642ae59 -1bd6d2a6 -45c5ee82 4851927d 5b016189 -56951d1 417b1dbc -4cef6141 -5fbf92b5 522bee48 -791e755d 748a09a0 67dafa54 -6a4e86a9 -345dba8d 39c9c670 2a993584 -270d491 5125dad3 -5cb1a630 -4fe155dc 42752927 -6940b234 64d4cecf 77843d3b -7a1041c8 -24037de4 2997011f 3ac7f2eb -37538e1 61c69362 -6c52ef9f -7f021c6b 72966096 -59a3fb83 5437877e 4767748a -4af30877 -14e03453 197448ae 0a24bb5a -07b0c7a 7198540d -7c0c28f2 -6f5cdb06 62c8a7f9 -49fd3cee 44694011 5739b3e5 -5aadcfa -04bef33e 092a8fc1 1a7a7c35 -17ee000 -7d09c488 709db87b 63cd4b8f -6e593774 456cac67 -48f8d09c -5ba82370 563c5f93 082f63b7 -05bb1f4c -16ebecc0 1b7f904 -6d5703e9 60c37f14 73938ce0 -7e07f01d 55326b08 -58a617f5 -4bf6e401 466298fc 1871a4d8 -15e5d825 -06b52bd1 0b21572 -5db44a5a 502036a5 4370c551 -4ee4b9ae 65d122b9 -68455e46 -7b15adb2 7681d14d 2892ed69 -25069196 -36566262 3bc21e9 -4dea8d37 407ef1ca 532e023e -5eba7ec3 758fe5d6 -781b992b -6b4b6adf 66df1622 38cc2a06 -355856fb -2608a50f 2b9cd91 -3c72d93c 31e6a5c7 22b65633 -2f222ad0 0417b1db -0983cd28 -1ad33ed4 1747422f 49547e0b -44c002f8 -5790f104 5a048d1 -2c2c1e55 21b862a8 32e8915c -3f7ceda1 144976b4 -19dd0a49 -0a8df9bd 07198540 590ab964 -549ec599 -47ce366d 4a5a4a9 -1ccf57e6 115b2b19 020bd8ed -0f9fa412 24aa3f05 -293e43fa -3a6eb00e 37faccf1 69e9f0d5 -647d8c2a -772d7fde 7ab9032 -0c91908b 0105ec76 12551f82 -1fc1637f 34f4f86a -39608497 -2a307763 27a40b9e 79b737ba -74234b47 -6773b8b3 6ae7c44	
	com.google.common.math.DoubleMath.<clinit>()<void>::array_0 (88 bytes)	

IMPROVEMENTS

- ▶ Use my own **customizable Dex format disassembler**
- ▶ **Yara rule with hook**
- ▶ Add native support (**r2 binding, QBDI, ...**)
- ▶ Add **fuzzing**
- ▶ And more ..