# **Androguard Documentation**

Release 3.4.0

**Anthony Desnos** 

## **CONTENTS**

1	<b>Documentation</b>	3
	1.1 Introduction	
	1.2 Tools	29
2	Commonly used APIs	45
3	Complete Python API	47
	3.1 androguard package	47
4 Indices and tables		185
Ру	ython Module Index	187
In	dex	189

Androguard is a full python tool to play with Android files. It is designed to work with Python 3 only.

- DEX, ODEX
- APK
- Android's binary xml
- Android resources
- Disassemble DEX/ODEX bytecodes
- Decompiler for DEX/ODEX files

You can either use the cli or graphical frontend for androguard, or use androguard purely as a library for your own tools and scripts.

CONTENTS 1

2 CONTENTS

**CHAPTER** 

ONE

## **DOCUMENTATION**

## 1.1 Introduction

## 1.1.1 Installation

There are several ways how to install androguard.

Before you start, make sure you are using a supported python version! For Windows, we recommend using the Anaconda python 3.6.x package.

**Warning:** The magic library might not work out of the box. If your magic library does not work, please refer to the installation instructions of python-magic.

#### PIP

The usual way to install a python packages is by using pypi.python.org and it's package installer pip. Just use

```
$ pip install -U androguard[magic,GUI]
```

to install androguard including the GUI and magic file type detection. In order to use features which use dot, you need Graphviz installed. This is not a python dependency but a binary package! Please follow the installation instructions for GraphvizInstall.

You can also make use of an virtualenv, to separate the installation from your system wide packages:

```
$ virtualenv venv-androguard
$ source venv-androguard/bin/activate
$ pip install -U androguard[magic,GUI]
```

pip should install all required packages too.

### **Debian / Ubuntu**

Debian has androguard in its repository. You can just install it using apt install androguard. All required dependencies are automatically installed.

#### **Install from Source**

Use git to fetch the sources, then install it. Please install git and python on your own. Androguard requires Python at least 3.4 to work. Pypy >= 5.9.0 should work as well but is not tested.

```
$ git clone --recursive https://github.com/androguard/androguard.git
$ cd androguard
$ virtualenv -p python3 venv-androguard
$ source venv-androguard/bin/activate
$ pip install .[magic,GUI]
```

The dependencies, defined in setup.py will be automatically installed.

For development purposes, you might want to install the extra dependecies for *docs* and *tests* as well:

```
$ git clone --recursive https://github.com/androguard/androguard.git
$ cd androguard
$ virtualenv -p python3 venv-androguard
$ source venv-androguard/bin/activate
$ pip install -e .[magic,GUI,tests,docs]
```

You can then create a local copy of the documentation:

```
$ python3 setup.py build_sphinx
```

Which is generated in build/sphinx/html.

## 1.1.2 Getting Started

#### **Using Androguard tools**

There are already some tools for specific purposes.

To just decode the AndroidManifest.xml or resources.arsc, there are androguard axml and androguard arsc. To get information about the certificates use androguard sign.

If you want to create call graphs, use androguard cg, or if you want control flow graphs, you can use androguard decompile.

## Using Androlyze and the python API

The easiest way to analyze APK files, is by using androguard analyze. It will start a iPython shell and has all modules loaded to get into action.

For analyzing and loading APK or DEX files, some wrapper functions exists. Use <code>AnalyzeAPK(filename)</code> or <code>AnalyzeDEX(filename)</code> to load a file and start analyzing. There are already plenty of APKs in the androguard repo, you can either use one of those, or start your own analysis.

```
$ androguard analyze
Androguard version 3.1.1 started
In [1]: a, d, dx = AnalyzeAPK("examples/android/abcore/app-prod-debug.apk")
# Depending on the size of the APK, this might take a while...
In [2]:
```

The three objects you get are a an APK object, d an array of DalvikVMFormat object and dx an Analysis object.

Inside the APK object, you can find all information about the APK, like package name, permissions, the AndroidManifest.xml or its resources.

The DalvikVMFormat corresponds to the DEX file found inside the APK file. You can get classes, methods or strings from the DEX file. But when using multi-DEX APK's it might be a better idea to get those from another place. The Analysis object should be used instead, as it contains special classes, which link information about the classes.dex and can even handle many DEX files at once.

## **Getting Information about an APK**

If you have successfully loaded your APK using AnalyzeAPK, you can now start getting information about the APK. For example, getting the permissions of the APK:

```
In [2]: a.get_permissions()
Out[2]:
['android.permission.INTERNET',
   'android.permission.WRITE_EXTERNAL_STORAGE',
   'android.permission.ACCESS_WIFI_STATE',
   'android.permission.ACCESS_NETWORK_STATE']
```

or getting a list of all activites, which are defined in the AndroidManifest.xml:

```
In [3]: a.get_activities()
Out[3]:
['com.greenaddress.abcore.MainActivity',
    'com.greenaddress.abcore.BitcoinConfEditActivity',
    'com.greenaddress.abcore.AboutActivity',
    'com.greenaddress.abcore.SettingsActivity',
    'com.greenaddress.abcore.DownloadSettingsActivity',
    'com.greenaddress.abcore.PeerActivity',
    'com.greenaddress.abcore.ProgressActivity',
    'com.greenaddress.abcore.LogActivity',
    'com.greenaddress.abcore.ConsoleActivity',
    'com.greenaddress.abcore.DownloadActivity']
```

Get the package name, app name and path of the icon:

```
In [4]: a.get_package()
Out[4]: 'com.greenaddress.abcore'

In [5]: a.get_app_name()
Out[5]: u'ABCore'

In [6]: a.get_app_icon()
Out[6]: u'res/mipmap-xxxhdpi-v4/ic_launcher.png'
```

Get the numeric version and the version string, and the minimal, maximal, target and effective SDK version:

```
In [7]: a.get_androidversion_code()
Out[7]: '2162'
In [8]: a.get_androidversion_name()
Out[8]: '0.62'
```

(continues on next page)

```
In [9]: a.get_min_sdk_version()
Out[9]: '21'
In [10]: a.get_max_sdk_version()
In [11]: a.get_target_sdk_version()
Out[11]: '27'
In [12]: a.get_effective_target_sdk_version()
Out[12]: 27
```

### You can even get the decoded XML for the AndroidManifest.xml:

```
In [15]: a.get_android_manifest_axml().get_xml()
Out[15]: '<manifest xmlns:android="http://schemas.android.com/apk/res/android"_
 →android:versionCode="2162" android:versionName="0.62" package="com.greenaddress.
 →abcore">\n<uses-sdk android:minSdkVersion="21" android:targetSdkVersion="27">\n</
 →uses-sdk>\n<uses-permission android:name="android.permission.INTERNET">\n</uses-
 →permission>\n<uses-permission android:name="android.permission.WRITE EXTERNAL_
 \rightarrowSTORAGE">\setminusn</uses-permission>\setminusn<uses-permission android:name="android.permission.
  \begin{tabular}{ll} $ \hookrightarrow ACCESS\_WIFI\_STATE" > \n < /uses-permission > \n < uses-permission and roid: name="and roid" | and roid: name="and roid: name="a
 \rightarrowpermission.ACCESS_NETWORK_STATE">\n</uses-permission>\n<application android:theme=
 →"@7F0F0006" android:label="@7F0E001D" android:icon="@7F0D0000" android:debuggable=
 →"true" android:allowBackup="false" android:supportsRtl="true">\n<activity_
 →android:name="com.greenaddress.abcore.MainActivity">\n<intent-filter>\n<action__
 →android:name="android.intent.action.MAIN">\n</action>\n<category android:name=
 \rightarrow "android.intent.category.LAUNCHER">\n</category>\n</intent-filter>\n</activity>\n
 →<service android:name="com.greenaddress.abcore.DownloadInstallCoreIntentService"
 \rightarrowandroid:exported="false">\n</service>\n<service android:name="com.greenaddress.
 →abcore.RPCIntentService" android:exported="false">\n</service>\n<service_
 →android:name="com.qreenaddress.abcore.ABCoreService" android:exported="false">\n</
 →service>\n<activity android:name="com.greenaddress.abcore.BitcoinConfEditActivity">
 →\n<intent-filter>\n<category android:name="android.intent.category.DEFAULT">\n</
 →category>\n<action android:name="com.greenaddress.abcore.BitcoinConfEditActivity">\n
 \rightarrow</activity>\n<activity>\n<activity} android:name="com.greenaddress."
 →abcore.AboutActivity">\n</activity>\n<activity android:label="@7F0E0038"
 →android:name="com.greenaddress.abcore.SettingsActivity" android:noHistory="true">\n
 →</activity>\n<activity \n<activity android:label="@7F0E0035" android:name="com.greenaddress.
 →abcore.DownloadSettingsActivity" android:noHistory="true">\n</activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<activity>\n<ac
 →android:theme="@7F0F0006" android:label="@7F0E0036" android:name="com.greenaddress.
 →abcore.PeerActivity">\n</activity>\n<activity android:theme="@7F0F00006"
 →android:label="@7F0E0037" android:name="com.greenaddress.abcore.ProgressActivity">\n
 →</activity>\n<activity android:name="com.greenaddress.abcore.LogActivity">\n</
 →activity>\n<activity \n<activity ">\n</
 →activity>\n<receiver android:name="com.greenaddress.abcore.PowerBroadcastReceiver">
 →\n<intent-filter>\n<action android:name="android.intent.action.ACTION_POWER_
  \verb|-CONNECTED">\n</action>\n<action and roid:name="and roid.intent.action.ACTION_POWER_roughly below the content of the cont
 →DISCONNECTED">\n</action>\n<action android:name="android.intent.action.ACTION_
 →SHUTDOWN">\n</action>\n<action android:name="android.intent.action.ACTION_BATTERY_
 →LOW">\n</action>\n<action>\n<action>\n
 →</intent-filter>\n</receiver>\n</application>\n</manifest>\n'
```

## Or if you like to use the AndroidManifest.xml as an ElementTree object, use the following method:

```
In [13]: a.get_android_manifest_xml()
Out[13]: <Element manifest at 0x7f9d01587b00>
```

There are many more methods to explore, just take a look at the API for APK.

## **Using the Analysis object**

The ~androguard.core.analysis.analysis.Analysis object has all information about the classes, methods, fields and strings inside one or multiple DEX files.

Additionally it enables you to get call graphs and crossreferences (XREFs) for each method, class, field and string.

This means you can investigate the application for certain API calls or create graphs to see the dependencies of different classes.

As a first example, we will get all classes from the Analysis:

As you can see, <code>get\_classes()</code> returns a list of <code>ClassAnalysis</code> objects. Some of them are marked as <code>EXTERNAL</code>, which means that the source code of this class is not defined within the DEX files that are loaded inside the Analysis. For example the first class <code>java.io.FileNotFoundException</code> is an API class.

A ClassAnalysis does not contain the actual code but the ClassDefItem can be loaded using the get\_vm\_class():

If the class is *EXTERNAL*, a *ExternalClass* is returned instead.

The ClassAnalysis also contains all the information about XREFs, which are explained in more detail in the next section.

#### **XREFs**

Consider the following Java source code:

```
class Foobar {
    public int afield = 23;

    public void somemethod() {
        String astring = "hello world";
     }
}
class Barfoo {
    (continues on next page)
```

```
public void othermethod() {
    Foobar x = new Foobar();

    x.somemethod();

    System.out.println(x.afield);
}
```

There are two classes and the class Barfoo instanciates the other class Foobar as well as calling methods and reading fields.

XREFs are generated for four things:

- Classes
- · Methods
- Fields
- Strings

XREFs work in two directions: *xref\_from* and *xref\_to*. *To* means, that the current object is calling another object. *From* means, that the current object is called by another object.

All XREFs can be visualized as an directed graph and if some object A is contained in the xref\_to, the called object will contain A in their xref from.

In the case of our Java example, the string astring is called in Foobar. somethod, therefore it will be contained in the xref\_to of Foobar.somethod.

The Field afield will be contained in the xref\_to of Barfoo.othermethod as well as the call to Foobar. somethod.

More on XREFs can be found in *xrefs*.

## 1.1.3 Crossreferences (XREFs)

Crossreferences or simply XREFs are the main thing which Analysis provides. XREFs are generated for Classes, Methods, Fields and Strings.

Next, we want to show a few usecases for XREFs and how they can be obtained.

Start up a ipython shell using androguard analyze in order to play through the example. We use an example from the androguard repo here:

```
$ androguard analyze examples/android/TestsAndroguard/bin/TestActivity.apk
Please be patient, this might take a while.
Found the provided file is of type 'APK'
[INFO ] androguard.analysis: End of creating cross references (XREF)
[INFO ] androguard.analysis: run time: Omin 00s
Added file to session:

$\times$SHA256::3bb32dd50129690bce850124ea120aa334e708eaa7987cf2329fd1ea0467a0eb
Loaded APK file...
>>> a

<androguard.core.bytecodes.apk.APK object at 0x000000000581D710>
>>> d
[<androguard.core.bytecodes.dvm.DalvikVMFormat object at 0x0000000000847400>]
>>> dx
```

```
<analysis.Analysis VMs: 1, Classes: 495, Strings: 496>
Androguard version 3.3.5 started
In [1]:
```

#### Get XREFs for method calls

The first example would be to query all called classes from the class tests.androguard.TestActivity. Remember, that you need to provide the class name as a type format with forward slashes instead of dots. In order to get the class, you can simply use classes or find\_classes():

```
In [4]: dx.classes['Ltests/androguard/TestActivity;']
Out[4]: <analysis.ClassAnalysis Ltests/androguard/TestActivity;>
```

This will return a ClassAnalysis object. Now you can iterate over all methods inside the class and query for the xrefs (the output is abbreviated):

```
In [10]: for meth in dx.classes['Ltests/androguard/TestActivity;'].get_methods():
             print("inside method {}".format(meth.name))
             for _, call, _ in meth.get_xref_to():
    . . . :
                 print(" calling -> {} -- {}".format(call.class_name, call.name))
    . . . :
    . . . :
inside method testCall1
  calling -> Ljava/lang/StringBuilder; -- toString
  calling -> Ljava/lang/StringBuilder; -- append
  calling -> Ljava/lang/StringBuilder; -- <init>
  calling -> Ljava/io/PrintStream; -- println
inside method testCalls
  calling -> Ljava/lang/Object; -- getClass
  calling -> Ljava/io/PrintStream; -- println
 calling -> Ltests/androguard/TestIfs; -- testIF
 calling -> Ltests/androguard/TestActivity; -- testCall2
[...]
```

Here you can see, that tests.androguard.TestActivity.testCall1 uses a StringBuilder as well as println. The method testCalls is calling other functions from the same package.

The other way around is also possible. Especially for Android API's, this is very interesting!

**Note:** External method, like the API calls, will not give any XREFs for xref to().

Lets say, you want all calls to the API class java.io.file:

(continues on next page)

```
called by -> Landroid/support/v4/util/AtomicFile; -- <init>
usage of method delete
 called by -> Landroid/support/v4/util/AtomicFile; -- failWrite
 called by -> Landroid/support/v4/util/AtomicFile; -- delete
 called by -> Landroid/support/v4/util/AtomicFile; -- delete
 called by -> Landroid/support/v4/util/AtomicFile; -- startWrite
 called by -> Landroid/support/v4/util/AtomicFile; -- openRead
 called by -> Landroid/support/v4/util/AtomicFile; -- finishWrite
usage of method renameTo
 called by -> Landroid/support/v4/util/AtomicFile; -- openRead
 called by -> Landroid/support/v4/util/AtomicFile; -- failWrite
 called by -> Landroid/support/v4/util/AtomicFile; -- startWrite
usage of method exists
 called by -> Landroid/support/v4/util/AtomicFile; -- startWrite
 called by -> Landroid/support/v4/util/AtomicFile; -- openRead
 called by -> Landroid/support/v4/util/AtomicFile; -- startWrite
usage of method getParentFile
 called by -> Landroid/support/v4/util/AtomicFile; -- startWrite
usage of method mkdir
 called by -> Landroid/support/v4/util/AtomicFile; -- startWrite
```

**Note:** An external class or method is simply a class or method which could not be found inside the loaded DEX files at the time the XREFs were created! Thus, it is important to always load all DEX files of a multidex file. On the other hand, beware that classes might not be defined as they could be loaded dynamically later. External does not automatically mean that this class/method is an Android or Java API!

## **Get XREFs for Strings**

Next, we want to see where certain strings are used. For example, you found the interesting String 'boom' and would like to know where it is used. You can use either strings or find\_strings() to get the proper object for the XREFs:

```
In [12]: dx.strings['boom']
Out[12]: <analysis.StringAnalysis 'boom'>
```

The resulting object is of type *StringAnalysis*.

**Note:** StringAnalysis does not have a xref\_to method, which is obvious, as a String does nothing but is always used.

Now we can call xref\_from() to get the usage of the String:

So, we know that this specific String is used once in the test\_base method.

### **Get XREFs for Fields**

The last XREF we can use are fields. Fields are a little bit different and do not use xref\_from and xref\_to but xref\_read() and xref\_write(). You can use the method find\_methods() in order to find fields.

Note: Calls to static fields are usually not tracked, as they are optimized by the compiler to const calls!

For example, you want to get the read's and write's to the field value inside tests.androguard. TestActivity:

```
In [25]: for field in dx.find_fields(classname='Ltests/androguard/TestActivity;',_
→fieldname='^value$'):
            print("Field: {}".format(field.name))
    . . . :
            for _, meth in field.get_xref_read():
                print(" read in {} -- {}".format(meth.class_name, meth.name))
    . . . :
             for _, meth in field.get_xref_write():
    . . . :
    . . . :
                 print(" write in {} -- {}".format(meth.class_name, meth.name))
    . . . :
Field: value
  read in Ltests/androguard/TestActivity; -- pouet
  read in Ltests/androquard/TestActivity; -- test1
  read in Ltests/androguard/TestActivity; -- test_base
  read in Ltests/androguard/TestActivity; -- testVars
  write in Ltests/androguard/TestActivity; -- <init>
  write in Ltests/androquard/TestActivity; -- pouet2
  write in Ltests/androguard/TestActivity; -- <init>
  write in Ltests/androquard/TestActivity; -- <init>
```

## 1.1.4 Basic Blocks

We already saw the concept of **xrefs**, which can be used to get references in the assembly. The next step is to look at the Control Flow Graph (CFG) of a method.

Such a CFG can be generated using the *decompile* command of the **androguard**\_ tool. Let's take the androguard example file and decompile it:

```
$ androguard decompile -d output_folder -f jpg --limit "LTestDefaultPackage.*"_
→examples/android/TestsAndroguard/bin/TestActivity.apk
[INFO ] androguard.analysis: End of creating cross references (XREF)
         ] androguard.analysis: run time: Omin 00s
Dump information examples/android/TestsAndroguard/bin/TestActivity.apk in output_
Create directory output_folder
Decompilation ... End
Dump LTestDefaultPackage$TestInnerClass$TestInnerInnerClass; <init>_
→ (LTestDefaultPackage$TestInnerClass; I I)V ... jpg ... source codes ... bytecodes ...
Dump LTestDefaultPackage$TestInnerClass$TestInnerInnerClass; <init>_
→ (LTestDefaultPackage$TestInnerClass; I I LTestDefaultPackage$TestInnerClass
→$TestInnerInnerClass;) V ... jpg ... bytecodes ...
Dump LTestDefaultPackage$TestInnerClass$TestInnerClass; Test (I)V ... jpg ....
⇒bytecodes ...
Dump LTestDefaultPackage$TestInnerClass; <init> (LTestDefaultPackage; I I)V ... jpg ..
→. source codes ... bytecodes ...
Dump LTestDefaultPackage$TestInnerClass; <init> (LTestDefaultPackage; I I_
→LTestDefaultPackage$TestInnerClass;)V ... jpg ... bytecodes ...
                                                                         (continues on next page)
```

```
Dump LTestDefaultPackage$TestInnerClass; access$1 (LTestDefaultPackage$TestInnerClass; 

) I ... jpg ... bytecodes ...

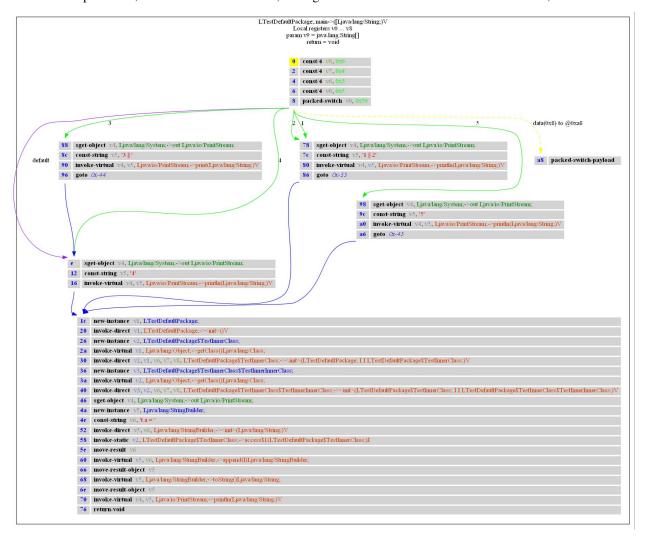
Dump LTestDefaultPackage$TestInnerClass; Test (I)V ... jpg ... bytecodes ...

Dump LTestDefaultPackage; <init> ()V ... jpg ... source codes ... bytecodes ...

Dump LTestDefaultPackage; main ([Ljava/lang/String;)V ... jpg ... bytecodes ...

Dump LTestDefaultPackage; const4 ()V ... jpg ... bytecodes ...
```

Note, that we only decompiled a certain subset of the file, as we are not interested in the other classes right now. Inside the output folder, we have now several files, among them some JPG files which show the CFG, like this one:



Each of the rectangles is a DVMBasicBlock. Each block is connected via an arrow, indicating the flow direction.

In this example, we can see that the *switch* instruction has six different ways to go, indicated by the green and purple arrows. Each green arrow is a specific check inside the *switch* instruction, i.e. what value results in which code block. The purple arrow is the default case. We can see that the *switch* only results in four different code blocks. There is a special block, with the yellow arrow, which is the pseudo instruction holding the switch payload.

Each of the switch blocks is followed by another, large basic block. If you look carefully, you can see that three of the blocks have *goto* commands at the end but the fourth block does not have one. First, take a look at the overall dissassembly of the method:

```
METHOD LTestDefaultPackage; public static main ([Ljava/lang/String; v9)V
main-BB@0x00000000 :
       0 (00000000) const/4
                                       v8, 0
       1 (00000002) const/4
                                       v7, 4
       2 (00000004) const/4
                                       v6, 3
                                        v0, 5
       3 (00000006) const/4
       4 (00000008) packed-switch
                                       v0, 80 [ D:main-BB@0x0000000e 1:main-
→BB@0x00000078 2:main-BB@0x00000078 3:main-BB@0x00000088 4:main-BB@0x00000000 5:main-
→BB@0x0000098 ]
       5 (0000000e) sget-object
                                       v4, Ljava/lang/System; ->out Ljava/io/
→PrintStream;
                                       v5, '4'
       6 (00000012) const-string
       7 (00000012) const String v3, 1 v3, Ljava/io/PrintStream;->
→println(Ljava/lang/String;)V [ main-BB@0x000001c ]
       8 (0000001c) new-instance v1, LTestDefaultPackage;
       9 (00000020) invoke-direct
                                       v1, LTestDefaultPackage; -> <init>() V
       10 (00000026) new-instance
                                       v2, LTestDefaultPackage$TestInnerClass;
       11 (0000002a) invoke-virtual
                                       v1, Ljava/lang/Object; ->getClass()Ljava/
→lang/Class;
       12 (00000030) invoke-direct
                                       v2, v1, v6, v7, v8, LTestDefaultPackage
→$TestInnerClass;-><init>(LTestDefaultPackage; I I LTestDefaultPackage
→$TestInnerClass;)V
       13 (00000036) new-instance
                                       v3, LTestDefaultPackage$TestInnerClass

⇒$TestInnerInnerClass;
       14 (0000003a) invoke-virtual
                                       v2, Ljava/lang/Object;->getClass()Ljava/
→lang/Class;
       15 (00000040) invoke-direct
                                       v3, v2, v6, v7, v8, LTestDefaultPackage
→$TestInnerClass$TestInnerInnerClass; -><init>(LTestDefaultPackage$TestInnerClass; I...
→I LTestDefaultPackage$TestInnerClass$TestInnerInnerClass;)V
                                       v4, Ljava/lang/System; ->out Ljava/io/
       16 (00000046) sget-object
→PrintStream;
       17 (0000004a) new-instance
                                       v5, Ljava/lang/StringBuilder;
       18 (0000004e) const-string
                                       v6, 't.a = '
       19 (00000052) invoke-direct
                                       v5, v6, Ljava/lang/StringBuilder; -><init>
→ (Ljava/lang/String;) V
       20 (00000058) invoke-static
                                    v2, LTestDefaultPackage$TestInnerClass;->
→access$1(LTestDefaultPackage$TestInnerClass;)I
       21 (0000005e) move-result
                                       v6
       22 (00000060) invoke-virtual
                                        v5, v6, Ljava/lang/StringBuilder;->
→append(I)Ljava/lang/StringBuilder;
       23 (00000066) move-result-object v5
       24 (00000068) invoke-virtual v5, Ljava/lang/StringBuilder;->
→toString()Ljava/lang/String;
       25 (0000006e) move-result-object v5
       26 (00000070) invoke-virtual v4, v5, Ljava/io/PrintStream;->
→println(Ljava/lang/String;)V
       27 (00000076) return-void
       28 (00000078) sget-object
                                       v4, Ljava/lang/System; ->out Ljava/io/
→PrintStream:
       29 (0000007c) const-string
                                       v5, '1 || 2'
       30 (00000080) invoke-virtual
                                       v4, v5, Ljava/io/PrintStream;->
→println(Ljava/lang/String;)V
       31 (00000086) goto
                                        -53 [ main-BB@0x000001c ]
       32 (00000088) sget-object
                                       v4, Ljava/lang/System; ->out Ljava/io/
→PrintStream;
       33 (0000008c) const-string
                                       v5, '3 || '
       34 (00000090) invoke-virtual
                                       v4, v5, Ljava/io/PrintStream; ->print(Ljava/
→lang/String;)V
                                                                      (continues on next page)
```

All these blocks are concatenated to each other. If you like, try to identify the basic blocks inside the dissasembly! Hint: The second column gives the offset inside the bytecode and matches the offset given in the CFG.

As you can see, the order of instructions in the bytecode does not match the execution order. For example, the *return* opcode is in the middle of the bytecode, while it is the end of the exection. Therefore some parts must have a *goto* to resume the execution at the correct point. For example, the basic block for the case that the argument of the switch opcode is 5 ends at offset 0xa6 and has a goto command to subtract 0x45 from the current offset. But that ends up beeing offset 0x61? No, it does not. To increase your confusion, you have to know, that offset arguments for opcodes are always in 16-bit units, while the offset used by androguard are counted in 8-bit units. That means, that you have to subtract 0x8a, which indeed returns to offset 0x1c in the bytecode.

**Warning:** The offset units used are sometimes a little bit inconsistent across androguard! If you find some inconsistent behaviour, please report it as an issue.

To conclude, let's take a look at the actual Java source code of this particular method:

```
public static void main(String [] z) {
    int a = 5;
    switch (a)
    case 1:
        System.out.println("1 || 2");
       break;
    case 3:
        System.out.print("3 || ");
    case 4:
    default:
        System.out.println("4");
        break;
    case 5:
        System.out.println("5");
   TestDefaultPackage p = new TestDefaultPackage();
    TestInnerClass t = p.new TestInnerClass(3, 4);
    TestInnerClass.TestInnerInnerClass t2 = t.new TestInnerInnerClass(3, 4);
    System.out.println("t.a = " + t.a);
```

Can you see how each Basic block belongs to a different path in the code?

## 1.1.5 Parsing Instructions and Bytecode

One often requested task is to parse the bytecode of all (or certain) methods. The bytecode can be used for various tasks, from creating simple statistics to machine learning.

The bytecode is stored for each method in the Dalvik file. Google provides some documentation about the bytecode format, which is very useful if you want to process it. Androguard can provide three different forms of the bytecode:

- · Raw bytes
- · disassembled representation
- · decompiled representation

All three serve different purposes and might be used at the same time.

First of all, we need to know a few things about the differences of representation. While the documentation says, that bytecode is structured in 16bit units, Androguard will use 8bit units to show the bytecode (i.e. bytes). If offsets are given in the bytecode, they are also presented as bytes. Also all indices are provided in byte length. Other than that, the mnemonic representation should follow in large parts the one provided in the documentation. Arguments are always shown in their "expanded" form, which is especially important for the few opcodes where only parts of the value are stores, like const/high16. In this case, the full value is shown including the 16 lower zero bits. As Dalvik is closely related to Java, all integer values are represented as signed int (32bit value) or long (64bit). Values are either given in decimal or hexadecimal representation. If the value is hexadecimal, the value is suffixed with a h, i.e. f7a0h or 63392.

In the following few examples, we will take the provided APK file examples\android\TestsAndroguard\bin\TestActiviapk and assume that you have loaded it via AnalyzeAPK and have the following objects:

```
>>> a
<androguard.core.bytecodes.apk.APK object at 0x0000000058DD240>
>>> d
[<androguard.core.bytecodes.dvm.DalvikVMFormat object at 0x000000004CE4CF8>]
>>> dx
<analysis.Analysis VMs: 1, Classes: 492, Strings: 496>
```

## Getting the raw bytecode

Our first task is to extract the raw bytecode of all methods.

```
for method in dx.get_methods():
    if method.is_external():
        continue

# Need to get the EncodedMethod from the MethodClassAnalysis object

m = method.get_method()

if m.get_code():
    # get_code() returns None or a DalvikCode object
    # get_bc() returns a DCode object
    # get_raw() returns bytearray
    print(m.get_code().get_bc().get_raw())
```

This will print a lot of bytearrays.

### **Getting disassembled instructions**

Next, we would like to get the disassembled instructions. The instruction itself have many different methods and you can find a detailed description in the documentation of the *Instruction* class.

```
for method in dx.get_methods():
    if method.is_external():
        continue
```

1.1. Introduction 15

```
m = method.get_method()
for idx, ins in m.get_instructions_idx():
    print(idx, ins.get_op_value(), ins.get_name(), ins.get_output())
```

This will print something like:

```
0 91 iput-object v1, v0, LTestDefaultPackage$TestInnerClass$TestInnerInnerClass;->this
→$1 LTestDefaultPackage$TestInnerClass;
4 112 invoke-direct v0, Ljava/lang/Object;-><init>()V
10 89 iput v2, v0, LTestDefaultPackage$TestInnerClass$TestInnerInnerClass;->a I
14 89 iput v3, v0, LTestDefaultPackage$TestInnerClass$TestInnerInnerClass;->c I
18 14 return-void
```

The variable idx is the index counted in bytes where the opcode starts. ins.get\_op\_value() returns the integer value of the opcode, ins.get\_name() the mnemonic and ins.get\_output() the parsed arguments.

As an example, let's count the number of individual opcodes and create some statistics:

```
from collections import defaultdict
from operator import itemgetter
c = defaultdict(int)

for method in dx.get_methods():
    if method.is_external():
        continue
    m = method.get_method()
    for ins in m.get_instructions():
        c[(ins.get_op_value(), ins.get_name())] += 1

for k, v in sorted(c.items(), key=itemgetter(1), reverse=True)[:10]:
        print(k, '-->', v)
```

This will output the top ten opcodes and the count:

```
(110, 'invoke-virtual') --> 3532

(84, 'iget-object') --> 2223

(12, 'move-result-object') --> 1749

(18, 'const/4') --> 1156

(112, 'invoke-direct') --> 1130

(10, 'move-result') --> 1111

(14, 'return-void') --> 1106

(56, 'if-eqz') --> 898

(26, 'const-string') --> 806

(113, 'invoke-static') --> 755
```

As another example, we will collect all constant integer values:

```
print('minimal:', min(c))
print('maximal:', max(c))
print('length: ', len(c))
```

#### This will print:

```
minimal: -4616189618054758400
maximal: 4707499256968118272
length: 205
```

## Get processed bytecode from decompiler

The last topic is how to get the processed bytecode from the decompiler. If you are only interested in the decompiled source code, you can use the source () function:

```
for method in dx.get_methods():
    if method.is_external():
        continue
    m = method.get_method()
    print(m.source())
```

It will print all sources of all methods.

But, you can also use DAD to compile abstract syntax trees (AST) for you. An AST can easily be used to do analysis on the code itself. Unfortunately, the method to get to the AST is a little bit awkward:

```
from pprint import pprint
from androguard.decompiler.dad.decompile import DvMethod
for method in dx.get_methods():
    if method.is_external():
        continue
    dv = DvMethod(dx.get_method(method.get_method()))
    dv.process(doAST=True)
    pprint(dv.get_ast())
```

The AST is a dictionary, wich might look like this one:

```
{ 'body': ['BlockStatement',
          None,
          [['ExpressionStatement',
            ['Assignment',
             [['FieldAccess',
               [['Local', 'this']],
               (TestDefaultPackage$TestInnerClass$TestInnerInnerClass,
                this$1.
                LTestDefaultPackage$TestInnerClass;)],
              ['Local', 'p1']],
             '']],
           ['ExpressionStatement',
            ['Assignment',
             [['FieldAccess',
               [['Local', 'this']],
               (TestDefaultPackage$TestInnerClass$TestInnerInnerClass, a, I)],
              ['Local', 'p2']],
             '']],
```

(continues on next page)

```
['ExpressionStatement',
            ['Assignment',
             [['FieldAccess',
               [['Local', 'this']],
               (TestDefaultPackage$TestInnerClass$TestInnerInnerClass, c, I)],
              ['Local', 'p3']],
             '']],
           ['ReturnStatement', None]]],
'comments': [],
'flags': ['private'],
'params': [[['TypeName', (TestDefaultPackage$TestInnerClass, 0)],
             ['Local', 'p1']],
            [['TypeName', ('.int', 0)], ['Local', 'p2']],
            [['TypeName', ('.int', 0)], ['Local', 'p3']]],
'ret': ['TypeName', ('.void', 0)],
'triple': (TestDefaultPackage$TestInnerClass$TestInnerInnerClass,
            <init>,
            (LTestDefaultPackage$TestInnerClass; II) V) }
```

This AST is the equivalent of the following source code:

## 1.1.6 Working with Sessions

If you are working on a larger APK, you might want to save your current work and come back later. Thats the reason for sessions: They allow you to save your work on disk and resume it at any point. Sessions could also be used to store the analysis on disk, for example if you do automated analysis and want to analyse certain files later.

There are several ways to work with sessions. The easiest way is to use AnalyzeAPK() with a session:

```
from androguard import misc
from androguard import session

# get a default session
sess = misc.get_default_session()

# Use the session
a, d, dx = misc.AnalyzeAPK("examples/android/abcore/app-prod-debug.apk", session=sess)

# Show the current Session information
sess.show()

# Do stuff...

# Save the session to disk
session.Save(sess, "androguard_session.ag")
```

```
# Load it again
sess = session.Load("androguard_session.ag")
```

The session information will look like this:

```
APKs in Session: 1
d5e26acca809e9cdfaece18afd8e63c60a26d7b6d566d70bd9f44d6934d5c433: [<androguard.
core.bytecodes.apk.APK object at 0x7fcecf4f3f10>]

DEXs in Session: 2
8bd7e9f48a6ed29e4c678633364e8bfd4e6ae76ef3e50c43a5ec3c00eb10a5bc: <analysis.
Analysis VMs: 2, Classes: 3092, Strings: 3293>
e2a1e46ecd03b701ce72c31057581e0104279d142fca06cdcdd000dd94a459e0: <analysis.
Analysis VMs: 2, Classes: 3092, Strings: 3293>

Analysis in Session: 1
d5e26acca809e9cdfaece18afd8e63c60a26d7b6d566d70bd9f44d6934d5c433: <analysis.
Analysis VMs: 2, Classes: 3092, Strings: 3293>
```

Similar functionality is available from the Session directly, but needs a second function to retrive the analyzed objects from the Session:

```
from androguard.session import Session

s = Session()
sha256 = s.add("examples/android/abcore/app-prod-debug.apk")

a, d, dx = s.get_objects_apk(digest=sha256)

s.show()

# When no filename is given, the Session will be saved at the current directory saved_file = s.save()
# ... and return the filename of the Session file
print(saved_file)
```

Note: Session objects store a lot of data and can get very big!

It is recommended not to use sessions in automated environments, where hundrets or thousands of APKs are loaded.

If you want to use sessions but keep the session alive only for one or multiple APKs, you can call the reset () method on a session, to remove all stored analysis data.

```
from androguard import misc
from androguard import session
import os

# get a default session
sess = misc.get_default_session()

for root, dirs, files in os.walk("examples")
    for f in files:
        if f.endswith(".apk"):
            # Use the session
            a, d, dx = misc.AnalyzeAPK(os.path.join(root, f), session=sess)

# Do your stuff
```

(continues on next page)

```
# Maybe save the session to disk...
# But now reset the session for the next analysis
sess.reset()
```

## 1.1.7 Use JADX as a Decompiler

Instead of using the internal decompiler DAD, you can also use JADX.

Install JADX as described at it's website. Make sure that the jadx executable is in \$PATH. Otherwise you might set the argument when calling <code>DecompilerJADX()</code>.

Here is a short demo code, how JADX can be used:

```
from androguard.core.bytecodes.apk import APK
from androquard.core.bytecodes.dvm import DalvikVMFormat
from androguard.core.analysis.analysis import Analysis
from androguard.decompiler.decompiler import DecompilerJADX
from androguard.core.androconf import show_logging
import logging
# Enable log output
show_logging(level=logging.DEBUG)
# Load our example APK
a = APK("examples/android/TestsAndroguard/bin/TestActivity.apk")
# Create DalvikVMFormat Object
d = DalvikVMFormat(a)
# Create Analysis Object
dx = Analysis(d)
# Load the decompiler
# Make sure that the jadx executable is found in $PATH
# or use the argument jadx="/path/to/jadx" to point to the executable
decompiler = DecompilerJADX(d, dx)
# propagate decompiler and analysis back to DalvikVMFormat
d.set_decompiler(decompiler)
d.set_vmanalysis(dx)
# Now you can do stuff like:
for m in d.get_methods()[:10]:
   print (m)
   print (decompiler.get_source_method(m))
```

## 1.1.8 Android Signing Certificates

Androguard has the ability to get information about the signing certificate found in APKs. Over the last versions of Androguard, different parsers has been used to get certificate information. The first parser was Chilkat, then a mixture of pyasn1 and cryptography was used, while the latest parser uses the asn1crypto library. Not all x509 parsers work with all certificates as there are plenty of examples where the certificate creator does not follow the RFCs for creating certificates. Some parsers do not accept such broken certificates and will fail to parse them.

The purpose of Androids signing process is not to provide verified information about the author, like with JAR signing, but only provide a way to check the integrity of the APK as well as check if an APK can be upgraded by comparing the certificate fingerprints. In some sense, the certificate information can be used to find other APKs from the same author - as long as the signing key was kept secret! There are also public available signing keys, like the ones from AOSP, thus the same fingerprint of two APKs does not always tell you it was signed by the same person.

If you like to know more about the APK signing process, please read the official documentation about Signing. There is also an official tool to verify and sign APKs called apksigner.

## Working with certificates

Inside the APK, there are two places for certificates:

- v1 aka JAR signing: PKCS#7 files in the META-INF folder
- v2 aka APK signing: a special section in the ZIP containing DER coded certificates

The easiest way to get to the certificate information is *androguard sign - Print Certificate Fingerprints*. It gives similar output to apksigner, but uses only androguard. It can not verify the integrity of the file though.

```
$ androsign.py --all --show examples/signing/apksig/golden-aligned-v1v2-out.apk
golden-aligned-v1v2-out.apk, package: 'android.appsecurity.cts.tinyapp'
Is signed v1: True
Is signed v2: True
Found 1 unique certificates
Issuer: CN=rsa-2048
Subject: CN=rsa-2048
Serial Number: 0x8e35306cdd0115f7L
Hash Algorithm: sha256
Signature Algorithm: rsassa_pkcs1v15
Valid not before: 2016-03-31 14:57:49+00:00
Valid not after: 2043-08-17 14:57:49+00:00
shal 0aa07c0f297b4ae834dc85a17eea8c2cf9380ff7
sha256 fb5dbd3c669af9fc236c6991e6387b7f11ff0590997f22d0f5c74ff40e04fca8
sha512.
md5 e995a5ed7137307661f854e66901ee9e
```

#### As a comparison, here is the output of apksigner:

```
$ apksigner verify -verbose --print-certs examples/signing/apksig/golden-aligned-v1v2-
→out.apk
Verified using v1 scheme (JAR signing): true
Verified using v2 scheme (APK Signature Scheme v2): true
Number of signers: 1
Signer #1 certificate DN: CN=rsa-2048
Signer #1 certificate SHA-256 digest:
 \hspace{2.5cm} \hookrightarrow \hspace{-.1cm} fb5dbd3c669af9fc236c6991e6387b7f11ff0590997f22d0f5c74ff40e04fca8 \\
Signer #1 certificate SHA-1 digest: 0aa07c0f297b4ae834dc85a17eea8c2cf9380ff7
Signer #1 certificate MD5 digest: e995a5ed7137307661f854e66901ee9e
Signer #1 key algorithm: RSA
Signer #1 key size (bits): 2048
Signer #1 public key SHA-256 digest:...
 \hspace*{-0.2cm} \leftarrow 8 cabaedf32f1052f6bc5edbeb84d1c500f8c1aa15f8944bf22c46e44c5c4f7e8 \\
Signer #1 public key SHA-1 digest: a708f9a777bac814e6634b02521224537ec3e019
Signer #1 public key MD5 digest: c0c8801fabf2ad970282be1c41584003
```

The most interesting part is probaby the fingerprint of the certificate (not of the public key!). You can use it to search for similar APKs. Sometimes there is a confusion about this fingerprint: The fingerprint is not the checksum of the whole PKCS#7 file, but only of a certain part of it! Calculating the hash of a PKCS#7 file from two different, but equally signed APKs will result in a different hash. The fingerprint will stay the same though.

Androguard offers methods in the androguard.core.bytecodes.apk.APK class to iterate over the certificates found there.

```
from androguard.core.bytecodes.apk import APK
a = APK('examples/signing/apksig/golden-aligned-v1v2-out.apk')
# first check if this APK is signed
print("APK is signed: {}".format(a.is_signed()))
if a.is_signed():
    # Test if signed v1 or v2 or both
   print("APK is signed with: {}".format("both" if a.is_signed_v1() and
   a.is_signed_v2() else "v1" if a.is_signed_v1() else "v2"))
# Iterate over all certificates
for cert in a.get_certificates():
    # Each cert is now a asn1crypt.x509.Certificate object
    # From the Certificate object, we can query stuff like:
   cert.shal # the shal fingerprint
   cert.sha256 # the sha256 fingerprint
   cert.issuer.human_friendly # issuer
   cert.subject.human_friendly # subject, usually the same
   cert.hash_algo # hash algorithm
   cert.signature_algo # Signature algorithm
   cert.serial_number # Serial number
    cert.contents # The DER coded bytes of the certificate itself
```

Please referr to the asn1crypto documentation for more information on the features of the Certificate class!

## 1.1.9 Android Binary XML Format

Android uses a special format to save XML and resource files. Also resource files are XML files in the source folder, but all resources are packed into a single resource file called resources. arsc. The underlying format is chunk based and is capable for storing several different information.

The most common AXML file is the AndroidManifest.xml. This file must be part of every APK, and contains the meta-information about the package.

Androguard is capable of decoding such files and two different tools exists for decoding:

- 1) androquard arsc for decoding resources.arsc.
- 2) androguard axml for decoding AndroidManifest.xml and all other XML files

### Decode the AndroidManifest.xml

Let's use one of the example files provided by androguard. To decode the AndroidManifest.xml of an APK file, simply give androguard axml the APK file as an argument:

```
$ androguard axml examples/android/TestsAndroguard/bin/TestActivity.apk
```

The output will look like this:

You can check with the original, uncompiled, XML file, which can be found here:

```
$ cat examples/android/TestsAndroguard/AndroidManifest.xml
```

The original file will print:

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"</pre>
   package="tests.androguard"
   android:versionCode="1"
   android:versionName="1.0" >
    <uses-sdk
       android:minSdkVersion="9"
        android:targetSdkVersion="16" />
    <application
       android:allowBackup="false"
        android:icon="@drawable/icon"
        android:label="@string/app_name" >
        <activity
            android:name="TestActivity"
           android:label="@string/app_name" >
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>
```

Note, that the overall structure is equal but there are certain differences.

- 1) Resource labels are hex numbers in the decompiled version but strings in the original one
- 2) Newlines and whitespaces are different.

Due to the compilation, this information is lost. But it does not matter, as the structure of the Manifest does not matter. To get some information about the resource IDs, we need information from the resources.arsc.

To retrive information about a single ID, simply run the following:

```
$ androguard arsc examples/android/TestsAndroguard/bin/TestActivity.apk --id 7F040001 @7f040001 resolves to '@tests.androguard:string/app_name' 
<default> = 'TestsAndroguardApplication'
```

You can see, that the ID 7F040001 was successfully resolved to the same string from the source file. To understand how Android handles resource configurations, you should read HandlingResources.

## Decode any other XML file

Also layout files or other XML files provided with the APK are compiled. To decompile them, just give the path inside the APK as an argument, or specify the binary XML file directly:

### Decode information from the resources.arsc

To get XML resource files out of the binary resources.arsc, use androquard arsc.

For example, get all string resources of an APK:

```
$ androguard arsc examples/android/TestsAndroguard/bin/TestActivity.apk --type string
```

will give the following output:

```
<resources>
<string name="hello">Hello World, TestActivity! kikoololmodif</string>
<string name="app_name">TestsAndroguardApplication</string>
</resources>
```

You can also list all resource types:

```
$ androguard arsc examples/android/TestsAndroguard/bin/TestActivity.apk --list-types
In Package: tests.androguard
In Locale: \x00\x00
    drawable
    layout
    public
    string
```

#### Working with AXML and Resource files from python

To load an AXML file, for example the AndroidManifest.xml, use the AXMLPrinter:

```
from androguard.core.bytecodes.axml import AXMLPrinter
with open("AndroidManifest.xml", "rb") as fp:
    a = AXMLPrinter(fp.read())

# Get the lxml.etree.Element from the AXMLPrinter:
xml = a.get_xml_obj()
```

```
# For example, get all uses-permission:
xml.findall("uses-permission")
```

In order to use resources, you need the ARSCParser:

## 1.1.10 Bulk Analysis

Androguard is capable of analysing probably thousand to millions of APKs. It is also possible to use tools like *multiprocessing* for this job and analyse APKs in parallel. Usually you want to put the results of your analysis somewhere, for example a database or some log file. It is also possile to use *Session* objects for this job, but you should be aware of some caveats!

1) Sessions take up a lot of space per APK. The resulting Session object can be more than 30 times larger than the original APK 2) Sessions should not be used to add unrelated APKs, again the size will blow up and you need to figure out which APK belongs to where

So the rule of thumb would be to not use Sessions for bulk analysis, only if you know what you are doing. Another way is to pickle the resulting objects. As the <code>DalvikVMFormat</code> are already stored in the <code>Analysis</code> object, there is no need to pickle them separately. Thus, it is only required to save the <code>APK</code> and <code>Analysis</code> object.

This is an example how to obtain the two objects and saving them to disk:

```
import sys
from pickle import dump
from hashlib import sha512
from androguard.misc import AnalyzeAPK

a, _, dx = AnalyzeAPK('examples/tests/a2dp.Vol_137.apk')
sha = sha512()
sha.update(a.get_raw())
with open("{}_apk.p".format(sha.hexdigest()), "wb") as fp:
    dump(a, fp)

with open("{}_analysis.p".format(sha.hexdigest()), "wb") as fp:
```

(continues on next page)

```
# It looks like here is the recursion problem...
sys.setrecursionlimit(50000)
dump(dx, fp)
```

But the resulting files are very large, especially the Analysis package:

But it is possible to compress both files to save disk space:

```
import sys
import lzma
from pickle import dump
from hashlib import sha512
from androguard.misc import AnalyzeAPK

a, _, dx = AnalyzeAPK('examples/tests/a2dp.Vol_137.apk')
sha = sha512()
sha.update(a.get_raw())
with lzma.open("{}_apk.p.lzma".format(sha.hexdigest()), "wb") as fp:
    dump(a, fp)

with lzma.open("{}_analysis.p.lzma".format(sha.hexdigest()), "wb") as fp:
    # It looks like here is the recursion problem...
    sys.setrecursionlimit(50000)
    dump(dx, fp)
```

which results in much smaller files:

```
$ du -sh *.lzma
4,5M_

$ 24a62690a770891a8f43d71e8f7beb24821d46a75e017ef4f4e6a04624105466621c96305d8e86f9900042e3ef1d5806a56

$ analysis.p.lzma
748K_

$ 24a62690a770891a8f43d71e8f7beb24821d46a75e017ef4f4e6a04624105466621c96305d8e86f9900042e3ef1d5806a56

$ apk.p.lzma
```

Obviously, as the APK is already packed, there is not much to compress anymore.

## **Using AndroAuto**

Another method is to use the framework *AndroAuto*. AndroAuto allows you to write small python classes which implement some method, which are then called by AndroAuto at certain points in time. AndroAuto is capable of

analysing thousands of apps, and uses threading to distribute the load to multiple CPUs. The results of your analysis can then be dumped to disk, or you could write your own method of saving them - for example, in a database.

The two key components are a Logger, for example <code>DefaultAndroLog</code> and an Analysis Runner, for example <code>DefaultAndroAnalysis</code>. Both are passed via a settings dictionary into <code>AndroAuto</code>.

Next, a minimal working example is given:

```
from androguard.core.analysis import auto
import sys
class AndroTest (auto.DirectoryAndroAnalysis):
   def __init__(self, path):
       super(AndroTest, self).__init__(path)
       self.has crashed = False
   def analysis_app(self, log, apkobj, dexobj, analysisobj):
        # Just print all objects to stdout
        print(log.id_file, log.filename, apkobj, dexobj, analysisobj)
    def finish(self, log):
       # This method can be used to save information in `log`
       # finish is called regardless of a crash, so maybe store the
       # information somewhere
       if self.has_crashed:
         print("Analysis of {} has finished with Errors".format(log))
          print("Analysis of {} has finished!".format(log))
   def crash(self, log, why):
       # If some error happens during the analysis, this method will be
       # called
       self.has_crashed = True
      print("Error during analysis of {}: {}".format(log, why), file=sys.stderr)
settings = {
    # The directory `some/directory` should contain some APK files
    "my": AndroTest('some/directory'),
    # Use the default Logger
   "log": auto.DefaultAndroLog,
    # Use maximum of 2 threads
    "max_fetcher": 2,
}
aa = auto.AndroAuto(settings)
aa.go()
```

In this example, the analysis\_app() function is used to get all created objects of the analysis and just print them to stdout.

More information can be found in the documentation of AndroAuto.

## 1.1.11 Debugging Broken APKs

Sometimes you will have troubles to get something done with androguard. This is usually the case if an APK uses some edge cases or deliberately tries to break parsers - which is not uncommon for malware.

Please feel free to open a bug report in such cases, so this error can be fixed. But before you do, try to gather some

more information about the APK. Sometimes not only androguard failes to decode the file, but the official tools do as well!

It is also always interesting to know, if such a broken file can still be installed on an Android system. If you like to test this, fire up an emulator and try to run the APK there.

#### AXML Parser / AndroidManifest.xml

Many errors happen in the parsing of the *AndroidManifest.xml*.

There are two official tools you can use to decode the *AndroidManifest.xml*:

- 1. aapt2
- 2. apkanalyzer

Both are available in the AndroidSDK. While aapt2 can only decode the structure of the file, apkanalyzer can give an actual XML:

```
$ apkanalyzer manifest print org.fdroid.fdroid_1002052.apk | head
<?xml version="1.0" encoding="utf-8"?>
<manifest
   xmlns:android="http://schemas.android.com/apk/res/android"
   android:versionCode="1002052"
   android:versionName="1.2.2"
   android:installLocation="0"
   package="org.fdroid.fdroid"
   platformBuildVersionCode="24"
   platformBuildVersionName="7.0">
$ aapt2 dump org.fdroid.fdroid_1002052.apk --file AndroidManifest.xml | head
Binary XML
N: android=http://schemas.android.com/apk/res/android (line=2)
 E: manifest (line=2)
   A: http://schemas.android.com/apk/res/android:versionCode(0x0101021b)=1002052
   A: http://schemas.android.com/apk/res/android:versionName(0x0101021c)="1.2.2"
A: http://schemas.android.com/apk/res/android:installLocation(0x010102b7)=0
   A: package="org.fdroid.fdroid" (Raw: "org.fdroid.fdroid")
   A: platformBuildVersionCode=24 (Raw: "24")
   A: platformBuildVersionName=7 (Raw: "7.0")
     E: uses-sdk (line=8)
```

Both outputs are actually useful, as aapt2 can provide much more detailed information about the format than apkanalyzer does.

### **Broken ZIP files**

As you might know, APK files are actually just ZIP files. You can test the zip file integrity using the ZIP command itself:

```
$ zip -T org.fdroid.fdroid_1002052.apk
test of org.fdroid.fdroid_1002052.apk OK
```

If there are any errors, like wrong CRC32, these get reported here. Other ZIP implementations have similar tools to check ZIP files.

## Verifying the APK Signature

You can check the signature of the file using apksigner from the AndroidSDK:

```
$ apksigner verify --verbose --print-certs org.fdroid.fdroid_1002052.apk
Verifies
Verified using v1 scheme (JAR signing): true
Verified using v2 scheme (APK Signature Scheme v2): false
Number of signers: 1
Signer #1 certificate DN: CN=Ciaran Gultnieks, OU=Unknown, O=Unknown, L=Wetherby,...
→ST=Unknown, C=UK
Signer #1 certificate SHA-256 digest:..
\rightarrow 43238d512c1e5eb2d6569f4a3afbf5523418b82e0a3ed1552770abb9a9c9ccab
Signer #1 certificate SHA-1 digest: 05f2e65928088981b317fc9a6dbfe04b0fa13b4e
Signer #1 certificate MD5 digest: 17c55c628056e193e95644e989792786
Signer #1 key algorithm: RSA
Signer #1 key size (bits): 2048
Signer #1 public key SHA-256 digest:
-e3d2cc87a245da2e84d4fb71e527c164e084d48bccf76ffad46ad17f1bfde388
Signer #1 public key SHA-1 digest: 26ef7882633282a9b04688178ee7f372fbec7c3d
Signer #1 public key MD5 digest: 9225fccafb33b605a86cfc09d7f38ec6
WARNING: META-INF/rxandroid.properties not protected by signature. Unauthorized_
→modifications to this JAR entry will not be detected. Delete or move the entry
→outside of META-INF/.
WARNING: META-INF/rxjava.properties not protected by signature. Unauthorized_
→modifications to this JAR entry will not be detected. Delete or move the entry
→outside of META-INF/.
WARNING: META-INF/services/com.fasterxml.jackson.core.JsonFactory not protected by,
→signature. Unauthorized modifications to this JAR entry will not be detected...
→Delete or move the entry outside of META-INF/.
WARNING: META-INF/services/com.fasterxml.jackson.core.ObjectCodec not protected by...
→ signature. Unauthorized modifications to this JAR entry will not be detected...
→Delete or move the entry outside of META-INF/.
WARNING: META-INF/buildserverid not protected by signature. Unauthorized
\hookrightarrowmodifications to this JAR entry will not be detected. Delete or move the entry,
→outside of META-INF/.
WARNING: META-INF/fdroidserverid not protected by signature. Unauthorized_
→modifications to this JAR entry will not be detected. Delete or move the entry...
→outside of META-INF/.
```

## 1.2 Tools

The only tool you need is androguard - The swiss army knife. It combines all old tools into a single command line interface.

You can still use the other tools as well, but note that they might get removed some day.

## 1.2.1 androguard - The swiss army knife

androguard is the new tool, which combines all the other tools into a single command line interface application.

```
Usage: androquard [OPTIONS] COMMAND [ARGS]...
 Androguard is a full Python tool to play with Android files.
```

1.2. Tools 29

```
Options:
                        Show the version and exit.
  --version
  --verbose, --debug Print more
  --quiet
                        Print less (only warnings and above)
  --silent
                       Print no log messages
  --help
                        Show this message and exit.
Commands:
 analyze Open a IPython Shell and start reverse engineering.

apkid Return the packageName/versionCode/versionName per APK as...

Decode resources.arsc either directly from a given file or...
  axml
               Parse the AndroidManifest.xml.
  cq
               Create a call graph and export it into a graph format.
  decompile Decompile an APK and create Control Flow Graphs.
  disassemble Disassemble Dalvik Code with size SIZE starting from an...
                Androguard GUI
  qui
  sign
                Return the fingerprint(s) of all certificates inside an APK.
```

Take a look at the detailed description of each tool in the next sections.

## 1.2.2 androguard analyze - Androguard Shell

androlyze is a tool that spawns an IPython shell.

```
Usage: androguard analyze [OPTIONS] [APK]

Open a IPython Shell and start reverse engineering.

Options:
--session PATH Previously saved session to load instead of a file
--help Show this message and exit.
```

## 1.2.3 androguard cg - Create Call Graph from APK

androcg can create files that can be read using graph visualization software, for example gephi.

## **Synopsis**

```
Usage: androguard cg [OPTIONS] APK

Create a call graph and export it into a graph format.

The default is to create a file called callgraph.gml in the current directory!

classnames are found in the type "Lfoo/bar/bla;".

Example:

$ androguard cg examples/tests/hello-world.apk
```

```
Options:
 -o, --output TEXT
                             Filename of the output file, the extension is
                             used to decide which format to use [default:
                             callgraph.gml]
 -s, --show
                             instead of saving the graph, print it with
                             mathplotlib (you might not see anything!)
 -v, --verbose
                             Print more output
  --classname TEXT
                             Regex to filter by classname [default: .*]
 --methodname TEXT
                             Regex to filter by methodname [default: .*]
 --descriptor TEXT
                             Regex to filter by descriptor [default: .*]
 --accessflag TEXT
                             Regex to filter by accessflags [default: .*]
 --no-isolated / --isolated Do not store methods which has no xrefs
 --help
                             Show this message and exit.
```

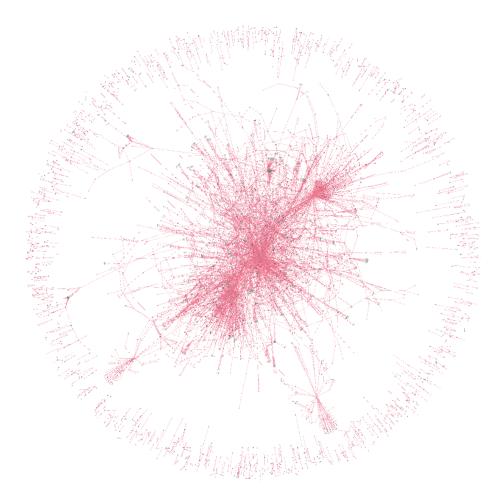
### **Examples**

The call graph is constructed from the Analysis object and then converted into a networkx *MultiDiGraph*. Currently supported formats are gml, gexf, gpickle, graphml, yaml and net.

The call graph contains methods as nodes and calls as edges. Each edge has the offset inside the method stored as an attribute and multiple calls between two methods result in multiple edges.

The methods to construct the callgraph from can be filtered. It is highly suggested to do that, as call graphs can get very large:

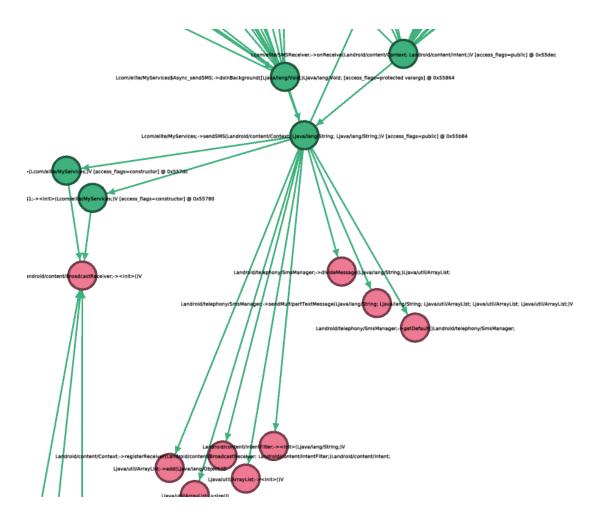
1.2. Tools 31



Of course, you can export the call graph with androguard and filter it later. If you filter with androguard, calls to this method will be stored, as well as calls from the method. For external methods only the first direction can be saved.

**Note:** External methods are not automatically API methods! It might be the case, that the method in question could not be found during disassembly time, hence it is stored as external.

Here is an example of an already filtered graph, visualized in gephi. Each node has an attribute to indicate if it is an internal (defined somewhere in the DEXs) or external (might be an API, but definetly not defined in the DEXs) method. In this case all green nodes are internal and all red ones are external. You can see the calls of some SMS Trojan to the API methods to write SMS.



# 1.2.4 androguard gui - Androguard GUI

Warning: The androgui is experimental and might not fully work!

```
Usage: androguard gui [OPTIONS]

Androguard GUI

Options:

-i, --input_file FILE Specify the inital file to load in the GUI
-p, --input_plugin PATH Additional Plugin (currently unused)
--help Show this message and exit.
```

## **Examples**

The androguard gui currently has functions to show disassmebled dalvik code, print all strings, methods, API usage and resources.

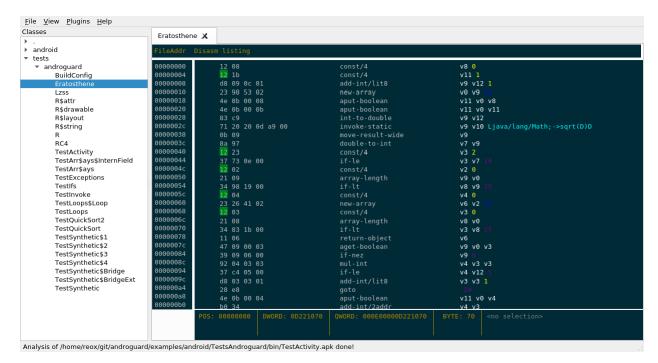
It uses Session in order to resume the work later.

1.2. Tools 33

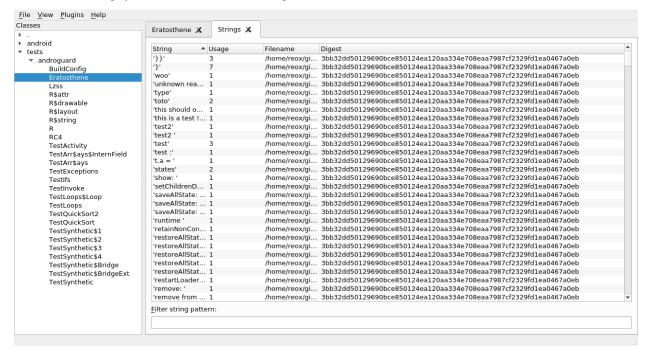
First, open up an APK using File, Open. If everything has worked, you will see all classes found inside the APK in the left tree view:



If you double click on one of the classes, you will get the disassembler view:

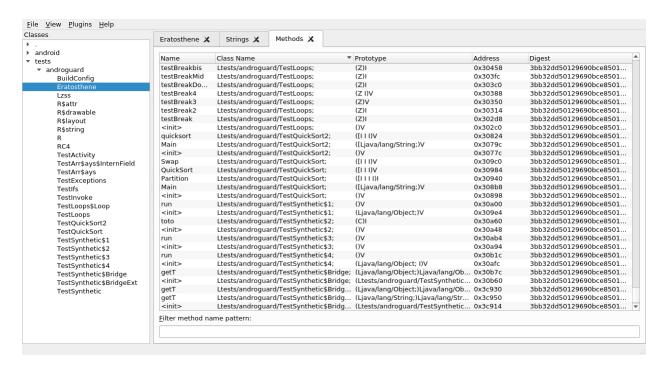


Under View, Strings you will find a list of all Strings inside the DEX file(s):

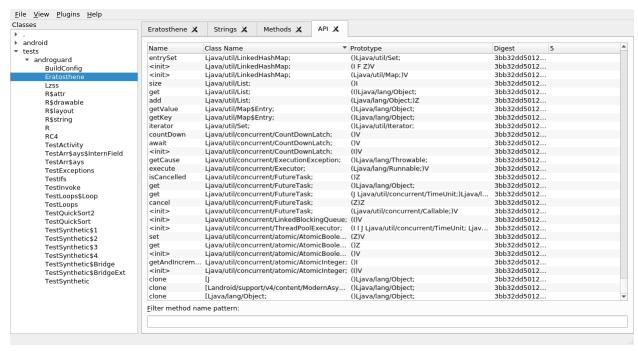


View, Methods shows all methods found in the DEX files(s):

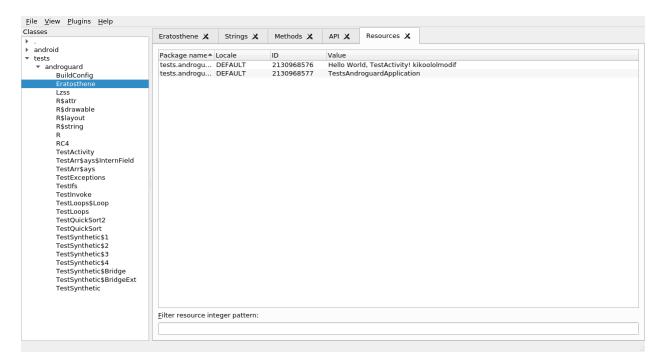
1.2. Tools 35



Using View, API you will get a list of all API methods (or bascically all external Methods) which are used inside the APK:



At last, you can get a list of all string resources from the resources.arsc file using View, Resources:



It is possible to add other APK or DEX files at any point using File, Add. In order to save the current state of the GUI and resume later, just go to File, Save and save the file as an .ag file. To resume later, just open the file with File, Open again.

# **Plugin System**

```
Warning: Plugins are not tested and there are no examples right now!
```

The androguard gui supports plugins to be loaded.

A plugin is a python file which implements the following class:

```
class PluginEntry:
    def __init__(self, session):
        """

        Session is a :class:`~androguard.session.Session` object.
        """
        self.session = session
```

# 1.2.5 androguard sign - Print Certificate Fingerprints

Get the fingerprints of the signing certificates inside an APK.

```
Usage: androguard sign [OPTIONS] [APK]...

Return the fingerprint(s) of all certificates inside an APK.

Options:
--hash [md5|sha1|sha256|sha512]
```

(continues on next page)

1.2. Tools 37

(continued from previous page)

```
Fingerprint Hash algorithm [default: shal]
-a, --all Print all supported hashes [default: False]
-s, --show Additionally of printing the fingerprints,
show more certificate information [default:
False]
--help Show this message and exit.
```

## **Examples**

```
$ androguard sign --all files/golden-aligned-v1v2-out.apk
golden-aligned-v1v2-out.apk, package: 'android.appsecurity.cts.tinyapp'
Is signed v1: True
Is signed v2: True
Found 1 unique certificates
md5 e995a5ed7137307661f854e66901ee9e
sha1 0aa07c0f297b4ae834dc85a17eea8c2cf9380ff7
sha512______4da6e6744a4dabef192b198be13b4492b0ce97469f3ce223dd9b7e8df2ee952328e06651e5e65dd3b60ac5e3946e16cf705
sha256 fb5dbd3c669af9fc236c6991e6387b7f11ff0590997f22d0f5c74ff40e04fca8
```

# 1.2.6 androguard axml - AndroidManifest.xml parser

Parse the AndroidManifest.xml from an APK and show/save the XML file.

```
Usage: androguard axml [OPTIONS] [FILE_]
 Parse the AndroidManifest.xml.
 Parsing is either direct or from a given APK and prints in XML format or
 saves to file.
 This tool can also be used to process any AXML encoded file, for example
 from the layout directory.
 Example:
     $ androguard axml AndroidManifest.xml
Options:
 -i, --input FILE
                      AndroidManifest.xml or APK to parse (legacy option)
 -o, --output TEXT
                      filename to save the decoded AndroidManifest.xml to,
                       default stdout
 -r, --resource TEXT Resource (any binary XML file) inside the APK to parse
                      instead of AndroidManifest.xml
                       Show this message and exit.
  --help
```

# 1.2.7 androguard arsc - resources.arsc parser

Parse the resources.arsc file from an APK and print human readable XML.

```
Usage: androguard arsc [OPTIONS] [FILE_]
 Decode resources.arsc either directly from a given file or from an APK.
 Example:
      $ androguard arsc app.apk
Options:
 -i, --input PATH
                      resources.arsc or APK to parse (legacy option)
 -o, --output TEXT
                      filename to save the decoded resources to
 -p, --package TEXT
                      Show only resources for the given package name
                      (default: the first package name found)
 -1, --locale TEXT
                      Show only resources for the given locale (default:
                      '\x00\x00')
 -t, --type TEXT
                      Show only resources of the given type (default: public)
 --id TEXT
                      Resolve the given ID for the given locale and package.
                      Provide the hex ID!
 -t, --list-packages List all package names and exit
 -t, --list-locales List all package names and exit
 -t, --list-types
                      List all types and exit
 --help
                      Show this message and exit.
```

# 1.2.8 androguard decompile - Decompile APKs and create CFG

androdd is a tool to create a decompiled version of an APK using the available decompilers.

## **Synopsis**

```
Usage: androguard decompile [OPTIONS] [FILE_]
 Decompile an APK and create Control Flow Graphs.
 Example:
     $ androquard resources.arsc
Options:
 -i, --input FILE
                      APK to parse (legacy option)
 -o, --output TEXT
                       output directory. If the output folder already
                       exsist, it will be overwritten! [required]
 -f, --format TEXT
                       Additionally write control flow graphs for each
                        method, specify the format for example png, jpg, raw
                        (write dot file), ...
 -j, --jar
                        Use DEX2JAR to create a JAR file
 -l, --limit TEXT
                        Limit to certain methods only by regex (default:
                        '.*')
 -d, --decompiler TEXT Use a different decompiler (default: DAD)
                        Show this message and exit.
```

It also can generate control flow graphs (CFG) for each method using the graphviz format. The CFGs can be exported as image file directly.

Additionally to the decompiled classes in .java format, each method is given in a SMALI like format (.ag files)

All filenames are sanatized, so they should work on most operating systems and filesystems.

1.2. Tools 39

# **Examples**

To get all CFG in png format and limit the processing only to a certain namespace, the following command can be used:

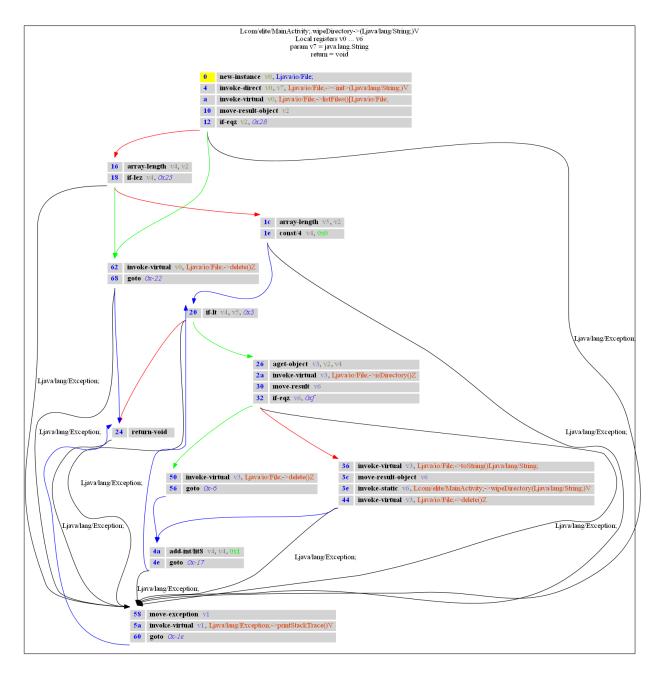
```
androguard decompile -o outputfolder -f png -i someapp.apk --limit "^Lcom/elite/.*"
```

Please make sure that graphviz and pydot are installed.

```
$ sudo apt-get install graphviz
$ pip install -U pydot
```

This will decompile the app *someapp.apk* into the folder *outputfolder* and limit the processing to all methods, where the classname starts with *com.elite*.

A CFG might look like this:



## while the .ag file has this content:

1.2. Tools 41

(continued from previous page)

```
(00000004) invoke-direct
                                          v0, v7, Ljava/io/File; -> <init>(Ljava/lang/
→String;) V
           (0000000a) invoke-virtual
                                          v0, Ljava/io/File; -> listFiles()[Ljava/io/
   2
→File:
            (00000010) move-result-object
   3
                                          v2
            (00000012) if-eqz
                                          v2, +28
   4
   0:55
    (Ljava/lang/Exception; -> 58 wipeDirectory-BB@0x58)
wipeDirectory-BB@0x16 : [ wipeDirectory-BB@0x1c wipeDirectory-BB@0x62 ]
          (00000016) array-length v4, v2
   6
           (00000018) if-lez
                                          v4, +25
   0:55
    (Ljava/lang/Exception; -> 58 wipeDirectory-BB@0x58)
wipeDirectory-BB@0x1c : [ wipeDirectory-BB@0x20 ]
           (0000001c) array-length
   7
                                          v5, v2
   8
           (0000001e) const/4
                                          v4, 0
   0:55
    (Ljava/lang/Exception; -> 58 wipeDirectory-BB@0x58)
wipeDirectory-BB@0x20 : [ wipeDirectory-BB@0x24 wipeDirectory-BB@0x26 ]
           (00000020) if-lt
   9
                                          v4, v5, +3
   0:55
    (Ljava/lang/Exception; -> 58 wipeDirectory-BB@0x58)
wipeDirectory-BB@0x24:
   10
          (00000024) return-void
    (Ljava/lang/Exception; -> 58 wipeDirectory-BB@0x58)
wipeDirectory-BB@0x26 : [ wipeDirectory-BB@0x36 wipeDirectory-BB@0x50 ]
                                          v3, v2, v4
        (00000026) aget-object
   12
           (0000002a) invoke-virtual
                                          v3, Ljava/io/File; ->isDirectory()Z
           (00000030) move-result
   13
                                          v6
           (00000032) if-eqz
   14
                                          v6, +f
   0:55
   (Ljava/lang/Exception; -> 58 wipeDirectory-BB@0x58)
wipeDirectory-BB@0x36 : [ wipeDirectory-BB@0x4a ]
          (00000036) invoke-virtual
   15
                                        v3, Ljava/io/File; ->toString()Ljava/lang/
→String:
           (0000003c) move-result-object v6
   16
           (0000003e) invoke-static
   17
                                         v6, Lcom/elite/MainActivity; ->
→wipeDirectory(Ljava/lang/String;)V
           (00000044) invoke-virtual v3, Ljava/io/File;->delete() Z
   18
   0:55
    (Ljava/lang/Exception; -> 58 wipeDirectory-BB@0x58)
wipeDirectory-BB@0x4a : [ wipeDirectory-BB@0x20 ]
   19 (0000004a) add-int/lit8 v4, v4, 1
   2.0
           (0000004e) goto
                                          -17
   0:55
    (Ljava/lang/Exception; -> 58 wipeDirectory-BB@0x58)
wipeDirectory-BB@0x50 : [ wipeDirectory-BB@0x4a ]
          (00000050) invoke-virtual
                                       v3, Ljava/io/File;->delete()Z
```

(continues on next page)

(continued from previous page)

```
22
            (00000056) goto
                                           -6
wipeDirectory-BB@0x58 : [ wipeDirectory-BB@0x24 ]
           (00000058) move-exception
   24
           (0000005a) invoke-virtual
                                           v1, Ljava/lang/Exception; ->
→printStackTrace()V
   25
           (00000060) goto
wipeDirectory-BB@0x62 : [ wipeDirectory-BB@0x24 ]
   26
       (00000062) invoke-virtual v0, Ljava/io/File;->delete()Z
    27
           (00000068) goto
    62:67
    (Ljava/lang/Exception; -> 58 wipeDirectory-BB@0x58)
```

# 1.2.9 androguard dissassemble - Disassembler for DEX

androdis is a disassembler for DEX files.

```
Usage: androguard disassemble [OPTIONS] DEX

Disassemble Dalvik Code with size SIZE starting from an offset

Options:

-o, --offset INTEGER Offset to start dissassembly inside the file
-s, --size INTEGER Number of bytes from offset to disassemble, 0 for whole file
--help Show this message and exit.
```

1.2. Tools 43

# CHAPTER

# **TWO**

# **COMMONLY USED APIS**

This is a just a selection of the most important top level API classes.

APK parser androguard.core.bytecodes.apk.APK

**DEX parser** androguard.core.bytecodes.dvm.DalvikVMFormat

AXML parser androguard.core.bytecodes.axml.AXMLPrinter

ARSC parser androguard.core.bytecodes.axml.ARSCParser

**Analysis** androguard.core.analysis.analysis.Analysis

Session androguard.session.Session

Automated Analysis androguard.core.analysis.auto.AndroAuto

Decompilers androguard.decompiler.decompiler

**CHAPTER** 

THREE

# **COMPLETE PYTHON API**

# 3.1 androguard package

# 3.1.1 Subpackages

androguard.core package

**Subpackages** 

# androguard.core.analysis package

The analysis module implements an abstraction layer for androguard.core.bytecodes.dvm. DalvikVMFormat objects. The the help of the androguard.core.analysis.analysis.Analsyis object, you can bundle several DEX files together. This is not only useful for multidex files, but also for a single dex, as Analysis offers many features to investigate DEX files. One of these features is crossreferencing (XREF). It allows you to build a graph of the methods inside the DEX files. You can then create callgraphs or find methods which use a specific API method.

#### **Submodules**

# androguard.core.analysis.analysis module

Warning: this feature is experimental and is currently not enabled by default! Use with caution!

Creates attributes for all classes, methods and fields on the Analysis object itself. This makes it easier to work with Analysis module in an iPython shell.

Classes can be search by typing dx.CLASS\_<tab>, as each class is added via this attribute name. Each class will have all methods attached to it via dx.CLASS\_Foobar.METHOD\_<tab>. Fields have a similar syntax: dx.CLASS Foobar.FIELD <tab>.

As Strings can contain nearly anything, use find\_strings() instead.

- Each CLASS\_item will return a ClassAnalysis
- Each METHOD\_item will return a MethodAnalysis
- Each FIELD item will return a FieldAnalysis

#### create\_xref()

Create Class, Method, String and Field crossreferences for all classes in the Analysis.

If you are using multiple DEX files, this function must be called when all DEX files are added. If you call the function after every DEX file, it will only work for the first time.

```
find_classes (name='.*', no_external=False)
```

Find classes by name, using regular expression This method will return all ClassAnalysis Object that match the name of the class.

#### **Parameters**

- name regular expression for class name (default ".\*")
- no\_external Remove external classes from the output (default False)

**Return type** Iterator[ClassAnalysis]

```
find_fields (classname='.*', fieldname='.*', fieldtype='.*', accessflags='.*')
find fields by regex
```

#### **Parameters**

- **classname** regular expression of the classname
- **fieldname** regular expression of the fieldname
- **fieldtype** regular expression of the fieldtype
- accessflags regular expression of the access flags

**Return type** Iterator[*FieldAnalysis*]

```
find_methods (classname='.*', methodname='.*', descriptor='.*', accessflags='.*', no external=False)
```

Find a method by name using regular expression. This method will return all MethodAnalysis objects, which match the classname, methodname, descriptor and accessflags of the method.

#### **Parameters**

- classname regular expression for the classname
- methodname regular expression for the method name
- descriptor regular expression for the descriptor
- accessflags regular expression for the accessflags
- no\_external Remove external method from the output (default False)

**Return type** Iterator[*MethodAnalysis*]

```
find_strings (string='.*')
Find strings by regex
```

Parameters string – regular expression for the string to search for

## **Return type** Iterator[StringAnalysis]

Generate a directed graph based on the methods found by the filters applied. The filters are the same as in find\_methods()

A networkx.MultiDiGraph is returned, containing all xrefs. That means a method which calls another method multiple times, will have multiple edges between them. Attached to the edge is the attribute *offset*, which gives the code offset inside the method of the call.

Specifying filters will not remove the methods if they are called by some other method.

The callgraph will check for both directions of edges. Thus, if you specify a single class as input, it will contain all classes which are called by this class (xref\_to), as well as all methods who calls the specified one (xref\_from).

Each node will contain the following meta information as attribute:

- external: is the method external or not (boolean)
- entrypoint: is the method a known entry point (boolean)
- native: is the method a native method by signature (boolean)
- public: is the method declared public (boolean)
- static: is the method declared static (boolean)
- vm: An ID of the DEX file where this method is declared or 0 if external (signed int)
- codesize: size of code of the method or zero if external (int)

#### **Parameters**

- classname regular expression of the classname (default: ".\*")
- methodname regular expression of the methodname (default: ".\*")
- **descriptor** regular expression of the descriptor (default: ".\*")
- accessflags regular expression of the access flags (default: ".\*")
- **no\_isolated** remove isolated nodes from the graph, e.g. methods which do not call anything (default: False)
- entry\_points A list of classes that are marked as entry point

Return type networkx.MultiDiGraph

## get\_class\_analysis (class\_name)

Returns the ClassAnalysis object for a given classname.

Parameters class\_name - classname like 'Ljava/lang/Object;' (including L and ;)

Returns ClassAnalysis

Return type ClassAnalysis

### get\_classes()

Returns a list of ClassAnalysis objects

Returns both internal and external classes (if any)

**Return type** Iterator[ClassAnalysis]

#### get external classes()

Returns all external classes, that means all classes that are not defined in the given set of DalvikVMObjects.

**Return type** Iterator[ClassAnalysis]

#### get field analysis(field)

Get the FieldAnalysis for a given fieldname

Parameters field (androguard.core.bytecodes.dvm.EncodedField) - the field

Returns FieldAnalysis

**Return type** FieldAnalysis

#### get\_fields()

Returns a list of *FieldAnalysis* objects

**Return type** Iterator[FieldAnalysis]

### get\_internal\_classes()

Returns all external classes, that means all classes that are defined in the given set of DalvikVMFormat.

**Return type** Iterator[ClassAnalysis]

## get\_method (method)

Get the MethodAnalysis object for a given EncodedMethod. This Analysis object is used to enhance EncodedMethods.

Parameters method - EncodedMethod to search for

Returns MethodAnalysis object for the given method, or None if method was not found

Return type MethodAnalysis

## get\_method\_analysis(method)

Get the MethodAnalysis object for a given EncodedMethod. This Analysis object is used to enhance EncodedMethods.

Parameters method - EncodedMethod to search for

Returns MethodAnalysis object for the given method, or None if method was not found

Return type MethodAnalysis

## get\_method\_analysis\_by\_name (class\_name, method\_name, method\_descriptor)

Returns the crossreferencing object for a given method.

This function is similar to <code>get\_method\_analysis()</code>, with the difference that you can look up the Method by name

### **Parameters**

- class\_name name of the class, for example 'Ljava/lang/Object;'
- method\_name name of the method, for example 'onCreate'
- method\_descriptor method descriptor, for example '(I I)V'

**Returns** MethodAnalysis

Return type MethodAnalysis

## get\_method\_by\_name (class\_name, method\_name, method\_descriptor)

Search for a EncodedMethod in all classes in this analysis

**Parameters** 

- class\_name name of the class, for example 'Ljava/lang/Object;'
- method name name of the method, for example 'onCreate'
- method\_descriptor descriptor, for example '(I I Ljava/lang/String)V

Returns EncodedMethod or None if method was not found

**Return type** androguard.core.bytecodes.dvm.EncodedMethod

#### get methods()

Returns a list of MethodAnalysis objects

**Return type** Iterator[*MethodAnalysis*]

## get\_permission\_usage (permission, apilevel=None)

Find the usage of a permission inside the Analysis.

example:: from androguard.misc import AnalyzeAPK a, d, dx = AnalyzeAPK("somefile.apk")

for meth in dx.get\_permission\_usage('android.permission.SEND\_SMS', a.get\_effective\_target\_sdk\_version()):
 print("Using API method { }".format(meth)) print("used in:") for \_, m, \_ in meth.get\_xref\_from():
 print(m.full\_name)

**Note:** The permission mappings might be incomplete! See also <code>get\_permissions()</code>.

#### **Parameters**

- permission the name of the android permission (usually 'android.permission.XXX')
- apilevel the requested API level or None for default

Returns yields MethodAnalysis objects for all using API methods

## get\_permissions (apilevel=None)

Returns the permissions and the API method based on the API level specified. This can be used to find usage of API methods which require a permission. Should be used in combination with an APK.

The returned permissions are a list, as some API methods require multiple permissions at once.

The following example shows the usage and how to get the calling methods using XREF:

example:: from androguard.misc import AnalyzeAPK a, d, dx = AnalyzeAPK("somefile.apk")

for meth, perm in dx.get\_permissions(a.get\_effective\_target\_sdk\_version()): print("Using
 API method {} for permission {}".format(meth, perm)) print("used in:") for \_, m, \_ in
 meth.get\_xref\_from():

```
print(m.full_name)
```

**..note::** This method might be unreliable and might not extract all used permissions. The permission mapping is based on [Axplorer](https://github.com/reddr/axplorer) and might be incomplete due to the nature of the extraction process. Unfortunately, there is no official API<->Permission mapping.

The output of this method relies also on the set API level. If the wrong API level is used, the results might be wrong.

Parameters apilevel – API level to load, or None for default

Returns yields tuples of MethodAnalysis (of the API method) and list of permission string

```
get strings()
          Returns a list of StringAnalysis objects
              Return type Iterator[StringAnalysis]
     get_strings_analysis()
          Returns a dictionary of strings and their corresponding StringAnalysis
              Return type Dict[str, StringAnalysis]
     is_class_present (class_name)
          Checks if a given class name is part of this Analysis.
              Parameters class_name - classname like 'Ljava/lang/Object;' (including L and ;)
              Returns True if class was found, False otherwise
              Return type bool
class androguard.core.analysis.analysis.BasicBlocks
     Bases: object
     This class represents all basic blocks of a method.
     It is a collection of many DVMBasicBlock.
     get()
              Returns yields each basic block (DVMBasicBlock object)
              Return type Iterator[DVMBasicBlock]
     get_basic_block (idx)
     get_basic_block_pos(item)
          Get the basic block at the index
              Parameters item - index
              Returns The basic block
              Return type DVMBasicBlock
     gets()
              Returns a list of basic blocks (DVMBasicBlock objects)
     pop(idx)
     push(bb)
          Adds another basic block to the collection
              Parameters bb (DVBMBasicBlock) - the DVMBasicBlock to add
class androguard.core.analysis.analysis.ClassAnalysis(classobj)
     Bases: object
     add_field_xref_read (method, classobj, field, off)
          Add a Field Read to this class
              Parameters
                 • method (MethodAnalysis) -
                  • classobj (ClassAnalysis) -
                  • field (androquard.code.bytecodes.dvm.EncodedField) -
                  • off (int) -
```

#### Returns

## add\_field\_xref\_write (method, classobj, field, off)

Add a Field Write to this class in a given method

#### **Parameters**

- method (MethodAnalysis) -
- classobj (ClassAnalysis) -
- field(androguard.core.bytecodes.dvm.EncodedField) -
- **off** (int) -

#### Returns

#### add\_method (method\_analysis)

Add the given method to this analyis. usually only called during Analysis.add and Analysis.\_resolve\_method

Parameters method\_analysis (MethodAnalysis) -

add\_method\_xref\_from (method1, classobj, method2, offset)

#### **Parameters**

- method1 (MethodAnalysis) -
- classobj (ClassAnalysis) -
- method2 (MethodAnalysis) -
- offset (int) -

add\_method\_xref\_to (method1, classobj, method2, offset)

### **Parameters**

- method1 (MethodAnalysis) the calling method
- classobj (ClassAnalysis) the calling class
- method2 (MethodAnalysis) the called method
- **offset** (*int*) offset in the bytecode of calling method

# add\_xref\_from(ref\_kind, classobj, methodobj, offset)

Creates a crossreference from this class. XrefFrom means, that the current class is called by another class.

#### **Parameters**

- ref kind (REF TYPE) type of call
- classobj (ClassAnalysis) ClassAnalysis object to link
- methodobj (MethodAnalysis) -
- offset (int) Offset in the methods bytecode, where the call happens

#### **Returns**

#### add\_xref\_to (ref\_kind, classobj, methodobj, offset)

Creates a crossreference to another class. XrefTo means, that the current class calls another class. The current class should also be contained in the another class' XrefFrom list.

Warning: The implementation of this specific method might not be what you expect! the parameter methodobj is the source method and not the destination in the case that ref\_kind is const-class or new-instance!

## **Parameters**

- ref\_kind (REF\_TYPE) type of call
- classobj (ClassAnalysis) ClassAnalysis object to link
- methodobj (MethodAnalysis) -
- offset (int) Offset in the Methods Bytecode, where the call happens

#### **Returns**

## property extends

Return the parent class

For external classes, this is not sure, thus we return always Object (which is the parent of all classes)

**Returns** a string of the parent class name

#### get\_class()

Returns the original Dalvik VM class or the external class object.

#### Returns

**Return type** Union[androguard.core.bytecodes.dvm.ClassDefItem, ExternalClass]

```
get_field_analysis (field)
```

#### get\_fields()

Return all FieldAnalysis objects of this class

## get\_method\_analysis(method)

Return the MethodAnalysis object for a given EncodedMethod

Parameters method - EncodedMethod

Returns MethodAnalysis

**Return type** *MethodAnalysis* 

## get\_methods()

Return all MethodAnalysis objects of this class

**Return type** Iterator[MethodAnalysis]

#### get\_nb\_methods()

Get the number of methods in this class

### get\_vm\_class()

Returns the original Dalvik VM class or the external class object.

#### Returns

**Return type** Union[androguard.core.bytecodes.dvm.ClassDefItem, ExternalClass]

#### get\_xref\_from()

Returns a dictionary of all classes calling the current class. This dictionary contains also information from which method the class is accessed.

Note: this method might contains wrong information about class usage!

The dictionary contains the classes as keys (stored as ClassAnalysis) and has a tuple as values, where the first item is the ref\_kind (which is an Enum of type REF\_TYPE), the second one is the method in which the class is called (MethodAnalysis) and the third the offset in the method where the call is originating.

```
example:: # dx is an Analysis object for cls in dx.find_classes('.*some/name.*'):
    print("Found class {} in Analysis".format(cls.name) for caller, refs in cls.get_xref_from().items():
    print(" called from {}".format(caller.name)) for ref_kind, ref_method, ref_offset in refs:
    print(" in method {} {}".format(ref_kind, ref_method))
```

**Return type** Iterator[Tuple[REF\_TYPE, MethodAnalysis, int]]

#### get xref to()

Returns a dictionary of all classes which are called by the current class. This dictionary contains also information about the method which is called.

**Note:** this method might contains wrong information about class usage!

The dictionary contains the classes as keys (stored as ClassAnalysis) and has a tuple as values, where the first item is the ref\_kind (which is an Enum of type REF\_TYPE), the second one is the method called (MethodAnalysis) and the third the offset in the method where the call is originating.

```
example:: # dx is an Analysis object for cls in dx.find_classes('.*some/name.*'):
```

```
print("Found class {} in Analysis".format(cls.name) for calling, refs in cls.get_xref_from().items():
```

print(" calling class {}".format(calling.name)) for ref\_kind, ref\_method, ref\_offset in refs:

print(" calling method { } { } ".format(ref\_kind, ref\_method))

**Return type** Iterator[Tuple[REF\_TYPE, MethodAnalysis, int]]

## property implements

Get a list of interfaces which are implemented by this class

Returns a list of Interface names

## is\_android\_api()

Tries to guess if the current class is an Android API class.

This might be not very precise unless an apilist is given, with classes that are in fact known APIs. Such a list might be generated by using the android.jar files.

Returns boolean

#### is\_external()

Tests if this class is an external class

**Returns** True if the Class is external, False otherwise

property name

```
Return the class name
              Returns
class androguard.core.analysis.analysis.DVMBasicBlock (start, vm, method, context)
     Bases: object
     A simple basic block of a dalvik method.
     A basic block consists of a series of Instruction which are not interrupted by branch or jump instructions
     such as goto, if, throw, return, switch etc.
     add_note (note)
     clear_notes()
     get_end()
          Get the end offset of this basic block
              Returns end offset
              Return type int
     get_exception_analysis()
     get_instructions()
          Get all instructions from a basic block.
              Returns Return all instructions in the current basic block
     get last()
          Get the last instruction in the basic block
              Returns androguard.core.bytecodes.dvm.Instruction
     get_last_length()
     get_method()
          Returns the originiating method
              Returns the method
              Return type androguard.core.bytecodes.dvm.EncodedMethod
     get_name()
     get_nb_instructions()
     get_next()
          Get next basic blocks
              Returns a list of the next basic blocks
              Return type DVMBasicBlock
     get_notes()
     get_prev()
          Get previous basic blocks
              Returns a list of the previous basic blocks
              Return type DVMBasicBlock
     get_special_ins (idx)
          Return the associated instruction to a specific instruction (for example a packed/sparse switch)
```

```
Parameters idx – the index of the instruction
              Return type None or an Instruction
     get_start()
          Get the starting offset of this basic block
              Returns starting offset
              Return type int
     push(i)
     set_childs (values)
     set_exception_analysis (exception_analysis)
     set_fathers(f)
     set_notes (value)
     show()
class androquard.core.analysis.analysis.ExceptionAnalysis(exception, bb)
     Bases: object
     get()
     show_buff()
class androquard.core.analysis.analysis.Exceptions
     Bases: object
     add (exceptions, basic_blocks)
     get()
     get_exception (addr_start, addr_end)
class androguard.core.analysis.analysis.ExternalClass(name)
     Bases: object
     The External Class is used for all classes that are not defined in the DEX file, thus are external classes.
          Parameters name – Name of the external class
     add_method (method)
     get_methods()
          Return the stored methods for this external class :return:
     get name()
          Returns the name of the ExternalClass object
class androguard.core.analysis.analysis.ExternalMethod(class_name, name, descrip-
                                                                      tor)
     Bases: object
     ExternalMethod is a stub class for methods that are not part of the current Analysis. There are two possibilities
```

ExternalMethod is a stub class for methods that are not part of the current Analysis. There are two possibilities for this:

- 1) The method is defined inside another DEX file which was not loaded into the Analysis
- 2) The method is an API method, hence it is defined in the Android system

External methods should have a similar API to *EncodedMethod* but obviously they have no code attached. The only known information about such methods are the class name, the method name and its descriptor.

#### **Parameters**

```
• class_name (str) - name of the class
```

```
• name (str) – name of the method
```

• **descriptor** (List[str]) – descriptor string

```
property full_name
```

Returns classname + name + descriptor, separated by spaces (no access flags)

```
get_access_flags_string()
```

Returns the access flags string.

Right now, this is always an empty strings, as we can not say what kind of access flags an external method might have.

```
get_class_name()
get_descriptor()
get_name()
property permission_api_name
```

Returns a name which can be used to look up in the permission maps

```
\textbf{class} \texttt{ androguard.core.analysis.analysis.FieldAnalysis} \ (\textit{field})
```

Bases: object

add\_xref\_read (classobj, methodobj, offset)

#### **Parameters**

- classobj (ClassAnalysis) -
- methodobj (MethodAnalysis) -
- offset (int) offset in the bytecode

add\_xref\_write(classobj, methodobj, offset)

#### **Parameters**

- classobj (ClassAnalysis) -
- methodobj (MethodAnalysis) -
- **offset** (*int*) offset in the bytecode

```
get_field()
```

Returns the actual field object

**Return type** androguard.core.bytecodes.dvm.EncodedField

```
get_xref_read (withoffset=False)
```

Returns a list of xrefs where the field is read.

The list contains tuples of the originating class and methods, where the class is represented as a ClassAnalysis, while the method is a MethodAnalysis.

**Parameters withoffset** (bool) – return the xrefs including the offset

```
get_xref_write (withoffset=False)
```

Returns a list of xrefs where the field is written to.

The list contains tuples of the originating class and methods, where the class is represented as a ClassAnalysis, while the method is a MethodAnalysis.

**Parameters withoffset** (bool) – return the xrefs including the offset

#### property name

```
class androguard.core.analysis.analysis.MethodAnalysis(vm, method)
```

Bases: object

This class analyses in details a method of a class/dex file It is a wrapper around a EncodedMethod and enhances it by using multiple DVMBasicBlock encapsulated in a BasicBlocks object.

## property access

Returns the access flags to the method as a string

## add\_xref\_from(classobj, methodobj, offset)

Add a crossrefernece from another method (this method is called by another method)

#### **Parameters**

- classobj ClassAnalysis
- methodobj EncodedMethod
- **offset** integer where in the method the call happens

# add\_xref\_to(classobj, methodobj, offset)

Add a crossreference to another method (this method calls another method)

#### **Parameters**

- classobj ClassAnalysis
- methodobj EncodedMethod
- offset integer where in the method the call happens

## property class\_name

Returns the name of the class of this method

# property descriptor

Returns the type descriptor for this method

#### property full\_name

Returns classname + name + descriptor, separated by spaces (no access flags)

## get\_access\_flags\_string()

Returns the concatenated access flags string

### get\_basic\_blocks()

Returns the <code>BasicBlocks</code> generated for this method. The <code>BasicBlocks</code> can be used to get a control flow graph (CFG) of the method.

Return type a BasicBlocks object

## get\_class\_name()

Return the class name of the method

#### get\_length()

**Returns** an integer which is the length of the code

Return type int

## get\_method()

**Return type** androguard.core.bytecodes.dvm.EncodedMethod

```
Returns
     get_vm()
              Return type androguard.core.bytecodes.dvm.DalvikVMFormat
              Returns
     get xref from()
          Returns a list of tuples containing the class, method and offset of the call, from where this object was
          called.
          The list of tuples has the form: (ClassAnalysis, EncodedMethod or ExternalMethod, int)
     get_xref_to()
          Returns a list of tuples containing the class, method and offset of the call, which are called by this method.
          The list of tuples has the form: (ClassAnalysis, EncodedMethod or ExternalMethod, int)
     is_android_api()
          Returns True if the method seems to be an Android API method.
          This method might be not very precise unless an list of known API methods is given.
              Returns boolean
     is external()
          Returns True if the underlying method is external
              Return type boolean
     property name
          Returns the name of this method
     show()
          Prints the content of this method to stdout.
          This will print the method signature and the decompiled code.
     show_xrefs()
class androguard.core.analysis.analysis.MethodClassAnalysis (meth)
     Bases: androquard.core.analysis.analysis.MethodAnalysis
     Deprecated since version 3.4.0: Always use MethodAnalysis! This method is just here for compatability
class androguard.core.analysis.analysis.REF_TYPE
     Bases: enum.IntEnum
     Stores the opcodes for the type of usage in an XREF.
     Used in ClassAnalysis to store the type of reference to the class.
     INVOKE DIRECT = 112
     INVOKE_DIRECT_RANGE = 118
     INVOKE_INTERFACE = 114
     INVOKE_INTERFACE_RANGE = 120
```

INVOKE\_STATIC = 113

INVOKE\_SUPER = 111

INVOKE\_STATIC\_RANGE = 119

INVOKE SUPER RANGE = 117

```
INVOKE_VIRTUAL = 110
INVOKE_VIRTUAL_RANGE = 116
REF_CLASS_USAGE = 28
REF_NEW_INSTANCE = 34
```

#### class androquard.core.analysis.analysis.StringAnalysis(value)

Bases: object

StringAnalysis contains the XREFs of a string.

As Strings are only used as a source, they only contain the XREF\_FROM set, i.e. where the string is used.

This Array stores the information in which method the String is used.

### add\_xref\_from(classobj, methodobj, off)

Adds a xref from the given method to this string

#### **Parameters**

- classobj (ClassAnalysis) -
- methodobj (MethodAnalysis) -
- **off** (*int*) offset in the bytecode of the call

#### get\_orig\_value()

Return the original, read only, value of the String

Returns the original value

## get\_value()

Return the (possible overwritten) value of the String

**Returns** the value of the string

```
get_xref_from(withoffset=False)
```

Returns a list of xrefs accessing the String.

The list contains tuples of the originating class and methods, where the class is represented as a ClassAnalysis, while the method is a MethodAnalysis.

#### is\_overwritten()

Returns True if the string was overwritten:return:

### set value(value)

Overwrite the current value of the String with a new value. The original value is not lost and can still be retrieved using <code>get\_orig\_value()</code>.

**Parameters value** (str) – new string value

```
androguard.core.analysis.analysis.is_ascii_obfuscation(vm)
```

Tests if any class inside a DalvikVMObject uses ASCII Obfuscation (e.g. UTF-8 Chars in Classnames)

Parameters vm – DalvikVMObject

**Returns** True if ascii obfuscation otherwise False

## androguard.core.analysis.auto module

```
class androguard.core.analysis.auto.AndroAuto(settings)
    Bases: object
```

The main class which analyse automatically android apps by calling methods from a specific object

Automatic analysis requires two objects to be created:

- 1) a Logger, found at key log in the settings
- 2) an Analysis runner, found at key my in the settings

Both are passed to AndroAuto via a dictionary. The setting dict understands the following keys:

- my: The Analysis runner (required)
- log: The Logger
- max\_fetcher: Maximum number of concurrent threads

DefaultAndroLog can be used as a baseclass for the Logger, while DefaultAndroAnalysis can be used a baseclass for the Analysis. There is also DirectoryAndroAnalysis which implements a fetcher which recursively reads a directory for files and can be used a baseclass as well.

example:

dump()

```
from androguard.core.analysis import auto

class AndroTest (auto.DirectoryAndroAnalysis):
    # This is the Test Runner

def analysis_app(self, log, apkobj, dexobj, analysisobj):
    # Just print all objects to stdout
    print(log.id_file, log.filename, apkobj, dexobj, analysisobj)

settings = {
    # The directory `some/directory` should contain some APK files
    "my": AndroTest('some/directory'),
    # Use the default Logger
    "log": auto.DefaultAndroLog,
    # Use maximum of 2 threads
    "max_fetcher": 2,
}

aa = auto.AndroAuto(settings)
aa.go()
```

# **Parameters** settings (dict) – the settings of the analysis

```
Dump the analysis

Calls dump() on the Analysis object

dump_file (filename)

Dump the analysis into a file

Calls dump_file(filename) on the Analysis object

go()

Launch the analysis.

this will start a total of max_fetcher threads.

class androguard.core.analysis.auto.DefaultAndroAnalysis

Bases: object

This class can be used as a template in order to analyse apps
```

The order of methods called in this class is the following:

- fetcher() is called to get files
- filter\_file() is called to get the filetype
- create\_apk() or create\_axml() or create\_arsc() and create\_dex() or create\_dey() depending on the filetype
- analysis\_apk() or analysis\_axml() or analysis\_arsc() and analysis\_dex() or analysis dey() depending on the filetype
- create\_adex() if at least one dex was found
- analysis\_app() with all the gathered objects so far
- finish() is called in any case after the analysis

crash () can be called during analysis if any Exception happens.

## analysis\_adex (log, adexobj)

This method is called in order to know if the analysis must continue

#### **Parameters**

- log an object which corresponds to a unique app
- adexobj (androguard.core.analysis.analysis.Analysis) a Analysis object

## Return type a boolean

## analysis\_apk (log, apkobj)

This method is called in order to know if the analysis must continue

#### **Parameters**

- log an object which corresponds to a unique app
- apkobj (androquard.core.bytecodes.apk.APK) a APK object

**Returns** True if a DEX file should be analyzed as well

## Return type bool

## analysis\_app (log, apkobj, dexobj, adexobj)

This method is called if you wish to analyse the final app

#### **Parameters**

- log an object which corresponds to a unique app
- apkobj (androquard.core.bytecodes.apk.APK) a APK object
- dexobj (androguard.core.bytecodes.dvm.DalvikVMFormat) a
   DalvikVMFormat object
- adexobj (androguard.core.analysis.analysis.Analysis) a Analysis object

#### analysis\_arsc(log, arscobj)

This method is called in order to know if the analysis must continue

## **Parameters**

• log – an object which corresponds to a unique app

arscobj (androguard.core.bytecodes.axml.ARSCParser) - a
 ARSCParser object

**Returns** True if the analysis should continue afterwards

## Return type bool

## analysis\_axml (log, axmlobj)

This method is called in order to know if the analysis must continue

#### **Parameters**

- log an object which corresponds to a unique app
- axmlobj (androguard.core.bytecodes.axml.AXMLPrinter) a
   AXMLPrinter object

**Returns** True if the analysis should continue afterwards

## Return type bool

### analysis\_dex (log, dexobj)

This method is called in order to know if the analysis must continue

#### **Parameters**

- log an object which corresponds to a unique app
- dexobj (androguard.core.bytecodes.dvm.DalvikVMFormat) a
   DalvikVMFormat object

Returns True if the analysis should continue with an analysis. Analysis

Return type bool

# analysis\_dey (log, deyobj)

This method is called in order to know if the analysis must continue

#### **Parameters**

- log an object which corresponds to a unique app
- deyobj (androguard.core.bytecodes.dvm.DalvikOdexVMFormat) a DalvikOdexVMFormat object

**Returns** True if the analysis should continue with an analysis. Analysis

# Return type bool

## crash (log, why)

This method is called if a crash happens

### **Parameters**

- log an object which corresponds to an unique app
- why the exception

# create\_adex (log, dexobj)

This method is called in order to create an Analysis object

#### **Parameters**

- log an object which corresponds to a unique app
- dexobj (androguard.core.bytecodes.dvm.DalvikVMFormat) a
   DalvikVMFormat object

## Rytpe a Analysis object

### create\_apk (log, fileraw)

This method is called in order to create a new APK object

#### **Parameters**

- log an object which corresponds to a unique app
- fileraw the raw apk (a string)

# Return type an APK object

## create\_arsc(log, fileraw)

This method is called in order to create a new ARSC object

#### **Parameters**

- log an object which corresponds to a unique app
- fileraw the raw arsc (a string)

## Return type an ARSCParser object

# create\_axml (log, fileraw)

This method is called in order to create a new AXML object

#### **Parameters**

- log an object which corresponds to a unique app
- fileraw the raw axml (a string)

## Return type an AXMLPrinter object

## create\_dex (log, dexraw)

This method is called in order to create a DalvikVMFormat object

#### **Parameters**

- log an object which corresponds to a unique app
- **dexraw** the raw classes.dex (a string)

#### Return type a DalvikVMFormat object

## create\_dey (log, dexraw)

This method is called in order to create a DalvikOdexVMFormat object

#### **Parameters**

- log an object which corresponds to a unique app
- **dexraw** the raw odex file (a string)

# Return type a DalvikOdexVMFormat object

#### dump()

This method is called to dump the result

#### dump\_file (filename)

This method is called to dump the result in a file

## Parameters filename – the filename to dump the result

# fetcher(q)

This method is called to fetch a new app in order to analyse it. The queue must be fill with the following format: (filename, raw)

must return False if the queue is filled, thus all files are read.

**Parameters q** – the Queue to put new app

```
filter_file (log, fileraw)
```

This method is called in order to filer a specific app

#### **Parameters**

- log an object which corresponds to a unique app
- fileraw (bytes) the raw file as bytes

**Return type** a tuple with 2 elements, the return value (boolean) if it is necessary to continue the analysis and the file type

finish(log)

This method is called before the end of the analysis

Parameters log – an object which corresponds to an unique app

```
class androguard.core.analysis.auto.DefaultAndroLog(id_file, filename)
```

Bases: object

A base class for the Androguard Auto Logger.

The Logger contains two attributes of the analyzed File: filename and id\_file, which is the Adler32 Checksum of the file.

The Logger can be extended to contain more attributes.

## class androquard.core.analysis.auto.DirectoryAndroAnalysis(directory)

Bases: androguard.core.analysis.auto.DefaultAndroAnalysis

A simple class example to analyse a whole directory with many APKs in it

## fetcher(q)

This method is called to fetch a new app in order to analyse it. The queue must be fill with the following format: (filename, raw)

must return False if the queue is filled, thus all files are read.

**Parameters q** – the Queue to put new app

## **Module contents**

## androguard.core.api\_specific\_resources package

## **Module contents**

## exception androguard.core.api\_specific\_resources.APILevelNotFoundError

Bases: Exception

```
androguard.core.api_specific_resources.load_permission_mappings(apilevel)
```

Load the API/Permission mapping for the requested API level. If the requested level was not found, None is returned.

**Parameters** apilevel – integer value of the API level, i.e. 24 for Android 7.0

**Returns** a dictionary of {MethodSignature: [List of Permissions]}

Load the Permissions for the given apilevel.

The permissions lists are generated using this tool: https://github.com/U039b/aosp\_permissions\_extraction

Has a fallback to select the maximum or minimal available API level. For example, if 28 is requested but only 26 is available, 26 is returned. If 5 is requested but 16 is available, 16 is returned.

If an API level is requested which is in between of two API levels we got, the lower level is returned. For example, if 5,6,7,10 is available and 8 is requested, 7 is returned instead.

#### **Parameters**

- apilevel integer value of the API level
- permtype either load permissions ('permissions') or

permission groups ('groups'):return: a dictionary of {Permission Name: {Permission info}

#### androguard.core.bytecodes package

The bytecodes modules are one very important core feature of Androguard. They contain parsers for APK, AXML, DEX, ODEX and DEY files as well for formats used inside these formats. These might be MUTF-8 for string encoding in DEX files as well as the widely used LEB128 encoding for numbers.

The most important modules might be androguard.core.bytecodes.apk.APK and androguard.core.bytecodes.dvm.DalvikVMFormat.

## **Submodules**

## androguard.core.bytecodes.apk module

```
 \textbf{class} \text{ androguard.core.bytecodes.apk.} \textbf{APK} (\textit{filename}, \textit{raw=False}, \textit{magic\_file=None}, \textit{skip\_analysis=False}, \textit{testzip=False})
```

Bases: object

## property files

Returns a dictionary of filenames and detected magic type

**Returns** dictionary of files and their mime type

```
find_tags (tag_name, **attribute_filter)
```

Return a list of all the matched tags in all available xml

**Parameters** tag(str) – specify the tag name

```
find_tags_from_xml (xml_name, tag_name, **attribute_filter)
```

Return a list of all the matched tags in a specific xml w :param str xml\_name: specify from which xml to pick the tag from :param str tag\_name: specify the tag name

#### get\_activities()

Return the android:name attribute of all activities

Return type a list of str

get\_all\_attribute\_value (tag\_name, attribute, format\_value=True, \*\*attribute\_filter)
Yields all the attribute values in xml files which match with the tag name and the specific attribute

**Parameters** 

- tag\_name (str) specify the tag name
- attribute (str) specify the attribute
- **format\_value** (bool) specify if the value needs to be formatted with packagename

## get\_all\_dex()

Return the raw data of all classes dex files

**Return type** a generator of bytes

## get\_android\_manifest\_axml()

Return the AXMLPrinter object which corresponds to the AndroidManifest.xml file

Return type AXMLPrinter

#### get\_android\_manifest\_xml()

Return the parsed xml object which corresponds to the AndroidManifest.xml file

Return type Element

### get\_android\_resources()

Return the ARSCParser object which corresponds to the resources arsc file

Return type ARSCParser

### get\_androidversion\_code()

Return the android version code

This information is read from the AndroidManifest.xml

Return type str

#### get\_androidversion\_name()

Return the android version name

This information is read from the AndroidManifest.xml

Return type str

## $get_app_icon(max_dpi=65536)$

Return the first icon file name, which density is not greater than max\_dpi, unless exact icon resolution is set in the manifest, in which case return the exact file.

This information is read from the AndroidManifest.xml

From https://developer.android.com/guide/practices/screens\_support.html and https://developer.android.com/ndk/reference/group\_\_\_configuration.html

- DEFAULT Odpi
- ldpi (low) 120dpi
- mdpi (medium) 160dpi
- TV 213dpi
- hdpi (high) 240dpi
- xhdpi (extra-high) 320dpi
- xxhdpi (extra-extra-high) 480dpi
- xxxhdpi (extra-extra-high) 640dpi
- anydpi 65534dpi (0xFFFE)
- nodpi 65535dpi (0xFFFF)

There is a difference between nodpi and anydpi: nodpi will be used if no other density is specified. Or the density does not match. nodpi is the fallback for everything else. If there is a resource that matches the DPI, this is used. anydpi is also valid for all densities but in this case, anydpi will overrule all other files! Therefore anydpi is usually used with vector graphics and with constraints on the API level. For example adaptive icons are usually marked as anydpi.

When it comes now to selecting an icon, there is the following flow:

- 1. is there an anydpi icon?
- 2. is there an icon for the dpi of the device?
- 3. is there a nodpi icon?
- 4. (only on very old devices) is there a icon with dpi 0 (the default)

For more information read here: https://stackoverflow.com/a/34370735/446140

#### Return type str

#### get\_app\_name()

Return the appname of the APK

This name is read from the AndroidManifest.xml using the application android:label. If no label exists, the android:label of the main activity is used.

If there is also no main activity label, an empty string is returned.

#### Return type str

# get\_attribute\_value (tag\_name, attribute, format\_value=False, \*\*attribute\_filter)

Return the attribute value in xml files which matches the tag name and the specific attribute

#### **Parameters**

- tag\_name (str) specify the tag name
- **attribute** (*str*) specify the attribute
- **format\_value** (bool) specify if the value needs to be formatted with packagename

### get\_certificate (filename)

Return a X.509 certificate object by giving the name in the apk file

Parameters filename – filename of the signature file in the APK

Returns a Certificate certificate

# ${\tt get\_certificate\_der} \ (\mathit{filename})$

Return the DER coded X.509 certificate from the signature file.

Parameters filename – Signature filename in APK

**Returns** DER coded X.509 certificate as binary

## get\_certificates()

Return a list of unique asn1crypto.x509.Certificate which are found in v1, v2 and v3 signing Note that we simply extract all certificates regardless of the signer. Therefore this is just a list of all certificates found in all signers.

### get\_certificates\_der\_v2()

Return a list of DER coded X.509 certificates from the v3 signature block

#### get\_certificates\_der\_v3()

Return a list of DER coded X.509 certificates from the v3 signature block

#### get certificates v1()

Return a list of asn1crypto.x509.Certificate which are found in the META-INF folder (v1 signing). Note that we simply extract all certificates regardless of the signer. Therefore this is just a list of all certificates found in all signers.

## get\_certificates\_v2()

Return a list of asn1crypto.x509.Certificate which are found in the v2 signing block. Note that we simply extract all certificates regardless of the signer. Therefore this is just a list of all certificates found in all signers.

#### get\_certificates\_v3()

Return a list of asn1crypto.x509.Certificate which are found in the v3 signing block. Note that we simply extract all certificates regardless of the signer. Therefore this is just a list of all certificates found in all signers.

## get\_declared\_permissions()

Returns list of the declared permissions.

**Return type** list of strings

### get\_declared\_permissions\_details()

Returns declared permissions with the details.

Return type dict

### get\_details\_permissions()

Return permissions with details.

THis can only return details about the permission, if the permission is defined in the AOSP.

**Return type** dict of {permission: [protectionLevel, label, description]}

### get\_dex()

Return the raw data of the classes dex file

This will give you the data of the file called *classes.dex* inside the APK. If the APK has multiple DEX files, you need to use  $qet\_all\_dex()$ .

Return type bytes

#### get\_dex\_names()

Return the names of all DEX files found in the APK. This method only accounts for "offical" dex files, i.e. all files in the root directory of the APK named classes.dex or classes[0-9]+.dex

Return type a list of str

### get\_effective\_target\_sdk\_version()

Return the effective targetSdkVersion, always returns int > 0.

If the targetSdkVersion is not set, it defaults to 1. This is set based on defaults as defined in: https://developer.android.com/guide/topics/manifest/uses-sdk-element.html

Return type int

### get\_element (tag\_name, attribute, \*\*attribute\_filter)

Deprecated since version 3.3.5: use get\_attribute\_value() instead

Return element in xml files which match with the tag name and the specific attribute

#### **Parameters**

- $tag_name(str)$  specify the tag name
- attribute (str) specify the attribute

#### Return type str

### get\_elements (tag\_name, attribute, with\_namespace=True)

Deprecated since version 3.3.5: use get\_all\_attribute\_value() instead

Return elements in xml files which match with the tag name and the specific attribute

#### **Parameters**

- $tag_name(str)$  a string which specify the tag name
- attribute (str) a string which specify the attribute

### get\_features()

Return a list of all android:names found for the tag uses-feature in the AndroidManifest.xml

#### Returns list

## get\_file (filename)

Return the raw data of the specified filename inside the APK

## Return type bytes

#### get\_filename()

Return the filename of the APK

### Return type str

### get\_files()

Return the file names inside the APK.

### **Return type** a list of str

## get\_files\_crc32()

Calculates and returns a dictionary of filenames and CRC32

**Returns** dict of filename: CRC32

# get\_files\_information()

Return the files inside the APK with their associated types and crc32

```
Return type str, str, int
```

#### get\_files\_types()

Return the files inside the APK with their associated types (by using python-magic)

At the same time, the CRC32 are calculated for the files.

### Return type a dictionnary

#### get intent filters(itemtype, name)

Find intent filters for a given item and name.

Intent filter are attached to activities, services or receivers. You can search for the intent filters of such items and get a dictionary of all attached actions and intent categories.

#### **Parameters**

- itemtype the type of parent item to look for, e.g. activity, service or receiver
- name the android:name of the parent item, e.g. activity name

**Returns** a dictionary with the keys *action* and *category* containing the *android:name* of those items

### get libraries()

Return the android:name attributes for libraries

#### **Return type** list

### get\_main\_activities()

Return names of the main activities

These values are read from the AndroidManifest.xml

**Return type** a set of str

#### get\_main\_activity()

Return the name of the main activity

This value is read from the AndroidManifest.xml

Return type str

#### get\_max\_sdk\_version()

Return the android:maxSdkVersion attribute

**Return type** string

#### get\_min\_sdk\_version()

Return the android:minSdkVersion attribute

**Return type** string

## get\_package()

Return the name of the package

This information is read from the AndroidManifest.xml

Return type str

### get\_permissions()

Return permissions names declared in the AndroidManifest.xml.

It is possible that permissions are returned multiple times, as this function does not filter the permissions, i.e. it shows you exactly what was defined in the AndroidManifest.xml.

Implied permissions, which are granted automatically, are not returned here. Use  $get\_uses\_implied\_permission\_list()$  if you need a list of implied permissions.

**Returns** A list of permissions

Return type list

## get\_providers()

Return the android:name attribute of all providers

Return type a list of string

#### get\_public\_keys\_der\_v2()

Return a list of DER coded X.509 public keys from the v3 signature block

## get\_public\_keys\_der\_v3()

Return a list of DER coded X.509 public keys from the v3 signature block

# get\_public\_keys\_v2()

Return a list of asn1crypto.keys.PublicKeyInfo which are found in the v2 signing block.

### get\_public\_keys\_v3()

Return a list of asn1crypto.keys.PublicKeyInfo which are found in the v3 signing block.

#### get\_raw()

Return raw bytes of the APK

Return type bytes

### get\_receivers()

Return the android:name attribute of all receivers

Return type a list of string

## get\_requested\_aosp\_permissions()

Returns requested permissions declared within AOSP project.

This includes several other permissions as well, which are in the platform apps.

Return type list of str

### get\_requested\_aosp\_permissions\_details()

Returns requested aosp permissions with details.

**Return type** dictionary

## get\_requested\_permissions()

Deprecated since version 3.1.0: use get\_permissions() instead.

Returns all requested permissions.

It has the same result as get\_permissions () and might be removed in the future

**Return type** list of str

## get\_requested\_third\_party\_permissions()

Returns list of requested permissions not declared within AOSP project.

**Return type** list of strings

#### get res value(name)

Return the literal value with a resource id

Return type str

## get\_services()

Return the android:name attribute of all services

Return type a list of str

### get\_signature()

Return the data of the first signature file found (v1 Signature / JAR Signature)

**Return type** First signature name or None if not signed

## get\_signature\_name()

Return the name of the first signature file found.

### get\_signature\_names()

Return a list of the signature file names (v1 Signature / JAR Signature)

**Return type** List of filenames matching a Signature

## get\_signatures()

Return a list of the data of the signature files. Only v1 / JAR Signing.

Return type list of bytes

#### get\_target\_sdk\_version()

Return the android:targetSdkVersion attribute

Return type string

### get uses implied permission list()

Return all permissions implied by the target SDK or other permissions.

## Return type list of string

#### get\_value\_from\_tag(tag, attribute)

Return the value of the android prefixed attribute in a specific tag.

This function will always try to get the attribute with a android: prefix first, and will try to return the attribute without the prefix, if the attribute could not be found. This is useful for some broken Android-Manifest.xml, where no android namespace is set, but could also indicate malicious activity (i.e. wrongly repackaged files). A warning is printed if the attribute is found without a namespace prefix.

If you require to get the exact result you need to query the tag directly:

## example::

### **Parameters**

- tag (lxml.etree.Element) specify the tag element
- **attribute** (*str*) specify the attribute name

**Returns** the attribute's value, or None if the attribute is not present

#### is androidtv()

Checks if this application does not require a touchscreen, as this is the rule to get into the TV section of the Play Store See: https://developer.android.com/training/tv/start/start.html for more information.

**Returns** True if 'android.hardware.touchscreen' is not required, False otherwise

## is\_leanback()

Checks if this application is build for TV (Leanback support) by checkin if it uses the feature 'android.software.leanback'

**Returns** True if leanback feature is used, false otherwise

## is\_multidex()

Test if the APK has multiple DEX files

Returns True if multiple dex found, otherwise False

#### is\_signed()

Returns true if either a v1 or v2 (or both) signature was found.

#### is\_signed\_v1()

Returns true if a v1 / JAR signature was found.

Returning *True* does not mean that the file is properly signed! It just says that there is a signature file which needs to be validated.

#### is\_signed\_v2()

Returns true of a v2 / APK signature was found.

Returning *True* does not mean that the file is properly signed! It just says that there is a signature file which needs to be validated.

#### is signed v3()

Returns true of a v3 / APK signature was found.

Returning *True* does not mean that the file is properly signed! It just says that there is a signature file which needs to be validated.

### is\_tag\_matched(tag, \*\*attribute\_filter)

Return true if the attributes matches in attribute filter.

An attribute filter is a dictionary containing: {attribute\_name: value}. This function will return True if and only if all attributes have the same value. This function allows to set the dictionary via kwargs, thus you can filter like this:

```
example:: a.is_tag_matched(tag, name="foobar", other="barfoo")
```

#### **Parameters**

- tag (lxml.etree.Element) specify the tag element
- attribute\_filter specify the attribute filter as dictionary

## is\_valid\_APK()

Return true if the APK is valid, false otherwise. An APK is seen as valid, if the AndroidManifest.xml could be successful parsed. This does not mean that the APK has a valid signature nor that the APK can be installed on an Android system.

### Return type boolean

#### is\_wearable()

Checks if this application is build for wearables by checking if it uses the feature 'android.hardware.type.watch' See: https://developer.android.com/training/wearables/apps/creating.html for more information.

Not every app is setting this feature (not even the example Google provides), so it might be wise to not 100% rely on this feature.

**Returns** True if wearable, False otherwise

```
new_zip (filename, deleted_files=None, new_files={})
```

Create a new zip file

#### **Parameters**

- filename (string) the output filename of the zip
- deleted\_files (None or a string) a regex pattern to remove specific file
- new\_files (a dictionnary (key:filename, value:content of the file)) a dictionnary of new files

## parse\_signatures\_or\_digests (digest\_bytes)

Parse digests

```
parse v2 signing block()
         Parse the V2 signing block and extract all features
     parse_v2_v3_signature()
     parse_v3_signing_block()
         Parse the V2 signing block and extract all features
     read_uint32_le (io_stream)
     show()
class androguard.core.bytecodes.apk.APKV2SignedData
     Bases: object
     This class holds all data associated with an APK V3 SigningBlock signed data. source: https://source.android.
     com/security/apksigning/v2.html
class androguard.core.bytecodes.apk.APKV2Signer
     Bases: object
     This class holds all data associated with an APK V2 SigningBlock signer, source: https://source.android.com/
     security/apksigning/v2.html
class androguard.core.bytecodes.apk.APKV3SignedData
     Bases: androguard.core.bytecodes.apk.APKV2SignedData
     This class holds all data associated with an APK V3 SigningBlock signed data. source: https://source.android.
     com/security/apksigning/v3.html
class androquard.core.bytecodes.apk.APKV3Signer
     Bases: androguard.core.bytecodes.apk.APKV2Signer
     This class holds all data associated with an APK V3 SigningBlock signer. source: https://source.android.com/
     security/apksigning/v3.html
exception androguard.core.bytecodes.apk.BrokenAPKError
     Bases: androguard.core.bytecodes.apk.Error
exception androguard.core.bytecodes.apk.Error
     Bases: Exception
     Base class for exceptions in this module.
exception androquard.core.bytecodes.apk.FileNotPresent
     Bases: androguard.core.bytecodes.apk.Error
androguard.core.bytecodes.apk.ensure_final_value(packageName, arsc, value)
     Ensure incoming value is always the value, not the resid
     androguard will sometimes return the Android "resId" aka Resource ID instead of the actual value. This checks
     whether the value is actually a resId, then performs the Android Resource lookup as needed.
     Read (appid, versionCode, versionName) from an APK
```

androquard.core.bytecodes.apk.get\_apkid(apkfile)

This first tries to do quick binary XML parsing to just get the values that are needed. It will fallback to full androguard parsing, which is slow, if it can't find the versionName value or versionName is set to a Android String Resource (e.g. an integer hex value that starts with @).

```
androguard.core.bytecodes.apk.parse_lxml_dom(tree)
androquard.core.bytecodes.apk.show Certificate(cert, short=False)
    Print Fingerprints, Issuer and Subject of an X509 Certificate.
```

#### **Parameters**

```
• cert (asn1crypto.x509.Certificate) - X509 Certificate to print
```

```
• short (Boolean) – Print in shortform for DN (Default: False)
```

## androguard.core.bytecodes.dvm module

```
class androguard.core.bytecodes.dvm.AnnotationElement (buff, cm)
     Bases: object
     This class can parse an annotation_element of a dex file
          Parameters
                • buff (Buff object) – a string which represents a Buff object of the annotation_element
                • cm (ClassManager) – a ClassManager object
     get_length()
     get_name_idx()
          Return the element name, represented as an index into the string_ids section
              Return type int
     get_obj()
     get_raw()
     get_value()
          Return the element value (EncodedValue)
              Return type a EncodedValue object
     show()
class androquard.core.bytecodes.dvm.AnnotationItem (buff, cm)
     Bases: object
     This class can parse an annotation_item of a dex file
          Parameters
               • buff (Buff object) – a string which represents a Buff object of the annotation_item
                • cm (ClassManager) – a ClassManager object
     get_annotation()
          Return the encoded annotation contents
              Return type a EncodedAnnotation object
     get_length()
     get_obj()
     get_off()
     get_raw()
     get_visibility()
          Return the intended visibility of this annotation
              Return type int
```

set off(off)

```
show()
class androguard.core.bytecodes.dvm.AnnotationOffItem (buff, cm)
     Bases: object
     This class can parse an annotation_off_item of a dex file
          Parameters
               • buff (Buff object) - a string which represents a Buff object of the annota-
                 tion off item
               • cm (ClassManager) - a ClassManager object
     get_annotation_item()
     get_annotation_off()
     get_length()
     get_obj()
     get_raw()
     show()
class androguard.core.bytecodes.dvm.AnnotationSetItem(buff, cm)
     Bases: object
     This class can parse an annotation set item of a dex file
          Parameters
               • buff – a string which represents a Buff object of the annotation_set_item
               • cm (ClassManager) - a ClassManager object
     get annotation off item()
          Return the offset from the start of the file to an annotation
             Return type a list of AnnotationOffItem
     get_length()
     get_obj()
     get_off()
     get_raw()
     set\_off(off)
     show()
class androquard.core.bytecodes.dvm.AnnotationSetRefItem(buff, cm)
     Bases: object
     This class can parse an annotation_set_ref_item of a dex file
          Parameters
               • buff (Buff object) - a string which represents a Buff object of the annota-
                 tion_set_ref_item
               • cm (ClassManager) - a ClassManager object
     get_annotations_off()
          Return the offset from the start of the file to the referenced annotation set or 0 if there are no annotations
          for this element.
```

```
Return type int
     get_obj()
     get_raw()
     show()
class androquard.core.bytecodes.dvm.AnnotationSetRefList(buff, cm)
     Bases: object
     This class can parse an annotation_set_ref_list_item of a dex file
         Parameters
               • buff (Buff object) - a string which represents a Buff object of the annota-
                 tion_set_ref_list_item
               • cm (ClassManager) - a ClassManager object
     get_length()
     get list()
         Return elements of the list
             Return type AnnotationSetRefItem
     get_obj()
     get off()
     get_raw()
     set_off(off)
     show()
class androquard.core.bytecodes.dvm.AnnotationsDirectoryItem(buff, cm)
     Bases: object
     This class can parse an annotations_directory_item of a dex file
         Parameters
               • buff (Buff object) - a string which represents a Buff object of the annota-
                 tions_directory_item
               • cm (ClassManager) – a ClassManager object
     get_annotated_fields_size()
         Return the count of fields annotated by this item
             Return type int
     get_annotated_methods_size()
         Return the count of methods annotated by this item
             Return type int
     get_annotated_parameters_size()
         Return the count of method parameter lists annotated by this item
             Return type int
     get_annotation_set_item()
```

```
get class annotations off()
          Return the offset from the start of the file to the annotations made directly on the class, or 0 if the class has
          no direct annotations
              Return type int
     get field annotations()
          Return the list of associated field annotations
              Return type a list of FieldAnnotation
     get_length()
     get_method_annotations()
          Return the list of associated method annotations
              Return type a list of MethodAnnotation
     get_obj()
     get_off()
     get_parameter_annotations()
          Return the list of associated method parameter annotations
              Return type a list of ParameterAnnotation
     get_raw()
     set off(off)
     show()
class androguard.core.bytecodes.dvm.ClassDataItem(buff, cm)
     Bases: object
     This class can parse a class_data_item of a dex file
          Parameters
                • buff (Buff object) – a string which represents a Buff object of the class_data_item
                • cm (ClassManager) – a ClassManager object
     get direct methods()
          Return the defined direct (any of static, private, or constructor) methods, represented as a sequence of
          encoded elements
              Return type a list of EncodedMethod objects
     get direct methods size()
          Return the number of direct methods defined in this item
              Return type int
     get_fields()
          Return static and instance fields
              Return type a list of EncodedField objects
     get_instance_fields()
          Return the defined instance fields, represented as a sequence of encoded elements
              Return type a list of EncodedField objects
     get instance fields size()
          Return the number of instance fields defined in this item
```

```
Return type int
     get_length()
     get_methods()
          Return direct and virtual methods
              Return type a list of EncodedMethod objects
     get_obj()
     get_off()
     get_raw()
     get_static_fields()
          Return the defined static fields, represented as a sequence of encoded elements
              Return type a list of EncodedField objects
     get_static_fields_size()
          Return the number of static fields defined in this item
              Return type int
     get_virtual_methods()
          Return the defined virtual (none of static, private, or constructor) methods, represented as a sequence of
          encoded elements
              Return type a list of EncodedMethod objects
     get virtual methods size()
          Return the number of virtual methods defined in this item
              Return type int
     set_off(off)
     set_static_fields(value)
     show()
class androguard.core.bytecodes.dvm.ClassDefItem(buff, cm)
     Bases: object
     This class can parse a class_def_item of a dex file
          Parameters
                • buff (Buff object) – a string which represents a Buff object of the class_def_item
                • cm (ClassManager) – a ClassManager object
     get access flags()
          Return the access flags for the class (public, final, etc.)
              Return type int
     get_access_flags_string()
          Return the access flags string of the class
              Return type str
     get_annotations()
     get_annotations_off()
          Return the offset from the start of the file to the annotations structure for this class, or 0 if there are no
```

annotations on this class.

```
Return type int
get_ast()
get_class_data()
     Return the associated class_data_item
         Return type a ClassDataItem object
get class data off()
     Return the offset from the start of the file to the associated class data for this item, or 0 if there is no class
     data for this class
         Return type int
get_class_idx()
     Return the index into the type_ids list for this class
         Return type int
get_fields()
     Return all fields of this class
         Return type a list of EncodedField objects
get_interfaces()
     Return the names of the interfaces
         Return type List[MUTF8String]
get_interfaces_off()
     Return the offset from the start of the file to the list of interfaces, or 0 if there are none
         Return type int
get_length()
get_methods()
     Return all methods of this class
         Return type a list of EncodedMethod objects
get_name()
     Return the name of this class
         Return type MUTF8String
get_obj()
get_raw()
get_source()
get_source_ext()
get_source_file_idx()
     Return the index into the string_ids list for the name of the file containing the original source for (at least
     most of) this class, or the special value NO_INDEX to represent a lack of this information
         Return type int
get_static_values_off()
     Return the offset from the start of the file to the list of initial values for static fields, or 0 if there are none
     (and all static fields are to be initialized with 0 or null)
```

Return type int

```
get_superclass_idx()
          Return the index into the type_ids list for the superclass
             Return type int
     get_superclassname()
         Return the name of the super class
             Return type MUTF8String
     reload()
     set_name (value)
     show()
     source()
         Return the source code of the entire class
             Return type string
class androquard.core.bytecodes.dvm.ClassHDefItem(size, buff, cm)
     Bases: object
     This class can parse a list of class_def_item of a dex file
          Parameters
               • buff (Buff object) - a string which represents a Buff object of the list of
                 class def item
               • cm (ClassManager) – a ClassManager object
     get_class_idx(idx)
     get_length()
     get_method (name_class, name_method)
     get_names()
     get_obj()
     get_off()
     get_raw()
     set_off(off)
     show()
class androquard.core.bytecodes.dvm.ClassManager(vm)
     Bases: object
     This class is used to access to all elements (strings, type, proto ...) of the dex format based on their offset or
     index.
     add_type_item (type_item, c_item, item)
     get_all_engine()
         Deprecated since version 3.3.5: do not use this function anymore!
     get_annotation_item(off)
     get_annotation_off_item(off)
     get_annotation_set_item(off)
     get_annotations_directory_item(off)
```

```
get_ascii_string(s)
get_class_data_item(off)
get\_code(idx)
get_debug_off(off)
get_encoded_array_item(off)
get engine()
    Deprecated since version 3.3.5: do not use this function anymore!
get_field(idx)
get_field_ref(idx)
get_item_by_offset (offset)
get_lazy_analysis()
     Deprecated since version 3.3.5: do not use this function anymore!
get method(idx)
get_method_ref (idx)
get_next_offset_item(idx)
get_obj_by_offset (offset)
     Returnes a object from as given offset inside the DEX file
get_odex_format()
     Returns True if the underlying VM is ODEX
get_proto(idx)
get_raw_string(idx)
     Return the (unprocessed) string from the string table at index idx.
         Parameters idx(int) – the index in the string section
get_string(idx)
     Return a string from the string table at index idx
     If string is hooked, the hooked string is returned.
         Parameters idx(int) – index in the string section
get_string_by_offset (offset)
get_type (idx)
    Return the resolved type name based on the index
     This returns the string associated with the type.
         Parameters idx (int) -
         Returns the type name
         Return type str
get_type_list(off)
get_type_ref (idx)
    Returns the string reference ID for a given type ID.
     This method is similar to get type () but does not resolve the string but returns the ID into the string
```

section.

```
If the type IDX is not found, -1 is returned.
     property packer
     set_decompiler(decompiler)
     set_hook_class_name (class_def, value)
     set_hook_field_name (encoded_field, value)
     set_hook_method_name(encoded_method, value)
     set_hook_string(idx, value)
class androquard.core.bytecodes.dvm.CodeItem(size, buff, cm)
     Bases: object
     get_code (off)
     get_length()
     get_obj()
     get_off()
     get_raw()
     set_off(off)
     show()
class androguard.core.bytecodes.dvm.DBGBytecode(cm, op_value)
     Bases: object
     add(value, ttype)
     get_obj()
     get_op_value()
     get_raw()
     get_value()
     show()
class androquard.core.bytecodes.dvm.DCode(class_manager, offset, size, buff)
     Bases: object
     This class represents the instructions of a method
          Parameters
               • class_manager (ClassManager object) - the ClassManager
               • offset (int) – the offset of the buffer
               • size (int) – the total size of the buffer
               • buff (string) – a raw buffer where are the instructions
     add_inote (msg, idx, off=None)
          Add a message to a specific instruction by using (default) the index of the address if specified
             Parameters
                 • msg(string) - the message
```

• idx (int) – index of the instruction (the position in the list of the instruction)

# 3.1. androguard package

```
• off (int) – address of the instruction
get_ins_off(off)
     Get a particular instruction by using the address
         Parameters off (int) – address of the instruction
         Return type an Instruction object
get_insn()
     Get the insn buffer
         Return type bytes
get_instruction (idx, off=None)
     Get a particular instruction by using (default) the index of the address if specified
         Parameters
             • idx (int) – index of the instruction (the position in the list of the instruction)
             • off (int) – address of the instruction
         Return type an Instruction object
get_instructions()
     Get the instructions
         Return type a generator of each Instruction (or a cached list of instructions if you have
             setup instructions)
get length()
     Return the length of this object
         Return type int
get raw()
     Return the raw buffer of this object
         Return type bytearray
is_cached_instructions()
off_to_pos(off)
     Get the position of an instruction by using the address
         Parameters off (int) – address of the instruction
         Return type int
set_idx(idx)
     Set the start address of the buffer
         Parameters idx (int) - the index
set_insn(insn)
     Set a new raw buffer to disassemble
         Parameters insn (bytes) – the buffer
set instructions (instructions)
     Set the instructions
         Parameters instructions (a list of Instruction) – the list of instructions
show()
     Display (with a pretty print) this object
```

```
class androguard.core.bytecodes.dvm.DalvikCode(buff, cm)
     Bases: object
     This class represents the instructions of a method
          Parameters
                • buff (BuffHandle) – a raw buffer where are the instructions
                • cm (ClassManager object) - the ClassManager
     add_inote (msg, idx, off=None)
          Add a message to a specific instruction by using (default) the index of the address if specified
               Parameters
                   • msg (string) - the message
                   • idx (int) – index of the instruction (the position in the list of the instruction)
                   • off (int) – address of the instruction
     get bc()
          Return the associated code object
               Return type DCode
     get debug()
          Return the associated debug object
               Return type DebugInfoItem
     get_debug_info_off()
          Get the offset from the start of the file to the debug info (line numbers + local variable info) sequence for
          this code, or 0 if there simply is no information
               Return type int
     get_handlers()
          Get the bytes representing a list of lists of catch types and associated handler addresses.
               Return type EncodedCatchHandlerList
     get ins size()
          Get the number of words of incoming arguments to the method that this code is for
               Return type int
     get insns size()
          Get the size of the instructions list, in 16-bit code units
               Return type int
     get_instruction (idx, off=None)
     get_length()
     get_obj()
     get_off()
     get_outs_size()
          Get the number of words of outgoing argument space required by this code for method invocation
```

get\_raw()

Return type int

Get the reconstructed code as bytearray

```
Return type bytearray
     get_registers_size()
          Get the number of registers used by this code
              Return type int
     get_size()
     get_tries()
          Get the array indicating where in the code exceptions are caught and how to handle them
              Return type a list of TryItem objects
     get_tries_size()
          Get the number of TryItem for this instance
              Return type int
     set_idx(idx)
     set_off(off)
     show()
class androguard.core.bytecodes.dvm.DalvikOdexVMFormat(buff,
                                                                               decompiler=None,
                                                                      config=None,
                                                                                            us-
                                                                      ing_api=None)
     Bases: androquard.core.bytecodes.dvm.DalvikVMFormat
     This class can parse an odex file
          Parameters
               • buff (string) – a string which represents the odex file
               • decompiler (object) – associate a decompiler object to display the java source code
          Example DalvikOdexVMFormat( read("classes.odex") )
     get buff()
          Return the whole buffer
              Return type bytearray
     get_dependencies()
          Return the odex dependencies object
              Return type an OdexDependencies object
     get_format_type()
          Return the type
              Return type a string
     save()
          Do not use!
class androguard.core.bytecodes.dvm.DalvikPacker(endian_tag)
     Bases: object
     Generic Packer class to unpack bytes based on different endianness
class androguard.core.bytecodes.dvm.DalvikVMFormat (buff,
                                                                        decompiler=None,
                                                                 fig=None, using_api=None)
     Bases: androguard.core.bytecode.BuffHandle
     This class can parse a classes.dex file of an Android application (APK).
```

#### **Parameters**

- buff (bytes) a string which represents the classes.dex file
- **decompiler** (object) associate a decompiler object to display the java source code

example:

```
d = DalvikVMFormat( read("classes.dex") )
create_python_export()
     Export classes/methods/fields' names in the python namespace
disassemble (offset, size)
     Disassembles a given offset in the DEX file
         Parameters
             • offset (int) – offset to disassemble in the file (from the beginning of the file)
             • size -
fix_checksums(buff)
     Fix a dex format buffer by setting all checksums
         Return type string
get_BRANCH_DVM_OPCODES()
     Deprecated since version 3.4.0: Will be removed!
get_all_fields()
     Return a list of field items
         Return type a list of FieldIdItem objects
get_api_version()
     This method returns api version that should be used for loading api specific resources.
         Return type int
get class(name)
    Return a specific class
         Parameters name – the name of the class
         Return type a ClassDefItem object
get class manager()
     This function returns a ClassManager object which allow you to get access to all index references (strings,
     methods, fields, ....)
         Return type ClassManager object
get_classes()
     Return all classes
         Return type a list of ClassDefItem objects
get_classes_def_item()
     This function returns the class def item
         Return type ClassHDefItem object
get_classes_names (update=False)
```

Return the names of classes

```
Parameters update – True indicates to recompute the list. Maybe needed after using a My-
            Class.set_name().
         Return type a list of string
get\_cm\_field(idx)
     Get a specific field by using an index
         Parameters idx(int) – index of the field
get\_cm\_method(idx)
     Get a specific method by using an index
         Parameters idx (int) - index of the method
get_cm_string(idx)
     Get a specific string by using an index
         Parameters idx(int) – index of the string
get\_cm\_type(idx)
     Get a specific type by using an index
         Parameters idx(int) – index of the type
get_codes_item()
     This function returns the code item
         Return type Code I tem object
get_debug_info_item()
     This function returns the debug info item
         Return type DebugInfoItem object
get_determineException()
     Deprecated since version 3.4.0: Will be removed!
get_determineNext()
    Deprecated since version 3.4.0: Will be removed!
get_field(name)
     Return a list all fields which corresponds to the regexp
         Parameters name – the name of the field (a python regexp)
         Return type a list with all EncodedField objects
get_field_descriptor (class_name, field_name, descriptor)
    Return the specific field
         Parameters
             • class_name (string) - the class name of the field
             • field_name (string) – the name of the field
             • descriptor (string) – the descriptor of the field
         Return type None or a EncodedField object
get_fields()
     Return all field objects
         Return type a list of EncodedField objects
```

```
get_fields_class (class_name)
    Return all fields of a specific class
        Parameters class_name (string) – the class name
        Return type a list with EncodedField objects
get fields id item()
    This function returns the field id item
        Return type FieldHIdItem object
get_format()
    Deprecated since version 3.4.0: Will be removed!
get_format_type()
    Return the type
        Return type a string
get_header_item()
    This function returns the header item
        Return type HeaderItem object
get_len_methods()
    Return the number of methods
        Return type int
get_method(name)
    Return a list all methods which corresponds to the regexp
        Parameters name – the name of the method (a python regexp)
        Return type a list with all EncodedMethod objects
get_method_by_idx (idx)
    Return a specific method by using an index :param idx: the index of the method :type idx: int
         Return type None or an EncodedMethod object
get_method_descriptor (class_name, method_name, descriptor)
    Return the specific method
        Parameters
             • class_name (string) - the class name of the method
             • method_name (string) - the name of the method
             • descriptor (string) – the descriptor of the method
        Return type None or a EncodedMethod object
get_methods()
    Return all method objects
        Return type a list of EncodedMethod objects
get_methods_class(class_name)
    Return all methods of a specific class
        Parameters class_name (string) - the class name
```

**Return type** a list with *EncodedMethod* objects

#### get\_methods\_descriptor(class\_name, method\_name)

Return the specific methods of the class

#### **Parameters**

- class\_name (string) the class name of the method
- **method\_name** (*string*) the name of the method

**Return type** None or a *EncodedMethod* object

### get\_methods\_id\_item()

This function returns the method id item

Return type MethodHIdItem object

## get\_regex\_strings (regular\_expressions)

Return all target strings matched the regex

**Parameters** regular\_expressions (string) – the python regex

**Return type** a list of strings matching the regex expression

#### get\_string\_data\_item()

This function returns the string data item

Return type StringDataItem object

### get\_strings()

Return all strings

The strings will have escaped surrogates, if only a single high or low surrogate is found. Complete surrogates are put together into the representing 32bit character.

**Return type** a list with all strings used in the format (types, names ...)

## get\_vmanalysis()

Deprecated since version 3.1.0: The *Analysis* is not loaded anymore into *DalvikVMFormat* in order to avoid cyclic dependencies. *Analysis* extends now *DalvikVMFormat*. This Method does nothing anymore!

The Analysis Object should contain all the information required, inclduing the DalvikVMFormats.

## list\_classes\_hierarchy()

Get a tree structure of the classes. The parent is always the superclass.

You can use pprint.pprint to print the dictionary in a pretty way.

**Returns** a dict with all the classnames

Return type dict

#### print\_classes\_hierarchy()

Deprecated since version 3.4.0: Will be removed!

#### save (

Return the dex (with the modifications) into raw format (fix checksums) (beta: do not use !)

Return type string

```
set_decompiler (decompiler)
```

### set\_vmanalysis (analysis)

Deprecated since version 3.1.0: The *Analysis* is not loaded anymore into *DalvikVMFormat* in order to avoid cyclic dependencies. *Analysis* extends now *DalvikVMFormat*. This Method does nothing anymore!

```
The Analysis Object should contain all the information required, inclduing the DalvikVMFormats.
     show()
         Show the all information in the object
     property version
         Returns the version number of the DEX Format
class androguard.core.bytecodes.dvm.DebugInfoItem(buff, cm)
     Bases: object
     get_bytecodes()
     get_line_start()
     get_off()
     get_parameter_names()
     get_parameters_size()
     get_raw()
     get_translated_parameter_names()
class androquard.core.bytecodes.dvm.DebugInfoItemEmpty (buff, cm)
     Bases: object
     get_length()
     get_obj()
     get_off()
     get_raw()
     reload()
     set\_off(off)
     show()
class androquard.core.bytecodes.dvm.EncodedAnnotation (buff, cm)
     Bases: object
     This class can parse an encoded_annotation of a dex file
         Parameters
               • buff (Buff object) – a string which represents a Buff object of the encoded annotation
               • cm (ClassManager) – a ClassManager object
     get_elements()
         Return the elements of the annotation, represented directly in-line (not as offsets)
             Return type a list of AnnotationElement objects
     get_length()
     get_obj()
     get_raw()
```

```
get size()
          Return the number of name-value mappings in this annotation
          :rtype:int
     get_type_idx()
          Return the type of the annotation. This must be a class (not array or primitive) type
              Return type int
     show()
class androguard.core.bytecodes.dvm.EncodedArray(buff, cm)
     Bases: object
     This class can parse an encoded_array of a dex file
          Parameters
                • buff (Buff object) – a string which represents a Buff object of the encoded_array
                • cm (ClassManager) – a ClassManager object
     get_length()
     get_obj()
     get_raw()
     get size()
          Return the number of elements in the array
              Return type int
     get_values()
          Return a series of size encoded_value byte sequences in the format specified by this section, concatenated
          sequentially
              Return type a list of EncodedValue objects
     show()
class androguard.core.bytecodes.dvm.EncodedArrayItem(buff, cm)
     Bases: object
     This class can parse an encoded_array_item of a dex file
          Parameters
                • buff (Buff object) - a string which represents a Buff object of the en-
                 coded array item
                • cm (ClassManager) – a ClassManager object
     get_length()
     get_obj()
     get_off()
     get_raw()
     get_value()
          Return the bytes representing the encoded array value
              Return type a EncodedArray object
     set_off(off)
```

```
show()
class androquard.core.bytecodes.dvm.EncodedCatchHandler(buff, cm)
     Bases: object
     This class can parse an encoded_catch_handler of a dex file
          Parameters
                • buff (Buff object) - a string which represents a Buff object of the en-
                 coded catch handler
                • cm (ClassManager) - a ClassManager object
     get_catch_all_addr()
          Return the bytecode address of the catch-all handler. This element is only present if size is non-positive.
              Return type int
     get_handlers()
          Return the stream of abs(size) encoded items, one for each caught type, in the order that the types should
          be tested.
              Return type a list of EncodedTypeAddrPair objects
     get_length()
     get_off()
     get_raw()
              Return type bytearray
     get_size()
          Return the number of catch types in this list
              Return type int
     set_off(off)
     show()
class androguard.core.bytecodes.dvm.EncodedCatchHandlerList(buff, cm)
     Bases: object
     This class can parse an encoded_catch_handler_list of a dex file
          Parameters
                • buff (Buff object) - a string which represents a Buff object of the en-
                 coded catch handler list
                • cm (ClassManager) – a ClassManager object
     get_length()
     get_list()
          Return the actual list of handler lists, represented directly (not as offsets), and concatenated sequentially
              Return type a list of EncodedCatchHandler objects
     get_obj()
     get_off()
     get_raw()
              Return type bytearray
```

```
get size()
          Return the size of this list, in entries
              Return type int
     set_off(off)
     show()
class androguard.core.bytecodes.dvm.EncodedField(buff, cm)
     Bases: object
     This class can parse an encoded_field of a dex file
          Parameters
                • buff (Buff object) – a string which represents a Buff object of the encoded field
                • cm (ClassManager) - a ClassManager object
     adjust_idx(val)
     get_access_flags()
          Return the access flags of the field
              Return type int
     get_access_flags_string()
          Return the access flags string of the field
              Return type string
     get_class_name()
          Return the class name of the field
              Return type string
     get_descriptor()
          Return the descriptor of the field
          The descriptor of a field is the type of the field.
              Return type string
     get field idx()
          Return the real index of the method
              Return type int
     get_field_idx_diff()
          Return the index into the field ids list for the identity of this field (includes the name and descriptor),
          represented as a difference from the index of previous element in the list
              Return type int
     get_init_value()
          Return the init value object of the field
              Return type EncodedValue
     get name()
          Return the name of the field
              Return type string
     get_obj()
     get_raw()
```

```
get_size()
     load()
     reload()
     set_init_value(value)
          Setup the init value object of the field
              Parameters value (EncodedValue) – the init value
     set_name (value)
     show()
          Display the information (with a pretty print) about the field
class androquard.core.bytecodes.dvm.EncodedMethod(buff, cm)
     Bases: object
     This class can parse an encoded_method of a dex file
          Parameters
                • buff (Buff object) – a string which represents a Buff object of the encoded_method
                • cm (ClassManager) – a ClassManager object
     access flags = None
          access flags of the method
     add_inote (msg, idx, off=None)
          Add a message to a specific instruction by using (default) the index of the address if specified
              Parameters
                  • msg (string) - the message
                  • idx (int) – index of the instruction (the position in the list of the instruction)
                  • off (int) – address of the instruction
     add_note (msg)
          Add a message to this method
              Parameters msg (string) – the message
     adjust_idx (val)
     code off = None
          offset of the code section
     property descriptor
          Get the descriptor of the method
     each_params_by_register (nb, proto)
          From the Dalvik Bytecode documentation:
```

> The N arguments to a method land in the last N registers > of the method's invocation frame, in order.

> Wide arguments consume two registers. > Instance methods are passed a this reference as their first argument.

This method will print a description of the register usage to stdout.

#### **Parameters**

- **nb** number of registers
- proto descriptor of method

#### property full\_name

Return class\_name + name + descriptor, separated by spaces (no access flags

### get\_access\_flags()

Return the access flags of the method

#### **Return type** int

### get\_access\_flags\_string()

Return the access flags string of the method

A description of all access flags can be found here: https://source.android.com/devices/tech/dalvik/dex-format#access-flags

#### Return type string

#### get\_address()

Return the offset from the start of the file to the code structure for this method, or 0 if this method is either abstract or native

### Return type int

#### get\_class\_name()

Return the class name of the method

### **Return type** string

#### get code()

Return the code object associated to the method

**Return type** *DalvikCode* object or None if no Code

#### get\_code\_off()

Return the offset from the start of the file to the code structure for this method, or 0 if this method is either abstract or native

## Return type int

# get\_debug()

Return the debug object associated to this method

#### Return type DebugInfoItem

### get\_descriptor()

Return the descriptor of the method A method descriptor will have the form (A A A ...)R Where A are the arguments to the method and R is the return type. Basic types will have the short form, i.e. I for integer, V for void and class types will be named like a classname, e.g. Ljava/lang/String;.

```
Typical descriptors will look like this: `(I)I // one integer argument, integer return (C)Z // one char argument, boolean as return (Ljava/lang/CharSequence; I)I // CharSequence and integer as argyument, integer as return (C)Ljava/lang/String; // char as argument, String as return.
```

More information about type descriptors are found here: https://source.android.com/devices/tech/dalvik/dex-format#typedescriptor

#### **Return type** string

#### get\_information()

Get brief information about the method's register use, parameters and return type.

The resulting dictionary has the form:

```
registers: (start, end),
params: [(reg_1, type_1), (reg_2, type_2), ..., (reg_n, type_n)],
return: type
)
```

The end register is not the last register used, but the last register used not for parameters. Hence, they represent the local registers. The start register is always zero. The register numbers for the parameters can be found in the tuples for each parameter.

**Returns** a dictionary with the basic information about the method

Return type dict

## get\_instruction(idx, off=None)

Get a particular instruction by using (default) the index of the address if specified

### **Parameters**

- idx (int) index of the instruction (the position in the list of the instruction)
- off (int) address of the instruction

Return type an Instruction object

## get\_instructions()

Get the instructions

**Return type** a generator of each *Instruction* (or a cached list of instructions if you have setup instructions)

### get\_instructions\_idx()

Iterate over all instructions of the method, but also return the current index. This is the same as using <code>get\_instructions()</code> and adding the instruction length to a variable each time.

#### **Returns**

**Return type** Iterator[(int, *Instruction*)]

## get\_length()

Return the length of the associated code of the method

Return type int

#### get\_locals()

Get the number of local registers used by the method

This number is equal to the number of registers minus the number of parameters minus 1.

**Returns** number of local registers

Return type int

#### get method idx()

Return the real index of the method

**Return type** int

## get\_method\_idx\_diff()

Return index into the method\_ids list for the identity of this method (includes the name and descriptor), represented as a difference from the index of previous element in the lis

Return type int

```
get name()
          Return the name of the method
              Return type string
     get_raw()
     get short string()
          Return a shorter formatted String which encodes this method. The returned name has the form: <class-
          name> <methodname> ([arguments . . . ]) < returntype>
            • All Class names are condensed to the actual name (no package).
            · Access flags are not returned.
            • <init> and <clinit> are NOT replaced by the classname!
          This name might not be unique!
              Returns str
     get_size()
     get_source()
     get_triple()
     is_cached_instructions()
     load()
     method_idx_diff = None
          method index diff in the corresponding section
     reload()
     set\_code\_idx(idx)
          Set the start address of the buffer to disassemble
              Parameters idx (int) – the index
     set_instructions (instructions)
          Set the instructions
              Parameters instructions (a list of Instruction) – the list of instructions
     set_name (value)
     show()
          Display the information (with a pretty print) about the method
     show info()
          Display the basic information about the method
     show_notes()
          Display the notes about the method
     source()
          Return the source code of this method
              Return type string
class androguard.core.bytecodes.dvm.EncodedTypeAddrPair(cm, buff)
     Bases: object
     This class can parse an encoded_type_addr_pair of a dex file
```

**Parameters** 

```
• buff (Buff object) - a string which represents a Buff object of the en-
                 coded_type_addr_pair
               • cm (ClassManager) – a ClassManager object
     get addr()
          Return the bytecode address of the associated exception handler
              Return type int
     get_length()
     get_obj()
     get_raw()
     get_type_idx()
          Return the index into the type_ids list for the type of the exception to catch
              Return type int
     show()
class androguard.core.bytecodes.dvm.EncodedValue(buff, cm)
     Bases: object
     This class can parse an encoded_value of a dex file
          Parameters
               • buff (Buff object) – a string which represents a Buff object of the encoded_value
                • cm (ClassManager) – a ClassManager object
     get_length()
     get_obj()
     get_raw()
     get_value()
          Return the bytes representing the value, variable in length and interpreted differently for different
          value_type bytes, though always little-endian
              Return type an object representing the value
     get_value_arg()
     get_value_type()
     show()
class androguard.core.bytecodes.dvm.ExportObject
     Bases: object
     Wrapper object for ipython exports
class androguard.core.bytecodes.dvm.FieldAnnotation(buff, cm)
     Bases: object
     This class can parse a field_annotation of a dex file
          Parameters
               • buff (Buff object) - a string which represents a Buff object of the field_annotation
                • cm (ClassManager) – a ClassManager object
```

```
get annotations off()
          Return the offset from the start of the file to the list of annotations for the field
              Return type int
     get field idx()
          Return the index into the field_ids list for the identity of the field being annotated
              Return type int
     get_length()
     get_obj()
     get_off()
     get_raw()
     set\_off(off)
     show()
class androquard.core.bytecodes.dvm.FieldHIdItem(size, buff, cm)
     Bases: object
     This class can parse a list of field_id_item of a dex file
          Parameters
                • buff (Buff object) – a string which represents a Buff object of the list of field_id_item
                • cm (ClassManager) – a ClassManager object
     get (idx)
     get_length()
     get_obj()
     get_off()
     get_raw()
     gets()
     set off(off)
     show()
class androguard.core.bytecodes.dvm.FieldIdItem(buff, cm)
     Bases: object
     This class can parse a field id item of a dex file
          Parameters
                • buff (Buff object) – a string which represents a Buff object of the field_id_item
                • cm (ClassManager) – a ClassManager object
     get_class_idx()
          Return the index into the type_ids list for the definer of this field
              Return type int
     get_class_name()
          Return the class name of the field
              Return type string
```

```
get_descriptor()
          Return the descriptor of the field
              Return type string
     get_length()
     get_list()
     get_name()
          Return the name of the field
              Return type string
     get_name_idx()
          Return the index into the string_ids list for the name of this field
              Return type int
     get_obj()
     get_raw()
     get_type()
          Return the type of the field
              Return type string
     get_type_idx()
          Return the index into the type_ids list for the type of this field
              Return type int
     reload()
     show()
class androguard.core.bytecodes.dvm.FieldIdItemInvalid
     Bases: object
     get_class_name()
     get_descriptor()
     get_list()
     get_name()
     get_type()
     show()
class androguard.core.bytecodes.dvm.FillArrayData(cm, buff)
     Bases: object
     This class can parse a FillArrayData instruction
          Parameters buff – a Buff object which represents a buffer where the instruction is stored
     add_note (msg)
          Add a note to this instruction
              Parameters msg(objects (string)) - the message
     disasm()
     get data()
          Return the data of this instruction (the payload)
```

```
Return type bytes
     get_formatted_operands()
     get_hex()
          Returns a HEX String, separated by spaces every byte
     get length()
          Return the length of the instruction
              Return type int
     get_name()
          Return the name of the instruction
              Return type string
     get_notes()
          Get all notes from this instruction
              Return type a list of objects
     get_op_value()
          Get the value of the opcode
              Return type int
     get_operands (idx=-1)
     get output (idx=-1)
          Return an additional output of the instruction
              Return type string
     get_raw()
     show (pos)
          Print the instruction
     show_buff(pos)
          Return the display of the instruction
              Return type string
class androguard.core.bytecodes.dvm.HeaderItem(size, buff, cm)
     Bases: object
     This class can parse an header_item of a dex file. Several checks are performed to detect if this is not an
     header_item. Also the Adler32 checksum of the file is calculated in order to detect file corruption. :param buff:
     a string which represents a Buff object of the header item :type androguard.core.bytecode.BuffHandle buff:
     Buff object :param cm: a ClassManager object :type cm: ClassManager
     get_length()
     get_obj()
     get_off()
     get_raw()
     set\_off(off)
class androquard.core.bytecodes.dvm.Instruction
     Bases: object
```

This class represents a Dalvik instruction

It can both handle normal instructions as well as optimized instructions.

**Warning:** There is not much documentation about the optimized opcodes! Hence, it relies on reverese engineered specification!

More information about the instruction format can be found in the official documentation: https://source.android.com/devices/tech/dalvik/instruction-formats.html

**Warning:** Values stored in the instructions are already interpreted at this stage.

The Dalvik VM has a eight opcodes to create constant integer values. There are four variants for 32bit values and four for 64bit. If floating point numbers are required, you have to use the conversion opcodes like int-to-float, int-to-double or the variants using long.

Androguard will always show the values as they are used in the opcode and also extend signs and shift values! As an example: The opcode <code>const/high16</code> can be used to create constant values where the lower 16 bits are all zero. In this case, androguard will process bytecode 15 00 CD AB as beeing <code>const/high16</code> v0, <code>0xABCD0000</code>. For the sign-extension, nothing is really done here, as it only affects the bit representation in the virtual machine. As androguard parses the values and uses python types internally, we are not bound to specific size.

### OP = 0

### disasm()

Some small line for disassembly view

# get\_formatted\_operands()

Returns the formatted operands, if any. This is a list with the parsed and interpreted operands of the opcode.

Returns None if no operands, otherwise a List

Deprecated since version 3.4.0: Will be removed! This method always returns None

# get\_hex()

Returns a HEX String, separated by spaces every byte

The hex string contains the raw bytes of the instruction, including the opcode and all arguments.

### Return type str

## get\_kind()

Return the 'kind' argument of the instruction

This is the type of the argument, i.e. in which kind of table you have to look up the argument in the ClassManager

## Return type int

## get\_length()

Return the length of the instruction in bytes

## **Return type** int

## get\_literals()

Return the associated literals

**Return type** list of int

```
get name()
          Return the mnemonic of the instruction
              Return type string
     get_op_value()
          Return the numerical value of the opcode
              Return type int
     get_operands (idx=-1)
          Return all operands
          This will return a list of tuples, containing the Enum Operand at the first position and the objects after-
          wards.
              Return type List[Tuple(Operand, object, ..)]
     get_output (idx=-1)
          Return an additional output of the instruction
              Return type string
     get_raw()
          Return the object in a raw format
              Return type string
     get ref kind()
          Return the value of the 'kind' argument
              Return type value
     get_translated_kind()
          Return the translated value of the 'kind' argument
              Return type string
     length = 0
     show(idx)
          Print the instruction
          No Line ending is printed.
     show buff (idx)
          Return the display of the instruction
              Return type string
class androquard.core.bytecodes.dvm.Instruction00x(cm, buff)
     Bases: androguard.core.bytecodes.dvm.Instruction
     A class for unused instructions, has zero length and raises an error on initialization
     length = 0
class androguard.core.bytecodes.dvm.Instruction10t (cm, buff)
     Bases: androquard.core.bytecodes.dvm.Instruction
     This class represents all instructions which have the 10t format
     get_operands(idx=-1)
          Return all operands
          This will return a list of tuples, containing the Enum Operand at the first position and the objects after-
```

wards.

```
Return type List[Tuple(Operand, object, ..)]
     get_output (idx=-1)
          Return an additional output of the instruction
              Return type string
     get raw()
          Return the object in a raw format
              Return type string
     get_ref_off()
     length = 2
class androguard.core.bytecodes.dvm.Instruction10x(cm, buff)
     Bases: androquard.core.bytecodes.dvm.Instruction
     This class represents all instructions which have the 10x format
     get_raw()
          Return the object in a raw format
              Return type string
     length = 2
class androguard.core.bytecodes.dvm.Instruction11n(cm, buff)
     Bases: androquard.core.bytecodes.dvm.Instruction
     This class represents all instructions which have the 11n format
     get_literals()
          Return the associated literals
              Return type list of int
     get_operands (idx=-1)
          Return all operands
          This will return a list of tuples, containing the Enum Operand at the first position and the objects after-
          wards.
              Return type List[Tuple(Operand, object, ..)]
     get_output (idx=-1)
          Return an additional output of the instruction
              Return type string
     get_raw()
          Return the object in a raw format
              Return type string
     length = 2
class androguard.core.bytecodes.dvm.Instruction11x(cm, buff)
     Bases: androquard.core.bytecodes.dvm.Instruction
     This class represents all instructions which have the 11x format
     get_operands(idx=-1)
          Return all operands
```

```
This will return a list of tuples, containing the Enum Operand at the first position and the objects after-
          wards.
              Return type List[Tuple(Operand, object, ..)]
     get output (idx=-1)
          Return an additional output of the instruction
              Return type string
     get_raw()
          Return the object in a raw format
              Return type string
     length = 2
class androguard.core.bytecodes.dvm.Instruction12x(cm, buff)
     Bases: androguard.core.bytecodes.dvm.Instruction
     This class represents all instructions which have the 12x format
     get operands (idx=-1)
          Return all operands
          This will return a list of tuples, containing the Enum Operand at the first position and the objects after-
              Return type List[Tuple(Operand, object, ..)]
     get_output (idx=-1)
          Return an additional output of the instruction
               Return type string
     get_raw()
          Return the object in a raw format
               Return type string
     length = 2
class androquard.core.bytecodes.dvm.Instruction20bc(cm, buff)
     Bases: androquard.core.bytecodes.dvm.Instruction
     This class represents all instructions which have the 20bc format
     get_operands (idx=-1)
          Return all operands
          This will return a list of tuples, containing the Enum Operand at the first position and the objects after-
          wards.
              Return type List[Tuple(Operand, object, ..)]
     get_output (idx=-1)
          Return an additional output of the instruction
              Return type string
     get_raw()
          Return the object in a raw format
               Return type string
     length = 4
```

```
class androguard.core.bytecodes.dvm.Instruction20t (cm, buff)
     Bases: androquard.core.bytecodes.dvm.Instruction
     This class represents all instructions which have the 20t format
     get_operands (idx=-1)
          Return all operands
          This will return a list of tuples, containing the Enum Operand at the first position and the objects after-
              Return type List[Tuple(Operand, object, ..)]
     get_output (idx=-1)
          Return an additional output of the instruction
              Return type string
     get_raw()
          Return the object in a raw format
              Return type string
     get_ref_off()
     length = 4
class androguard.core.bytecodes.dvm.Instruction21c(cm, buff)
     Bases: androquard.core.bytecodes.dvm.Instruction
     This class represents all instructions which have the 21c format
     get_operands (idx=-1)
          Return all operands
          This will return a list of tuples, containing the Enum Operand at the first position and the objects after-
          wards.
              Return type List[Tuple(Operand, object, ..)]
     get_output (idx=-1)
          Return an additional output of the instruction
              Return type string
     get raw()
          Return the object in a raw format
              Return type string
     get_raw_string()
     get ref kind()
          Return the value of the 'kind' argument
              Return type value
     get_string()
     length = 4
class androguard.core.bytecodes.dvm.Instruction21h(cm, buff)
     Bases: androguard.core.bytecodes.dvm.Instruction
     This class represents all instructions which have the 21h format
```

```
get literals()
          Return the associated literals
              Return type list of int
     get_operands (idx=-1)
          Return all operands
          This will return a list of tuples, containing the Enum Operand at the first position and the objects after-
              Return type List[Tuple(Operand, object, ..)]
     get_output (idx=-1)
          Return an additional output of the instruction
              Return type string
     get_raw()
          Return the object in a raw format
              Return type string
     length = 4
class androguard.core.bytecodes.dvm.Instruction21s(cm, buff)
     Bases: androguard.core.bytecodes.dvm.Instruction
     This class represents all instructions which have the 21s format
     get_literals()
          Return the associated literals
              Return type list of int
     get_operands (idx=-1)
          Return all operands
          This will return a list of tuples, containing the Enum Operand at the first position and the objects after-
          wards.
              Return type List[Tuple(Operand, object, ..)]
     get output (idx=-1)
          Return an additional output of the instruction
              Return type string
     get_raw()
          Return the object in a raw format
              Return type string
     length = 4
class androguard.core.bytecodes.dvm.Instruction21t(cm, buff)
     Bases: androquard.core.bytecodes.dvm.Instruction
     This class represents all instructions which have the 21t format
     get_operands (idx=-1)
          Return all operands
          This will return a list of tuples, containing the Enum Operand at the first position and the objects after-
          wards.
              Return type List[Tuple(Operand, object, ..)]
```

```
get_output (idx=-1)
          Return an additional output of the instruction
              Return type string
     get raw()
          Return the object in a raw format
              Return type string
     get_ref_off()
     length = 4
class androguard.core.bytecodes.dvm.Instruction22b (cm, buff)
     Bases: androquard.core.bytecodes.dvm.Instruction
     This class represents all instructions which have the 22b format
     get_literals()
          Return the associated literals
              Return type list of int
     get_operands (idx=-1)
          Return all operands
          This will return a list of tuples, containing the Enum Operand at the first position and the objects after-
          wards.
              Return type List[Tuple(Operand, object, ..)]
     get_output (idx=-1)
          Return an additional output of the instruction
              Return type string
     get_raw()
          Return the object in a raw format
              Return type string
     length = 4
class androguard.core.bytecodes.dvm.Instruction22c(cm, buff)
     Bases: androquard.core.bytecodes.dvm.Instruction
     This class represents all instructions which have the 22c format
     get_operands (idx=-1)
          Return all operands
          This will return a list of tuples, containing the Enum Operand at the first position and the objects after-
              Return type List[Tuple(Operand, object, ..)]
     get_output (idx=-1)
          Return an additional output of the instruction
              Return type string
     get_raw()
          Return the object in a raw format
              Return type string
```

```
get_ref_kind()
          Return the value of the 'kind' argument
              Return type value
     length = 4
class androquard.core.bytecodes.dvm.Instruction22cs(cm, buff)
     Bases: androguard.core.bytecodes.dvm.Instruction
     This class represents all instructions which have the 22cs format
     get_operands (idx=-1)
          Return all operands
          This will return a list of tuples, containing the Enum Operand at the first position and the objects after-
          wards.
              Return type List[Tuple(Operand, object, ..)]
     get_output (idx=-1)
          Return an additional output of the instruction
              Return type string
     get_raw()
          Return the object in a raw format
              Return type string
     get_ref_kind()
          Return the value of the 'kind' argument
              Return type value
     length = 4
class androguard.core.bytecodes.dvm.Instruction22s(cm, buff)
     Bases: androquard.core.bytecodes.dvm.Instruction
     This class represents all instructions which have the 22s format
     get_literals()
          Return the associated literals
              Return type list of int
     get_operands (idx=-1)
          Return all operands
          This will return a list of tuples, containing the Enum Operand at the first position and the objects after-
          wards.
              Return type List[Tuple(Operand, object, ..)]
     get_output (idx=-1)
          Return an additional output of the instruction
              Return type string
     get_raw()
          Return the object in a raw format
              Return type string
     length = 4
```

```
class androguard.core.bytecodes.dvm.Instruction22t (cm, buff)
     Bases: androquard.core.bytecodes.dvm.Instruction
     This class represents all instructions which have the 22t format
     get_operands (idx=-1)
          Return all operands
          This will return a list of tuples, containing the Enum Operand at the first position and the objects after-
              Return type List[Tuple(Operand, object, ..)]
     get_output (idx=-1)
          Return an additional output of the instruction
              Return type string
     get_raw()
          Return the object in a raw format
              Return type string
     get_ref_off()
     length = 4
class androguard.core.bytecodes.dvm.Instruction22x(cm, buff)
     Bases: androquard.core.bytecodes.dvm.Instruction
     This class represents all instructions which have the 22x format
     get_operands (idx=-1)
          Return all operands
          This will return a list of tuples, containing the Enum Operand at the first position and the objects after-
          wards.
              Return type List[Tuple(Operand, object, ..)]
     get_output (idx=-1)
          Return an additional output of the instruction
              Return type string
     get raw()
          Return the object in a raw format
              Return type string
     length = 4
class androquard.core.bytecodes.dvm.Instruction23x(cm, buff)
     Bases: androguard.core.bytecodes.dvm.Instruction
     This class represents all instructions which have the 23x format
     get_operands (idx=-1)
          Return all operands
          This will return a list of tuples, containing the Enum Operand at the first position and the objects after-
              Return type List[Tuple(Operand, object, ..)]
     get_output (idx=-1)
          Return an additional output of the instruction
```

```
Return type string
     get_raw()
          Return the object in a raw format
              Return type string
     length = 4
class androguard.core.bytecodes.dvm.Instruction30t (cm, buff)
     Bases: androguard.core.bytecodes.dvm.Instruction
     This class represents all instructions which have the 30t format
     get_operands (idx=-1)
          Return all operands
          This will return a list of tuples, containing the Enum Operand at the first position and the objects after-
          wards.
              Return type List[Tuple(Operand, object, ..)]
     get output (idx=-1)
          Return an additional output of the instruction
              Return type string
     get_raw()
          Return the object in a raw format
              Return type string
     get_ref_off()
     length = 6
class androguard.core.bytecodes.dvm.Instruction31c(cm, buff)
     Bases: androquard.core.bytecodes.dvm.Instruction
     This class represents all instructions which have the 31c format
     get_operands(idx=-1)
          Return all operands
          This will return a list of tuples, containing the Enum Operand at the first position and the objects after-
          wards.
              Return type List[Tuple(Operand, object, ..)]
     get_output (idx=-1)
          Return an additional output of the instruction
              Return type string
     get_raw()
          Return the object in a raw format
              Return type string
     get_raw_string()
     get_ref_kind()
          Return the value of the 'kind' argument
              Return type value
```

```
get_string()
          Return the string associated to the 'kind' argument
              Return type string
     length = 6
class androquard.core.bytecodes.dvm.Instruction31i(cm, buff)
     Bases: androguard.core.bytecodes.dvm.Instruction
     This class represents all instructions which have the 31i format
     get_literals()
          Return the associated literals
              Return type list of int
     get_operands (idx=-1)
          Return all operands
          This will return a list of tuples, containing the Enum Operand at the first position and the objects after-
          wards.
              Return type List[Tuple(Operand, object, ..)]
     get_output (idx=-1)
          Return an additional output of the instruction
              Return type string
     get_raw()
          Return the object in a raw format
              Return type string
     length = 6
class androguard.core.bytecodes.dvm.Instruction31t (cm, buff)
     Bases: androquard.core.bytecodes.dvm.Instruction
     This class represents all instructions which have the 31t format
     get_operands (idx=-1)
          Return all operands
          This will return a list of tuples, containing the Enum Operand at the first position and the objects after-
          wards.
              Return type List[Tuple(Operand, object, ..)]
     get output (idx=-1)
          Return an additional output of the instruction
              Return type string
     get_raw()
          Return the object in a raw format
              Return type string
     get_ref_off()
     length = 6
class androquard.core.bytecodes.dvm.Instruction32x(cm, buff)
     Bases: androguard.core.bytecodes.dvm.Instruction
```

```
This class represents all instructions which have the 32x format
     get_operands (idx=-1)
          Return all operands
          This will return a list of tuples, containing the Enum Operand at the first position and the objects after-
          wards.
              Return type List[Tuple(Operand, object, ..)]
     get_output (idx=-1)
          Return an additional output of the instruction
              Return type string
     get_raw()
          Return the object in a raw format
              Return type string
     length = 6
class androquard.core.bytecodes.dvm.Instruction35c(cm, buff)
     Bases: androguard.core.bytecodes.dvm.Instruction
     This class represents all instructions which have the 35c format
     get_operands (idx=-1)
          Return all operands
          This will return a list of tuples, containing the Enum Operand at the first position and the objects after-
               Return type List[Tuple(Operand, object, ..)]
     get_output (idx=-1)
          Return an additional output of the instruction
               Return type string
     get_raw()
          Return the object in a raw format
              Return type string
     get ref kind()
          Return the value of the 'kind' argument
              Return type value
     length = 6
class androquard.core.bytecodes.dvm.Instruction35mi(cm, buff)
     Bases: androguard.core.bytecodes.dvm.Instruction
     This class represents all instructions which have the 35mi format
     get_operands (idx=-1)
          Return all operands
          This will return a list of tuples, containing the Enum Operand at the first position and the objects after-
              Return type List[Tuple(Operand, object, ..)]
     get_output (idx=-1)
          Return an additional output of the instruction
```

```
Return type string
     get_raw()
          Return the object in a raw format
              Return type string
     get ref kind()
          Return the value of the 'kind' argument
              Return type value
     length = 6
class androquard.core.bytecodes.dvm.Instruction35ms(cm, buff)
     Bases: androquard.core.bytecodes.dvm.Instruction
     This class represents all instructions which have the 35ms format
     get_operands (idx=-1)
          Return all operands
          This will return a list of tuples, containing the Enum Operand at the first position and the objects after-
          wards.
              Return type List[Tuple(Operand, object, ..)]
     get_output (idx=-1)
          Return an additional output of the instruction
              Return type string
     get_raw()
          Return the object in a raw format
              Return type string
     get_ref_kind()
          Return the value of the 'kind' argument
              Return type value
     length = 6
class androguard.core.bytecodes.dvm.Instruction3rc(cm, buff)
     Bases: androquard.core.bytecodes.dvm.Instruction
     This class represents all instructions which have the 3rc format
     get_operands (idx=-1)
          Return all operands
          This will return a list of tuples, containing the Enum Operand at the first position and the objects after-
              Return type List[Tuple(Operand, object, ..)]
     get_output (idx=-1)
          Return an additional output of the instruction
              Return type string
     get_raw()
          Return the object in a raw format
              Return type string
```

```
get ref kind()
          Return the value of the 'kind' argument
              Return type value
     length = 6
class androquard.core.bytecodes.dvm.Instruction3rmi(cm, buff)
     Bases: androguard.core.bytecodes.dvm.Instruction
     This class represents all instructions which have the 3rmi format
     Note, this instruction is similar to 3rc but holds an inline
     get_operands (idx=-1)
          Return all operands
          This will return a list of tuples, containing the Enum Operand at the first position and the objects after-
          wards.
              Return type List[Tuple(Operand, object, ..)]
     get output (idx=-1)
          Return an additional output of the instruction
              Return type string
     get_raw()
          Return the object in a raw format
              Return type string
     get_ref_kind()
          Return the value of the 'kind' argument
              Return type value
     length = 6
class androguard.core.bytecodes.dvm.Instruction3rms(cm, buff)
     Bases: androguard.core.bytecodes.dvm.Instruction
     This class represents all instructions which have the 3rms format
     Note, this instruction is similar to 3rc but holds a vtaboff
     get_operands (idx=-1)
          Return all operands
          This will return a list of tuples, containing the Enum Operand at the first position and the objects after-
              Return type List[Tuple(Operand, object, ..)]
     get_output (idx=-1)
          Return an additional output of the instruction
               Return type string
     get_raw()
          Return the object in a raw format
              Return type string
     get_ref_kind()
          Return the value of the 'kind' argument
```

```
Return type value
     length = 6
class androguard.core.bytecodes.dvm.Instruction40sc(cm, buff)
     Bases: androguard.core.bytecodes.dvm.Instruction
     This class represents all instructions which have the 40sc format
     This instruction is only used in ODEX
     get_operands (idx=-1)
          Return all operands
          This will return a list of tuples, containing the Enum Operand at the first position and the objects after-
              Return type List[Tuple(Operand, object, ..)]
     get_output (idx=-1)
          Return an additional output of the instruction
              Return type string
     get raw()
          Return the object in a raw format
              Return type string
     get ref kind()
          Return the value of the 'kind' argument
              Return type value
     length = 8
class androguard.core.bytecodes.dvm.Instruction41c(cm, buff)
     Bases: \verb| and roguard.core.by tecodes.dvm.Instruction| \\
     This class represents all instructions which have the 41c format
     This instruction is only used in ODEX
     get_operands (idx=-1)
          Return all operands
          This will return a list of tuples, containing the Enum Operand at the first position and the objects after-
          wards.
              Return type List[Tuple(Operand, object, ..)]
     get_output (idx=-1)
          Return an additional output of the instruction
              Return type string
     get_raw()
          Return the object in a raw format
              Return type string
     get_ref_kind()
          Return the value of the 'kind' argument
              Return type value
     length = 8
```

```
class androquard.core.bytecodes.dvm.Instruction45cc(cm, buff)
     Bases: androguard.core.bytecodes.dvm.Instruction
     get_operands()
          Return all operands
          This will return a list of tuples, containing the Enum Operand at the first position and the objects after-
          wards.
              Return type List[Tuple(Operand, object, ..)]
     get_output (idx=-1)
          Return an additional output of the instruction
              Return type string
     get_raw()
          Return the object in a raw format
              Return type string
     length = 8
class androguard.core.bytecodes.dvm.Instruction4rcc(cm, buff)
     Bases: androguard.core.bytecodes.dvm.Instruction
     get_operands()
          Return all operands
          This will return a list of tuples, containing the Enum Operand at the first position and the objects after-
              Return type List[Tuple(Operand, object, ..)]
     get_output (idx=-1)
          Return an additional output of the instruction
              Return type string
     get_raw()
          Return the object in a raw format
              Return type string
     length = 8
class androguard.core.bytecodes.dvm.Instruction511(cm, buff)
     Bases: androguard.core.bytecodes.dvm.Instruction
     This class represents all instructions which have the 511 format
     get_literals()
          Return the associated literals
              Return type list of int
     get_operands (idx=-1)
          Return all operands
          This will return a list of tuples, containing the Enum Operand at the first position and the objects after-
          wards.
              Return type List[Tuple(Operand, object, ..)]
     get output (idx=-1)
          Return an additional output of the instruction
```

```
Return type string
     get_raw()
          Return the object in a raw format
              Return type string
     length = 10
class androguard.core.bytecodes.dvm.Instruction52c(cm, buff)
     Bases: androguard.core.bytecodes.dvm.Instruction
     This class represents all instructions which have the 52c format
     This instruction is only used in ODEX
     get_operands (idx=-1)
          Return all operands
          This will return a list of tuples, containing the Enum Operand at the first position and the objects after-
              Return type List[Tuple(Operand, object, ..)]
     get output (idx=-1)
          Return an additional output of the instruction
              Return type string
     get raw()
          Return the object in a raw format
              Return type string
     get_ref_kind()
          Return the value of the 'kind' argument
              Return type value
     length = 10
class androguard.core.bytecodes.dvm.Instruction5rc(cm, buff)
     Bases: androquard.core.bytecodes.dvm.Instruction
     This class represents all instructions which have the 5rc format
     This instruction is only used in ODEX
     get_operands (idx=-1)
          Return all operands
          This will return a list of tuples, containing the Enum Operand at the first position and the objects after-
          wards.
              Return type List[Tuple(Operand, object, ..)]
     get_output (idx=-1)
          Return an additional output of the instruction
              Return type string
     get_raw()
          Return the object in a raw format
              Return type string
```

```
get ref kind()
          Return the value of the 'kind' argument
              Return type value
     length = 10
exception androquard.core.bytecodes.dvm.InvalidInstruction
     Bases: Exception
class androguard.core.bytecodes.dvm.LinearSweepAlgorithm
     Bases: object
     This class is used to disassemble a method. The algorithm used by this class is linear sweep.
     static get_instructions (cm, size, insn, idx)
          Yields all instructions for the given bytecode sequence. If unknown/corrupt/unused instructions are en-
          countered, the loop will stop and an error is written to the log.
          That means that the bytecode read might be corrupt or was crafted in this way, to break parsers.
              Parameters
                  • cm (ClassManager) – a ClassManager object
                  • size (int) – the total size of the buffer in 16-bit units
                  • insn (bytearray) – a raw buffer where are the instructions
                  • idx (int) - a start address in the buffer
                  • raise_errors (bool) – True to raise errors instead of simply logging them
              Return type Iterator[Instruction]
class androguard.core.bytecodes.dvm.MapItem(buff, cm)
     Bases: object
     get_item()
          Return the associated item itself. Might return None, if parse () was not called yet.
          This method is the same as get_item().
     get_length()
     get obj()
          Return the associated item itself. Might return None, if parse () was not called yet.
          This method is the same as get_item().
     get off()
          Gets the offset of the map item itself inside the DEX file
     get offset()
          Gets the offset of the item of the map item
     get_raw()
     get_size()
          Returns the number of items found at the location indicated by get_offset ().
     get_type()
     parse()
     set item(item)
```

show()

```
class androguard.core.bytecodes.dvm.MapList(cm, off, buff)
     Bases: object
     This class can parse the "map_list" of the dex format
     https://source.android.com/devices/tech/dalvik/dex-format#map-list
     get class manager()
     get_item_type (ttype)
          Get a particular item type
              Parameters ttype – a string which represents the desired type
              Return type None or the item object
     get_length()
     get_obj()
     get_off()
     get_raw()
     set_off(off)
     show()
          Print with a pretty display the MapList object
class androquard.core.bytecodes.dvm.MethodAnnotation(buff, cm)
     Bases: object
     This class can parse a method_annotation of a dex file
          Parameters
                • buff (Buff object) – a string which represents a Buff object of the method_annotation
                • cm (ClassManager) – a ClassManager object
     get_annotations_off()
          Return the offset from the start of the file to the list of annotations for the method
              Return type int
     get_length()
     get method idx()
          Return the index into the method_ids list for the identity of the method being annotated
              Return type int
     get_obj()
     get_off()
     get_raw()
     set_off(off)
     show()
class androguard.core.bytecodes.dvm.MethodHIdItem(size, buff, cm)
     Bases: object
     This class can parse a list of method_id_item of a dex file
          Parameters
```

```
• buff (Buff object) - a string which represents a Buff object of the list of
                 method_id_item
                • cm (ClassManager) – a ClassManager object
     get (idx)
     get_length()
     get_obj()
     get_off()
     get_raw()
     reload()
     set_off(off)
     show()
class androquard.core.bytecodes.dvm.MethodIdItem(buff, cm)
     Bases: object
     This class can parse a method_id_item of a dex file
          Parameters
                • buff (Buff object) – a string which represents a Buff object of the method_id_item
                • cm (ClassManager) – a ClassManager object
     get class idx()
          Return the index into the type_ids list for the definer of this method
              Return type int
     get class name()
          Return the class name of the method
              Return type string
     get_descriptor()
          Return the descriptor
              Return type string
     get_length()
     get_list()
     get name()
          Return the name of the method
              Return type string
     get_name_idx()
          Return the index into the string_ids list for the name of this method
              Return type int
     get_obj()
     get_proto()
          Return the prototype of the method
              Return type string
```

```
get_proto_idx()
         Return the index into the proto_ids list for the prototype of this method
             Return type int
     get_raw()
     get real descriptor()
         Return the real descriptor (i.e. without extra spaces)
             Return type string
     get_triple()
     reload()
     show()
class androguard.core.bytecodes.dvm.MethodIdItemInvalid
     Bases: object
     get_class_name()
     get_descriptor()
     get_list()
     get_name()
     get_proto()
     show()
class androguard.core.bytecodes.dvm.OdexDependencies(buff)
     Bases: object
     This class can parse the odex dependencies
         Parameters buff – a Buff object string which represents the odex dependencies
     get_dependencies()
         Return the list of dependencies
             Return type a list of strings
     get_raw()
class androguard.core.bytecodes.dvm.OdexHeaderItem(buff)
     Bases: object
     This class can parse the odex header
         Parameters buff – a Buff object string which represents the odex dependencies
     get_raw()
     show()
class androquard.core.bytecodes.dvm.OffObj(o)
     Bases: object
class androguard.core.bytecodes.dvm.PackedSwitch(cm, buff)
     Bases: object
     This class can parse a PackedSwitch instruction
```

**Parameters** buff – a Buff object which represents a buffer where the instruction is stored

3.1. androguard package

```
add note (msg)
          Add a note to this instruction
              Parameters msg(objects (string)) - the message
     disasm()
     get_formatted_operands()
     get_hex()
          Returns a HEX String, separated by spaces every byte
     get_keys()
          Return the keys of the instruction
              Return type a list of long
     get_length()
     get_name()
          Return the name of the instruction
              Return type string
     get notes()
          Get all notes from this instruction
              Return type a list of objects
     get_op_value()
          Get the value of the opcode
              Return type int
     get_operands(idx=-1)
          Return an additional output of the instruction
              Return type string
     get_output (idx=-1)
          Return an additional output of the instruction
                  rtype string
     get_raw()
     get_targets()
          Return the targets (address) of the instruction
              Return type a list of long
     get_values()
     show (pos)
          Print the instruction
     show_buff(pos)
          Return the display of the instruction
              Return type string
class androguard.core.bytecodes.dvm.ParameterAnnotation(buff, cm)
     Bases: object
     This class can parse a parameter_annotation of a dex file
          Parameters
```

```
• buff (Buff object) - a string which represents a Buff object of the parame-
                 ter annotation
               • cm (ClassManager) – a ClassManager object
     get_annotations_off()
          Return the offset from the start of the file to the list of annotations for the method parameters
              Return type int
     get_length()
     get_method_idx()
          Return the index into the method_ids list for the identity of the method whose parameters are being anno-
          tated
              Return type int
     get_obj()
     get_off()
     get_raw()
     set\_off(off)
     show()
class androguard.core.bytecodes.dvm.ProtoHIdItem(size, buff, cm)
     Bases: object
     This class can parse a list of proto_id_item of a dex file
          Parameters
                • buff (Buff object) - a string which represents a Buff object of the list of
                 proto id item
               • cm (ClassManager) – a ClassManager object
     get (idx)
     get_length()
     get_obj()
     get_off()
     get_raw()
     set_off(off)
     show()
class androguard.core.bytecodes.dvm.ProtoIdItem(buff, cm)
     Bases: object
     This class can parse a proto_id_item of a dex file
          Parameters
               • buff (Buff object) – a string which represents a Buff object of the proto_id_item
               • cm (ClassManager) - a ClassManager object
     get_length()
     get_obj()
```

```
get_parameters_off()
          Return the offset from the start of the file to the list of parameter types for this prototype, or 0 if this
          prototype has no parameters
              Return type int
     get_parameters_off_value()
          Return the string associated to the parameters off
              Return type MUTF8String
     get_raw()
     get_return_type_idx()
          Return the index into the type_ids list for the return type of this prototype
              Return type int
     get_return_type_idx_value()
          Return the string associated to the return_type_idx
              Return type string
     get_shorty_idx()
          Return the index into the string_ids list for the short-form descriptor string of this prototype
              Return type int
     get shorty idx value()
          Return the string associated to the shorty_idx
              Return type string
     show()
class androquard.core.bytecodes.dvm.ProtoIdItemInvalid
     Bases: object
     get_params()
     get_return_type()
     get_shorty()
     show()
class androguard.core.bytecodes.dvm.SparseSwitch(cm, buff)
     Bases: object
     This class can parse a SparseSwitch instruction
          Parameters buff – a Buff object which represents a buffer where the instruction is stored
     add note (msg)
          Add a note to this instruction
              Parameters msg(objects (string)) - the message
     disasm()
     get_formatted_operands()
     get_hex()
          Returns a HEX String, separated by spaces every byte
     get_keys()
          Return the keys of the instruction
```

```
Return type a list of long
     get_length()
     get_name()
          Return the name of the instruction
              Return type string
     get notes()
          Get all notes from this instruction
              Return type a list of objects
     get_op_value()
          Get the value of the opcode
              Return type int
     get_operands (idx=-1)
          Return an additional output of the instruction
              Return type string
     get output (idx=-1)
          Return an additional output of the instruction
              Return type string
     get_raw()
     get targets()
          Return the targets (address) of the instruction
              Return type a list of long
     get_values()
     show (pos)
          Print the instruction
     show_buff(pos)
          Return the display of the instruction
              Return type string
class androguard.core.bytecodes.dvm.StringDataItem (buff, cm)
     Bases: object
```

This class can parse a string\_data\_item of a dex file

Strings in Dalvik files might not be representable in python! This is due to the fact, that you can store any UTF-16 character inside a Dalvik file, but this string might not be decodeable in python as it can contain invalid surrogate-pairs.

To circumvent this issue, this class has different methods how to access the string. There are also some fallbacks implemented to make a "invalid" string printable in python. Dalvik uses MUTF-8 as encoding for the strings. This encoding has the advantage to allow for null terminated strings in UTF-8 encoding, as the null character maps to something else. Therefore you can use  $get\_data()$  to retrieve the actual data of the string and can handle encoding yourself. Or you use  $get\_unicode()$  to return a decoded UTF-16 string, which might cause problems during printing or saving. If you want a representation of the string, which should be printable in python you cause get() which escapes invalid characters.

#### **Parameters**

• buff (BuffHandle) - a string which represents a Buff object of the string data item

```
• cm (ClassManager) – a ClassManager object
     get()
          Returns a MUTF8String object
     get_data()
          Return a series of MUTF-8 code units (a.k.a. octets, a.k.a. bytes) followed by a byte of value 0
              Return type string
     get_length()
          Get the length of the raw string including the ULEB128 coded length and the null byte terminator
              Returns int
     get_obj()
     get_off()
     get_raw()
          Returns the raw string including the ULEB128 coded length and null byte string terminator
              Returns bytes
     get utf16 size()
          Return the size of this string, in UTF-16 code units
          :rtype:int
     set_off(off)
     show()
class androguard.core.bytecodes.dvm.StringIdItem(buff, cm)
     Bases: object
     This class can parse a string_id_item of a dex file
          Parameters
                • buff (Buff object) - a string which represents a Buff object of the string_id_item
                • cm (ClassManager) - a ClassManager object
     get_length()
     get_obj()
     get_off()
     get_raw()
     get_string_data_off()
          Return the offset from the start of the file to the string data for this item
              Return type int
     set\_off(off)
     show()
class androguard.core.bytecodes.dvm.TryItem(buff, cm)
     Bases: object
     This class represents the try_item format
          Parameters
                • buff (BuffHandle) – a raw buffer where are the try item format
```

```
• cm (ClassManager) - the ClassManager
     get_handler_off()
          Get the offset in bytes from the start of the associated EncodedCatchHandlerList to the
          EncodedCatchHandler for this entry.
              Return type int
     get_insn_count()
          Get the number of 16-bit code units covered by this entry
              Return type int
     get_length()
     get_off()
     get_raw()
     get_start_addr()
          Get the start address of the block of code covered by this entry. The address is a count of 16-bit code units
          to the start of the first covered instruction.
              Return type int
     set_off(off)
class androguard.core.bytecodes.dvm.TypeHIdItem(size, buff, cm)
     Bases: object
     This class can parse a list of type_id_item of a dex file
          Parameters
                • buff (Buff object) - a string which represents a Buff object of the list of type_id_item
                • cm (ClassManager) – a ClassManager object
     get (idx)
     get_length()
     get_obj()
     get_off()
     get_raw()
     get_type()
          Return the list of type_id_item
              Return type a list of TypeIdItem objects
     set_off(off)
     show()
class androguard.core.bytecodes.dvm.TypeIdItem(buff, cm)
     Bases: object
     This class can parse a type_id_item of a dex file
          Parameters
                • buff (Buff object) - a string which represents a Buff object of the type_id_item
                • cm (ClassManager) – a ClassManager object
```

```
get_descriptor_idx()
          Return the index into the string_ids list for the descriptor string of this type
              Return type int
     get_descriptor_idx_value()
          Return the string associated to the descriptor
              Return type string
     get_length()
     get_obj()
     get_raw()
     show()
class androguard.core.bytecodes.dvm.TypeItem(buff, cm)
     Bases: object
     This class can parse a type_item of a dex file
          Parameters
                • buff (Buff object) – a string which represents a Buff object of the type_item
                • cm (ClassManager) – a ClassManager object
     get_length()
     get_obj()
     get_raw()
     get_string()
          Return the type string
              Return type string
     get_type_idx()
          Return the index into the type_ids list
              Return type int
     show()
class androguard.core.bytecodes.dvm.TypeList(buff, cm)
     Bases: object
     This class can parse a type_list of a dex file
          Parameters
                • buff (Buff object) – a string which represents a Buff object of the type_list
                • cm (ClassManager) – a ClassManager object
     get_length()
     get_list()
          Return the list of TypeItem
              Return type a list of TypeItem objects
     get_obj()
     get_off()
```

```
get_pad()
         Return the alignment string
             Return type string
     get_raw()
     get size()
          Return the size of the list, in entries
             Return type int
     get_string()
          Return the concatenation of all strings
              Return type string
     get_type_list_off()
          Return the offset of the item
             Return type int
     set_off(off)
     show()
androguard.core.bytecodes.dvm.clean_name_instruction(instruction)
     USED IN ELSIM
androquard.core.bytecodes.dvm.determineException(vm, m)
     Returns try-catch handler inside the method.
```

### **Parameters**

- **vm** a DalvikVMFormat
- ullet  $\mathbf{m}-a$  EncodedMethod

### Returns

```
and roguard.core.bytecodes.dvm.determineNext(i, cur\_idx, m)
```

Determine the next offsets inside the bytecode of an *EncodedMethod*. The offsets are calculated in number of bytes from the start of the method. Note, that offsets inside the bytecode are denoted in 16bit units but this method returns actual bytes!

Offsets inside the opcode are counted from the beginning of the opcode.

The returned type is a list, as branching opcodes will have multiple paths. *if* and *switch* opcodes will return more than one item in the list, while *throw*, *return* and *goto* opcodes will always return a list with length one.

An offset of -1 indicates that the method is exited, for example by throw or return.

If the entered opcode is not branching or jumping, an empty list is returned.

#### **Parameters**

- i (Instruction) the current Instruction
- cur\_idx (int) Index of the instruction
- m (EncodedMethod) the current method

# Returns

# Return type list

```
androguard.core.bytecodes.dvm.get_access_flags_string(value)
Transform an access flag field to the corresponding string
```

```
Parameters value (int) – the value of the access flags
```

### **Return type** string

```
androguard.core.bytecodes.dvm.get_byte(cm, buff)

androguard.core.bytecodes.dvm.get_bytecodes_method(dex_object, ana_object, method)

androguard.core.bytecodes.dvm.get_bytecodes_methodx(method, mx)

androguard.core.bytecodes.dvm.get_instruction(cm, op_value, buff)

Return the Instruction for the given opcode
```

#### **Parameters**

- cm (ClassManager) ClassManager to propagate to Instruction
- op\_value (int) integer value of the instruction
- **buff** (*bytearray*) Bytecode starting with the instruction

**Returns** the parsed Instruction

Return type Instruction

```
androguard.core.bytecodes.dvm.get_instruction_payload(op_value, cm, buff)
androguard.core.bytecodes.dvm.get_kind(cm, kind, value)
    Return the value of the 'kind' argument
```

#### **Parameters**

- cm (ClassManager) a ClassManager object
- kind (int) the type of the 'kind' argument
- **value** (*int*) the value of the 'kind' argument

# Return type string

```
androguard.core.bytecodes.dvm.get_optimized_instruction(cm, op_value, buff)
androguard.core.bytecodes.dvm.get_params_info(nb, proto)
androguard.core.bytecodes.dvm.get_sbyte(cm, buff)
androguard.core.bytecodes.dvm.get_type(atype, size=None)
    Retrieve the type of a descriptor(e.g: I)
androguard.core.bytecodes.dvm.read_null_terminated_string(f)
    Read a null terminated string from a file-like object.:param f: file-like object:rtype: bytearray
androguard.core.bytecodes.dvm.readsleb128(cm, buff)
    Read a signed LEB128 at the current position of the buffer.
```

Parameters buff – a file like object

Returns decoded sLEB128

```
androguard.core.bytecodes.dvm.readuleb128 (cm, buff)
Read an unsigned LEB128 at the current position of the buffer
```

Parameters buff - a file like object

Returns decoded unsigned LEB128

```
androguard.core.bytecodes.dvm.readuleb128p1 (cm, buff)
```

Read an unsigned LEB128p1 at the current position of the buffer. This format is the same as uLEB128 but has the ability to store the value -1.

Parameters buff – a file like object

Returns decoded uLEB128p1

 $\verb"androguard.core.bytecodes.dvm.static_operand_instruction" (instruction)$ 

**USED IN ELSIM** 

androguard.core.bytecodes.dvm.writesleb128(cm, value)

Convert an integer value to the corresponding signed LEB128

Parameters value – integer value

**Returns** bytes

androguard.core.bytecodes.dvm.writeuleb128(cm, value)

Convert an integer value to the corresponding unsigned LEB128.

Raises a value error, if the given value is negative.

**Parameters** value – non-negative integer

Returns bytes

# androguard.core.bytecodes.axml module

class androguard.core.bytecodes.axml.ARSCComplex (buff, parent=None)

Bases: object

This is actually a ResTable\_map\_entry

It contains a set of {name: value} mappings, which are of type *ResTable\_map*. A *ResTable\_map* contains two items: *ResTable\_ref* and *Res\_value*.

See http://androidxref.com/9.0.0\_r3/xref/frameworks/base/libs/androidfw/include/androidfw/ResourceTypes.h#1485 for *ResTable\_map\_entry* and http://androidxref.com/9.0.0\_r3/xref/frameworks/base/libs/androidfw/include/androidfw/ResourceTypes.h#1498 for *ResTable\_map* 

class androguard.core.bytecodes.axml.ARSCHeader(buff, expected\_type=None)

Bases: object

Object which contains a Resource Chunk. This is an implementation of the ResChunk\_header.

It will throw an ResParserError if the header could not be read successfully.

It is not checked if the data is outside the buffer size nor if the current chunk fits into the parent chunk (if any)!

The parameter *expected\_type* can be used to immediately check the header for the type or raise a *ResParserError*. This is useful if you know what type of chunk must follow.

See http://androidxref.com/9.0.0\_r3/xref/frameworks/base/libs/androidfw/include/androidfw/ResourceTypes. h#196 :raises: ResParserError

SIZE = 8

## property end

Get the absolute offset inside the file, where the chunk ends. This is equal to ARSCHeader.start + ARSC-Header.size.

## property header\_size

Size of the chunk header (in bytes). Adding this value to the address of the chunk allows you to find its associated data (if any).

## property size

Total size of this chunk (in bytes). This is the chunkSize plus the size of any data associated with the chunk. Adding this value to the chunk allows you to completely skip its contents (including any child chunks). If this value is the same as chunkSize, there is no data associated with the chunk.

### property type

Type identifier for this chunk

```
class androguard.core.bytecodes.axml.ARSCParser(raw_buff)
    Bases: object
```

Parser for resource.arsc files

The ARSC File is, like the binary XML format, a chunk based format. Both formats are actually identical but use different chunks in order to store the data.

The most outer chunk in the ARSC file is a chunk of type RES\_TABLE\_TYPE. Inside this chunk is a StringPool and at least one package.

Each package is a chunk of type RES\_TABLE\_PACKAGE\_TYPE. It contains again many more chunks.

```
class ResourceResolver(android_resources, config=None)
```

Bases: object

Resolves resources by ID and configuration. This resolver deals with complex resources as well as with references.

```
put_ate_value (result, ate, config)
```

Put a ResTableEntry into the list of results :param list result: results array :param ARSCResTableEntry ate: :param ARSCResTableConfig config: :return:

```
put_item_value (result, item, config, parent, complex_)
```

Put the tuple (ARSCResTableConfig, resolved string) into the result set

## **Parameters**

- result (list) the result set
- item (ARSCResStringPoolRef) -
- config (ARSCResTableConfig) -
- parent (ARSCResTableEntry) the originating entry
- complex (bool) True if the originating ARSCResTableEntry was complex

# Returns

## resolve (res\_id)

the given ID into the Resource and returns a list of matching resources.

Parameters res id (int) – numerical ID of the resource

Returns a list of tuples of (ARSCResTableConfig, str)

### get bool resources (package name, locale='\x00\x00')

Get the XML (as string) of all resources of type 'bool'.

Read more about bool resources: https://developer.android.com/guide/topics/resources/more-resources. html#Bool

### **Parameters**

- package\_name the package name to get the resources for
- locale the locale to get the resources for (default: ")

# get\_color\_resources (package\_name, locale='\x00\x00')

Get the XML (as string) of all resources of type 'color'.

Read more about color resources: https://developer.android.com/guide/topics/resources/more-resources. html#Color

### **Parameters**

- package\_name the package name to get the resources for
- locale the locale to get the resources for (default: ")

## get\_dimen\_resources (package\_name, locale='\x00\x00')

Get the XML (as string) of all resources of type 'dimen'.

Read more about Dimension resources: https://developer.android.com/guide/topics/resources/more-resources.html#Dimension

#### **Parameters**

- package\_name the package name to get the resources for
- locale the locale to get the resources for (default: ")

# get\_id (package\_name, rid, locale='\x00\x00')

Returns the tuple (resource\_type, resource\_name, resource\_id) for the given resource\_id.

#### **Parameters**

- package\_name package name to query
- rid the resource id
- locale specific locale

**Returns** tuple of (resource\_type, resource\_name, resource\_id)

## get\_id\_resources (package\_name, locale='\x00\x00')

Get the XML (as string) of all resources of type 'id'.

Read more about ID resources: https://developer.android.com/guide/topics/resources/more-resources.html#Id

### **Parameters**

- package\_name the package name to get the resources for
- locale the locale to get the resources for (default: ")

### get\_integer\_resources (package\_name, locale='\x00\x00')

Get the XML (as string) of all resources of type 'integer'.

Read more about integer resources: https://developer.android.com/guide/topics/resources/more-resources. html#Integer

#### **Parameters**

- package name the package name to get the resources for
- locale the locale to get the resources for (default: ")

```
get_items (package_name)
```

# get\_locales (package\_name)

Retrieve a list of all available locales in a given packagename.

Parameters package\_name - the package name to get locales of

## get\_packages\_names()

Retrieve a list of all package names, which are available in the given resources.arsc.

# get\_public\_resources (package\_name, locale='\x00\x00')

Get the XML (as string) of all resources of type 'public'.

The public resources table contains the IDs for each item.

#### **Parameters**

- package\_name the package name to get the resources for
- locale the locale to get the resources for (default: ")

```
get res configs(rid, config=None, fallback=True)
```

Return the resources found with the ID *rid* and select the right one based on the configuration, or return all if no configuration was set.

But we try to be generous here and at least try to resolve something: This method uses a fallback to return at least one resource (the first one in the list) if more than one items are found and the default config is used and no default entry could be found.

This is usually a bad sign (i.e. the developer did not follow the android documentation: https://developer.android.com/guide/topics/resources/localization.html#failing2) In practise an app might just be designed to run on a single locale and thus only has those locales set.

You can disable this fallback behaviour, to just return exactly the given result.

#### **Parameters**

- rid resource id as int
- config a config to resolve from, or None to get all results
- fallback Enable the fallback for resolving default configuration (default: True)

Returns a list of ARSCResTableConfig: ARSCResTableEntry

```
get_res_id_by_key (package_name, resource_type, key)
get_resolved_res_configs (rid, config=None)
```

Return a list of resolved resource IDs with their corresponding configuration. It has a similar return type as <code>get\_res\_configs()</code> but also handles complex entries and references. Also instead of returning <code>ARSCResTableEntry</code> in the tuple, the actual values are resolved.

This is the preferred way of resolving resource IDs to their resources.

## **Parameters**

- rid (int) the numerical ID of the resource
- config (ARSCTableResConfig) the desired configuration or None to retrieve all

**Returns** A list of tuples of (ARSCResTableConfig, str)

```
get_resolved_strings()
get_resource_bool(ate)
get_resource_color(ate)
get_resource_dimen(ate)
get_resource_id(ate)
get_resource_integer(ate)
get_resource_string(ate)
get_resource_style(ate)
get_resource_style(ate)
```

Returns the XML name for a resource, including the package name if package is None. A full name might look like @com.example:string/foobar Otherwise the name is only looked up in the specified package and

is returned without the package name. The same example from about without the package name will read as @string/foobar.

If the ID could not be found, *None* is returned.

A description of the XML name can be found here: https://developer.android.com/guide/topics/resources/providing-resources#ResourcesFromXml

#### **Parameters**

- r\_id numerical ID if the resource
- package package name

Returns XML name identifier

```
get_string (package_name, name, locale='\x00\x00')
```

```
get_string_resources (package_name, locale='\x00\x00')
```

Get the XML (as string) of all resources of type 'string'.

Read more about string resources: https://developer.android.com/guide/topics/resources/string-resource.html

#### **Parameters**

- package\_name the package name to get the resources for
- locale the locale to get the resources for (default: ")

### get\_strings\_resources()

Get the XML (as string) of all resources of type 'string'. This is a combined variant, which has all locales and all package names stored.

```
get_type_configs (package_name, type_name=None)
```

```
get_types (package_name, locale='\x00\x00')
```

Retrieve a list of all types which are available in the given package and locale.

#### **Parameters**

- package\_name the package name to get types of
- locale the locale to get types of (default: ")

## static parse\_id(name)

Resolves an id from a binary XML file in the form "@[package:]DEADBEEF" and returns a tuple of package name and resource id. If no package name was given, i.e. the ID has the form "@DEADBEEF", the package name is set to None.

Raises a ValueError if the id is malformed.

**Parameters** name – the string of the resource, as in the binary XML file

**Returns** a tuple of (resource\_id, package\_name).

```
class androguard.core.bytecodes.axml.ARSCResStringPoolRef(buff, parent=None)
    Bases: object
```

This is actually a *Res\_value* It holds information about the stored resource value

 $\textbf{See:} \quad http://androidxref.com/9.0.0\_r3/xref/frameworks/base/libs/androidfw/include/androidfw/ResourceTypes. \\ \quad h\#262$ 

# format\_value()

Return the formatted (interpreted) data according to data\_type.

```
get_data()
     get_data_type()
     get_data_type_string()
     get_data_value()
     is reference()
          Returns True if the Res_value is actually a reference to another resource
class androguard.core.bytecodes.axml.ARSCResTableConfig(buff=None, **kwargs)
     Bases: object
     ARSCResTableConfig contains the configuration for specific resource selection. This is used on the device to
     determine which resources should be loaded based on different properties of the device like locale or displaysize.
     See the definition of ResTable_config in http://androidxref.com/9.0.0_r3/xref/frameworks/base/libs/androidfw/
     include/androidfw/ResourceTypes.h#911
     classmethod default_config()
     get config name friendly()
          Here for legacy reasons.
          use get_qualifier() instead.
     get_country()
     get_density()
     get_language()
     get_language_and_region()
          Returns the combined language+region string or for the default locale :return:
     get qualifier()
          Return resource name qualifier for the current configuration. for example * ldpi-v4 * hdpi-v4
          All possible qualifiers are listed in table 2 of https://developer.android.com/guide/topics/resources/
          providing-resources
          ..todo:: This name might not have all properties set! Therefore returned values might not reflect the true
          qualifier name! :return: str
     is default()
          Test if this is a default resource, which matches all
          This is indicated that all fields are zero. :return: True if default, False otherwise
class androquard.core.bytecodes.axml.ARSCResTableEntry(buff, mResId, parent=None)
     Bases: object
     A ResTable_entry.
     See http://androidxref.com/9.0.0_r3/xref/frameworks/base/libs/androidfw/include/androidfw/ResourceTypes.
     h#1458
     FLAG_COMPLEX = 1
     FLAG PUBLIC = 2
     FLAG_WEAK = 4
     get_index()
     get_key_data()
```

```
get_value()
     is_complex()
     is_public()
     is_weak()
class androquard.core.bytecodes.axml.ARSCResTablePackage (buff, header)
     Bases: object
     A ResTable_package
     See http://androidxref.com/9.0.0_r3/xref/frameworks/base/libs/androidfw/include/androidfw/ResourceTypes.
     h#861
     get_name()
class androguard.core.bytecodes.axml.ARSCResType (buff, parent=None)
     Bases: object
     This is a ResTable_type without it's ResChunk_header. It contains a ResTable_config
     See http://androidxref.com/9.0.0 r3/xref/frameworks/base/libs/androidfw/include/androidfw/ResourceTypes.
     h#1364
     get_package_name()
     get_type()
class androguard.core.bytecodes.axml.ARSCResTypeSpec(buff, parent=None)
     Bases: object
     See http://androidxref.com/9.0.0_r3/xref/frameworks/base/libs/androidfw/include/androidfw/ResourceTypes.
     h#1327
class androquard.core.bytecodes.axml.AXMLParser(raw_buff)
```

AXMLParser reads through all chunks in the AXML file and implements a state machine to return information about the current chunk, which can then be read by <code>AXMLPrinter</code>.

An AXML file is a file which contains multiple chunks of data, defined by the *ResChunk\_header*. There is no real file magic but as the size of the first header is fixed and the *type* of the *ResChunk\_header* is set to *RES\_XML\_TYPE*, a file will usually start with *0x03000800*. But there are several examples where the *type* is set to something else, probably in order to fool parsers.

Typically the AXMLParser is used in a loop which terminates if  $m\_event$  is set to  $END\_DOCUMENT$ . You can use the next() function to get the next chunk. Note that not all chunk types are yielded from the iterator! Some chunks are processed in the AXMLParser only. The parser will set  $is\_valid()$  to False if it parses something not valid. Messages what is wrong are logged.

See http://androidxref.com/9.0.0\_r3/xref/frameworks/base/libs/androidfw/include/androidfw/ResourceTypes. h#563

# property comment

Bases: object

Return the comment at the current position or None if no comment is given

This works only for Tags, as the comments of Namespaces are silently dropped. Currently, there is no way of retrieving comments of namespaces.

```
getAttributeCount()
```

Return the number of Attributes for a Tag or -1 if not in a tag

### getAttributeName (index)

Returns the String which represents the attribute name

# getAttributeNamespace (index)

Return the Namespace URI (if any) for the attribute

### getAttributeUri (index)

Returns the numeric ID for the namespace URI of an attribute

### getAttributeValue(index)

This function is only used to look up strings All other work is done by <code>format\_value()</code> # FIXME should unite those functions :param index: index of the attribute :return:

# getAttributeValueData(index)

Return the data of the attribute at the given index

**Parameters** index – index of the attribute

## getAttributeValueType (index)

Return the type of the attribute at the given index

**Parameters** index – index of the attribute

## getName()

Legacy only! use name instead

# getPrefix()

Legacy only! use namespace instead

### getText()

Legacy only! use text instead

### is\_valid(

Get the state of the AXMLPrinter. if an error happend somewhere in the process of parsing the file, this flag is set to False.

# property name

Return the String assosciated with the tag name

# property namespace

Return the Namespace URI (if any) as a String for the current tag

### property nsmap

Returns the current namespace mapping as a dictionary

there are several problems with the map and we try to guess a few things here:

- 1) a URI can be mapped by many prefixes, so it is to decide which one to take
- 2) a prefix might map to an empty string (some packers)
- 3) uri+prefix mappings might be included several times
- 4) prefix might be empty

# property text

Return the String assosicated with the current text

```
class androguard.core.bytecodes.axml.AXMLPrinter(raw_buff)
```

Bases: object

Converter for AXML Files into a lxml ElementTree, which can easily be converted into XML.

```
get_buff()
          Returns the raw XML file without prettification applied.
              Returns bytes, encoded as UTF-8
     get_xml (pretty=True)
          Get the XML as an UTF-8 string
              Returns bytes encoded as UTF-8
     get_xml_obj()
          Get the XML as an ElementTree object
              Returns lxml.etree.Element
     is_packed()
          Returns True if the AXML is likely to be packed
          Packers do some weird stuff and we try to detect it. Sometimes the files are not packed but simply broken
          or compiled with some broken version of a tool. Some file corruption might also be appear to be a packed
          file.
              Returns True if packer detected, False otherwise
     is_valid()
          Return the state of the AXMLParser. If this flag is set to False, the parsing has failed, thus the resulting
          XML will not work or will even be empty.
class androquard.core.bytecodes.axml.PackageContext(current_package,
                                                                                             string-
                                                                     pool_main,
                                                                                      mTableStrings,
                                                                     mKeyStrings)
     Bases: object
     get_mResId()
     get_package_name()
     set_mResId (mResId)
exception androquard.core.bytecodes.axml.ResParserError
     Bases: Exception
     Exception for the parsers
class androguard.core.bytecodes.axml.StringBlock(buff, header)
     Bases: object
     StringBlock is a CHUNK inside an AXML File: ResStringPool_header It contains all strings, which are used
     by referecing to ID's
     See http://androidxref.com/9.0.0_r3/xref/frameworks/base/libs/androidfw/include/androidfw/ResourceTypes.
     h#436
     getString(idx)
          Return the string at the index in the string table
              Parameters idx – index in the string table
              Returns str
     getStyle(idx)
          Return the style associated with the index
              Parameters idx – index of the style
              Returns
```

```
show()
```

Print some information on stdout about the string table

```
androguard.core.bytecodes.axml.complexToFloat(xcomplex)
```

Convert a complex unit into float

Format a value based on type and data. By default, no strings are looked up and "<string>" is returned. You need to define *lookup\_string* in order to actually lookup strings from the string table.

#### **Parameters**

- \_type The numeric type of the value
- \_data The numeric data of the value
- **lookup\_string** A function how to resolve strings from integer IDs

```
androguard.core.bytecodes.axml.get_arsc_info(arscobj)
```

Return a string containing all resources packages ordered by packagename, locale and type.

```
Parameters arscobj - ARSCParser
```

Returns a string

# androguard.core.mutf8 module

```
class androguard.core.mutf8.MUTF8String(b)
```

Bases: bytes

```
find(sub[, start[, end]]) \rightarrow int
```

Return the lowest index in B where subsection sub is found, such that sub is contained within B[start,end]. Optional arguments start and end are interpreted as in slice notation.

Return -1 on failure.

```
classmethod from str(s)
```

```
classmethod join (data, spacing=b")
```

Concatenate any number of bytes objects.

The bytes whose method is called is inserted in between each pair.

The result is returned as a new bytes object.

```
Example: b'.'.join([b'ab', b'pq', b'rs']) -> b'ab.pq.rs'.
```

# lstrip(sub)

Strip leading bytes contained in the argument.

If the argument is omitted or None, strip leading ASCII whitespace.

```
replace (old, new, count=None)
```

Return a copy with all occurrences of substring old replaced by new.

count Maximum number of occurrences to replace. -1 (the default value) means replace all occurrences.

If the optional argument count is given, only the first count occurrences are replaced.

```
rsplit (sep=None, maxsplit=-1)
```

Return a list of the sections in the bytes, using sep as the delimiter.

**sep** The delimiter according which to split the bytes. None (the default value) means split on ASCII whitespace characters (space, tab, return, newline, formfeed, vertical tab).

maxsplit Maximum number of splits to do. -1 (the default value) means no limit.

Splitting is done starting at the end of the bytes and working to the front.

```
split (sep=None, maxsplit=-1)
```

Return a list of the sections in the bytes, using sep as the delimiter.

**sep** The delimiter according which to split the bytes. None (the default value) means split on ASCII whitespace characters (space, tab, return, newline, formfeed, vertical tab).

maxsplit Maximum number of splits to do. -1 (the default value) means no limit.

```
startswith(prefix[,start[,end]]) \rightarrow bool
```

Return True if B starts with the specified prefix, False otherwise. With optional start, test B beginning at that position. With optional end, stop comparing B at that position. prefix can also be a tuple of bytes to try.

```
androguard.core.mutf8.decode(b) androguard.core.mutf8.encode(s)
```

## **Module contents**

# androguard.core.resources package

## **Submodules**

# androguard.core.resources.public module

### Module contents

### **Submodules**

# androguard.core.androconf module

```
class androguard.core.androconf.Color
    Bases: object

Black = '\x1b[30m'

Blue = '\x1b[34m'

Bold = '\x1b[1m'

Cyan = '\x1b[36m'

Green = '\x1b[32m'

Grey = '\x1b[37m'

Normal = '\x1b[0m'

Purple = '\x1b[35m'

Red = '\x1b[31m'

Yellow = '\x1b[33m'
```

```
class androquard.core.androconf.Configuration
     Bases: object
     instance = {'BIN_DED': 'ded.sh', 'BIN_DEX2JAR': 'dex2jar.sh', 'BIN_FERNFLOWER': 'fernf
exception androguard.core.androconf.InvalidResourceError
     Bases: Exception
     Invalid Resource Erorr is thrown by load_api_specific_resource_module
androguard.core.androconf.color_range(startcolor, goalcolor, steps)
     wrapper for interpolate_tuple that accepts colors as html ("#CCCCC" and such)
androguard.core.androconf.default_colors(obj)
androquard.core.androconf.disable_colors()
     Disable colors from the output (color = normal)
androguard.core.androconf.enable_colors(colors)
androquard.core.androconf.interpolate_tuple(startcolor, goalcolor, steps)
     Take two RGB color sets and mix them over a specified number of steps. Return the list
androguard.core.androconf.is_android(filename)
     Return the type of the file
     :param filename : the filename :returns: "APK", "DEX", None
androguard.core.androconf.is android raw(raw)
     Returns a string that describes the type of file, for common Android specific formats
androguard.core.androconf.is_ascii_problem(s)
     Test if a string contains other chars than ASCII
         Parameters s (MUTF8Strin) – a string to test
         Returns True if string contains other chars than ASCII, False otherwise
androquard.core.androconf.load_api_specific_resource_module (resource_name,
                                                                          api=None)
     Load the module from the JSON files and return a dict, which might be empty if the resource could not be
     If no api version is given, the default one from the CONF dict is used.
         Parameters
               • resource name - Name of the resource to load
               • api – API version
         Returns dict
androguard.core.androconf.make_color_tuple(color)
     turn something like "#000000" into 0,0,0 or "#FFFFFF into "255,255,255"
androguard.core.androconf.remove_colors()
     Remove colors from the output (no escape sequences)
androguard.core.androconf.rrmdir(directory)
     Recursivly delete a directory
         Parameters directory – directory to remove
androguard.core.androconf.save_colors()
```

```
androguard.core.androconf.set_options(key, value)
     Deprecated since version 3.3.5: Use CONF [key] = value instead
and roguard.core.and roconf.show_logging ( level=20 )
     enable log messages on stdout
     We will catch all messages here! From all loggers...
androguard.core.bytecode module
class androguard.core.bytecode.Buff(offset, buff)
     Bases: object
class androguard.core.bytecode.BuffHandle(buff)
     Bases: object
     BuffHandle is a wrapper around bytes. It gives the ability to jump in the byte stream, just like with BytesIO.
     add_idx(idx)
          Advance the current offset by idx
              Parameters idx(int) – number of bytes to advance
     end()
          Test if the current offset is at the end or over the buffer boundary
              Return type bool
     get buff()
          Return the whole buffer
              Return type bytearray
     get idx()
          Get the current offset in the buffer
              Return type int
     length_buff()
          Alias for size ()
     peek (size)
          Alias for read b()
     read(size)
          Read from the current offset a total number of size bytes and increment the offset by size
              Parameters size (int) – length of bytes to read
              Return type bytearray
     readNullString(size)
          Read a String with length size at the current offset
              Parameters size (int) - length of the string
              Return type bytearray
     read_at (offset, size)
          Read bytes from the given offset with length size without incrementing the current offset
              Parameters
                   • offset (int) – offset to start reading
```

```
• size (int) – length of bytes to read
                                   Return type bytearray
             read_b (size)
                         Read bytes with length size without incrementing the current offset
                                   Parameters size (int) – length to read in bytes
                                   Return type bytearray
             readat (off)
                         Read all bytes from the start of off until the end of the buffer
                         This method can be used to determine a checksum of a buffer from a given point on.
                                   Parameters off (int) – starting offset
                                   Return type bytearray
             save (filename)
                         Save the current buffer to filename
                         Exisiting files with the same name will be overwritten.
                                   Parameters filename (str) – the name of the file to save to
             set buff(buff)
                         Overwrite the current buffer with the content of buff
                                   Parameters buff (bytearray) – the new buffer
             set idx(idx)
                         Set the current offset in the buffer
                                   Parameters idx (int) - offset to set
             size()
                         Get the total size of the buffer
                                   Return type int
             tell()
                         Alias for get idx().
                                   Return type int
androguard.core.bytecode.FormatClassToJava(i)
             Transform a java class name into the typed variant found in DEX files.
             example:
             >>> FormatClassToJava('java.lang.Object')
              'Ljava/lang/Object;'
                         Parameters i – the input class name
                         Return type str
and rogular = 1 and rogula
             Transform a typed class name into a form which can be used as a python attribute
             example:
```

```
>>> FormatClassToPython('Lfoo/bar/foo/Barfoo$InnerClass;')
'Lfoo_bar_foo_Barfoo_InnerClass'
```

**Parameters** i – classname to transform

Return type str

androguard.core.bytecode.FormatDescriptorToPython(i)

Format a descriptor into a form which can be used as a python attribute

example:

```
>>> FormatDescriptorToPython('(Ljava/lang/Long; Ljava/lang/Long; Z Z)V')
'Ljava_lang_LongLjava_lang_LongZZV
```

**Parameters** i – name to transform

**Return type** str

```
androquard.core.bytecode.FormatNameToPython(i)
```

Transform a (method) name into a form which can be used as a python attribute

example:

```
>>> FormatNameToPython('<clinit>')
'clinit'
```

**Parameters** i – name to transform

Return type str

If no package could be found, the package is an empty string.

If the name is an array type, the array is discarded.

example:

```
>>> get_package_class_name('Ljava/lang/Object;')
('java.lang', 'Object')
>>> get_package_class_name('[[Ljava/lang/Object;')
('java.lang', 'Object')
```

(continues on next page)

(continued from previous page)

```
>>> get_package_class_name('LSomeClass;')
('', 'SomeClass')
```

Parameters name - the name

Return type tuple

Returns

androguard.core.bytecode.method2dot(mx, colors=None)

Export analysis method to dot format.

A control flow graph is created by using the concept of BasicBlocks. Each BasicBlock is a sequence of opcode without any jumps or branch.

#### **Parameters**

- mx MethodAnalysis
- colors dict of colors to use, if colors is None the default colors are used

**Returns** a string which contains the dot graph

```
androguard.core.bytecode.method2format(output,_format='png', mx=None, raw=None)
```

Export method structure as a graph to a specific file format using dot from the graphviz package. The result is written to the file specified via output.

There are two possibilites to give input for this method:

1) use raw argument and pass a dictionary containing the keys name, nodes and edges. This can be created using method2dot(). 2) give a MethodAnalysis.

This function requires pydot!

There is a special format raw which saves the dot buffer before it is handled by pydot.

## **Parameters**

- output (str) output filename
- **\_format** (*str*) format type (png, jpg ...). Can use all formats which are understood by pydot.
- mx (androguard.core.analysis.analysis.MethodAnalysis) specify the MethodAnalysis object
- raw (dict) use directly a dot raw buffer if None

androguard.core.bytecode.method2jpg(output, mx, raw=False)

Export method to a jpg file format

# **Parameters**

- **output** (*string*) **output** filename
- mx (MethodAnalysis object) specify the MethodAnalysis object
- raw (string) use directly a dot raw buffer (optional)

 $\verb"androguard.core.bytecode.method2json" (\textit{mx}, \textit{directed\_graph} = \textit{False})$ 

Create directed or undirected graph in the json format.

# **Parameters**

• mx - MethodAnalysis

```
• directed_graph – True if a directed graph should be created (default: False)
         Returns
androguard.core.bytecode.method2json_direct(mx)
         Parameters mx - MethodAnalysis
         Returns
androguard.core.bytecode.method2json_undirect(mx)
         Parameters mx - MethodAnalysis
         Returns
androguard.core.bytecode.method2png(output, mx, raw=False)
     Export method to a png file format
         Parameters
               • output (string) – output filename
               • mx (MethodAnalysis object) - specify the MethodAnalysis object
               • raw (string) – use directly a dot raw buffer
androguard.core.bytecode.object_to_bytes(obj)
     Convert a object to a bytearray or call get_raw() of the object if no useful type was found.
androquard.core.bytecode.vm2json(vm)
     Get a JSON representation of a DEX file
         Parameters vm - DalvikVMFormat
         Returns
Module contents
androguard.decompiler package
Subpackages
androguard.decompiler.dad package
Submodules
androguard.decompiler.dad.dast module
This file is a simplified version of writer.py that outputs an AST instead of source code.
class androguard.decompiler.dad.dast.JSONWriter(graph, method)
     Bases: object
     add(val)
     get_ast()
     get_cond (node)
     visit_cond_node(cond)
```

visit\_ins(op)

```
visit loop node (loop)
    visit node (node)
    visit_return_node(ret)
    visit_statement_node(stmt)
    visit switch node(switch)
    visit throw node(throw)
    visit_try_node (try_node)
androquard.decompiler.dad.dast.array_access(arr, ind)
androquard.decompiler.dad.dast.array_creation(tn, params, dim)
androquard.decompiler.dad.dast.array_initializer(params, tn=None)
androquard.decompiler.dad.dast.assignment(lhs, rhs, op=")
androquard.decompiler.dad.dast.binary_infix(op, left, right)
androguard.decompiler.dad.dast.cast(tn, arg)
androguard.decompiler.dad.dast.dummy(*args)
androguard.decompiler.dad.dast.expression_stmt (expr)
androquard.decompiler.dad.dast.field access (triple, left)
androguard.decompiler.dad.dast.if_stmt(cond_expr, scopes)
androguard.decompiler.dad.dast.jump_stmt(keyword)
androquard.decompiler.dad.dast.literal(result, tt)
androguard.decompiler.dad.dast.literal bool(b)
androquard.decompiler.dad.dast.literal_class(desc)
androguard.decompiler.dad.dast.literal_double(f)
androguard.decompiler.dad.dast.literal_float(f)
androquard.decompiler.dad.dast.literal hex int(b)
androquard.decompiler.dad.dast.literal int(b)
androguard.decompiler.dad.dast.literal_long(b)
androguard.decompiler.dad.dast.literal_null()
androquard.decompiler.dad.dast.literal string(s)
androguard.decompiler.dad.dast.local(name)
androquard.decompiler.dad.dast.local_decl_stmt(expr, decl)
androquard.decompiler.dad.dast.loop_stmt(isdo,cond_expr,body)
androquard.decompiler.dad.dast.method_invocation(triple, name, base, params)
androquard.decompiler.dad.dast.parenthesis(expr)
androguard.decompiler.dad.dast.parse_descriptor(desc)
androguard.decompiler.dad.dast.return_stmt(expr)
androguard.decompiler.dad.dast.statement_block()
```

```
androquard.decompiler.dad.dast.switch_stmt(cond_expr, ksv_pairs)
androguard.decompiler.dad.dast.throw_stmt(expr)
androguard.decompiler.dad.dast.try_stmt(tryb, pairs)
androguard.decompiler.dad.dast.typen(baset, dim)
androquard.decompiler.dad.dast.unary postfix(left, op)
androguard.decompiler.dad.dast.unary_prefix(op, left)
androguard.decompiler.dad.dast.var_decl(typen, var)
androguard.decompiler.dad.dast.visit_arr_data(value)
androguard.decompiler.dad.dast.visit_decl(var, init_expr=None)
androguard.decompiler.dad.dast.visit_expr(op)
androguard.decompiler.dad.dast.visit_ins(op, isCtor=False)
androquard.decompiler.dad.dast.write_inplace_if_possible(lhs, rhs)
androguard.decompiler.dad.basic_blocks module
class androguard.decompiler.dad.basic_blocks.BasicBlock (name, block_ins)
    Bases: androguard.decompiler.dad.node.Node
    add_ins (new_ins_list)
    add_variable_declaration(variable)
    get_ins()
    get_loc_with_ins()
    number_ins(num)
    remove_ins (loc, ins)
    set_catch_type (_type)
class androquard.decompiler.dad.basic blocks.CatchBlock (node)
    Bases: androguard.decompiler.dad.basic_blocks.BasicBlock
    visit (visitor)
    visit_exception(visitor)
class androquard.decompiler.dad.basic blocks.CondBlock (name, block ins)
    Bases: androguard.decompiler.dad.basic_blocks.BasicBlock
    neg()
    update_attribute_with(n_map)
    visit (visitor)
    visit_cond(visitor)
class androguard.decompiler.dad.basic_blocks.Condition(cond1, cond2, isand, isnot)
    Bases: object
    get ins()
    get_loc_with_ins()
```

```
neg()
    visit (visitor)
class androguard.decompiler.dad.basic_blocks.LoopBlock(name, cond)
    Bases: androguard.decompiler.dad.basic_blocks.CondBlock
    get ins()
    get_loc_with_ins()
    neg()
    update_attribute_with(n_map)
    visit (visitor)
    visit_cond(visitor)
class androguard.decompiler.dad.basic_blocks.ReturnBlock(name, block_ins)
    Bases: androguard.decompiler.dad.basic_blocks.BasicBlock
    visit (visitor)
class androguard.decompiler.dad.basic_blocks.ShortCircuitBlock(name, cond)
    Bases: androguard.decompiler.dad.basic_blocks.CondBlock
    get_ins()
    get loc with ins()
    neg()
    visit_cond(visitor)
class androguard.decompiler.dad.basic_blocks.StatementBlock(name, block_ins)
    Bases: androquard.decompiler.dad.basic_blocks.BasicBlock
    visit (visitor)
class androguard.decompiler.dad.basic_blocks.SwitchBlock(name, switch, block_ins)
    Bases: androquard.decompiler.dad.basic_blocks.BasicBlock
    add_case (case)
    copy_from (node)
    order_cases()
    update_attribute_with(n_map)
    visit (visitor)
class androguard.decompiler.dad.basic_blocks.ThrowBlock (name, block_ins)
    Bases: androguard.decompiler.dad.basic_blocks.BasicBlock
    visit (visitor)
class androguard.decompiler.dad.basic_blocks.TryBlock(node)
    Bases: androquard.decompiler.dad.basic_blocks.BasicBlock
    add_catch_node (node)
    property num
    visit (visitor)
```

```
androquard.decompiler.dad.basic blocks.build node from block (block,
                                                                                  vmap,
                                                                       gen ret,
                                                                                 excep-
                                                                       tion type=None)
androguard.decompiler.dad.control flow module
androguard.decompiler.dad.control_flow.catch_struct(graph, idoms)
androguard.decompiler.dad.control_flow.derived_sequence(graph)
    Compute the derived sequence of the graph G The intervals of G are collapsed into nodes, intervals of these
    nodes are built, and the process is repeated iteratively until we obtain a single node (if the graph is not irre-
    ducible)
androguard.decompiler.dad.control_flow.identify_structures(graph, idoms)
androguard.decompiler.dad.control_flow.if_struct(graph, idoms)
androguard.decompiler.dad.control_flow.intervals(graph)
    Compute the intervals of the graph Returns interval_graph: a graph of the intervals of G interv_heads: a dict of
    (header node, interval)
androquard.decompiler.dad.control_flow.loop_follow(start, end, nodes_in_loop)
androquard.decompiler.dad.control_flow.loop_struct(graphs_list, intervals_list)
androguard.decompiler.dad.control_flow.loop_type(start, end, nodes_in_loop)
androguard.decompiler.dad.control_flow.mark_loop(graph, start, end, interval)
androguard.decompiler.dad.control_flow.mark_loop_rec(graph, node, s_num, e_num, in-
                                                             terval, nodes_in_loop)
androguard.decompiler.dad.control_flow.short_circuit_struct(graph,
                                                                                  idom,
                                                                      node_map)
androguard.decompiler.dad.control_flow.switch_struct(graph, idoms)
androguard.decompiler.dad.control_flow.update_dom(idoms, node_map)
androquard.decompiler.dad.control_flow.while_block_struct(graph, node_map)
androguard.decompiler.dad.dataflow module
class androquard.decompiler.dad.dataflow.BasicReachDef(graph, params)
    Bases: object
    run()
class androguard.decompiler.dad.dataflow.DummyNode(name)
    Bases: androguard.decompiler.dad.node.Node
    get loc with ins()
androguard.decompiler.dad.dataflow.build_def_use(graph, lparams)
    Builds the Def-Use and Use-Def (DU/UD) chains of the variables of the method.
```

androguard.decompiler.dad.dataflow.clear\_path\_node(graph, reg, loc1, loc2)

Check that the path from loc1 to loc2 is clear. We have to check that there is no side effect between the two location points. We also have to check that the variable *reg* is not redefined along one of the possible pathes

androguard.decompiler.dad.dataflow.clear\_path(graph, reg, loc1, loc2)

from loc1 to loc2.

```
androquard.decompiler.dad.dataflow.dead_code_elimination(graph, du, ud)
```

Run a dead code elimination pass. Instructions are checked to be dead. If it is the case, we remove them and we update the DU & UD chains of its variables to check for further dead instructions.

```
androguard.decompiler.dad.dataflow.group_variables(lvars, DU, UD) androguard.decompiler.dad.dataflow.place_declarations(graph, dvars, du, ud) androguard.decompiler.dad.dataflow.reach_def_analysis(graph, lparams) androguard.decompiler.dad.dataflow.register_propagation(graph, du, ud)
```

Propagate the temporary registers between instructions and remove them if necessary. We process the nodes of the graph in reverse post order. For each instruction in the node, we look at the variables that it uses. For each of these variables we look where it is defined and if we can replace it with its definition. We have to be careful to the side effects some instructions may have. To do the propagation, we use the computed DU and UD chains.

```
androguard.decompiler.dad.dataflow.split_variables(graph, lvars, DU, UD) androguard.decompiler.dad.dataflow.update_chain(graph, loc, du, ud)
```

Updates the DU chain of the instruction located at loc such that there is no more reference to it so that we can remove it. When an instruction is found to be dead (i.e it has no side effect, and the register defined is not used) we have to update the DU chain of all the variables that may me used by the dead instruction.

# androguard.decompiler.dad.decompile module

```
class androguard.decompiler.dad.decompile.DvClass(dvclass, vma)
    Bases: object
```

This is a wrapper for ClassDefItem inside the decompiler.

At first, methods contains a list of EncodedMethods, which are successively replaced by DvMethod in the process of decompilation.

```
get_ast()
get_methods()
get_source()
get_source_ext()
process(doAST=False)
process_method(num, doAST=False)
show_source()
class androguard.decompiler.dad.decompile.DvMachine(name)
Bases: object
```

Wrapper class for a Dalvik Object, like a DEX or ODEX file.

The wrapper allows to take a Dalvik file and get a list of Classes out of it. The <code>DvMachine</code> can take either an APK file directly, where all DEX files from the multidex are used, or a single DEX or ODEX file as an argument.

At first, classes contains only ClassDefItem as values. Then these objects are replaced by DvClass items successively.

```
get_ast()
```

Processes each class with AST enabled and returns a dictionary with all single ASTs Classnames as keys.

Returns an dictionary for all classes

Return type dict

```
get_class(class_name)
         Return the DvClass with the given name
         The name is partially matched against the known class names and the first result is returned. For example,
         the input foobar will match on Lfoobar/bla/foo;
             Parameters class_name (str)-
             Returns the class matching on the name
             Return type DvClass
     get_classes()
         Return a list of classnames contained in this machine. The format of each name is Lxxx;
             Returns list of class names
     process()
         Process all classes inside the machine.
         This calls process () on each DvClass.
     process and show()
         Run process () and show_source () after each other.
     show_source()
         Calls show_source on all classes inside the machine. This prints the source to stdout.
         This calls show source () on each DvClass.
class androguard.decompiler.dad.decompile.DvMethod(methanalysis)
     Bases: object
     This is a wrapper around MethodAnalysis and EncodedMethod inside the decompiler.
     get_ast()
     get_source()
     get_source_ext()
     process (doAST=False)
     show source()
androguard.decompiler.dad.decompile.get_field_ast (field)
androguard.decompiler.dad.decompile.main()
androguard.decompiler.dad.graph module
class androguard.decompiler.dad.graph.GenInvokeRetName
     Bases: object
     last()
     new()
     set_to(ret)
class androguard.decompiler.dad.graph.Graph
     Bases: object
     Stores a CFG (Control Flow Graph), which is a directed graph.
```

```
The CFG defines an entry node entry, a single exit node exit, a list of nodes nodes and a list of edges
     edges.
     add_catch_edge(e1, e2)
     add_edge(e1, e2)
     add node (node)
          Adds the given node to the graph, without connecting it to anything else.
              Parameters node (androguard.decompiler.dad.node.Node) - node to add
     all_preds (node)
     all_sucs (node)
     compute_rpo()
          Number the nodes in reverse post order. An RPO traversal visit as many predecessors of a node as possible
          before visiting the node itself.
     draw (name, dname, draw_branches=True)
          Writes the current graph as a PNG file
              Parameters
                  • name (str) – filename (without .png)
                  • dname (str) – directory of the output png
                  • draw branches -
              Returns
     \texttt{get\_ins\_from\_loc}(loc)
     get_node_from_loc(loc)
     immediate_dominators()
     number_ins()
     post_order()
          Yields the :class'~androguard.decompiler.dad.node.Node's of the graph in post-order i.e we visit all the
          children of a node before visiting the node itself.
     preds (node)
     remove\_ins(loc)
     remove\_node(node)
          Remove the node from the graph, removes also all connections.
              Parameters node (androquard.decompiler.dad.node.Node) - the node to remove
     sucs (node)
androguard.decompiler.dad.graph.bfs (start)
     Breadth first search
     Yields all nodes found from the starting point
          Parameters start – start node
androguard.decompiler.dad.graph.construct(start_block, vmap, exceptions)
     Constructs a CFG
```

## **Parameters**

- start\_block (androguard.core.analysis.analysis.DVMBasicBlock) The startpoint
- vmap variable mapping
- exceptions list of androguard.core.analysis.analysis.ExceptionAnalysis

### **Return type** *Graph*

```
androguard.decompiler.dad.graph.dom_lt(graph)

Dominator algorithm from Lengauer-Tarjan
```

```
androguard.decompiler.dad.graph.make_node(graph, block, block_to_node, vmap, gen_ret)
```

```
androquard.decompiler.dad.graph.simplify(graph)
```

Simplify the CFG by merging/deleting statement nodes when possible: If statement B follows statement A and if B has no other predecessor besides A, then we can merge A and B into a new statement node. We also remove nodes which do nothing except redirecting the control flow (nodes which only contains a goto).

```
androguard.decompiler.dad.graph.split_if_nodes(graph)
```

Split IfNodes in two nodes, the first node is the header node, the second one is only composed of the jump condition.

# androguard.decompiler.dad.instruction module

```
class androquard.decompiler.dad.instruction.ArrayExpression
    Bases: androguard.decompiler.dad.instruction.IRForm
class androguard.decompiler.dad.instruction.ArrayLengthExpression(array)
    Bases: androguard.decompiler.dad.instruction.ArrayExpression
    get_type()
    get_used_vars()
    replace (old, new)
    replace_var (old, new)
    visit (visitor)
class androguard.decompiler.dad.instruction.ArrayLoadExpression(arg,
                                                                            index.
                                                                      _type)
    Bases: androquard.decompiler.dad.instruction.ArrayExpression
    get_type()
    get_used_vars()
    replace (old, new)
    replace_var(old, new)
    visit (visitor)
class androguard.decompiler.dad.instruction.ArrayStoreInstruction(rhs,
                                                                              ar-
                                                                        ray, index,
                                                                        _type)
    Bases: androguard.decompiler.dad.instruction.IRForm
    get used vars()
    has_side_effect()
    replace(old, new)
```

```
replace_var(old, new)
    visit (visitor)
class androguard.decompiler.dad.instruction.AssignExpression(lhs, rhs)
    Bases: androguard.decompiler.dad.instruction.IRForm
    get lhs()
    get_rhs()
    get_used_vars()
    has_side_effect()
    is call()
    is_propagable()
    remove_defined_var()
    replace (old, new)
    replace_lhs (new)
    replace_var(old, new)
    visit (visitor)
class androguard.decompiler.dad.instruction.BaseClass(name, descriptor=None)
    Bases: androquard.decompiler.dad.instruction.IRForm
    is const()
    visit (visitor)
class androguard.decompiler.dad.instruction.BinaryCompExpression(op,
                                                                             arg1,
                                                                       arg2, _type)
    Bases: androguard.decompiler.dad.instruction.BinaryExpression
    visit (visitor)
class androquard.decompiler.dad.instruction.BinaryExpression(op, arg1,
                                                                             arg2,
                                                                  _type)
    Bases: androquard.decompiler.dad.instruction.IRForm
    get_used_vars()
    has_side_effect()
    replace (old, new)
    replace_var (old, new)
    visit (visitor)
class androguard.decompiler.dad.instruction.BinaryExpression2Addr(op,
                                                                             dest.
                                                                        arg, _type)
    Bases: androguard.decompiler.dad.instruction.BinaryExpression
class androquard.decompiler.dad.instruction.BinaryExpressionLit (op,
                                                                             arg1,
    Bases: androguard.decompiler.dad.instruction.BinaryExpression
class androquard.decompiler.dad.instruction.CastExpression(op, atype, arg)
    Bases: androguard.decompiler.dad.instruction.UnaryExpression
    get_type()
```

```
get_used_vars()
    is_const()
    visit (visitor)
class androquard.decompiler.dad.instruction.CheckCastExpression(arg,
                                                                              \_type,
                                                                       descrip-
                                                                       tor=None)
    Bases: androguard.decompiler.dad.instruction.IRForm
    get_used_vars()
    is const()
    replace (old, new)
    replace_var (old, new)
    visit (visitor)
class androguard.decompiler.dad.instruction.ConditionalExpression(op,
    Bases: androquard.decompiler.dad.instruction.IRForm
    get_lhs()
    get_used_vars()
    is cond()
    neg()
    replace (old, new)
    replace_var (old, new)
    visit (visitor)
class androguard.decompiler.dad.instruction.ConditionalZExpression(op, arg)
    Bases: androguard.decompiler.dad.instruction.IRForm
    get_lhs()
    get_used_vars()
    is_cond()
    neg()
    replace (old, new)
    replace_var (old, new)
    visit (visitor)
class androguard.decompiler.dad.instruction.Constant (value, atype, int_value=None,
                                                           descriptor=None)
    Bases: androquard.decompiler.dad.instruction.IRForm
    get_int_value()
    get_type()
    get_used_vars()
    is const()
    visit (visitor)
```

```
class androquard.decompiler.dad.instruction.FillArrayExpression (reg, value)
    Bases: androguard.decompiler.dad.instruction.ArrayExpression
    get_rhs()
    get_used_vars()
    is_propagable()
    replace (old, new)
    replace_var (old, new)
    visit (visitor)
class androguard.decompiler.dad.instruction.FilledArrayExpression (asize, atype,
                                                                         args)
    Bases: androquard.decompiler.dad.instruction.ArrayExpression
    get_used_vars()
    replace (old, new)
    replace_var (old, new)
    visit (visitor)
class androguard.decompiler.dad.instruction.IRForm
    Bases: object
    get_lhs()
    get_rhs()
    get_type()
    get_used_vars()
    has_side_effect()
    is_call()
    is cond()
    is const()
    is_ident()
    is_propagable()
    remove_defined_var()
    replace (old, new)
    replace_lhs (new)
    replace_var(old, new)
    set_type (_type)
    visit (visitor)
class androguard.decompiler.dad.instruction.InstanceExpression(arg, klass, ftype,
                                                                     name)
    Bases: androquard.decompiler.dad.instruction.IRForm
    get_type()
    get_used_vars()
```

```
replace (old, new)
    replace_var(old, new)
    visit (visitor)
class androguard.decompiler.dad.instruction.InstanceInstruction (rhs, lhs, klass,
                                                                        atype, name)
    Bases: androguard.decompiler.dad.instruction.IRForm
    get_lhs()
    get_used_vars()
    has_side_effect()
    replace (old, new)
    replace_var (old, new)
    visit (visitor)
class androguard.decompiler.dad.instruction.InvokeDirectInstruction(clsname,
                                                                             name,
                                                                             base,
                                                                             rtype,
                                                                             ptype,
                                                                             args,
                                                                             triple)
    Bases: androquard.decompiler.dad.instruction.InvokeInstruction
class androguard.decompiler.dad.instruction.InvokeInstruction(clsname, name,
                                                                      base, rtype, ptype,
                                                                      args, triple)
    Bases: androguard.decompiler.dad.instruction.IRForm
    get_type()
    get_used_vars()
    has_side_effect()
    is_call()
    replace (old, new)
    replace_var (old, new)
    visit (visitor)
class androguard.decompiler.dad.instruction.InvokeRangeInstruction(clsname,
                                                                            name,
                                                                            rtype,
                                                                            ptype,
                                                                            args,
                                                                            triple)
    Bases: androquard.decompiler.dad.instruction.InvokeInstruction
class androguard.decompiler.dad.instruction.InvokeStaticInstruction(clsname,
                                                                             name,
                                                                             base,
                                                                             rtype,
                                                                             ptype,
                                                                             args,
                                                                             triple)
```

```
Bases: androguard.decompiler.dad.instruction.InvokeInstruction
    get_used_vars()
{f class} and {f coup} and {f coup} ider. dad. instruction. Monitor {f Ent} {f coup} respectively.
    Bases: androguard.decompiler.dad.instruction.RefExpression
class androguard.decompiler.dad.instruction.MonitorExitExpression(ref)
    Bases: androquard.decompiler.dad.instruction.RefExpression
    visit (visitor)
class androquard.decompiler.dad.instruction.MoveExceptionExpression(ref,
                                                                           _type)
    Bases: androguard.decompiler.dad.instruction.RefExpression
    get_lhs()
    get_used_vars()
    has_side_effect()
    replace lhs(new)
    visit (visitor)
class androquard.decompiler.dad.instruction.MoveExpression(lhs, rhs)
    Bases: androquard.decompiler.dad.instruction.IRForm
    get_lhs()
    get_rhs()
    get_used_vars()
    has_side_effect()
    is_call()
    replace (old, new)
    replace_lhs (new)
    replace_var (old, new)
    visit (visitor)
class androquard.decompiler.dad.instruction.MoveResultExpression(lhs, rhs)
    Bases: androguard.decompiler.dad.instruction.MoveExpression
    has_side_effect()
    is_propagable()
    visit (visitor)
class androquard.decompiler.dad.instruction.NewArrayExpression(asize, atype)
    Bases: androguard.decompiler.dad.instruction.ArrayExpression
    get_used_vars()
    is_propagable()
    replace (old, new)
    replace_var (old, new)
    visit (visitor)
```

```
class androquard.decompiler.dad.instruction.NewInstance(ins_type)
    Bases: androquard.decompiler.dad.instruction.IRForm
    get_type()
    get_used_vars()
    replace (old, new)
    visit (visitor)
class androguard.decompiler.dad.instruction.NopExpression
    Bases: androquard.decompiler.dad.instruction.IRForm
    get_lhs()
    get_used_vars()
    visit (visitor)
class androguard.decompiler.dad.instruction.Param(value, atype)
    Bases: androquard.decompiler.dad.instruction.Variable
    is_const()
    visit (visitor)
class androguard.decompiler.dad.instruction.RefExpression(ref)
    Bases: androguard.decompiler.dad.instruction.IRForm
    get_used_vars()
    is_propagable()
    replace (old, new)
    replace_var (old, new)
class androquard.decompiler.dad.instruction.ReturnInstruction(arg)
    Bases: androquard.decompiler.dad.instruction.IRForm
    get_lhs()
    get_used_vars()
    replace (old, new)
    replace_var (old, new)
    visit (visitor)
class androquard.decompiler.dad.instruction.StaticExpression(cls name,
                                                                  field_type,
                                                                  field name)
    Bases: androquard.decompiler.dad.instruction.IRForm
    get_type()
    replace (old, new)
    visit (visitor)
class androguard.decompiler.dad.instruction.StaticInstruction(rhs, klass, ftype,
                                                                    name)
    Bases: androguard.decompiler.dad.instruction.IRForm
    get_lhs()
    get_used_vars()
```

```
has side effect()
    replace (old, new)
    replace_var (old, new)
    visit (visitor)
class androquard.decompiler.dad.instruction.SwitchExpression(src, branch)
    Bases: androguard.decompiler.dad.instruction.IRForm
    get_used_vars()
    replace (old, new)
    replace_var(old, new)
    visit (visitor)
class androguard.decompiler.dad.instruction.ThisParam(value, atype)
    Bases: androguard.decompiler.dad.instruction.Param
    visit (visitor)
class androguard.decompiler.dad.instruction.ThrowExpression(ref)
    Bases: androguard.decompiler.dad.instruction.RefExpression
    visit (visitor)
class androquard.decompiler.dad.instruction.UnaryExpression(op, arg, type)
    Bases: androguard.decompiler.dad.instruction.IRForm
    get_type()
    get_used_vars()
    replace (old, new)
    replace_var (old, new)
    visit (visitor)
class androguard.decompiler.dad.instruction.Variable(value)
    Bases: androquard.decompiler.dad.instruction.IRForm
    get_used_vars()
    is_ident()
    value()
    visit (visitor)
    visit_decl (visitor)
androguard.decompiler.dad.node module
class androguard.decompiler.dad.node.Interval(head)
    Bases: object
    add_node (node)
    compute_end(graph)
    get_end()
    get_head()
```

```
class androguard.decompiler.dad.node.LoopType
    Bases: object
    copy()
    property is_endless
    property is_posttest
    property is_pretest
class androguard.decompiler.dad.node.MakeProperties(name, bases, dct)
    Bases: type
class androquard.decompiler.dad.node.Node(name)
    Bases: object
    copy_from (node)
    get_end()
    get_head()
    update_attribute_with(n_map)
class androguard.decompiler.dad.node.NodeType
    Bases: object
    copy()
    property is_cond
    property is_return
    property is_stmt
    property is_switch
    property is_throw
androguard.decompiler.dad.opcode_ins module
class androguard.decompiler.dad.opcode_ins.Op
    Bases: object
    ADD = '+'
    AND = '&'
    CMP = 'cmp'
    DIV = '/'
    EQUAL = '=='
    GEQUAL = '>='
    GREATER = '>'
    INTSHL = '<<'
    INTSHR = '>>'
    LEQUAL = '<='
    LONGSHL = '<<'
```

```
LONGSHR = '>>'
    LOWER = '<'
    MOD = '%'
    MUL = '*'
    NEG = '-'
    NEQUAL = '!='
    NOT = '~'
    OR = '|'
    SUB = '-'
    XOR = '^'
androquard.decompiler.dad.opcode_ins.adddouble(ins, vmap)
androquard.decompiler.dad.opcode_ins.adddouble2addr(ins, vmap)
androguard.decompiler.dad.opcode_ins.addfloat(ins, vmap)
androguard.decompiler.dad.opcode_ins.addfloat2addr(ins, vmap)
androguard.decompiler.dad.opcode_ins.addint(ins, vmap)
androquard.decompiler.dad.opcode ins.addint2addr(ins, vmap)
androquard.decompiler.dad.opcode ins.addintlit16(ins, vmap)
androguard.decompiler.dad.opcode_ins.addintlit8(ins, vmap)
androquard.decompiler.dad.opcode_ins.addlong(ins, vmap)
androquard.decompiler.dad.opcode_ins.addlong2addr(ins, vmap)
androquard.decompiler.dad.opcode_ins.aget (ins, vmap)
androquard.decompiler.dad.opcode_ins.agetboolean(ins, vmap)
androguard.decompiler.dad.opcode_ins.agetbyte(ins, vmap)
androquard.decompiler.dad.opcode ins.agetchar(ins, vmap)
androquard.decompiler.dad.opcode ins.aqetobject(ins, vmap)
androguard.decompiler.dad.opcode_ins.agetshort(ins, vmap)
androguard.decompiler.dad.opcode_ins.agetwide(ins, vmap)
androguard.decompiler.dad.opcode_ins.andint(ins, vmap)
androquard.decompiler.dad.opcode ins.andint2addr(ins, vmap)
androquard.decompiler.dad.opcode_ins.andintlit16(ins, vmap)
androquard.decompiler.dad.opcode_ins.andintlit8(ins, vmap)
androguard.decompiler.dad.opcode_ins.andlong(ins, vmap)
androquard.decompiler.dad.opcode_ins.andlong2addr(ins, vmap)
androguard.decompiler.dad.opcode_ins.aput (ins, vmap)
androguard.decompiler.dad.opcode_ins.aputboolean(ins, vmap)
androguard.decompiler.dad.opcode_ins.aputbyte(ins, vmap)
```

```
androquard.decompiler.dad.opcode ins.aputchar(ins, vmap)
androguard.decompiler.dad.opcode_ins.aputobject(ins, vmap)
androguard.decompiler.dad.opcode_ins.aputshort(ins, vmap)
androguard.decompiler.dad.opcode_ins.aputwide(ins, vmap)
androquard.decompiler.dad.opcode ins.arraylength(ins, vmap)
androquard.decompiler.dad.opcode ins.assign binary 2addr exp(ins,
                                                                          val op,
                                                                 op_type, vmap)
androguard.decompiler.dad.opcode_ins.assign_binary_exp(ins, val_op, op_type, vmap)
androquard.decompiler.dad.opcode_ins.assign_cast_exp(val_a, val_b, val_op, op_type,
                                                         vman)
androguard.decompiler.dad.opcode_ins.assign_cmp(val_a, val_b, val_c, cmp_type, vmap)
androquard.decompiler.dad.opcode_ins.assign_const(dest_reg, cst, vmap)
androguard.decompiler.dad.opcode_ins.assign_lit(op_type, val_cst, val_a, val_b, vmap)
androguard.decompiler.dad.opcode_ins.checkcast (ins, vmap)
androguard.decompiler.dad.opcode_ins.cmpqdouble(ins, vmap)
androquard.decompiler.dad.opcode ins.cmpqfloat(ins,vmap)
androguard.decompiler.dad.opcode_ins.cmpldouble(ins, vmap)
androquard.decompiler.dad.opcode_ins.cmplfloat(ins, vmap)
androguard.decompiler.dad.opcode_ins.cmplong(ins, vmap)
androquard.decompiler.dad.opcode ins.const (ins, vmap)
androquard.decompiler.dad.opcode ins.const16(ins, vmap)
androguard.decompiler.dad.opcode_ins.const4 (ins, vmap)
androguard.decompiler.dad.opcode_ins.constclass(ins, vmap)
androquard.decompiler.dad.opcode ins.consthigh16(ins, vmap)
androguard.decompiler.dad.opcode_ins.conststring(ins, vmap)
androquard.decompiler.dad.opcode_ins.conststringjumbo(ins, vmap)
androguard.decompiler.dad.opcode_ins.constwide(ins, vmap)
androquard.decompiler.dad.opcode ins.constwide16(ins, vmap)
androquard.decompiler.dad.opcode ins.constwide32(ins, vmap)
androguard.decompiler.dad.opcode_ins.constwidehigh16 (ins, vmap)
androquard.decompiler.dad.opcode ins.divdouble(ins, vmap)
androquard.decompiler.dad.opcode ins.divdouble2addr(ins, vmap)
androguard.decompiler.dad.opcode_ins.divfloat(ins, vmap)
androguard.decompiler.dad.opcode_ins.divfloat2addr(ins, vmap)
androquard.decompiler.dad.opcode_ins.divint(ins, vmap)
androguard.decompiler.dad.opcode_ins.divint2addr(ins, vmap)
androguard.decompiler.dad.opcode_ins.divintlit16(ins, vmap)
```

```
androquard.decompiler.dad.opcode ins.divintlit8(ins, vmap)
androguard.decompiler.dad.opcode_ins.divlong(ins, vmap)
androguard.decompiler.dad.opcode_ins.divlong2addr(ins, vmap)
androguard.decompiler.dad.opcode_ins.doubletofloat(ins, vmap)
androquard.decompiler.dad.opcode ins.doubletoint(ins, vmap)
androquard.decompiler.dad.opcode ins.doubletolong(ins, vmap)
androquard.decompiler.dad.opcode_ins.fillarraydata(ins, vmap, value)
androquard.decompiler.dad.opcode_ins.fillarraydatapayload(ins, vmap)
androquard.decompiler.dad.opcode_ins.fillednewarray(ins, vmap, ret)
androquard.decompiler.dad.opcode_ins.fillednewarrayrange(ins, vmap, ret)
androquard.decompiler.dad.opcode_ins.floattodouble(ins, vmap)
androquard.decompiler.dad.opcode_ins.floattoint(ins, vmap)
androguard.decompiler.dad.opcode_ins.floattolong(ins, vmap)
androquard.decompiler.dad.opcode_ins.get_args(vmap, param_type, largs)
androquard.decompiler.dad.opcode ins.qet variables(vmap, *variables)
androquard.decompiler.dad.opcode ins.goto(ins, vmap)
androquard.decompiler.dad.opcode ins.goto16 (ins, vmap)
androquard.decompiler.dad.opcode_ins.goto32 (ins, vmap)
androguard.decompiler.dad.opcode_ins.ifeq(ins, vmap)
androquard.decompiler.dad.opcode ins.ifeqz(ins, vmap)
androquard.decompiler.dad.opcode_ins.ifge (ins, vmap)
androguard.decompiler.dad.opcode_ins.ifgez(ins, vmap)
androguard.decompiler.dad.opcode_ins.ifgt (ins, vmap)
androquard.decompiler.dad.opcode ins.ifqtz(ins, vmap)
androquard.decompiler.dad.opcode ins.ifle(ins, vmap)
androguard.decompiler.dad.opcode_ins.iflez(ins, vmap)
androguard.decompiler.dad.opcode_ins.iflt(ins, vmap)
androquard.decompiler.dad.opcode ins.ifltz(ins, vmap)
androquard.decompiler.dad.opcode ins.ifne(ins, vmap)
androquard.decompiler.dad.opcode_ins.ifnez(ins, vmap)
androquard.decompiler.dad.opcode_ins.iget (ins, vmap)
androguard.decompiler.dad.opcode_ins.igetboolean(ins, vmap)
androquard.decompiler.dad.opcode_ins.igetbyte(ins, vmap)
androguard.decompiler.dad.opcode_ins.igetchar(ins, vmap)
androguard.decompiler.dad.opcode_ins.igetobject(ins, vmap)
androguard.decompiler.dad.opcode_ins.igetshort(ins, vmap)
```

```
androquard.decompiler.dad.opcode ins.igetwide(ins, vmap)
androquard.decompiler.dad.opcode ins.instanceof(ins, vmap)
androguard.decompiler.dad.opcode_ins.inttobyte(ins, vmap)
androguard.decompiler.dad.opcode_ins.inttochar(ins, vmap)
androquard.decompiler.dad.opcode ins.inttodouble(ins, vmap)
androquard.decompiler.dad.opcode ins.inttofloat(ins, vmap)
androquard.decompiler.dad.opcode_ins.inttolong(ins, vmap)
androguard.decompiler.dad.opcode_ins.inttoshort(ins, vmap)
androquard.decompiler.dad.opcode_ins.invokedirect(ins, vmap, ret)
androquard.decompiler.dad.opcode_ins.invokedirectrange(ins, vmap, ret)
androquard.decompiler.dad.opcode_ins.invokeinterface(ins, vmap, ret)
androquard.decompiler.dad.opcode_ins.invokeinterfacerange(ins, vmap, ret)
androguard.decompiler.dad.opcode_ins.invokestatic(ins, vmap, ret)
androguard.decompiler.dad.opcode_ins.invokestaticrange(ins, vmap, ret)
androguard.decompiler.dad.opcode_ins.invokesuper(ins, vmap, ret)
androquard.decompiler.dad.opcode ins.invokesuperrange(ins, vmap, ret)
androquard.decompiler.dad.opcode ins.invokevirtual(ins, vmap, ret)
androguard.decompiler.dad.opcode_ins.invokevirtualrange(ins, vmap, ret)
androguard.decompiler.dad.opcode_ins.iput (ins, vmap)
androquard.decompiler.dad.opcode ins.iputboolean(ins, vmap)
androguard.decompiler.dad.opcode_ins.iputbyte(ins, vmap)
androguard.decompiler.dad.opcode_ins.iputchar(ins, vmap)
androguard.decompiler.dad.opcode_ins.iputobject(ins, vmap)
androquard.decompiler.dad.opcode ins.iputshort(ins, vmap)
androquard.decompiler.dad.opcode ins.iputwide(ins, vmap)
androguard.decompiler.dad.opcode_ins.load_array_exp(val_a, val_b, val_c, ar_type,
androquard.decompiler.dad.opcode_ins.longtodouble(ins, vmap)
androquard.decompiler.dad.opcode_ins.longtofloat(ins, vmap)
androquard.decompiler.dad.opcode_ins.longtoint(ins, vmap)
androquard.decompiler.dad.opcode_ins.monitorenter(ins, vmap)
androguard.decompiler.dad.opcode_ins.monitorexit(ins, vmap)
androguard.decompiler.dad.opcode_ins.move(ins, vmap)
androquard.decompiler.dad.opcode ins.move16 (ins, vmap)
androguard.decompiler.dad.opcode_ins.moveexception(ins, vmap, _type)
androguard.decompiler.dad.opcode_ins.movefrom16 (ins, vmap)
androquard.decompiler.dad.opcode_ins.moveobject(ins, vmap)
```

```
androquard.decompiler.dad.opcode ins.moveobject16 (ins, vmap)
androguard.decompiler.dad.opcode_ins.moveobjectfrom16(ins, vmap)
androguard.decompiler.dad.opcode_ins.moveresult (ins, vmap, ret)
androguard.decompiler.dad.opcode_ins.moveresultobject(ins, vmap, ret)
androquard.decompiler.dad.opcode ins.moveresultwide(ins, vmap, ret)
androquard.decompiler.dad.opcode ins.movewide(ins, vmap)
androquard.decompiler.dad.opcode_ins.movewide16(ins, vmap)
androquard.decompiler.dad.opcode_ins.movewidefrom16(ins, vmap)
androquard.decompiler.dad.opcode_ins.muldouble(ins, vmap)
androquard.decompiler.dad.opcode_ins.muldouble2addr(ins, vmap)
androguard.decompiler.dad.opcode_ins.mulfloat(ins, vmap)
androquard.decompiler.dad.opcode_ins.mulfloat2addr(ins, vmap)
androguard.decompiler.dad.opcode_ins.mulint(ins, vmap)
androquard.decompiler.dad.opcode ins.mulint2addr(ins, vmap)
androguard.decompiler.dad.opcode_ins.mulintlit16(ins, vmap)
androquard.decompiler.dad.opcode ins.mulintlit8(ins, vmap)
androquard.decompiler.dad.opcode ins.mullong(ins, vmap)
androguard.decompiler.dad.opcode_ins.mullong2addr(ins, vmap)
androguard.decompiler.dad.opcode_ins.negdouble(ins, vmap)
androquard.decompiler.dad.opcode ins.negfloat(ins, vmap)
androguard.decompiler.dad.opcode_ins.negint(ins, vmap)
androquard.decompiler.dad.opcode_ins.neglong(ins, vmap)
androguard.decompiler.dad.opcode_ins.newarray(ins, vmap)
androquard.decompiler.dad.opcode ins.newinstance(ins, vmap)
androquard.decompiler.dad.opcode ins.nop(ins, vmap)
androguard.decompiler.dad.opcode_ins.notint(ins, vmap)
androguard.decompiler.dad.opcode_ins.notlong(ins, vmap)
androquard.decompiler.dad.opcode ins.orint(ins, vmap)
androquard.decompiler.dad.opcode ins.orint2addr(ins, vmap)
androguard.decompiler.dad.opcode_ins.orintlit16(ins, vmap)
androquard.decompiler.dad.opcode_ins.orintlit8(ins, vmap)
androquard.decompiler.dad.opcode_ins.orlong(ins, vmap)
androguard.decompiler.dad.opcode_ins.orlong2addr(ins, vmap)
androquard.decompiler.dad.opcode_ins.packedswitch(ins, vmap)
androguard.decompiler.dad.opcode_ins.remdouble(ins, vmap)
androquard.decompiler.dad.opcode ins.remdouble2addr(ins, vmap)
```

```
androguard.decompiler.dad.opcode ins.remfloat(ins, vmap)
androquard.decompiler.dad.opcode ins.remfloat2addr(ins, vmap)
androguard.decompiler.dad.opcode_ins.remint(ins, vmap)
androguard.decompiler.dad.opcode_ins.remint2addr(ins, vmap)
androquard.decompiler.dad.opcode ins.remintlit16(ins, vmap)
androquard.decompiler.dad.opcode ins.remintlit8(ins, vmap)
androquard.decompiler.dad.opcode_ins.remlong(ins, vmap)
androquard.decompiler.dad.opcode_ins.remlong2addr(ins, vmap)
androquard.decompiler.dad.opcode ins.return reg(ins, vmap)
androquard.decompiler.dad.opcode_ins.returnobject(ins, vmap)
androquard.decompiler.dad.opcode_ins.returnvoid(ins, vmap)
androquard.decompiler.dad.opcode_ins.returnwide(ins, vmap)
androquard.decompiler.dad.opcode ins.rsubint(ins, vmap)
androquard.decompiler.dad.opcode ins.rsubintlit8(ins, vmap)
androquard.decompiler.dad.opcode ins.sqet (ins, vmap)
androquard.decompiler.dad.opcode ins.sqetboolean(ins, vmap)
androquard.decompiler.dad.opcode ins.sqetbyte(ins, vmap)
androguard.decompiler.dad.opcode ins.sqetchar(ins, vmap)
androquard.decompiler.dad.opcode_ins.sgetobject(ins, vmap)
androquard.decompiler.dad.opcode ins.sqetshort(ins, vmap)
androquard.decompiler.dad.opcode_ins.sqetwide(ins, vmap)
androguard.decompiler.dad.opcode_ins.shlint(ins, vmap)
androguard.decompiler.dad.opcode_ins.shlint2addr(ins, vmap)
androquard.decompiler.dad.opcode ins.shlintlit8(ins, vmap)
androquard.decompiler.dad.opcode ins.shllong(ins, vmap)
androguard.decompiler.dad.opcode_ins.shllong2addr(ins, vmap)
androguard.decompiler.dad.opcode_ins.shrint(ins, vmap)
androquard.decompiler.dad.opcode ins.shrint2addr(ins, vmap)
androquard.decompiler.dad.opcode ins.shrintlit8(ins, vmap)
androquard.decompiler.dad.opcode_ins.shrlong(ins, vmap)
androquard.decompiler.dad.opcode_ins.shrlong2addr(ins, vmap)
androquard.decompiler.dad.opcode_ins.sparseswitch(ins, vmap)
androquard.decompiler.dad.opcode_ins.sput (ins, vmap)
androquard.decompiler.dad.opcode_ins.sputboolean(ins, vmap)
androguard.decompiler.dad.opcode_ins.sputbyte(ins, vmap)
androquard.decompiler.dad.opcode ins.sputchar(ins, vmap)
```

```
androguard.decompiler.dad.opcode_ins.sputobject(ins, vmap)
androguard.decompiler.dad.opcode_ins.sputshort(ins, vmap)
androguard.decompiler.dad.opcode_ins.sputwide(ins, vmap)
androguard.decompiler.dad.opcode_ins.store_array_inst(val_a, val_b, val_c, ar_type,
androguard.decompiler.dad.opcode_ins.subdouble(ins, vmap)
androquard.decompiler.dad.opcode_ins.subdouble2addr(ins, vmap)
androguard.decompiler.dad.opcode_ins.subfloat(ins, vmap)
androquard.decompiler.dad.opcode_ins.subfloat2addr(ins, vmap)
androguard.decompiler.dad.opcode_ins.subint(ins, vmap)
androguard.decompiler.dad.opcode_ins.subint2addr(ins, vmap)
androguard.decompiler.dad.opcode_ins.sublong(ins, vmap)
androguard.decompiler.dad.opcode_ins.sublong2addr(ins, vmap)
androquard.decompiler.dad.opcode ins.throw(ins, vmap)
androguard.decompiler.dad.opcode_ins.ushrint(ins, vmap)
androguard.decompiler.dad.opcode_ins.ushrint2addr(ins, vmap)
androguard.decompiler.dad.opcode_ins.ushrintlit8(ins, vmap)
androguard.decompiler.dad.opcode_ins.ushrlong(ins, vmap)
androguard.decompiler.dad.opcode_ins.ushrlong2addr(ins, vmap)
androguard.decompiler.dad.opcode_ins.xorint(ins, vmap)
androguard.decompiler.dad.opcode_ins.xorint2addr(ins, vmap)
androguard.decompiler.dad.opcode_ins.xorintlit16(ins, vmap)
androquard.decompiler.dad.opcode ins.xorintlit8(ins, vmap)
androguard.decompiler.dad.opcode_ins.xorlong(ins, vmap)
androguard.decompiler.dad.opcode_ins.xorlong2addr(ins, vmap)
```

# androguard.decompiler.dad.util module

```
androguard.decompiler.dad.util.build_path (graph, node1, node2, path=None)

Build the path from node1 to node2. The path is composed of all the nodes between node1 and node2, node1 excluded. Although if there is a loop starting from node1, it will be included in the path.

androguard.decompiler.dad.util.common_dom(idom, cur, pred)
```

androguard.decompiler.dad.util.create\_png(cls\_name, medir\_name='graphs2')

meth\_name,

graph,

Creates a PNG from a given Graph.

### **Parameters**

- cls name (str) name of the class
- **meth\_name** (str) name of the method
- graph (androguard.decompiler.dad.graph.Graph) -

```
• dir_name (str) - output directory
androguard.decompiler.dad.util.get_access_class(access)
androguard.decompiler.dad.util.get_access_field(access)
androguard.decompiler.dad.util.get_access_method(access)
androguard.decompiler.dad.util.get_params_type(descriptor)
     Return the parameters type of a descriptor (e.g (IC)V)
androguard.decompiler.dad.util.get_type (atype, size=None)
     Retrieve the java type of a descriptor (e.g : I)
androquard.decompiler.dad.util.get_type_size(param)
     Return the number of register needed by the type @param
androquard.decompiler.dad.util.merge_inner(clsdict)
     Merge the inner class(es) of a class: e.g class A { ... } class A$foo{ ... } class A$bar{ ... } ==> class A {
     class foo\{\ldots\} class bar\{\ldots\}... \}
androguard.decompiler.dad.writer module
class androguard.decompiler.dad.writer.Writer(graph, method)
     Bases: object
     Transforms a method into Java code.
     dec ind(i=1)
     end_ins()
     inc\_ind(i=1)
     space()
     str_ext()
     visit_alength(array)
     visit_aload(array, index)
     visit_assign (lhs, rhs)
     visit astore (array, index, rhs, data=None)
     visit base class(cls, data=None)
     visit binary expression (op, arg1, arg2)
     visit_cast (op, arg)
     visit_catch_node (catch_node)
     visit_check_cast (arg, atype)
     visit_cond_expression(op, arg1, arg2)
     visit_cond_node(cond)
     visit_condz_expression(op, arg)
     visit_constant (cst)
     visit decl(var)
     visit_fill_array (array, value)
```

```
visit_filled_new_array (atype, size, args)
visit_get_instance (arg, name, data=None)
visit_get_static(cls, name)
visit_ins (ins)
visit_invoke (name, base, ptype, rtype, args, invokeInstr)
visit_loop_node (loop)
visit_monitor_enter(ref)
visit_monitor_exit (ref)
visit_move (lhs, rhs)
visit_move_exception (var, data=None)
visit_move_result (lhs, rhs)
visit_new (atype, data=None)
visit_new_array (atype, size)
visit_node (node)
visit_nop()
visit param(param, data=None)
visit_put_instance (lhs, name, rhs, data=None)
visit_put_static(cls, name, rhs)
visit_return(arg)
visit_return_node(ret)
visit_return_void()
visit_short_circuit_condition (nnot, aand, cond1, cond2)
visit_statement_node(stmt)
visit_super()
visit_switch (arg)
visit_switch_node (switch)
visit_this()
visit_throw(ref)
visit_throw_node(throw)
visit_try_node (try_node)
visit_unary_expression(op, arg)
visit_variable(var)
write(s, data=None)
write_ext(t)
write_ind()
write_ind_visit_end(lhs, s, rhs=None, data=None)
```

```
write_ind_visit_end_ext(lhs, before, s, after, rhs=None, data=None, subsec-
tion='UNKNOWN_SUBSECTION')
write_inplace_if_possible(lhs, rhs)
write_method()
androguard.decompiler.dad.writer.string(s)
Convert a string to a escaped ASCII representation including quotation marks :param s: a string :return: ASCII escaped string
```

#### **Module contents**

#### **Submodules**

## androguard.decompiler.decompiler module

```
class androguard.decompiler.decompiler.DecompilerDAD (vm, vmx)
    Bases: object
    display_all(_class)
    display_source(m)
    get_all (class_name)
    get_ast_class(_class)
    get_ast_method(m)
    get_source_class(_class)
    get_source_class_ext(_class)
    get_source_method(m)
class androguard.decompiler.decompiler.DecompilerDed(vm,
                                                                        bin ded='ded.sh',
                                                             tmp_dir='/tmp/')
    Bases: object
    display_all (_class)
    display_source (method)
    get_all(class_name)
    get_source_class(_class)
    get_source_method(method)
class androguard.decompiler.decompiler.DecompilerDex2Fernflower(vm,
                                                                           bin dex2jar='dex2jar.sh',
                                                                          bin_fernflower='fernflower.jar',
                                                                          op-
                                                                          tions_fernflower={'asc':
                                                                           '1',
                                                                                  'dgs':
                                                                           '1'},
                                                                          tmp dir='/tmp/')
    Bases: object
    display_all (_class)
    display_source (method)
```

```
get_all (class_name)
     get_source_class(_class)
     get_source_method(method)
class androguard.decompiler.decompiler.DecompilerDex2Jad(vm,
                                                                     bin dex2jar='dex2jar.sh',
                                                                     bin_jad='jad',
                                                                     tmp_dir='/tmp/')
     Bases: object
     display_all (_class)
     display_source (method)
     get_all(class_name)
     get_source_class(_class)
     get_source_method(method)
class androguard.decompiler.decompiler.DecompilerDex2WineJad(vm,
                                                                          bin_dex2jar='dex2jar.sh',
                                                                          bin jad='jad',
                                                                          tmp dir='/tmp/')
     Bases: object
     display_all (_class)
     display_source (method)
     get_all(class_name)
     get_source_class(_class)
     get_source_method(method)
class androguard.decompiler.decompiler.DecompilerJADX (vm, vmx, jadx='jadx', keep-
                                                                 files=False)
     Bases: object
     display_all (_class)
         ???
             Parameters _class -
             Returns
     display_source(m)
         This method does the same as get_source_method but prints the result directly to stdout
             Parameters m – EncodedMethod to print
             Returns
     get_all(class_name)
         ???
             Parameters class_name -
             Returns
     get_source_class(_class)
         Return the Java source code of a whole class
             Parameters _class - ClassDefItem object, to get the source from
```

#### Returns

#### get\_source\_method(m)

Return the Java source of a single method

Parameters m – EncodedMethod Object

#### Returns

#### **Module contents**

#### 3.1.2 Submodules

## 3.1.3 androguard.misc module

```
androguard.misc.AnalyzeAPK(_file, session=None, raw=False)
```

Analyze an android application and setup all stuff for a more quickly analysis! If session is None, no session is used at all. This is the default behaviour. If you like to continue your work later, it might be a good idea to use a session. A default session can be created by using get\_default\_session().

## **Parameters**

- **\_file**(string (for filename) or bytes (for raw)) the filename of the android application or a buffer which represents the application
- session A session (default: None)
- raw boolean if raw bytes are supplied instead of a filename

Return type return the APK, list of DalvikVMFormat, and Analysis objects

androguard.misc.AnalyzeDex(filename, session=None)

Analyze an android dex file and setup all stuff for a more quickly analysis!

#### **Parameters**

- **filename** (*string*) the filename of the android dex file or a buffer which represents the dex file
- session A session (Default None)

**Return type** return a tuple of (sha256hash, DalvikVMFormat, Analysis)

androquard.misc.AnalyzeODex(filename, session=None)

Analyze an android odex file and setup all stuff for a more quickly analysis!

#### **Parameters**

- **filename** (string) the filename of the android dex file or a buffer which represents the dex file
- **session** The Androguard Session to add the ODex to (default: None)

Return type return a tuple of (sha256hash, DalvikOdexVMFormat, Analysis)

androquard.misc.RunDecompiler (d, dx, decompiler name)

Run the decompiler on a specific analysis

#### **Parameters**

- d (DalvikVMFormat object) the DalvikVMFormat object
- dx (VMAnalysis object) the analysis of the format
- **decompiler** (string) the type of decompiler to use ("dad", "dex2jad", "ded")

androguard.misc.clean\_file\_name (filename, unique=True, replace='\_', force\_nt=False)

Return a filename version, which has no characters in it which are forbidden. On Windows these are for example <, /, ?, ...

The intention of this function is to allow distribution of files to different OSes.

#### **Parameters**

- filename string to clean
- unique check if the filename is already taken and append an integer to be unique (default:
- replace replacement character. (default: '')
- **force\_nt** Force shortening of paths like on NT systems (default: False)

Returns clean string

```
androguard.misc.get_default_session()
```

Return the default Session from the configuration or create a new one, if the session in the configuration is None.

Return type androguard.session.Session

```
androguard.misc.init_print_colors()
androguard.misc.sign apk (filename, keystore, storepass)
     Use jarsigner to sign an APK file.
```

#### **Parameters**

- **filename** APK file on disk to sign (path)
- **keystore** path to keystore
- **storepass** your keystorage passphrase

## 3.1.4 androguard.session module

```
androquard.session.Load(filename)
     load your session!
```

example:

```
s = session.Load("mysession.ag")
```

**Parameters filename** (string) – the filename where the session has been saved

#### Return type the elements of your session:)

```
androguard.session.Save (session, filename=None) save your session to use it later.
```

Returns the filename of the written file. If not filename is given, a file named *androguard\_session\_<DATE>.ag* will be created in the current working directory. *<DATE>* is a timestamp with the following format: *%Y-%m-%d %H%M%S*.

This function will overwrite existing files without asking.

If the file could not written, None is returned.

example:

```
s = session.Session()
session.Save(s, "msession.ag")
```

#### **Parameters**

- session A Session object to save
- **filename** (*string*) output filename to save the session

```
class androguard.session.Session(export_ipython=False)
    Bases: object
```

A Session is able to store multiple APK, DEX or ODEX files and can be pickled to disk in order to resume work later.

The main function used in Sessions is probably add(), which adds files to the session and performs analysis on them.

Afterwards, the files can be gathered using methods such as  $get\_objects\_apk()$ ,  $get\_objects\_dex()$  or  $get\_classes()$ .

example:

```
s = Session()
digest = s.add("some.apk")

print("SHA256 of the file: {}".format(digest))

a, d, dx = s.get_objects_apk("some.apk", digest)
print(a.get_package())

# Reset the Session for a fresh set of files
s.reset()

digest2 = s.add("classes.dex")
print("SHA256 of the file: {}".format(digest2))
for h, d, dx in s.get_objects_dex():
    print("SHA256 of the DEX file: {}".format(h))
```

add (filename, raw\_data=None, dx=None)

Generic method to add a file to the session.

This is the main method to use when adding files to a Session!

If an APK file is supplied, all DEX files are analyzed too. For DEX and ODEX files, only this file is analyzed (what else should be analyzed).

Returns the SHA256 of the analyzed file.

#### **Parameters**

- filename filename to load
- raw\_data bytes of the file, or None to load the file from filename
- dx An already exiting Analysis object

**Returns** the sha256 of the file or None on failure

#### addAPK (filename, data)

Add an APK file to the Session and run analysis on it.

#### **Parameters**

- **filename** (file)name of APK file
- data binary data of the APK file

Returns a tuple of SHA256 Checksum and APK Object

addDEX (filename, data, dx=None, postpone\_xref=False)

Add a DEX file to the Session and run analysis.

#### **Parameters**

- filename the (file)name of the DEX file
- data binary data of the dex file
- dx an existing Analysis Object (optional)
- postpone\_xref True if no xref shall be created, and will be called manually

Returns A tuple of SHA256 Hash, DalvikVMFormat Object and Analysis object

#### addDEY (filename, data, dx=None)

Add an ODEX file to the session and run the analysis

#### get\_all\_apks()

Yields a list of tuples of SHA256 hash of the APK and APK objects of all analyzed APKs in the Session.

#### get\_analysis (current\_class)

Returns the *Analysis* object which contains the *current\_class*.

## Parameters current\_class

(androguard.core.bytecodes.dvm.

 ${\tt ClassDefItem)}-The~class~to~search~for$ 

**Return type** androguard.core.analysis.analysis.Analysis

#### qet classes()

Returns all Java Classes from the DEX objects as an array of DEX files.

## get\_digest\_by\_class (current\_class)

Return the SHA256 hash of the object containing the ClassDefItem

Returns the first digest this class was present. For example, if you analyzed an APK, this should return the digest of the APK and not of the DEX file.

#### get\_filename\_by\_class (current\_class)

Returns the filename of the DEX file where the class is in.

Returns the first filename this class was present. For example, if you analyzed an APK, this should return the filename of the APK and not of the DEX file.

Parameters current class - ClassDefItem

Returns None if class was not found or the filename

```
get_format (current_class)
```

Returns the DalvikVMFormat of a given ClassDefItem.

Parameters current\_class - A ClassDefItem

```
get_nb_strings()
```

Return the total number of strings in all Analysis objects

```
get_objects_apk (filename=None, digest=None)
```

Returns APK, DalvikVMFormat and Analysis of a specified APK.

You must specify either *filename* or *digest*. It is possible to use both, but in this case only *digest* is used.

example:

```
s = Session()
digest = s.add("some.apk")
a, d, dx = s.get_objects_apk(digest=digest)
```

#### example:

```
s = Session()
filename = "some.apk"
digest = s.add(filename)
a, d, dx = s.get_objects_apk(filename=filename)
```

#### **Parameters**

- **filename** the filename of the APK file, only used of digest is None
- digest the sha256 hash, as returned by add () for the APK

**Returns** a tuple of (APK, [DalvikVMFormat], Analysis)

```
get_objects_dex()
```

Yields all dex objects inclduing their Analysis objects

**Returns** tuple of (sha256, DalvikVMFormat, Analysis)

```
get_strings()
```

Yields all StringAnalysis for all unique Analysis objects

## isOpen()

Test if any file was analyzed in this session

**Returns** *True* if any file was analyzed, *False* otherwise

#### reset()

Reset the current session, delete all added files.

```
save (filename=None)
```

Save the current session, see also Save ().

#### show()

Print information to stdout about the current session. Gets all APKs, all DEX files and all Analysis objects.

## 3.1.5 androguard.util module

```
androguard.util.get_certificate_name_string(name, short=False, delimiter=', ') Format the Name type of a X509 Certificate in a human readable form.
```

#### **Parameters**

- name (dict or asn1crypto.x509.Name) Name object to return the DN from
- **short** (boolean) Use short form (default: False)
- **delimiter** (str) Delimiter string or character between two parts (default: ', ')

## Return type str

androguard.util.read(filename, binary=True)
Open and read a file

#### **Parameters**

- **filename** filename to open and read
- binary True if the file should be read as binary

**Returns** bytes if binary is True, str otherwise

## 3.1.6 Module contents

## CHAPTER

# **FOUR**

# **INDICES AND TABLES**

- genindex
- modindex
- search

## **PYTHON MODULE INDEX**

## а androquard, 184 androguard.core, 151 androguard.core.analysis,66 androquard.core.analysis.analysis,47 androquard.core.analysis.auto,61 androguard.core.androconf, 145 androguard.core.api\_specific\_resources, androquard.core.bytecode, 147 androguard.core.bytecodes, 145 androquard.core.bytecodes.apk, 67 androguard.core.bytecodes.axml, 135 androquard.core.bytecodes.dvm, 77 androguard.core.mutf8, 144 androquard.core.resources, 145 androguard.core.resources.public, 145 androquard.decompiler, 179 androquard.decompiler.dad, 177 androguard.decompiler.dad.basic\_blocks, androguard.decompiler.dad.control\_flow, 155 androguard.decompiler.dad.dast, 151 androguard.decompiler.dad.dataflow, 155 androquard.decompiler.dad.decompile, 156 androguard.decompiler.dad.graph, 157 androguard.decompiler.dad.instruction, 159 androquard.decompiler.dad.node, 166 androquard.decompiler.dad.opcode ins, 167 androguard.decompiler.dad.util, 174 androguard.decompiler.dad.writer, 175

androquard.decompiler.decompiler, 177

androguard.misc, 179 androguard.session, 180 androguard.util, 183

188 Python Module Index

## **INDEX**

A	method), 97	
access() (androguard.core.analysis.analysis.MethodAnaproperty), 59	nafysis_ins() (androguard.decompiler.dad.basic_blocks.Basic method), 153	:Block
access_flags (andro-	add_method() (andro-	
guard.core.bytecodes.dvm.EncodedMethod attribute), 97	guard.core.analysis.analysis.ClassAnalysis method), 53	
ADD (androguard.decompiler.dad.opcode_ins.Op attribute), 167	add_method() (andro- guard.core.analysis.analysis.ExternalClass	
add() (androguard.core.analysis.analysis.Analysis method), 47	<pre>method), 57 add_method_xref_from() (andro-</pre>	
add() (androguard.core.analysis.analysis.Exceptions method), 57	guard.core.analysis.analysis.ClassAnalysis method), 53	
add() (androguard.core.bytecodes.dvm.DBGBytecode method), 85	<pre>add_method_xref_to() (andro- guard.core.analysis.analysis.ClassAnalysis</pre>	
add() (androguard.decompiler.dad.dast.JSONWriter method), 151	<pre>method), 53 add_node() (androguard.decompiler.dad.graph.Graph</pre>	
add () (androguard session Session method) 181	method), 158	
<pre>add_case() (androguard.decompiler.dad.basic_blocks.S</pre>	SwitchBlocke () (androguard.decompiler.dad.node.Interval method), 166	
add_catch_edge() (andro-	add_note() (androguard.core.analysis.analysis.DVMBasic	:Block
guard.decompiler.dad.graph.Graph method), 158	<pre>method), 56 add_note() (androguard.core.bytecodes.dvm.EncodedMet.</pre>	hod
add_catch_node() (andro- guard.decompiler.dad.basic_blocks.TryBlock method), 154	<pre>method), 97 add_note() (androguard.core.bytecodes.dvm.FillArrayDat method), 103</pre>	
add_edge() (androguard.decompiler.dad.graph.Graph method), 158	<pre>add_note() (androguard.core.bytecodes.dvm.PackedSwitch</pre>	h
add_field_xref_read() (andro- guard.core.analysis.analysis.ClassAnalysis	<pre>add_note() (androguard.core.bytecodes.dvm.SparseSwitch</pre>	h
method), 52	add_type_item() (andro-	
add_field_xref_write() (andro- guard.core.analysis.analysis.ClassAnalysis	guard.core.bytecodes.dvm.ClassManager method), 83	
method), 53	add_variable_declaration() (andro-	
add_idx() (androguard.core.bytecode.BuffHandle method), 147	guard.decompiler.dad.basic_blocks.BasicBlock method), 153	
add_inote() (andro-	add_xref_from() (andro-	
guard.core.bytecodes.dvm.DalvikCode method), 87	guard.core.analysis.analysis.ClassAnalysis method), 53	
add_inote()(androguard.core.bytecodes.dvm.DCode	add_xref_from() (andro-	
method), 85	guard.core.analysis.analysis.MethodAnalysis	
add_inote() (andro-	<pre>method), 59 add xref from() (andro-</pre>	
guard.core.bytecodes.dvm.EncodedMethod	add_xref_from() (andro-	

guard.core. analysis. analysis. String Analysis	guard.decompiler.dad.opcode_ins), 168
method), 61	agetwide() (in module andro-
add_xref_read() (andro-	guard.decompiler.dad.opcode_ins), 168
guard.core. analysis. analysis. Field Analysis	all_preds() (andro-
method), 58	guard.decompiler.dad.graph.Graph method),
add_xref_to() (andro-	158
guard.core.analysis.analysis.ClassAnalysis	all_sucs() (androguard.decompiler.dad.graph.Graph
method), 53	method), 158
<pre>add_xref_to()</pre> (andro-	Analysis (class in androguard.core.analysis.analysis),
guard.core.analysis.analysis.MethodAnalysis	47
method), 59	analysis_adex() (andro-
<pre>add_xref_write()</pre> (andro-	guard.core.analysis.auto.DefaultAndroAnalysis
guard.core.analysis.analysis.FieldAnalysis	method), 63
method), 58	analysis_apk() (andro-
addAPK() (androguard.session.Session method), 182	guard.core.analysis.auto.DefaultAndroAnalysis
addDEX() (androguard.session.Session method), 182	method), 63
addDEY() (androguard.session.Session method), 182	analysis_app() (andro-
adddouble() (in module andro-	guard.core.analysis.auto.DefaultAndroAnalysis
guard.decompiler.dad.opcode_ins), 168	method), 63
adddouble2addr() (in module andro-	analysis_arsc() (andro-
guard.decompiler.dad.opcode_ins), 168	guard.core.analysis.auto.DefaultAndroAnalysis
addfloat() (in module andro-	method), 63
guard.decompiler.dad.opcode_ins), 168	analysis_axml() (andro-
addfloat2addr() (in module andro-	guard.core.analysis.auto.DefaultAndroAnalysis
guard.decompiler.dad.opcode_ins), 168	method), 64
addint() (in module andro-	analysis_dex() (andro-
guard.decompiler.dad.opcode_ins), 168	guard.core.analysis.auto.DefaultAndroAnalysis
addint2addr() (in module andro-	method), 64
guard.decompiler.dad.opcode_ins), 168	analysis_dey() (andro-
addintlit16() (in module andro-	guard.core.analysis.auto.DefaultAndroAnalysis
guard.decompiler.dad.opcode_ins), 168	method), 64
	AnalyzeAPK() (in module androguard.misc), 179
guard.decompiler.dad.opcode_ins), 168	AnalyzeDex() (in module androguard.misc), 179
addlong() (in module andro-	AnalyzeODex() (in module androguard.misc), 179
guard.decompiler.dad.opcode_ins), 168	AND (androguard.decompiler.dad.opcode_ins.Op at-
addlong2addr() (in module andro-	tribute), 167
guard.decompiler.dad.opcode_ins), 168	andint() (in module andro-
adjust_idx() (andro-	guard.decompiler.dad.opcode_ins), 168
guard.core.bytecodes.dvm.EncodedField	andint2addr() (in module andro-
method), 96	guard.decompiler.dad.opcode_ins), 168
adjust_idx() (andro-	andintlit16() (in module andro-
guard.core.bytecodes.dvm.EncodedMethod	guard.decompiler.dad.opcode_ins), 168
method), 97	andintlit8() (in module andro-
aget() (in module andro-	guard.decompiler.dad.opcode_ins), 168
guard.decompiler.dad.opcode_ins), 168	andlong() (in module andro-
agetboolean() (in module andro-	guard.decompiler.dad.opcode_ins), 168
<pre>guard.decompiler.dad.opcode_ins), 168</pre>	andlong2addr() (in module andro-
agetbyte() (in module andro-	guard.decompiler.dad.opcode_ins), 168
guard.decompiler.dad.opcode_ins), 168	AndroAuto (class in androguard.core.analysis.auto),
agetchar() (in module andro-	61
guard.decompiler.dad.opcode_ins), 168	androguard (module), 184
agetobject() (in module andro-	androguard.core (module), 151
guard.decompiler.dad.opcode_ins), 168	androguard.core.analysis(module),66
agetshort() (in module andro-	

ule), 47	Annatation Cot Dof Itom (alass in	an dua
androguard.core.analysis.auto( <i>module</i> ),61	AnnotationSetRefItem (class in guard.core.bytecodes.dvm), 78	andro-
androguard.core.androconf (module), 145	•	andro-
androguard.core.api_specific_resources	guard.core.bytecodes.dvm), 79	
( <i>module</i> ), 66	APILevelNotFoundError,66	
androguard.core.bytecode ( <i>module</i> ), 147	APK (class in androguard.core.bytecodes.apk), 67	
androguard.core.bytecodes (module), 145	`	andro-
androguard.core.bytecodes.apk (module), 67	guard.core.bytecodes.apk), 76	_
androguard.core.bytecodes.axml (module),	`	andro-
135	guard.core.bytecodes.apk), 76	,
androguard.core.bytecodes.dvm (module),77	` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` `	andro-
androguard.core.mutf8 (module), 144	guard.core.bytecodes.apk), 76	J
androguard.core.resources(module), 145 androguard.core.resources.public (mod-	`	andro-
ule), 145	guard.core.bytecodes.apk), 76 aput () (in module	andro-
androguard.decompiler( <i>module</i> ), 179	guard.decompiler.dad.opcode_ins), 168	unuro-
androguard.decompiler.dad( <i>module</i> ), 177		andro-
androguard.decompiler.dad.mount, 177 androguard.decompiler.dad.basic_blocks	guard.decompiler.dad.opcode_ins), 168	anaro
(module), 153		andro-
androguard.decompiler.dad.control_flow	guard.decompiler.dad.opcode_ins), 168	
(module), 155		andro-
androguard.decompiler.dad.dast (module),	guard.decompiler.dad.opcode_ins), 168	
151	•	andro-
androguard.decompiler.dad.dataflow(mod-	guard.decompiler.dad.opcode_ins), 169	
ule), 155		andro-
androguard.decompiler.dad.decompile	guard.decompiler.dad.opcode_ins), 169	
(module), 156	aputwide() (in module	andro-
androguard.decompiler.dad.graph (module),	guard.decompiler.dad.opcode_ins), 169	
157	<del></del>	andro-
androguard.decompiler.dad.instruction	guard.decompiler.dad.dast), 152	
(module), 159	<del>-</del> · · · · · · · · · · · · · · · · · · ·	andro-
androguard.decompiler.dad.node (module),	guard.decompiler.dad.dast), 152	
166	<del></del>	andro-
androguard.decompiler.dad.opcode_ins	guard.decompiler.dad.dast), 152	,
(module), 167	1 1	andro-
androguard.decompiler.dad.util ( <i>module</i> ),  174	guard.decompiler.dad.instruction), 159	an dno
androguard.decompiler.dad.writer (mod-	arraylength() (in module guard.decompiler.dad.opcode_ins), 169	andro-
ule), 175		andro-
androguard.decompiler.decompiler (mod-	ArrayLengthExpression (class in guard.decompiler.dad.instruction), 159	апато-
ule), 177		andro-
androguard.misc ( <i>module</i> ), 179	guard.decompiler.dad.instruction), 159	unuro-
androguard.mise ( <i>module</i> ), 179 androguard.session ( <i>module</i> ), 180	•	andro-
androguard.util (module), 183	guard.decompiler.dad.instruction), 159	anaro
AnnotationElement (class in andro-	1	andro-
guard.core.bytecodes.dvm), 77	guard.core.bytecodes.axml), 135	ciriciro
AnnotationItem (class in andro-	•	andro-
guard.core.bytecodes.dvm), 77	guard.core.bytecodes.axml), 135	
AnnotationOffItem (class in andro-		andro-
guard.core.bytecodes.dvm), 78	guard.core.bytecodes.axml), 136	
•	ARSCParser.ResourceResolver (class in	andro-
guard.core.bytecodes.dvm), 79	guard.core.bytecodes.axml), 136	
AnnotationSetItem (class in andro-	ARSCResStringPoolRef (class in	andro-
guard.core.bytecodes.dym), 78	guard.core.bytecodes.axml), 139	

ARSCResTableConfig (class in andro-	
	Bold (androguard.core.androconf.Color attribute), 145
guard.core.bytecodes.axml), 140 ARSCResTableEntry (class in andro-	BrokenAPKError, 76 Buff (class in androguard.core.bytecode), 147
guard.core.bytecodes.axml), 140	BuffHandle (class in androguard.core.bytecode), 147
	build_def_use() (in module andro-
guard.core.bytecodes.axml), 141	guard.decompiler.dad.dataflow), 155
= = · · · · · · · · · · · · · · · · · ·	<pre>build_node_from_block() (in module andro-</pre>
guard.core.bytecodes.axml), 141  ARSCResTypeSpec (class in andro-	<pre>guard.decompiler.dad.basic_blocks), 154 build_path() (in module andro-</pre>
ARSCResTypeSpec (class in androguard.core.bytecodes.axml), 141	guard.decompiler.dad.util), 174
assign_binary_2addr_exp() (in module andro-	guira.accompiler.aua.uui), 174
guard.decompiler.dad.opcode_ins), 169	C
assign_binary_exp() (in module andro-	cast() (in module androguard.decompiler.dad.dast),
guard.decompiler.dad.opcode_ins), 169	152
<pre>assign_cast_exp() (in module andro-</pre>	CastExpression (class in andro-
guard.decompiler.dad.opcode_ins), 169	guard.decompiler.dad.instruction), 160
assign_cmp() (in module andro-	catch_struct() (in module andro-
guard.decompiler.dad.opcode_ins), 169	guard.decompiler.dad.control_flow), 155
assign_const() (in module andro- guard.decompiler.dad.opcode_ins), 169	CatchBlock (class in andro-
assign_lit() (in module andro-	guard.decompiler.dad.basic_blocks), 153 checkcast() (in module andro-
guard.decompiler.dad.opcode_ins), 169	checkcast() (in module andro- guard.decompiler.dad.opcode_ins), 169
AssignExpression (class in andro-	CheckCastExpression (class in andro-
guard.decompiler.dad.instruction), 160	guard.decompiler.dad.instruction), 161
assignment() (in module andro-	class_name() (andro-
guard.decompiler.dad.dast), 152	guard.core.analysis.analysis.MethodAnalysis
AXMLParser (class in andro-	property), 59
guard.core.bytecodes.axml), 141	ClassAnalysis (class in andro-
AXMLPrinter (class in andro-	guard.core.analysis.analysis), 52
guard.core.bytecodes.axml), 142	ClassDataItem (class in andro-
В	guard.core.bytecodes.dvm), 80
_	ClassDefItem (class in andro-
BaseClass (class in andro-	ClassDefItem (class in androguard.core.bytecodes.dvm), 81
_	ClassDefItem (class in androguard.core.bytecodes.dvm), 81
BaseClass (class in andro- guard.decompiler.dad.instruction), 160	ClassDefItem (class in androguard.core.bytecodes.dvm), 81 ClassHDefItem (class in andro-
BaseClass (class in andro- guard.decompiler.dad.instruction), 160 BasicBlock (class in andro-	ClassDefItem (class in andro- guard.core.bytecodes.dvm), 81 ClassHDefItem (class in andro- guard.core.bytecodes.dvm), 83
BaseClass (class in andro- guard.decompiler.dad.instruction), 160  BasicBlock (class in andro- guard.decompiler.dad.basic_blocks), 153  BasicBlocks (class in andro- guard.core.analysis.analysis), 52	ClassDefItem (class in andro- guard.core.bytecodes.dvm), 81 ClassHDefItem (class in andro- guard.core.bytecodes.dvm), 83 ClassManager (class in andro- guard.core.bytecodes.dvm), 83 clean_file_name() (in module androguard.misc),
BaseClass (class in andro- guard.decompiler.dad.instruction), 160  BasicBlock (class in andro- guard.decompiler.dad.basic_blocks), 153  BasicBlocks (class in andro- guard.core.analysis.analysis), 52  BasicReachDef (class in andro-	ClassDefItem (class in andro- guard.core.bytecodes.dvm), 81  ClassHDefItem (class in andro- guard.core.bytecodes.dvm), 83  ClassManager (class in andro- guard.core.bytecodes.dvm), 83  clean_file_name() (in module androguard.misc), 180
BaseClass (class in andro- guard.decompiler.dad.instruction), 160  BasicBlock (class in andro- guard.decompiler.dad.basic_blocks), 153  BasicBlocks (class in andro- guard.core.analysis.analysis), 52  BasicReachDef (class in andro- guard.decompiler.dad.dataflow), 155	ClassDefItem (class in andro- guard.core.bytecodes.dvm), 81  ClassHDefItem (class in andro- guard.core.bytecodes.dvm), 83  ClassManager (class in andro- guard.core.bytecodes.dvm), 83  clean_file_name() (in module androguard.misc), 180  clean_name_instruction() (in module andro-
BaseClass (class in andro- guard.decompiler.dad.instruction), 160  BasicBlock (class in andro- guard.decompiler.dad.basic_blocks), 153  BasicBlocks (class in andro- guard.core.analysis.analysis), 52  BasicReachDef (class in andro- guard.decompiler.dad.dataflow), 155  bfs() (in module androguard.decompiler.dad.graph),	ClassDefItem (class in andro- guard.core.bytecodes.dvm), 81  ClassHDefItem (class in andro- guard.core.bytecodes.dvm), 83  ClassManager (class in andro- guard.core.bytecodes.dvm), 83  clean_file_name() (in module androguard.misc),
BaseClass (class in andro- guard.decompiler.dad.instruction), 160  BasicBlock (class in andro- guard.decompiler.dad.basic_blocks), 153  BasicBlocks (class in andro- guard.core.analysis.analysis), 52  BasicReachDef (class in andro- guard.decompiler.dad.dataflow), 155  bfs() (in module androguard.decompiler.dad.graph), 158	ClassDefItem (class in andro- guard.core.bytecodes.dvm), 81  ClassHDefItem (class in andro- guard.core.bytecodes.dvm), 83  ClassManager (class in andro- guard.core.bytecodes.dvm), 83  clean_file_name() (in module androguard.misc),
BaseClass (class in andro- guard.decompiler.dad.instruction), 160  BasicBlock (class in andro- guard.decompiler.dad.basic_blocks), 153  BasicBlocks (class in andro- guard.core.analysis.analysis), 52  BasicReachDef (class in andro- guard.decompiler.dad.dataflow), 155  bfs() (in module androguard.decompiler.dad.graph), 158  binary_infix() (in module andro-	ClassDefItem (class in andro- guard.core.bytecodes.dvm), 81  ClassHDefItem (class in andro- guard.core.bytecodes.dvm), 83  ClassManager (class in andro- guard.core.bytecodes.dvm), 83  clean_file_name() (in module androguard.misc), 180  clean_name_instruction() (in module andro- guard.core.bytecodes.dvm), 133  clear_notes() (andro- guard.core.analysis.analysis.DVMBasicBlock
BaseClass (class in andro- guard.decompiler.dad.instruction), 160  BasicBlock (class in andro- guard.decompiler.dad.basic_blocks), 153  BasicBlocks (class in andro- guard.core.analysis.analysis), 52  BasicReachDef (class in andro- guard.decompiler.dad.dataflow), 155  bfs() (in module androguard.decompiler.dad.graph), 158  binary_infix() (in module andro- guard.decompiler.dad.dast), 152	ClassDefItem (class in andro- guard.core.bytecodes.dvm), 81  ClassHDefItem (class in andro- guard.core.bytecodes.dvm), 83  ClassManager (class in andro- guard.core.bytecodes.dvm), 83  clean_file_name() (in module androguard.misc), 180  clean_name_instruction() (in module andro- guard.core.bytecodes.dvm), 133  clear_notes() (andro- guard.core.analysis.analysis.DVMBasicBlock method), 56
BaseClass (class in andro- guard.decompiler.dad.instruction), 160  BasicBlock (class in andro- guard.decompiler.dad.basic_blocks), 153  BasicBlocks (class in andro- guard.core.analysis.analysis), 52  BasicReachDef (class in andro- guard.decompiler.dad.dataflow), 155  bfs() (in module androguard.decompiler.dad.graph), 158  binary_infix() (in module andro-	ClassDefItem (class in andro- guard.core.bytecodes.dvm), 81  ClassHDefItem (class in andro- guard.core.bytecodes.dvm), 83  ClassManager (class in andro- guard.core.bytecodes.dvm), 83  clean_file_name() (in module androguard.misc), 180  clean_name_instruction() (in module andro- guard.core.bytecodes.dvm), 133  clear_notes() (andro- guard.core.analysis.analysis.DVMBasicBlock method), 56  clear_path() (in module andro-
BaseClass (class in andro- guard.decompiler.dad.instruction), 160  BasicBlock (class in andro- guard.decompiler.dad.basic_blocks), 153  BasicBlocks (class in andro- guard.core.analysis.analysis), 52  BasicReachDef (class in andro- guard.decompiler.dad.dataflow), 155  bfs() (in module androguard.decompiler.dad.graph), 158  binary_infix() (in module andro- guard.decompiler.dad.dast), 152  BinaryCompExpression (class in andro-	ClassDefItem (class in andro- guard.core.bytecodes.dvm), 81  ClassHDefItem (class in andro- guard.core.bytecodes.dvm), 83  ClassManager (class in andro- guard.core.bytecodes.dvm), 83  clean_file_name() (in module androguard.misc), 180  clean_name_instruction() (in module andro- guard.core.bytecodes.dvm), 133  clear_notes() (andro- guard.core.analysis.analysis.DVMBasicBlock method), 56
BaseClass (class in andro- guard.decompiler.dad.instruction), 160  BasicBlock (class in andro- guard.decompiler.dad.basic_blocks), 153  BasicBlocks (class in andro- guard.core.analysis.analysis), 52  BasicReachDef (class in andro- guard.decompiler.dad.dataflow), 155  bfs () (in module androguard.decompiler.dad.graph), 158  binary_infix() (in module andro- guard.decompiler.dad.dast), 152  BinaryCompExpression (class in andro- guard.decompiler.dad.instruction), 160  BinaryExpression (class in andro- guard.decompiler.dad.instruction), 160	ClassDefItem (class in andro- guard.core.bytecodes.dvm), 81  ClassHDefItem (class in andro- guard.core.bytecodes.dvm), 83  ClassManager (class in andro- guard.core.bytecodes.dvm), 83  clean_file_name() (in module androguard.misc), 180  clean_name_instruction() (in module andro- guard.core.bytecodes.dvm), 133  clear_notes() (andro- guard.core.analysis.analysis.DVMBasicBlock method), 56  clear_path() (in module andro- guard.decompiler.dad.dataflow), 155  clear_path_node() (in module andro- guard.decompiler.dad.dataflow), 155
BaseClass (class in andro- guard.decompiler.dad.instruction), 160  BasicBlock (class in andro- guard.decompiler.dad.basic_blocks), 153  BasicBlocks (class in andro- guard.core.analysis.analysis), 52  BasicReachDef (class in andro- guard.decompiler.dad.dataflow), 155  bfs() (in module androguard.decompiler.dad.graph), 158  binary_infix() (in module andro- guard.decompiler.dad.dast), 152  BinaryCompExpression (class in andro- guard.decompiler.dad.instruction), 160  BinaryExpression (class in andro- guard.decompiler.dad.instruction), 160  BinaryExpression2Addr (class in andro-	ClassDefItem (class in andro- guard.core.bytecodes.dvm), 81  ClassHDefItem (class in andro- guard.core.bytecodes.dvm), 83  ClassManager (class in andro- guard.core.bytecodes.dvm), 83  clean_file_name() (in module androguard.misc),
BaseClass (class in andro- guard.decompiler.dad.instruction), 160  BasicBlock (class in andro- guard.decompiler.dad.basic_blocks), 153  BasicBlocks (class in andro- guard.core.analysis.analysis), 52  BasicReachDef (class in andro- guard.decompiler.dad.dataflow), 155  bfs() (in module androguard.decompiler.dad.graph), 158  binary_infix() (in module andro- guard.decompiler.dad.dast), 152  BinaryCompExpression (class in andro- guard.decompiler.dad.instruction), 160  BinaryExpression (class in andro- guard.decompiler.dad.instruction), 160  BinaryExpression2Addr (class in andro- guard.decompiler.dad.instruction), 160	ClassDefItem (class in andro- guard.core.bytecodes.dvm), 81  ClassHDefItem (class in andro- guard.core.bytecodes.dvm), 83  ClassManager (class in andro- guard.core.bytecodes.dvm), 83  clean_file_name() (in module androguard.misc), 180  clean_name_instruction() (in module andro- guard.core.bytecodes.dvm), 133  clear_notes() (andro- guard.core.analysis.analysis.DVMBasicBlock method), 56  clear_path() (in module andro- guard.decompiler.dad.dataflow), 155  clear_path_node() (in module andro- guard.decompiler.dad.dataflow), 155  CMP (androguard.decompiler.dad.opcode_ins.Op at- tribute), 167
BaseClass (class in andro- guard.decompiler.dad.instruction), 160  BasicBlock (class in andro- guard.decompiler.dad.basic_blocks), 153  BasicBlocks (class in andro- guard.core.analysis.analysis), 52  BasicReachDef (class in andro- guard.decompiler.dad.dataflow), 155  bfs() (in module androguard.decompiler.dad.graph), 158  binary_infix() (in module andro- guard.decompiler.dad.dast), 152  BinaryCompExpression (class in andro- guard.decompiler.dad.instruction), 160  BinaryExpression (class in andro- guard.decompiler.dad.instruction), 160  BinaryExpression2Addr (class in andro- guard.decompiler.dad.instruction), 160  BinaryExpressionLit (class in andro- guard.decompiler.dad.instruction), 160  BinaryExpressionLit (class in andro-	ClassDefItem (class in androguard.core.bytecodes.dvm), 81  ClassHDefItem (class in androguard.core.bytecodes.dvm), 83  ClassManager (class in androguard.core.bytecodes.dvm), 83  clean_file_name() (in module androguard.misc), 180  clean_name_instruction() (in module androguard.core.bytecodes.dvm), 133  clear_notes() (androguard.core.analysis.analysis.DVMBasicBlock method), 56  clear_path() (in module androguard.decompiler.dad.dataflow), 155  clear_path_node() (in module androguard.decompiler.dad.dataflow), 155  CMP (androguard.decompiler.dad.opcode_ins.Op attribute), 167  cmpgdouble() (in module androguard.opcode_ins.Op attribute), 167  cmpgdouble() (in module androguard.opcode_ins.Op androguard.opcode_ins.Op attribute), 167
BaseClass (class in andro- guard.decompiler.dad.instruction), 160  BasicBlock (class in andro- guard.decompiler.dad.basic_blocks), 153  BasicBlocks (class in andro- guard.core.analysis.analysis), 52  BasicReachDef (class in andro- guard.decompiler.dad.dataflow), 155  bfs () (in module androguard.decompiler.dad.graph), 158  binary_infix() (in module andro- guard.decompiler.dad.dast), 152  BinaryCompExpression (class in andro- guard.decompiler.dad.instruction), 160  BinaryExpression (class in andro- guard.decompiler.dad.instruction), 160  BinaryExpressionLit (class in andro- guard.decompiler.dad.instruction), 160  BinaryExpressionLit (class in andro- guard.decompiler.dad.instruction), 160  BinaryExpressionLit (class in andro- guard.decompiler.dad.instruction), 160	ClassDefItem (class in androguard.core.bytecodes.dvm), 81 ClassHDefItem (class in androguard.core.bytecodes.dvm), 83 ClassManager (class in androguard.core.bytecodes.dvm), 83 Clean_file_name() (in module androguard.misc), 180 Clean_name_instruction() (in module androguard.core.bytecodes.dvm), 133 Clear_notes() (androguard.core.analysis.analysis.DVMBasicBlock method), 56 Clear_path() (in module androguard.decompiler.dad.dataflow), 155 Clear_path_node() (in module androguard.decompiler.dad.dataflow), 155 CMP (androguard.decompiler.dad.opcode_ins.Op attribute), 167 Cmpgdouble() (in module androguard.decompiler.dad.opcode_ins), 169
BaseClass (class in andro- guard.decompiler.dad.instruction), 160  BasicBlock (class in andro- guard.decompiler.dad.basic_blocks), 153  BasicBlocks (class in andro- guard.core.analysis.analysis), 52  BasicReachDef (class in andro- guard.decompiler.dad.dataflow), 155  bfs() (in module androguard.decompiler.dad.graph), 158  binary_infix() (in module andro- guard.decompiler.dad.dast), 152  BinaryCompExpression (class in andro- guard.decompiler.dad.instruction), 160  BinaryExpression2Addr (class in andro- guard.decompiler.dad.instruction), 160  BinaryExpressionLit (class in andro- guard.decompiler.dad.instruction), 160  BinaryExpressionLit (class in andro-	ClassDefItem (class in androguard.core.bytecodes.dvm), 81  ClassHDefItem (class in androguard.core.bytecodes.dvm), 83  ClassManager (class in androguard.core.bytecodes.dvm), 83  clean_file_name() (in module androguard.misc), 180  clean_name_instruction() (in module androguard.core.bytecodes.dvm), 133  clear_notes() (androguard.core.analysis.analysis.DVMBasicBlock method), 56  clear_path() (in module androguard.decompiler.dad.dataflow), 155  clear_path_node() (in module androguard.decompiler.dad.dataflow), 155  CMP (androguard.decompiler.dad.opcode_ins.Op attribute), 167  cmpgdouble() (in module androguard.opcode_ins.Op attribute), 167  cmpgdouble() (in module androguard.opcode_ins.Op androguard.opcode_ins.Op attribute), 167

cmpldouble()	(in	module	andro-	guard.decompiler.dad.opcode_ins), 169
guard.deco	ompiler.dad.o	pcode_ins), 1	69	constwide16() (in module andro-
cmplfloat()	(in	module	andro-	guard.decompiler.dad.opcode_ins), 169
=		pcode_ins), 1		constwide32() (in module andro-
cmplong()			andro-	guard.decompiler.dad.opcode_ins), 169
		pcode_ins), 1		constwidehigh16() (in module andro-
	•			nod guard.decompiler.dad.opcode_ins), 169
attribute),		200000000000000000000000000000000000000		copy () (androguard.decompiler.dad.node.LoopType
CodeItem (class		rd core bytec	odes dvm)	method), 167
85	in anarogua	ra.corc.oyicc	oues.avm),	copy () (androguard.decompiler.dad.node.NodeType
Color (class in and	droquard core	androconf)	145	method), 167
color_range()				copy_from() (andro-
	e.androconf),		anaro-	guard.decompiler.dad.basic_blocks.SwitchBlock
			I AVMI Dans	
comment () (andro	-	ytecoaes.axm	ı.AAMLFars	
property),		11 -	1	copy_from() (androguard.decompiler.dad.node.Node
	(in	module	andro-	method), 167
	ompiler.dad.u		7	crash() (androguard.core.analysis.auto.DefaultAndroAnalysis
complexToFloat			andro-	method), 64
	e.bytecodes.a.	xml), 144		create_adex() (andro-
compute_end()			(andro-	guard.core.analysis.auto.DefaultAndroAnalysis
	ompiler.dad.n	ode.Interval	method),	method), 64
166				create_apk() (andro-
compute_rpo()			(andro-	guard.core. analysis. auto. Default Andro Analysis
guard.dece	ompiler.dad.g	raph.Graph	method),	method), 65
158				create_arsc() (andro-
CondBlock	(class	in	andro-	guard.core.analysis.auto.DefaultAndroAnalysis
guard.dece	ompiler.dad.b	asic_blocks),	153	method), 65
Condition	(class	in	andro-	create_axml() (andro-
guard.dece	ompiler.dad.b	asic_blocks),	153	guard.core.analysis.auto.DefaultAndroAnalysis
ConditionalExp	_		andro-	method), 65
				create_dex() (andro-
ConditionalZEx	_			guard.core.analysis.auto.DefaultAndroAnalysis
		nstruction), 1		method), 65
Configuration				create_dey() (andro-
	e.androconf),		anaro	guard.core.analysis.auto.DefaultAndroAnalysis
			andro-	method), 65
	•			create_ipython_exports() (andro-
				guard.core.analysis.analysis.Analysis method),
		pcode_ins), 1		guara.core.anarysis.anarysis.Anarysis method), 47
	•	•		• ,
const4()	`	module		create_png() (in module andro-
	•	pcode_ins), 1		guard.decompiler.dad.util), 174
Constant	(class	in	andro-	create_python_export() (andro-
~	•	nstruction), 1	_	guard.core.bytecodes.dvm.DalvikVMFormat
constclass()	(in	module	andro-	method), 89
~	•	pcode_ins), 1	.69	create_xref() (andro-
consthigh16()	(in	module	andro-	guard.core.analysis.analysis.Analysis method),
guard.dec	ompiler.dad.o	pcode_ins), 1		48
construct()	(in	module	andro-	Cyan (androguard.core.androconf.Color attribute), 145
guard.dece	ompiler.dad.g	raph), 158		D
<pre>conststring()</pre>	(in	module	andro-	D
guard.deco	ompiler.dad.o	pcode_ins), 1	.69	DalvikCode (class in andro-
conststringjum	nbo() (ii	n module	andro-	guard.core.bytecodes.dvm), 86
		pcode_ins), 1	.69	DalvikOdexVMFormat (class in andro-
constwide()	(in	module	andro-	guard.core.bytecodes.dvm), 88

DalvikPacker (class in and		guard.core.androconf), 146
guard.core.bytecodes.dvm), 88		disable_print_colors() (in module andro-
DalvikVMFormat (class in and		guard.core.bytecode), 149
guard.core.bytecodes.dvm), 88	C	disasm() (androguard.core.bytecodes.dvm.FillArrayData
DBGBytecode (class in and	lro-	method), 103
guard.core.bytecodes.dvm), 85		disasm() (androguard.core.bytecodes.dvm.Instruction
DCode (class in androguard.core.bytecodes.dvm), 85		method), 105
dead_code_elimination() (in module and	dro-	disasm()(androguard.core.bytecodes.dvm.PackedSwitch
guard.decompiler.dad.dataflow), 155		method), 126
	dro-	disasm() (androguard.core.bytecodes.dvm.SparseSwitch
guard.core.bytecodes.dvm), 93		method), 128
	dro-	disassemble() (andro-
guard.core.bytecodes.dvm), 93		guard.core.bytecodes.dvm.DalvikVMFormat
dec_ind() (androguard.decompiler.dad.writer.Wri	iter	method), 89
method), 175		display_all() (andro-
decode () (in module androguard.core.mutf8), 145		guard.decompiler.decompiler.DecompilerDAD
DecompilerDAD (class in and	dro-	method), 177
guard.decompiler.decompiler), 177		display_all() (andro-
DecompilerDed (class in and		guard.decompiler.decompiler.DecompilerDed
guard.decompiler.decompiler), 177	110	method), 177
DecompilerDex2Fernflower (class in and	dro e	
	110-	
guard.decompiler.decompiler), 177	J	guard.decompiler.decompiler.DecompilerDex2Fernflower
DecompilerDex2Jad (class in and		method), 177
guard.decompiler.decompiler), 178		display_all() (andro-
DecompilerDex2WineJad (class in and	tro-	guard.decompiler.decompiler.DecompilerDex2Jad
guard.decompiler.decompiler), 178		method), 178
	tro-	display_all() (andro-
guard.decompiler.decompiler), 178	_	guard.decompiler.decompiler.DecompilerDex2WineJad
default_colors() (in module and		method), 178
guard.core.androconf), 146		display_all() (andro-
default_config() (and		guard.decompiler.decompiler.DecompilerJADX
guard.core.by tecodes.axml. ARSCR es Table College and the c	Config	method), 178
class method), 140	C	display_source() (andro-
DefaultAndroAnalysis (class in and	lro-	guard.decompiler.decompiler.DecompilerDAD
guard.core.analysis.auto), 62		method), 177
DefaultAndroLog (class in and	dro- c	display_source() (andro-
guard.core.analysis.auto), 66		guard.decompiler.decompiler.DecompilerDed
derived_sequence() (in module and	tro-	method), 177
guard.decompiler.dad.control_flow), 155		display_source() (andro-
descriptor() (and		guard.decompiler.decompiler.DecompilerDex2Fernflower
guard.core.analysis.analysis.MethodAnalysis	is	method), 177
property), 59		display_source() (andro-
descriptor() (and		guard.decompiler.decompiler.DecompilerDex2Jad
guard.core.bytecodes.dvm.EncodedMethod		method), 178
property), 97		display_source() (andro-
determineException() (in module and		guard.decompiler.decompiler.DecompilerDex2WineJad
guard.core.bytecodes.dvm), 133	110	method), 178
determineNext() (in module and	dro-	display_source() (andro-
guard.core.bytecodes.dvm), 133		guard.decompiler.decompiler.DecompilerJADX
Dex2Jar (class in androguard.decompiler.decompile	er).	method), 178
179		DIV (androguard.decompiler.dad.opcode_ins.Op at-
DirectoryAndroAnalysis (class in and	_	tribute), 167
guard.core.analysis.auto), 66		divdouble() (in module andro-
disable_colors() (in module and		guard.decompiler.dad.opcode_ins), 169

divdouble2addr() (in module andro-	method), 97
guard.decompiler.dad.opcode_ins), 169	enable_colors() (in module andro-
divfloat() (in module andro-	guard.core.androconf), 146
guard.decompiler.dad.opcode_ins), 169	<pre>enable_print_colors() (in module andro-</pre>
divfloat2addr() (in module andro-	guard.core.bytecode), 149
<pre>guard.decompiler.dad.opcode_ins), 169</pre>	encode () (in module androguard.core.mutf8), 145
divint() (in module andro-	EncodedAnnotation (class in andro-
<pre>guard.decompiler.dad.opcode_ins), 169</pre>	guard.core.bytecodes.dvm), 93
divint2addr() (in module andro-	EncodedArray (class in andro-
guard.decompiler.dad.opcode_ins), 169	guard.core.bytecodes.dvm), 94
	EncodedArrayItem (class in andro-
guard.decompiler.dad.opcode_ins), 169	guard.core.bytecodes.dvm), 94
	EncodedCatchHandler (class in andro-
guard.decompiler.dad.opcode_ins), 169	guard.core.bytecodes.dvm), 95
- · · · · · · · · · · · · · · · · · · ·	EncodedCatchHandlerList (class in andro-
guard.decompiler.dad.opcode_ins), 170	guard.core.bytecodes.dvm), 95
- · · · · · · · · · · · · · · · · · · ·	EncodedField (class in andro-
guard.decompiler.dad.opcode_ins), 170	guard.core.bytecodes.dvm), 96
	EncodedMethod (class in andro-
guard.decompiler.dad.graph), 159	guard.core.bytecodes.dvm), 97
	EncodedTypeAddrPair (class in andro-
guard.decompiler.dad.opcode_ins), 170	guard.core.bytecodes.dvm), 100
	EncodedValue (class in andro-
guard.decompiler.dad.opcode_ins), 170	guard.core.bytecodes.dvm), 101
	end() (androguard.core.bytecode.BuffHandle method),
guard.decompiler.dad.opcode_ins), 170	147
draw() (androguard.decompiler.dad.graph.Graph	
method), 158	property), 135
dummy () (in module androguard.decompiler.dad.dast),	
152	method), 175
DummyNode (class in andro-	ensure_final_value() (in module andro-
guard.decompiler.dad.dataflow), 155	guard.core.bytecodes.apk), 76
dump() (androguard.core.analysis.auto.AndroAuto	
method), 62	tribute), 167
dump () (androguard.core.analysis.auto.DefaultAndroAna	· ·
method), 65	ExceptionAnalysis (class in andro-
dump_file() (andro-	guard.core.analysis.analysis), 57
guard.core.analysis.auto.AndroAuto method),	
62	guard.core.analysis.analysis), 57
<pre>dump_file() (andro-</pre>	ExportObject (class in androguard.core.bytecodes.dvm), 101
method), 65	expression_stmt() (in module andro-
	guard.decompiler.dad.dast), 152
DvClass (class in andro- guard.decompiler.dad.decompile), 156	extends () (androguard.core.analysis.analysis.ClassAnalysis
DvMachine (class in andro-	property), 54
guard.decompiler.dad.decompile), 156	ExternalClass (class in andro-
DVMBasicBlock (class in andro-	guard.core.analysis.analysis), 57
guard.core.analysis.analysis), 56	ExternalMethod (class in andro-
DvMethod (class in andro-	guard.core.analysis.analysis), 57
guard.decompiler.dad.decompile), 157	zuuru.core.unui ysis.unui ysis), 57
znara.accompner.aaa.accompne), 137	F
E	•
each_params_by_register() (andro-	fetcher() (androguard.core.analysis.auto.DefaultAndroAnalysis method), 65
guard.core.bytecodes.dvm.EncodedMethod	memous, 03
zuara.core.o jiecoaes.avin.inicoaeamentoa	

<pre>fetcher() (androguard.core.analysis.auto.DirectoryAn</pre>	droAnalysis67
method), 66	$\verb finish()  (and roguard.core.analysis.auto.Default And roAnalysis $
field_access() (in module andro-	method), 66
guard.decompiler.dad.dast), 152	fix_checksums() (andro-
FieldAnalysis (class in andro-	guard.core.bytecodes.dvm.DalvikVMFormat
guard.core.analysis.analysis), 58	method), 89
	FLAG_COMPLEX (andro-
guard.core.bytecodes.dvm), 101	guard.core.bytecodes.axml.ARSCResTableEntry
FieldHIdItem (class in andro-	attribute), 140
guard.core.bytecodes.dvm), 102	FLAG_PUBLIC (andro-
FieldIdItem (class in androguard.core.bytecodes.dvm), 102	guard.core.bytecodes.axml.ARSCResTableEntry attribute), 140
· · · · · · · · · · · · · · · · · · ·	FLAG_WEAK (androguard.core.bytecodes.axml.ARSCResTableEntry
guard.core.bytecodes.dvm), 103	attribute), 140
FileNotPresent, 76	floattodouble() (in module andro-
files() (androguard.core.bytecodes.apk.APK prop-	guard.decompiler.dad.opcode_ins), 170
erty), 67	floattoint() (in module andro-
FillArrayData (class in andro-	guard.decompiler.dad.opcode_ins), 170
guard.core.bytecodes.dvm), 103	floattolong() (in module andro-
fillarraydata() (in module andro-	
guard.decompiler.dad.opcode_ins), 170	
fillarraydatapayload() (in module andro-	guard.core.bytecodes.axml.ARSCResStringPoolRef
guard.decompiler.dad.opcode_ins), 170	method), 139
	format_value() (in module andro-
	guard.core.bytecodes.axml), 144
FilledArrayExpression (class in andro- guard.decompiler.dad.instruction), 162	
	FormatClassToPython() (in module androguard.core.bytecode), 148
	FormatDescriptorToPython() (in module andro-
	guard.core.bytecode), 149
filter() (androguard.decompiler.decompiler.MethodFi	· · · · · · · · · · · · · · · · · · ·
method), 179	guard.core.bytecode), 149
	from_str() (androguard.core.mutf8.MUTF8String
guard.core.analysis.auto.DefaultAndroAnalysis	class method), 144
method), 66	full_name() (andro-
find() (androguard.core.mutf8.MUTF8String method),	
	property), 58
find_classes() (andro-	full_name() (andro-
guard.core.analysis.analysis.Analysis method), 48	guard.core.analysis.analysis.MethodAnalysis property), 59
find_fields() (andro-	full_name() (andro-
guard.core.analysis.analysis.Analysis method),	guard.core.bytecodes.dvm.EncodedMethod
48	property), 97
find_methods() (andro-	property), 51
guard.core.analysis.analysis.Analysis method),	G
48	
find_strings() (andro-	GenInvokeRetName (class in andro-
guard.core.analysis.analysis.Analysis method),	guard.decompiler.dad.graph), 157
48	GEQUAL (androguard.decompiler.dad.opcode_ins.Op at-
find_tags() (androguard.core.bytecodes.apk.APK	tribute), 167 get () (androguard.core.analysis.analysis.BasicBlocks
method), 67	method), 52
find_tags_from_xml() (andro- guard.core.bytecodes.apk.APK method),	get () (androguard.core.analysis.analysis.ExceptionAnalysis method), 57

(androguard.core.analysis.analysis.Exceptions get\_all() (androguard.decompiler.decompiler.DecompilerDed get () method), 57 method), 177 get() (androguard.core.bytecodes.dvm.FieldHIdItem get\_all() (androguard.decompiler.decompiler.DecompilerDex2Fernflov method), 102 method), 177 get() (androguard.core.bytecodes.dvm.MethodHIdItem get\_all() (androguard.decompiler.decompiler.DecompilerDex2Jad method), 124 method), 178 get\_all() (androguard.decompiler.decompiler.DecompilerDex2WineJac (androguard.core.bytecodes.dvm.ProtoHIdItem get() method), 127 method), 178 get() (androguard.core.bytecodes.dvm.StringDataItem get\_all() (androguard.decompiler.decompiler.DecompilerJADX *method*), 130 method), 178 (and roguard. core. by tecodes. dvm. Type HIdI temget\_all\_apks() (androguard.session.Session get() method), 131 method), 182 module get\_access\_class() (in androget\_all\_attribute\_value() (androguard.decompiler.dad.util), 175 guard.core.bytecodes.apk.APK method), get\_access\_field() module andro-67 (in guard.decompiler.dad.util), 175 get\_all\_dex() (androguard.core.bytecodes.apk.APK get\_access\_flags() (andromethod), guard.core.bytecodes.dvm.ClassDefItem method), 81 get\_all\_engine() (androget\_access\_flags() (androguard.core.bytecodes.dvm.ClassManager guard.core.bytecodes.dvm.EncodedField method), 83 method), 96 get all fields() (androget\_access\_flags() (androguard.core.bytecodes.dvm.DalvikVMFormat guard.core.bytecodes.dvm.EncodedMethod method), 89 method), 98 get\_analysis() (androguard.session.Session get\_access\_flags\_string() (andromethod), 182 guard.core.analysis.analysis.ExternalMethod get\_android\_manifest\_axml() (andromethod), 58 guard.core.bytecodes.apk.APK method), get\_access\_flags\_string() (androguard.core.analysis.analysis.MethodAnalysis get\_android\_manifest\_xml() (andromethod), 59 guard.core.bytecodes.apk.APK method), get\_access\_flags\_string() (andro-68 guard.core.bytecodes.dvm.ClassDefItem get\_android\_resources() (andromethod), 81 guard.core.bytecodes.apk.APK method), get\_access\_flags\_string() (androguard.core.bytecodes.dvm.EncodedField get\_androidversion\_code() (andromethod), 96 guard.core.bytecodes.apk.APK method), get\_access\_flags\_string() (androguard.core.bytecodes.dvm.EncodedMethod get\_androidversion\_name() (andromethod), 98 guard.core.bytecodes.apk.APK method), get\_access\_flags\_string() (in module androguard.core.bytecodes.dvm), 133 get\_annotated\_fields\_size() (androguard.core.bytecodes.dvm.AnnotationsDirectoryItem get\_access\_method() (in module androguard.decompiler.dad.util), 175 method), 79 (androget\_activities() get\_annotated\_methods\_size() (androguard.core.bytecodes.apk.APK method), guard.core.bytecodes.dvm.AnnotationsDirectoryItem method), 79 get\_addr() (androguard.core.bytecodes.dvm.EncodedTypeAddxPaiotated\_parameters\_size() (andromethod), 101 guard.core.bytecodes.dvm.AnnotationsDirectoryItem get\_address() (andromethod), 79 guard.core.bytecodes.dvm.EncodedMethod get\_annotation() (andromethod), 98 guard.core.bytecodes.dvm.AnnotationItem get\_all() (androguard.decompiler.decompiler.DecompilerDAD method),77 method), 177

Index 197

get\_annotation\_item()

(andro-

guard.core.bytecodes.dvm.AnnotationOffItem method), 78	<pre>get_arsc_info() (in module andro- guard.core.bytecodes.axml), 144</pre>
get_annotation_item() (andro-	get_ascii_string() (andro-
guard.core.bytecodes.dvm.ClassManager method), 83	guard.core.bytecodes.dvm.ClassManager method), 83
<pre>get_annotation_off() (andro- guard.core.bytecodes.dvm.AnnotationOffItem</pre>	<pre>get_ast() (androguard.core.bytecodes.dvm.ClassDefItem</pre>
<pre>method), 78 get_annotation_off_item() (andro-</pre>	<pre>get_ast() (androguard.decompiler.dad.dast.JSONWriter</pre>
guard.core.bytecodes.dvm.AnnotationSetItem method), 78	get_ast() (androguard.decompiler.dad.decompile.DvClass method), 156
<pre>get_annotation_off_item() (andro- guard.core.bytecodes.dvm.ClassManager</pre>	<pre>get_ast() (androguard.decompiler.dad.decompile.DvMachine</pre>
method), 83	<pre>get_ast() (androguard.decompiler.dad.decompile.DvMethod</pre>
<pre>get_annotation_set_item() (andro-</pre>	method), 157
guard.core.bytecodes.dvm.AnnotationsDirectory.	
method), 79	guard.decompiler.decompiler.DecompilerDAD
<pre>get_annotation_set_item() (andro-</pre>	method), 177
guard.core.bytecodes.dvm.ClassManager	get_ast_method() (andro-
method), 83	guard.decompiler.decompiler.DecompilerDAD
get_annotations() (andro-	method), 177
guard.core.bytecodes.dvm.ClassDefItem	get_attribute_value() (andro-
method), 81	guard.core.bytecodes.apk.APK method),
get_annotations_directory_item() (an-	69
droguard.core.bytecodes.dvm.ClassManager	get_basic_block() (andro-
method), 83	guard.core.analysis.analysis.BasicBlocks
get_annotations_off() (andro-	<pre>method), 52 get_basic_block_pos() (andro-</pre>
guard.core.bytecodes.dvm.AnnotationSetRefItem method), 78	guard.core.analysis.analysis.BasicBlocks
get_annotations_off() (andro-	method), 52
	get_basic_blocks() (andro-
guard.core.bytecodes.dvm.ClassDefItem method), 81	guard.core.analysis.analysis.MethodAnalysis
get_annotations_off() (andro-	method), 59
guard.core.bytecodes.dvm.FieldAnnotation	get_bc() (androguard.core.bytecodes.dvm.DalvikCode
method), 101	method), 87
get_annotations_off() (andro-	get_bool_resources() (andro-
guard.core.bytecodes.dvm.MethodAnnotation	guard.core.bytecodes.axml.ARSCParser
method), 123	method), 136
get_annotations_off() (andro-	get_BRANCH_DVM_OPCODES() (andro-
guard.core.bytecodes.dvm.ParameterAnnotation	guard.core.bytecodes.dvm.DalvikVMFormat
method), 127	method), 89
<pre>get_api_version() (andro-</pre>	<pre>get_buff() (androguard.core.bytecode.BuffHandle</pre>
guard.core.bytecodes.dvm.DalvikVMFormat	method), 147
method), 89	<pre>get_buff() (androguard.core.bytecodes.axml.AXMLPrinter</pre>
<pre>get_apkid() (in module andro-</pre>	method), 142
guard.core.bytecodes.apk), 76	<pre>get_buff() (androguard.core.bytecodes.dvm.DalvikOdexVMFormat</pre>
<pre>get_app_icon() (andro-</pre>	method), 88
guard.core.bytecodes.apk.APK method),	get_byte() (in module andro-
68	guard.core.bytecodes.dvm), 134
<pre>get_app_name()</pre> (andro-	get_bytecodes() (andro-
guard.core.bytecodes.apk.APK method),	guard.core.bytecodes.dvm.DebugInfoItem
69	method), 93
<pre>get_args() (in module andro-</pre>	<pre>get_bytecodes_method() (in module andro-</pre>
guard.decompiler.dad.opcode_ins), 170	guard.core.bytecodes.dvm), 134

get_bytecodes_methodx() (in module andro-	method), 84
<pre>guard.core.bytecodes.dvm), 134 get_call_graph() (andro-</pre>	<pre>get_class_data_off() (andro- guard.core.bytecodes.dvm.ClassDefItem</pre>
guard.core.analysis.analysis.Analysis method),	method), 82
49	get_class_idx() (andro-
<pre>get_catch_all_addr()</pre>	guard.core.bytecodes.dvm.ClassDefItem er method), 82
method), 95	get_class_idx() (andro-
get_certificate() (andro- guard.core.bytecodes.apk.APK method),	guard.core.bytecodes.dvm.ClassHDefItem method), 83
guara.core.byiecodes.apk.A1 K meinou), 69	get_class_idx() (andro-
<pre>get_certificate_der() (andro-</pre>	guard.core.bytecodes.dvm.FieldIdItem
guard.core.bytecodes.apk.APK method), 69	method), 102
<pre>get_certificate_name_string() (in module</pre>	<pre>get_class_idx() (andro- guard.core.bytecodes.dvm.MethodIdItem</pre>
androguard.util), 183	method), 124 get class manager() (andro-
get_certificates() (andro- guard.core.bytecodes.apk.APK method), 69	<pre>get_class_manager() (andro- guard.core.bytecodes.dvm.DalvikVMFormat method), 89</pre>
<pre>get_certificates_der_v2() (andro-</pre>	get_class_manager() (andro-
guard.core.bytecodes.apk.APK method), 69	guard.core.bytecodes.dvm.MapList method), 123
<pre>get_certificates_der_v3() (andro-</pre>	<pre>get_class_name() (andro-</pre>
guard.core.bytecodes.apk.APK method), 69	guard.core.analysis.analysis.ExternalMethod method), 58
get_certificates_v1() (andro-	get_class_name() (andro-
guard.core.bytecodes.apk.APK method), 69	guard.core.analysis.analysis.MethodAnalysis method), 59
<pre>get_certificates_v2() (andro-</pre>	get_class_name() (andro-
guard.core.bytecodes.apk.APK method), 70	guard.core.bytecodes.dvm.EncodedField method), 96
get_certificates_v3() (andro-	get_class_name() (andro-
guard.core.bytecodes.apk.APK method), 70	guard.core.bytecodes.dvm.EncodedMethod method), 98
get_class() (andro-	get_class_name() (andro-
guard.core.analysis.analysis.ClassAnalysis method), 54	guard.core.bytecodes.dvm.FieldIdItem method), 102
get_class() (andro-	get_class_name() (andro-
guard.core.bytecodes.dvm.DalvikVMFormat method), 89	guard.core.bytecodes.dvm.FieldIdItemInvalid method), 103
get_class() (andro-	get_class_name() (andro-
guard.decompiler.dad.decompile.DvMachine method), 156	guard.core.bytecodes.dvm.MethodIdItem method), 124
get_class_analysis() (andro-	get_class_name() (andro-
guard.core.analysis.analysis.Analysis method), 49	guard.core.bytecodes.dvm.MethodIdItemInvalid method), 125
<pre>get_class_annotations_off() (andro-</pre>	<pre>get_classes() (andro-</pre>
guard.core.bytecodes.dvm.AnnotationsDirectory. method), 79	Item guard.core.analysis.analysis.Analysis method), 49
get_class_data() (andro-	<pre>get_classes() (andro-</pre>
guard.core.bytecodes.dvm.ClassDefItem method), 82	guard.core.bytecodes.dvm.DalvikVMFormat method), 89
get_class_data_item() (andro-	get_classes() (andro-
guard.core.bytecodes.dvm.ClassManager	guard.decompiler.dad.decompile.DvMachine

method), 157 get_classes() (androguard.session.Sessio	guard.core.bytecodes.axml.ARSCResStringPoolRef method), 140
method), 182	get_data_value() (andro-
get_classes_def_item() (andro	y
guard.core.bytecodes.dvm.DalvikVMFormat	method), 140
method), 89	get_debug() (andro-
get_classes_names() (andro	
guard.core.bytecodes.dvm.DalvikVMFormat	method), 87
method), 89	get_debug() (andro-
get_cm_field() (andro	
guard.core.bytecodes.dvm.DalvikVMFormat	method), 98
method), 90	<pre>get_debug_info_item() (andro-</pre>
get_cm_method() (andro	
guard.core.bytecodes.dvm.DalvikVMFormat	method), 90
method), 90	get_debug_info_off() (andro-
<pre>get_cm_string() (andro</pre>	
guard.core.bytecodes.dvm.DalvikVMFormat	method), 87
method), 90	get_debug_off() (andro-
get_cm_type() (andro	
guard.core.bytecodes.dvm.DalvikVMFormat	method), 84
method), 90	<pre>get_declared_permissions() (andro-</pre>
<pre>get_code() (androguard.core.bytecodes.dvm.ClassM</pre>	
method), 84	70
<pre>get_code() (androguard.core.bytecodes.dvm.CodeIte</pre>	m get_declared_permissions_details() (an-
method), 85	droguard.core.bytecodes.apk.APK method),
<pre>get_code() (androguard.core.bytecodes.dvm.Encode</pre>	dMethod 70
method), 98	<pre>get_default_session() (in module andro-</pre>
<pre>get_code_off()</pre> (andro	- guard.misc), 180
guard.core.bytecodes.dvm.EncodedMethod	<pre>get_density() (andro-</pre>
guard.core.bytecodes.dvm.EncodedMethod method), 98	<pre>get_density() (andro- guard.core.bytecodes.axml.ARSCResTableConfig</pre>
	guard.core.by tecodes.axml. ARSCR es Table Config
<pre>method), 98 get_codes_item() (andro</pre>	guard.core.bytecodes.axml.ARSCResTableConfig method), 140 get_dependencies() (andro-
<pre>method), 98 get_codes_item() (andro</pre>	guard.core.bytecodes.axml.ARSCResTableConfig method), 140 get_dependencies() (andro- guard.core.bytecodes.dvm.DalvikOdexVMFormat
<pre>method), 98 get_codes_item() (andro</pre>	guard.core.bytecodes.axml.ARSCResTableConfig method), 140 get_dependencies() (andro- guard.core.bytecodes.dvm.DalvikOdexVMFormat method), 88
<pre>method), 98 get_codes_item()</pre>	guard.core.bytecodes.axml.ARSCResTableConfig method), 140 get_dependencies() (andro- guard.core.bytecodes.dvm.DalvikOdexVMFormat method), 88 get_dependencies() (andro-
method), 98  get_codes_item() (andro	guard.core.bytecodes.axml.ARSCResTableConfig method), 140 get_dependencies() (andro- guard.core.bytecodes.dvm.DalvikOdexVMFormat method), 88 get_dependencies() (andro- guard.core.bytecodes.dvm.OdexDependencies
method), 98  get_codes_item() (andro	guard.core.bytecodes.axml.ARSCResTableConfig method), 140 get_dependencies() (andro- guard.core.bytecodes.dvm.DalvikOdexVMFormat method), 88 get_dependencies() (andro- guard.core.bytecodes.dvm.OdexDependencies Writer method), 125
method), 98  get_codes_item() (andro	guard.core.bytecodes.axml.ARSCResTableConfig method), 140 get_dependencies() (andro- guard.core.bytecodes.dvm.DalvikOdexVMFormat method), 88 get_dependencies() (andro- guard.core.bytecodes.dvm.OdexDependencies Writer method), 125 get_descriptor() (andro-
method), 98  get_codes_item() (andro	guard.core.bytecodes.axml.ARSCResTableConfig method), 140 get_dependencies() (andro- guard.core.bytecodes.dvm.DalvikOdexVMFormat method), 88 get_dependencies() (andro- guard.core.bytecodes.dvm.OdexDependencies Vriter method), 125 get_descriptor() (andro- guard.core.analysis.analysis.ExternalMethod
method), 98  get_codes_item() (andro	guard.core.bytecodes.axml.ARSCResTableConfig method), 140 get_dependencies() (andro- guard.core.bytecodes.dvm.DalvikOdexVMFormat method), 88 get_dependencies() (andro- guard.core.bytecodes.dvm.OdexDependencies Vriter method), 125 get_descriptor() (andro- guard.core.analysis.analysis.ExternalMethod fig method), 58
method), 98  get_codes_item() (andro	guard.core.bytecodes.axml.ARSCResTableConfig method), 140 get_dependencies() (andro- guard.core.bytecodes.dvm.DalvikOdexVMFormat method), 88 get_dependencies() (andro- guard.core.bytecodes.dvm.OdexDependencies Vriter method), 125 get_descriptor() (andro- guard.core.analysis.analysis.ExternalMethod fig method), 58 get_descriptor() (andro-
method), 98  get_codes_item() (andro	guard.core.bytecodes.axml.ARSCResTableConfig method), 140 get_dependencies() (andro- guard.core.bytecodes.dvm.DalvikOdexVMFormat method), 88 get_dependencies() (andro- guard.core.bytecodes.dvm.OdexDependencies Vriter method), 125 get_descriptor() (andro- guard.core.analysis.analysis.ExternalMethod fig method), 58 get_descriptor() (andro- guard.core.bytecodes.dvm.EncodedField
method), 98  get_codes_item() (andro	guard.core.bytecodes.axml.ARSCResTableConfig method), 140 get_dependencies() (andro- guard.core.bytecodes.dvm.DalvikOdexVMFormat method), 88 get_dependencies() (andro- guard.core.bytecodes.dvm.OdexDependencies Vriter method), 125 get_descriptor() (andro- guard.core.analysis.analysis.ExternalMethod fig method), 58 get_descriptor() (andro- guard.core.bytecodes.dvm.EncodedField fig method), 96
method), 98  get_codes_item() (andro	guard.core.bytecodes.axml.ARSCResTableConfig method), 140 get_dependencies() (andro- guard.core.bytecodes.dvm.DalvikOdexVMFormat method), 88 get_dependencies() (andro- guard.core.bytecodes.dvm.OdexDependencies Vriter method), 125 get_descriptor() (andro- guard.core.analysis.analysis.ExternalMethod fig method), 58 get_descriptor() (andro- guard.core.bytecodes.dvm.EncodedField fig method), 96 get_descriptor() (andro-
method), 98  get_codes_item() (andro	guard.core.bytecodes.axml.ARSCResTableConfig method), 140 get_dependencies() (andro- guard.core.bytecodes.dvm.DalvikOdexVMFormat method), 88 get_dependencies() (andro- guard.core.bytecodes.dvm.OdexDependencies Vriter method), 125 get_descriptor() (andro- guard.core.analysis.analysis.ExternalMethod fig method), 58 get_descriptor() (andro- guard.core.bytecodes.dvm.EncodedField fig method), 96 get_descriptor() (andro- ResStringPool@ufird.core.bytecodes.dvm.EncodedMethod
method), 98  get_codes_item() (andro	guard.core.bytecodes.axml.ARSCResTableConfig method), 140 get_dependencies() (andro- guard.core.bytecodes.dvm.DalvikOdexVMFormat method), 88 get_dependencies() (andro- guard.core.bytecodes.dvm.OdexDependencies Vriter method), 125 get_descriptor() (andro- guard.core.analysis.analysis.ExternalMethod fig method), 58 get_descriptor() (andro- guard.core.bytecodes.dvm.EncodedField fig method), 96 get_descriptor() (andro- ResStringPool@ufird.core.bytecodes.dvm.EncodedMethod method), 98
method), 98 get_codes_item() (andro guard.core.bytecodes.dvm.DalvikVMFormat method), 90 get_color_resources() (andro guard.core.bytecodes.axml.ARSCParser method), 136 get_cond() (androguard.decompiler.dad.dast.JSONV method), 151 get_config_name_friendly() (andro guard.core.bytecodes.axml.ARSCResTableCon method), 140 get_country() (andro guard.core.bytecodes.axml.ARSCResTableCon method), 140 get_data() (androguard.core.bytecodes.axml.ARSCR method), 139 get_data() (androguard.core.bytecodes.dvm.FillArro	guard.core.bytecodes.axml.ARSCResTableConfig method), 140 get_dependencies() (andro- guard.core.bytecodes.dvm.DalvikOdexVMFormat method), 88 get_dependencies() (andro- guard.core.bytecodes.dvm.OdexDependencies Vriter method), 125 get_descriptor() (andro- guard.core.analysis.analysis.ExternalMethod fig method), 58 get_descriptor() (andro- guard.core.bytecodes.dvm.EncodedField fig method), 96 get_descriptor() (andro- ResStringPoolRufard.core.bytecodes.dvm.EncodedMethod method), 98 ayDatat_descriptor() (andro-
method), 98  get_codes_item() (andro	guard.core.bytecodes.axml.ARSCResTableConfig method), 140 get_dependencies() (andro- guard.core.bytecodes.dvm.DalvikOdexVMFormat method), 88 get_dependencies() (andro- guard.core.bytecodes.dvm.OdexDependencies Vriter method), 125 get_descriptor() (andro- guard.core.analysis.analysis.ExternalMethod fig method), 58 get_descriptor() (andro- guard.core.bytecodes.dvm.EncodedField fig method), 96 get_descriptor() (andro- ResStringPoolRufird.core.bytecodes.dvm.EncodedMethod method), 98 tyDatat_descriptor() (andro- guard.core.bytecodes.dvm.FieldIdItem
method), 98  get_codes_item() (andro	guard.core.bytecodes.axml.ARSCResTableConfig method), 140 get_dependencies() (andro- guard.core.bytecodes.dvm.DalvikOdexVMFormat method), 88 get_dependencies() (andro- guard.core.bytecodes.dvm.OdexDependencies Vriter method), 125 get_descriptor() (andro- guard.core.analysis.analysis.ExternalMethod fig method), 58 get_descriptor() (andro- guard.core.bytecodes.dvm.EncodedField fig method), 96 get_descriptor() (andro- ResStringPoolRufird.core.bytecodes.dvm.EncodedMethod method), 98 ayDatat_descriptor() (andro- guard.core.bytecodes.dvm.FieldIdItem method), 102
method), 98  get_codes_item() (andro	guard.core.bytecodes.axml.ARSCResTableConfig method), 140 get_dependencies() (andro- guard.core.bytecodes.dvm.DalvikOdexVMFormat method), 88 get_dependencies() (andro- guard.core.bytecodes.dvm.OdexDependencies Vriter method), 125 get_descriptor() (andro- guard.core.analysis.analysis.ExternalMethod fig method), 58 get_descriptor() (andro- guard.core.bytecodes.dvm.EncodedField fig method), 96 get_descriptor() (andro- ResStringPoolRufrd.core.bytecodes.dvm.EncodedMethod method), 98 nyDatat_descriptor() (andro- guard.core.bytecodes.dvm.FieldIdItem lataItem method), 102 get_descriptor() (andro- andro- lataItem method), 102 get_descriptor() (andro- lataItem method), 102
method), 98  get_codes_item() (andro	guard.core.bytecodes.axml.ARSCResTableConfig method), 140 get_dependencies() (andro- guard.core.bytecodes.dvm.DalvikOdexVMFormat method), 88 get_dependencies() (andro- guard.core.bytecodes.dvm.OdexDependencies Vriter method), 125 get_descriptor() (andro- guard.core.analysis.analysis.ExternalMethod fig method), 58 get_descriptor() (andro- guard.core.bytecodes.dvm.EncodedField fig method), 96 get_descriptor() (andro- ResStringPoolRufird.core.bytecodes.dvm.EncodedMethod method), 98 ayDatat_descriptor() (andro- guard.core.bytecodes.dvm.FieldIdItem method), 102 get_descriptor() (andro- guard.core.bytecodes.dvm.FieldIdItemInvalid
method), 98  get_codes_item() (andro	guard.core.bytecodes.axml.ARSCResTableConfig method), 140  get_dependencies() (andro- guard.core.bytecodes.dvm.DalvikOdexVMFormat method), 88  get_dependencies() (andro- guard.core.bytecodes.dvm.OdexDependencies Vriter method), 125  get_descriptor() (andro- guard.core.analysis.analysis.ExternalMethod fig method), 58  get_descriptor() (andro- guard.core.bytecodes.dvm.EncodedField fig method), 96  get_descriptor() (andro- ResStringPoolRefird.core.bytecodes.dvm.EncodedMethod method), 98  ayDatat_descriptor() (andro- guard.core.bytecodes.dvm.FieldIdItem method), 102  get_descriptor() (andro- guard.core.bytecodes.dvm.FieldIdItemInvalid filef method), 103
method), 98  get_codes_item() (andro	guard.core.bytecodes.axml.ARSCResTableConfig method), 140 get_dependencies() (andro- guard.core.bytecodes.dvm.DalvikOdexVMFormat method), 88 get_dependencies() (andro- guard.core.bytecodes.dvm.OdexDependencies Vriter method), 125 get_descriptor() (andro- guard.core.analysis.analysis.ExternalMethod fig method), 58 get_descriptor() (andro- guard.core.bytecodes.dvm.EncodedField fig method), 96 get_descriptor() (andro- ResStringPoolRafird.core.bytecodes.dvm.EncodedMethod method), 98 ayDatat_descriptor() (andro- guard.core.bytecodes.dvm.FieldIdItem DataItem method), 102 get_descriptor() (andro- guard.core.bytecodes.dvm.FieldIdItemInvalid method), 103 get_descriptor() (andro-

method), 124		get_end()	(androguard	.decompiler.dad.n	ode.Node
<pre>get_descriptor()</pre>	(andro-	method	<i>d</i> ), 167		
guard.core.bytecodes.dvm.MethodIdIt	emInvalid				(andro-
method), 125		_		s.dvm.ClassMana	iger
<pre>get_descriptor_idx()</pre>	(andro-	method	//		
guard.core.bytecodes.dvm.TypeIdItem	method),				(andro-
131	( 1	_		analysis.Exceptio	ns
get_descriptor_idx_value()	(andro-	method	* *	÷ = ()	( an dua
guard.core.bytecodes.dvm.TypeIdItem 132		guard.	core.analysis.	analysis.DVMBas	(andro- sicBlock
get_details_permissions()	(andro-	method get_external		()	(andro-
70		guard.		analysis.Analysis	,
<pre>get_determineException()</pre>	(andro-	49			
guard.core.bytecodes.dvm.DalvikVMF	ormat	get_feature:		LADE	(andro-
method), 90	( an dua	guara.a 71	core.bytecode	es.apк.APK	method),
<pre>get_determineNext()</pre>	(andro-	get_field()			(andro-
method), 90	отти	-	core analysis	analysis.FieldAnd	
get_dex() (androguard.core.bytecodes.	.apk.APK	method		anaiysis.i iciai in	uysus
method), 70		<pre>get_field()</pre>	.,,		(andro-
<pre>get_dex_names()</pre>	(andro-		core.bytecode	s.dvm.ClassMana	,
	method),	method			
70		<pre>get_field()</pre>			(andro-
<pre>get_digest_by_class()</pre>	(andro-	guard.o method		s.dvm.DalvikVMI	Format
get_dimen_resources()	(andro-	get_field_a			(andro-
guard.core.bytecodes.axml.ARSCPars				analysis.Analysis	method),
method), 137		50			
<pre>get_direct_methods()</pre>		get_field_a			(andro-
guard.core.bytecodes.dvm.ClassDatal method), 80	tem	guard.o method		analysis.ClassAn	alysis
<pre>get_direct_methods_size()</pre>	(andro-	get_field_a	nnotation	s()	(andro-
guard.core.bytecodes.dvm.ClassDatal method), 80	tem	guard.o method		s.dvm.Annotation	sDirectoryItem
<pre>get_effective_target_sdk_version</pre>	() (an-				andro-
droguard.core.bytecodes.apk.APK	method),			ad.decompile), 15	
70		get_field_de			(andro-
<pre>get_element()</pre>	(andro-	_		s.dvm.DalvikVMI	Format
guard.core.bytecodes.apk.APK	method),	method			. 1
70	(an dro	get_field_ic		ng dum EngadadEi	(andro-
<pre>get_elements()</pre>	(andro- method),	method		s.dvm.EncodedFi	ени
71	memou),	get_field_id			(andro-
<pre>get_elements()</pre>	(andro-			s.dvm.FieldAnnoi	`
guard.core.bytecodes.dvm.EncodedAn	`	method	· ·		
method), 93		get_field_id			(andro-
<pre>get_encoded_array_item()</pre>	(andro-			s.dvm.EncodedFi	eld
guard.core.bytecodes.dvm.ClassMana	ger	method	1), 96		
method), 84		get_field_re			(andro-
<pre>get_end() (androguard.core.analysis.analysi     method), 56</pre>	s.DVMBas	icBlock guard.o method		es.dvm.ClassMana	iger
<pre>get_end() (androguard.decompiler.dad.node</pre>	e.Interval				(andro-
method), 166		guard.	core.analysis.	analysis.Analysis	method),

50 get_fields() (androguard.core.analysis.analysis.ClassAnalysis	<pre>get_formatted_operands() (andro- guard.core.bytecodes.dvm.PackedSwitch method), 126</pre>
method), 54 get_fields() (andro-guard.core.bytecodes.dvm.ClassDataItem	get_formatted_operands() (andro- guard.core.bytecodes.dvm.SparseSwitch method), 128
method), 80 get_fields() (andro-	get_handler_off() (andro- guard.core.bytecodes.dvm.TryItem method),
<pre>guard.core.bytecodes.dvm.ClassDefItem     method), 82 get fields() (andro-</pre>	131 get_handlers() (androguard.core.bytecodes.dvm.DalvikCode
get_fields() (andro- guard.core.bytecodes.dvm.DalvikVMFormat method), 90	method), 87 get_handlers() (andro-
${\tt get\_fields\_class()} \qquad \qquad (and ro-\\ \textit{guard.core.bytecodes.dvm.DalvikVMFormat}$	guard.core.bytecodes.dvm.EncodedCatchHandler method), 95
<pre>method), 90 get_fields_id_item() (andro- guard.core.bytecodes.dvm.DalvikVMFormat</pre>	<pre>get_head() (androguard.decompiler.dad.node.Interval</pre>
method), 91 get_file() (androguard.core.bytecodes.apk.APK	method), 167
<pre>method),71 get_filename() (andro- guard.core.bytecodes.apk.APK method),</pre>	guard.core.bytecodes.dvm.DalvikVMFormat method), 91
<pre>guard.core.bytecodes.apk.APK method), 71 get_filename_by_class() (andro-</pre>	get_hex() (androguard.core.bytecodes.dvm.FillArrayData method), 104 get_hex() (androguard.core.bytecodes.dvm.Instruction
	method), 105 get_hex() (androguard.core.bytecodes.dvm.PackedSwitch
<pre>method),71 get_files_crc32()</pre>	method), 126 get_hex() (androguard.core.bytecodes.dvm.SparseSwitch method), 128
71 get_files_information() (andro-	<pre>get_id() (androguard.core.bytecodes.axml.ARSCParser</pre>
<pre>guard.core.bytecodes.apk.APK method), 71 get_files_types() (andro-</pre>	<pre>get_id_resources() (andro- guard.core.bytecodes.axml.ARSCParser method), 137</pre>
guard.core.bytecodes.apk.APK method), 71	
<pre>get_format() (andro- guard.core.bytecodes.dvm.DalvikVMFormat method), 91</pre>	<pre>get_index() (andro- guard.core.bytecodes.axml.ARSCResTableEntry method), 140</pre>
${\tt get\_format()} \ \ (and roguard.session.Session\ method),\\ 183$	<pre>get_information() (andro- guard.core.bytecodes.dvm.EncodedMethod</pre>
<pre>get_format_type() (andro- guard.core.bytecodes.dvm.DalvikOdexVMFormate</pre> <pre>method), 88</pre>	method), 98  It get_init_value() (andro- guard.core.bytecodes.dvm.EncodedField
<pre>get_format_type() (andro- guard.core.bytecodes.dvm.DalvikVMFormat method),91</pre>	<pre>method), 96 get_ins() (androguard.decompiler.dad.basic_blocks.BasicBlock     method), 153</pre>
${\tt get\_formatted\_operands()} \qquad \qquad (and ro-\\ \textit{guard.core.bytecodes.dvm.FillArrayData}$	<pre>get_ins() (androguard.decompiler.dad.basic_blocks.Condition</pre>
<pre>method), 104 get_formatted_operands() (andro- guard.core.bytecodes.dvm.Instruction method),</pre>	
105	method), 154

<pre>get_ins_from_loc() (andro-</pre>	<pre>get_integer_resources() (andro- guard.core.bytecodes.axml.ARSCParser method), 137</pre>
	get_intent_filters() (andro-
guard.core.bytecodes.dvm.DCode method), 86	guard.core.bytecodes.apk.APK method), 71
<pre>get_ins_size() (andro-</pre>	<pre>get_interfaces() (andro-</pre>
guard.core.bytecodes.dvm.DalvikCode method), 87	guard.core.bytecodes.dvm.ClassDefItem method), 82
<pre>get_insn() (androguard.core.bytecodes.dvm.DCode</pre>	<pre>get_interfaces_off() (andro-</pre>
method), 86	guard.core.bytecodes.dvm.ClassDefItem
get_insn_count() (andro-	method), 82
guard.core.bytecodes.dvm.TryItem method),	get_internal_classes() (andro-
131	guard.core.analysis.analysis.Analysis method),
get_insns_size() (andro-	50
guard.core.bytecodes.dvm.DalvikCode method), 87	<pre>get_item() (androguard.core.bytecodes.dvm.MapItem</pre>
get_instance_fields() (andro-	<pre>get_item_by_offset() (andro-</pre>
guard.core.bytecodes.dvm.ClassDataItem method), 80	guard.core.bytecodes.dvm.ClassManager method), 84
<pre>get_instance_fields_size() (andro-</pre>	get_item_type() (andro-
guard.core.bytecodes.dvm.ClassDataItem	guard.core.bytecodes.dvm.MapList method),
method), 80	123
<pre>get_instruction() (andro-</pre>	get_items() (andro-
guard.core.bytecodes.dvm.DalvikCode method), 87	guard.core.bytecodes.axml.ARSCParser method), 137
<pre>get_instruction() (andro-</pre>	<pre>get_jar() (androguard.decompiler.decompiler.Dex2Jar</pre>
guard.core.bytecodes.dvm.DCode method),	method), 179
86	get_key_data() (andro-
get_instruction() (andro-	guard.core.bytecodes.axml.ARSCResTableEntry
guard.core.bytecodes.dvm.EncodedMethod	method), 140
method), 99	get_keys() (androguard.core.bytecodes.dvm.PackedSwitch
get_instruction() (in module andro-	method), 126
guard.core.bytecodes.dvm), 134	get_keys() (androguard.core.bytecodes.dvm.SparseSwitch
get_instruction_payload() (in module andro- guard.core.bytecodes.dvm), 134	<pre>method), 128 get_kind() (androguard.core.bytecodes.dvm.Instruction</pre>
get_instructions() (andro-	method), 105
guard.core.analysis.analysis.DVMBasicBlock	get_kind() (in module andro-
method), 56	guard.core.bytecodes.dvm), 134
get_instructions() (andro-	get_language() (andro-
guard.core.bytecodes.dvm.DCode method),	guard.core.bytecodes.axml.ARSCResTableConfig
86	method), 140
<pre>get_instructions() (andro-</pre>	<pre>get_language_and_region() (andro-</pre>
guard.core.bytecodes.dvm.EncodedMethod method), 99	guard.core.bytecodes.axml.ARSCResTableConfig method), 140
get_instructions() (andro-	get_last()(androguard.core.analysis.analysis.DVMBasicBlock
guard.core.bytecodes.dvm.LinearSweepAlgorithi	
static method), 122	get_last_length() (andro-
<pre>get_instructions_idx() (andro-</pre>	guard.core.analysis.analysis.DVMBasicBlock
guard.core.bytecodes.dvm.EncodedMethod	method), 56
method), 99	get_lazy_analysis() (andro-
<pre>get_int_value() (andro-</pre>	guard.core.bytecodes.dvm.ClassManager
guard. decompiler. dad. instruction. Constant	method), 84
method), 161	get_len_methods() (andro-

guard.core.bytecodes.dvm.DalvikVMF	Format		guard.core.bytecodes.dvm.EncodedCa	tchHandler
method), 91	( 1		method), 95	
<pre>get_length()</pre>		get_ler		(andro-
guard.core.analysis.analysis.MethodA method), 59	nalysis		guard.core.bytecodes.dvm.EncodedCa method), 95	tchHandlerList
get_length()	(andro-	get_ler	ngth()	(andro-
guard.core.bytecodes.dvm.Annotation method), 77	Element		guard.core.bytecodes.dvm.EncodedMe method), 99	rthod
get_length()	(andro-	get_ler	ngth()	(andro-
guard.core.bytecodes.dvm.Annotation method), 77	Item		guard.core.bytecodes.dvm.EncodedTypmethod), 101	peAddrPair
get_length()	(andro-	get_ler	ngth()	(andro-
guard.core.bytecodes.dvm.Annotation method), 78	OffItem		guard.core.bytecodes.dvm.EncodedValmethod), 101	lue
get_length()	(andro-	get_ler	ngth()	(andro-
guard.core.bytecodes.dvm.Annotations method), 80	sDirectory.		guard.core.bytecodes.dvm.FieldAnnote method), 102	ation
get_length()	(andro-	get_ler	ngth()	(andro-
guard.core.bytecodes.dvm.Annotation method), 78	SetItem		guard.core.bytecodes.dvm.FieldHIdIte method), 102	m
get_length()		get_ler	ngth()	(andro-
guard.core.bytecodes.dvm.Annotation method), 79	SetRefList		guard.core.bytecodes.dvm.FieldIdItem method), 103	
get_length()		get_ler		(andro-
guard.core.bytecodes.dvm.ClassDataI method), 81	tem		guard.core.bytecodes.dvm.FillArrayDomethod), 104	ata
get_length()	(andro-	get_ler	ngth()	(andro-
guard.core.bytecodes.dvm.ClassDefIte method), 82	em		guard.core.bytecodes.dvm.HeaderItem method), 104	l.
get_length()	(andro-	get_ler	ngth()	(andro-
guard.core.bytecodes.dvm.ClassHDefl method), 83	Item		guard.core.bytecodes.dvm.Instruction 105	method),
get_length()	(andro-	get_ler	ngth()	(andro-
guard.core.bytecodes.dvm.CodeItem 85	method),		guard.core.bytecodes.dvm.MapItem 122	method),
get_length()	(andro-	get_ler	ngth()	(andro-
guard.core.bytecodes.dvm.DalvikCode method), 87	?		guard.core.bytecodes.dvm.MapList 123	method),
get_length()	(andro-	get_ler	ngth()	(andro-
guard.core.bytecodes.dvm.DCode 86	method),		guard.core.bytecodes.dvm.MethodAnn method), 123	otation
get_length()	(andro-	get_ler	ngth()	(andro-
guard.core.bytecodes.dvm.DebugInfoI method), 93	temEmpty		guard.core.bytecodes.dvm.MethodHId method), 124	Item
get_length()	(andro-	get_ler	ngth()	(andro-
guard.core.bytecodes.dvm.EncodedAn method), 93	notation		guard.core.bytecodes.dvm.MethodIdItamethod), 124	em
get_length()	(andro-	get_ler	ngth()	(andro-
guard.core.bytecodes.dvm.EncodedArd method), 94	ray		guard.core.bytecodes.dvm.PackedSwittmethod), 126	ch
get_length()	(andro-	get_ler		(andro-
guard.core.bytecodes.dvm.EncodedArd method), 94	rayItem		guard.core.bytecodes.dvm.ParameterA method), 127	nnotation
get_length()	(andro-	get_ler		(andro-

guard.core.bytecodes.dvm.ProtoHIdI	em	method), 93	
method), 127		<pre>get_list() (androguard.core.bytecodes.dvm.Ar</pre>	nnotation Set Ref List
<pre>get_length()</pre>	(andro-	method), 79	
guard.core.bytecodes.dvm.ProtoIdIte method), 127	n	<pre>get_list() (androguard.core.bytecodes.dvm.Ex method), 95</pre>	ncodedCatchHandlerList
<pre>get_length()</pre>	(andro-	<pre>get_list() (androguard.core.bytecodes.dvm.Fa</pre>	ieldIdItem
guard.core.bytecodes.dvm.SparseSwi	tch	method), 103	
method), 129		<pre>get_list() (androguard.core.bytecodes.dvm.Fr</pre>	ieldIdItemInvalid
<pre>get_length()</pre>	(andro-	method), 103	
guard.core.bytecodes.dvm.StringData method), 130	Item	<pre>get_list() (androguard.core.bytecodes.dvm.M</pre>	lethodIdItem
<pre>get_length()</pre>	(andro-	<pre>get_list() (androguard.core.bytecodes.dvm.M</pre>	lethodIdItemInvalid
guard.core.bytecodes.dvm.StringIdIte	m	method), 125	
method), 130		<pre>get_list() (androguard.core.bytecodes.dvm.Ty</pre>	ypeList
<pre>get_length()</pre>	(andro-	<i>method</i> ), 132	
guard.core.bytecodes.dvm.TryItem	method),	<pre>get_literals()</pre>	andro-
131		guard.core.bytecodes.dvm.Instruction m	ethod),
<pre>get_length()</pre>	(andro-	105	
guard.core.bytecodes.dvm.TypeHIdIte	em	· · · · · · · · · · · · · · · · · ·	(andro-
method), 131		guard.core.bytecodes.dvm.Instruction111	n
<pre>get_length()</pre>	(andro-	method), 107	
guard.core.bytecodes.dvm.TypeIdIten	n method),		(andro-
132		guard.core.bytecodes.dvm.Instruction211	h
<pre>get_length()</pre>	(andro-	method), 109	
guard.core.bytecodes.dvm.TypeItem	method),		(andro-
132		guard.core.bytecodes.dvm.Instruction21s	S
<pre>get_length()</pre>	(andro-	method), 110	
guard.core.bytecodes.dvm.TypeList	method),		(andro-
132		guard.core.bytecodes.dvm.Instruction22l	b
<pre>get_lhs() (androguard.decompiler.dad.instr</pre>	uction.Assi		
method), 160		- · · · · · · · · · · · · · · · · · · ·	andro-
	uction.Con	ditionalExp <b>gussidn</b> ore.bytecodes.dvm.Instruction22s	S
method), 161		method), 112	
<pre>get_lhs() (androguard.decompiler.dad.instr</pre>	uction.Con		andro-
method), 161		guard.core.bytecodes.dvm.Instruction311	į
<pre>get_lhs() (androguard.decompiler.dad.instr</pre>	uction.Insta		
method), 163			andro-
	uction.IRF	orm guard.core.bytecodes.dvm.Instruction511	,
method), 162		method), 120	,
<pre>get_lhs() (androguard.decompiler.dad.instr</pre>	uction.Mov		andro-
method), 164		guard.decompiler.dad.basic_blocks.Basic_	cBlock
<pre>get_lhs() (androguard.decompiler.dad.instr</pre>	uction.Mov	*	. 1
method), 164			(andro-
	испоп.мор.	Expression guard.decompiler.dad.basic_blocks.Cond	аноп
method), 165		method), 153	· I
get_lhs() (androguard.decompiler.dad.instr	испоп.кен		(andro-
method), 165		guard.decompiler.dad.basic_blocks.Loop	ОВІОСК
get_lhs() (androguard.decompiler.dad.instr	исноп.ман		· I
method), 165	(an J		andro-
get_libraries()	(andro-	guard.decompiler.dad.basic_blocks.Shor	истсиныноск
guard.core.bytecodes.apk.APK	method),	method), 154	andro
71	(an Inc		andro-
get_line_start()	(andro-	guard.decompiler.dad.dataflow.DummyN	ioue
guard.core.bytecodes.dvm.DebugInfo	uem	method), 155	

<pre>get_locales() (andro</pre>	- get_method_idx() (andro- guard.core.bytecodes.dvm.EncodedMethod method), 99
<pre>get_locals() (andro</pre>	
<pre>get_main_activities() (andro</pre>	<del>-</del>
<pre>get_main_activity() (andro</pre>	
<pre>get_max_sdk_version() (andro</pre>	
<pre>get_method() (andro</pre>	· · · · · · · · · · · · · · · · · · ·
<pre>get_method()</pre>	guard.core.analysis.analysis.ClassAnalysis method), 54
<pre>get_method()</pre>	- get_methods() (andro- guard.core.analysis.analysis.ExternalClass method), 57
<pre>get_method()</pre>	- get_methods() (andro- guard.core.bytecodes.dvm.ClassDataItem method), 81
<pre>get_method() (andro</pre>	guard.core.bytecodes.dvm.ClassDefItem method), 82  (andro- guard.core.bytecodes.dvm.ClassDefItem
<pre>get_method()</pre>	get_methods() (androguard.core.bytecodes.dvm.DalvikVMFormat method), 91
<pre>get_method_analysis() (andro</pre>	
guard.core.analysis.analysis.Analysis method) 50	method), 156
<pre>get_method_analysis() (andro</pre>	- get_methods_class() (andro- guard.core.bytecodes.dvm.DalvikVMFormat method), 91
<pre>get_method_analysis_by_name() (andro</pre>	
<pre>get_method_annotations() (andro</pre>	
<pre>get_method_by_idx() (andro</pre>	- get_min_sdk_version() (andro- guard.core.bytecodes.apk.APK method), 72
<pre>get_method_by_name() (andro</pre>	· · · · · · · · · · · · · · · · · · ·
<pre>get_method_descriptor() (andro</pre>	- get_name() (androguard.core.analysis.analysis.DVMBasicBlock method), 56 get_name() (androguard.core.analysis.analysis.ExternalClass

method),				le_from_loc()	(andro-
<pre>get_name() (an     method),</pre>		.ExternalMeth		guard.decompiler.dad.graph.Graph 158	method),
get_name()	(androguard.core.bytecode.Tm	<i>pBlock</i> get_	_not	es()	(andro-
method),	. 149			guard.core.analysis.analysis.DVMBas	ricBlock
get_name()(an	droguard.core.bytecodes.axml.A	RSCResTable	Packa	ngethod), 56	
method),	-			es()	(andro-
get_name()(an	droguard.core.bytecodes.dvm.C	_		guard.core.bytecodes.dvm.FillArrayD	ata
method),		v		method), 104	
get_name()(an	droguard.core.bytecodes.dvm.E	ncodedFi <b>eJe</b> t_	_not	es()	(andro-
method),	. 96	_		guard.core.bytecodes.dvm.PackedSwit	tch .
get name()(an	droguard.core.bytecodes.dvm.E	ncodedMethod		method), 126	
method),				es()	(andro-
· · · · · · · · · · · · · · · · · · ·	droguard.core.bytecodes.dvm.F	_		guard.core.bytecodes.dvm.SparseSwit	,
method),	-			method), 129	
		ieldIdItem <b>In</b> va		() (androguard.core.bytecodes.dvm.A	AnnotationElement
method),	-			method), 77	
		illArravD <b>ate</b> t		() (androguard.core.bytecodes.dvm.A	AnnotationItem
method),	-	warray bagacae_		method), 77	Thirto tall to the control of the co
		struction a=+		() (androguard.core.bytecodes.dvm.A	Annotation OffItem
method),	-	wiruciiongec_		method), 78	ппошиоподлет
		lethedldltem+		method), 76 j () (androguard.core.bytecodes.dvm.1	AnnotationsDirectoryItem
method),	•	remourangmc_	_	method), 80	AnnoiditonsDirectoryment
/ /		lathadiditam <del>i</del> n		method), 80   () (androguard.core.bytecodes.dvm.1	Annotation SatItam
=		етоатап <b>у</b> ни <u></u>	_		Аппошионзениет
method),		a als a dConitals +		method), 78	Ann at ation Cat Dallton
		ackeaswiigæ L_		() (androguard.core.bytecodes.dvm.A	AnnotationSetRejItem
method),				method), 79	Ann at ation Cat Dallist
	-	parseswiiget_		() (androguard.core.bytecodes.dvm.A	AnnotationSetRejList
method),		, ,		method), 79	
get_name_idx				() (androguard.core.bytecodes.dvm.	JassDataItem
	re.bytecodes.dvm.AnnotationEle			method), 81	CI D O
method),				() (androguard.core.bytecodes.dvm.	ClassDefItem
get_name_idx		(andro-		method), 82	
-	re.bytecodes.dvm.FieldIdItem	get_	_	() (androguard.core.bytecodes.dvm.	ClassHDefItem
method),				method), 83	
<pre>get_name_idx</pre>			_obj	() (androguard.core.bytecodes.dvm.	CodeItem
-	re.bytecodes.dvm.MethodIdItem			method), 85	
method),	. 124	get_		() (androguard.core.bytecodes.dvm.l	DalvikCode
get_names()		(andro-		method), 87	
guard.co	re.bytecodes.dvm.ClassHDefIte	$m$ get_	_obj	() (androguard.core.bytecodes.dvm.l	DBGBytecode
method),	. 83		i	method), 85	
get_nb_instru	actions() (	(andro- get_	_obj	() (androguard.core.bytecodes.dvm.l	DebugInfoItemEmpty
guard.co	re.analysis.analysis.DVMBasicI	Block	i	method), 93	
method),	56	get_	_obj	() (androguard.core.bytecodes.dvm.l	EncodedAnnotation
get_nb_method	ls()	(andro-	i	method), 93	
guard.co	re.analysis.analysis.ClassAnaly	sis get_	_obj	() (androguard.core.bytecodes.dvm.l	EncodedArray
method),	. 54		i	method), 94	
get_nb_string	gs () (androguard.session.S	Session get_	_obj	() (androguard.core.bytecodes.dvm.l	EncodedArrayItem
method),	183			method), 94	
		.DVMBasiæBlo	<i>o.d</i> ab i	() (androguard.core.bytecodes.dvm.l	Encoded Catch Handler Lis
method),		_		method), 95	
get_next_offs		andro- get_		() (androguard.core.bytecodes.dvm.l	EncodedField
	re.bytecodes.dvm.ClassManage			method), 96	
method),	-			() (androguard.core.bytecodes.dvm.l	EncodedTypeAddrPair

method), 101	method), 78
<pre>get_obj() (androguard.core.bytecodes.dvm.EncodedVal</pre>	uget_off() (androguard.core.bytecodes.dvm.AnnotationSetRefList method), 79
	ntjæt_off() (androguard.core.bytecodes.dvm.ClassDataItem method), 81
	nget_off() (androguard.core.bytecodes.dvm.ClassHDefItem method), 83
get_obj() (androguard.core.bytecodes.dvm.FieldIdItem method), 103	
	get_off() (androguard.core.bytecodes.dvm.DalvikCode method), 87
	get_off() (androguard.core.bytecodes.dvm.DebugInfoItem method), 93
	get_off() (androguard.core.bytecodes.dvm.DebugInfoItemEmpty method), 93
	ogationoff () (androguard.core.bytecodes.dvm.EncodedArrayItem method), 94
	tgent_off() (androguard.core.bytecodes.dvm.EncodedCatchHandler method), 95
	nget_off() (androguard.core.bytecodes.dvm.EncodedCatchHandlerList method), 95
	rapota <u>ti</u> on f ( ) (androguard.core.bytecodes.dvm.FieldAnnotation method), 102
	nget_off() (androguard.core.bytecodes.dvm.FieldHIdItem method), 102
	<pre>get_off() (androguard.core.bytecodes.dvm.HeaderItem</pre>
<pre>get_obj() (androguard.core.bytecodes.dvm.StringDataI</pre>	
<pre>get_obj() (androguard.core.bytecodes.dvm.StringIdIten</pre>	
	nget_off() (androguard.core.bytecodes.dvm.MethodAnnotation method), 123
	get_off() (androguard.core.bytecodes.dvm.MethodHIdItem method), 124
	get_off() (androguard.core.bytecodes.dvm.ParameterAnnotation method), 127
	get_off() (androguard.core.bytecodes.dvm.ProtoHIdItem method), 127
<pre>get_obj_by_offset() (andro- guard.core.bytecodes.dvm.ClassManager</pre>	<pre>get_off() (androguard.core.bytecodes.dvm.StringDataItem</pre>
method), 84	<pre>get_off() (androguard.core.bytecodes.dvm.StringIdItem</pre>
<pre>get_objects_apk() (androguard.session.Session     method), 183</pre>	method), 130 get_off() (androguard.core.bytecodes.dvm.TryItem
get_objects_dex() (androguard.session.Session	method), 131
method), 183	<pre>get_off() (androguard.core.bytecodes.dvm.TypeHIdItem</pre>
<pre>get_odex_format()</pre>	method), 131
method), 84	<pre>get_off() (androguard.core.bytecodes.dvm.TypeList     method), 132</pre>
<pre>get_off() (androguard.core.bytecodes.dvm.AnnotationI</pre>	
method), 77	guard.core.bytecodes.dvm.MapItem method),
get_off() (androguard.core.bytecodes.dvm.Annotations	
<pre>method), 80 get_off() (androguard.core.bytecodes.dvm.AnnotationS</pre>	get_op_value() (andro- SetItem guard.core.bytecodes.dym.DBGBytecode
	- /

method), 85		method), 111	
<pre>get_op_value()</pre>	(andro-	get_operands()	(andro-
guard.core.bytecodes.dvm.FillArrayDat	a	guard.core.bytecodes.dvm.Instruction2	2cs
method), 104		method), 112	
<pre>get_op_value()</pre>	(andro-	get_operands()	(andro-
guard.core.bytecodes.dvm.Instruction n	nethod),	guard.core.bytecodes.dvm.Instruction2	22s
106		method), 112	
<pre>get_op_value()</pre>	(andro-	<pre>get_operands()</pre>	(andro-
guard.core.bytecodes.dvm.PackedSwitch method), 126	'n	guard.core.bytecodes.dvm.Instruction2 method), 113	?2t
<pre>get_op_value()</pre>	(andro-	get_operands()	(andro-
guard.core.bytecodes.dvm.SparseSwitch	i	guard.core.bytecodes.dvm.Instruction2	22x
method), 129		method), 113	
		get_operands()	(andro-
guard.core.bytecodes.dvm.FillArrayDat method), 104	а	guard.core.bytecodes.dvm.Instruction2 method), 113	?3x
<pre>get_operands()</pre>	(andro-	get_operands()	(andro-
guard.core.bytecodes.dvm.Instruction n 106	nethod),	guard.core.bytecodes.dvm.Instruction3 method), 114	POt
<pre>get_operands()</pre>	(andro-	get_operands()	(andro-
guard.core.by tecodes.dvm. Instruction 10	)t	guard.core.bytecodes.dvm.Instruction3	?1c
method), 106		method), 114	
		get_operands()	(andro-
guard.core.bytecodes.dvm.Instruction11 method), 107	'n	guard.core.bytecodes.dvm.Instruction3 method), 115	?1i
		<pre>get_operands()</pre>	(andro-
guard.core.bytecodes.dvm.Instruction11 method), 107	!x	guard.core.bytecodes.dvm.Instruction3 method), 115	?1t
	(andro-	get_operands()	(andro-
guard.core.bytecodes.dvm.Instruction12		guard.core.bytecodes.dvm.Instruction3	22x
method), 108		method), 116	
<pre>get_operands()</pre>	(andro-	get_operands()	(andro-
guard.core.bytecodes.dvm.Instruction20		guard.core.bytecodes.dvm.Instruction3	?5c
method), 108		method), 116	
<pre>get_operands()</pre>	(andro-	get_operands()	(andro-
guard.core.bytecodes.dvm.Instruction20	)t	guard.core.bytecodes.dvm.Instruction3	5mi
method), 109		method), 116	
			(andro-
guard.core.bytecodes.dvm.Instruction21	c	guard.core.bytecodes.dvm.Instruction3	5ms
method), 109		method), 117	
3 <del>-</del> 1	(andro-	<pre>get_operands()</pre>	(andro-
guard.core.bytecodes.dvm.Instruction21	'h	guard.core.bytecodes.dvm.Instruction3	rc
method), 110	, 1	method), 117	
· ·	(andro-	<pre>get_operands()</pre>	(andro-
guard.core.bytecodes.dvm.Instruction21	S	guard.core.bytecodes.dvm.Instruction3	rmı
method), 110	( <b>1</b>	method), 118	( <b>1</b>
3 <del>-</del> 1	(andro-	get_operands()	(andro-
guard.core.bytecodes.dvm.Instruction21	ı	guard.core.bytecodes.dvm.Instruction3	rms
method), 110	(andro-	method), 118	(andro-
<pre>get_operands()</pre>	`	<pre>get_operands()</pre>	`
method), 111		method), 119	OBC
	(andro-	get_operands()	(andro-
guard.core.bytecodes.dvm.Instruction22	`	guard.core.bytecodes.dvm.Instruction4	`
•			

method), 119		get_output()	(andro-
get_operands()	(andro-	guard.core.bytecodes.dvm.Instruction2	21h
guard.core.bytecodes.dvm.Instruction	on45cc	method), 110	
method), 120		get_output()	(andro-
get_operands()	(andro-	guard.core.bytecodes.dvm.Instruction2	21s
guard.core.bytecodes.dvm.Instruction	on4rcc	method), 110	
method), 120		get_output()	(andro-
get_operands()	(andro-	guard.core.bytecodes.dvm.Instruction2	?1t
guard.core.bytecodes.dvm.Instruction	on51l	method), 110	
method), 120		<pre>get_output()</pre>	(andro-
<pre>get_operands()</pre>	(andro-	guard.core.bytecodes.dvm.Instruction2	22 <i>b</i>
guard.core.bytecodes.dvm.Instruction	on52c	method), 111	
method), 121		<pre>get_output()</pre>	(andro-
<pre>get_operands()</pre>	(andro-	guard.core.bytecodes.dvm.Instruction2	2 <i>c</i>
guard.core.bytecodes.dvm.Instruction	on3rc	method), 111	( 1
method), 121	( 1	<pre>get_output()</pre>	(andro-
get_operands()	(andro-	guard.core.bytecodes.dvm.Instruction2	ZCS
guard.core.bytecodes.dvm.PackedSv	witch	method), 112	( <b>1</b>
method), 126	( an dua	get_output()	(andro-
get_operands()	(andro-	guard.core.bytecodes.dvm.Instruction2	128
guard.core.bytecodes.dvm.SparseSv	viicn	method), 112	(an dua
method), 129	modulo an	get_output()	(andro-
<pre>get_optimized_instruction() (in</pre>		guard.core.bytecodes.dvm.Instruction2	2 <i>l</i>
get_orig_value()		<pre>method), 113 get_output()</pre>	(andro-
guard.core.analysis.analysis.Stringa		guard.core.bytecodes.dvm.Instruction2	,
method), 61	inaiysis	method), 113	21
get_output()	(andro-	get_output()	(andro-
guard.core.bytecodes.dvm.FillArray		guard.core.bytecodes.dvm.Instruction2	`
method), 104	, 2	method), 113	
get_output()	(andro-	get_output()	(andro-
guard.core.bytecodes.dvm.Instruction		guard.core.bytecodes.dvm.Instruction3	`
106	,,	method), 114	
get_output()	(andro-	get_output()	(andro-
guard.core.bytecodes.dvm.Instruction		guard.core.bytecodes.dvm.Instruction3	21c
method), 107		method), 114	
get_output()	(andro-	get_output()	(andro-
guard.core.bytecodes.dvm.Instruction	on11n	guard.core.bytecodes.dvm.Instruction3	21i
method), 107		method), 115	
get_output()	(andro-	get_output()	(andro-
guard.core.bytecodes.dvm.Instruction	on11x	guard.core.bytecodes.dvm.Instruction3	21t
method), 108		method), 115	
get_output()	(andro-	<pre>get_output()</pre>	(andro-
guard.core.bytecodes.dvm.Instruction	on12x	guard.core.bytecodes.dvm.Instruction3	2x
method), 108		method), 116	
get_output()	(andro-	get_output()	(andro-
guard.core.bytecodes.dvm.Instruction	on20bc	guard.core.bytecodes.dvm.Instruction3	35c
method), 108		method), 116	
get_output()	(andro-	<pre>get_output()</pre>	(andro-
guard.core.bytecodes.dvm.Instruction	on20t	guard.core.bytecodes.dvm.Instruction3	5mi
method), 109	, 1	method), 116	, ,
<pre>get_output()</pre>	(andro-	<pre>get_output()</pre>	(andro-
guard.core.bytecodes.dvm.Instruction method), 109	on21c	guard.core.bytecodes.dvm.Instruction3 method), 117	Sms

act output ()	(andro-	method), 132	
<pre>get_output()</pre>		get_parameter_annotations()	(andro-
method), 117	570	guard.core.bytecodes.dvm.Annotation	,
get_output()	(andro-	method), 80	isDirectory frem
guard.core.bytecodes.dvm.Instruction.	`		(andro-
method), 118	311111	guard.core.bytecodes.dvm.DebugInfo	`
get_output()	(andro-	method), 93	11cm
guard.core.bytecodes.dvm.Instruction.	`		(andro-
method), 118	311113	guard.core.bytecodes.dvm.ProtoIdIter	`
get_output()	(andro-	method), 127	116
guard.core.bytecodes.dvm.Instruction	•		(andro-
method), 119	1050	guard.core.bytecodes.dvm.ProtoIdIter	
get_output()	(andro-	method), 128	
guard.core.bytecodes.dvm.Instruction	*		(andro-
method), 119	,,,,	guard.core.bytecodes.dvm.DebugInfo	`
get_output()	(andro-	method), 93	11cm
guard.core.bytecodes.dvm.Instruction	•		(andro-
method), 120	,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,	guard.core.bytecodes.dvm.ProtoIdIter	,
get_output()	(andro-	method), 128	
guard.core.bytecodes.dvm.Instruction			andro-
method), 120		guard.core.bytecodes.dvm), 134	
get_output()	(andro-	get_params_type() (in module	andro-
guard.core.bytecodes.dvm.Instruction.		guard.decompiler.dad.util), 175	
method), 120	J 1 v	<pre>get_permission_usage()</pre>	(andro-
get_output()	(andro-	guard.core.analysis.analysis.Analysis	`
guard.core.bytecodes.dvm.Instruction.		51	memou),
method), 121	. <b>2</b> C	get_permissions()	(andro-
get_output()	(andro-	guard.core.analysis.analysis.Analysis	`
guard.core.bytecodes.dvm.Instruction.		51	,,,
method), 121		get_permissions()	(andro-
get_output()	(andro-	guard.core.bytecodes.apk.APK	method),
guard.core.bytecodes.dvm.PackedSwit		72	<i>"</i>
method), 126		<pre>get_prev() (androguard.core.analysis.analy</pre>	ysis.DVMBasicBlock
get_output()	(andro-		
guard.core.bytecodes.dvm.SparseSwite			(andro-
method), 129		guard.core.bytecodes.dvm.ClassMana	ager
get_outs_size()	(andro-	method), 84	
guard.core.bytecodes.dvm.DalvikCode	ę	get_proto()	(andro-
method), 87		guard.core.bytecodes.dvm.MethodIdI	tem
get_package()	(andro-	method), 124	
guard.core.bytecodes.apk.APK	method),	get_proto()	(andro-
72		guard.core.bytecodes.dvm.MethodIdI	temInvalid
<pre>get_package_class_name() (in modul</pre>	le andro-	method), 125	
guard.core.bytecode), 149		<pre>get_proto_idx()</pre>	(andro-
get_package_name()	(andro-	guard.core.bytecodes.dvm.MethodIdI	tem
guard.core.bytecodes.axml.ARSCResT	уре	method), 124	
method), 141		<pre>get_providers()</pre>	(andro-
get_package_name()	(andro-	guard.core.bytecodes.apk.APK	method),
guard.core.bytecodes.axml.PackageCo	ontext	72	
method), 143		get_public_keys_der_v2()	(andro-
get_packages_names()	(andro-	guard.core.bytecodes.apk.APK	method),
guard.core.bytecodes.axml.ARSCPars	er	72	
method), 137		<pre>get_public_keys_der_v3()</pre>	(andro-
get pad() (androguard.core.bytecodes.dvm	ı.TypeList	guard.core.bytecodes.apk.APK	method),

72		method), 95
<pre>get_public_keys_v2()</pre>	(andro-	<pre>get_raw() (androguard.core.bytecodes.dvm.EncodedCatchHandlerList</pre>
guard.core.bytecodes.apk.APK	method),	method), 95
72		<pre>get_raw() (androguard.core.bytecodes.dvm.EncodedField</pre>
<pre>get_public_keys_v3()</pre>	(andro-	method), 96
guard.core.bytecodes.apk.APK 72	method),	<pre>get_raw() (androguard.core.bytecodes.dvm.EncodedMethod</pre>
<pre>get_public_resources()</pre>	(andro-	<pre>get_raw() (androguard.core.bytecodes.dvm.EncodedTypeAddrPair</pre>
guard.core.bytecodes.axml.ARSCPar	rser	method), 101
method), 137		<pre>get_raw() (androguard.core.bytecodes.dvm.EncodedValue</pre>
<pre>get_qualifier()</pre>	(andro-	method), 101
	TableConfig	get_raw() (androguard.core.bytecodes.dvm.FieldAnnotation
method), 140	LADE	method), 102
	s.apk.APK	get_raw() (androguard.core.bytecodes.dvm.FieldHIdItem
method), 72	Annotation	method), 102
	.Annotation1	Ebentent aw () (androguard.core.bytecodes.dvm.FieldIdItem
method), 77	Annotation	method), 103
method), 77		(temet_raw() (androguard.core.bytecodes.dvm.FillArrayData method), 104
<pre>get_raw() (androguard.core.bytecodes.dvm     method), 78</pre>	.Annotation	Offtemraw() (androguard.core.bytecodes.dvm.HeaderItem method), 104
<pre>get_raw() (androguard.core.bytecodes.dvm</pre>	.Annotations	Directory Man (androguard.core.bytecodes.dvm.Instruction method), 106
<pre>get_raw() (androguard.core.bytecodes.dvm</pre>	.AnnotationS	SegHemraw () (androguard.core.bytecodes.dvm.Instruction10t method), 107
	.AnnotationS	SegRefItenw() (androguard.core.bytecodes.dvm.Instruction10x method), 107
<pre>get_raw() (androguard.core.bytecodes.dvm</pre>	.AnnotationS	SegRefListw() (androguard.core.bytecodes.dvm.Instruction11n method), 107
	.ClassDataI	taget_raw() (androguard.core.bytecodes.dvm.Instruction11x method), 108
**	.ClassDefIte	mget_raw() (androguard.core.bytecodes.dvm.Instruction12x method), 108
	.ClassHDefI	tegnet_raw() (androguard.core.bytecodes.dvm.Instruction20bc method), 108
	.CodeItem	get_raw() (androguard.core.bytecodes.dvm.Instruction20t method), 109
<pre>get_raw() (androguard.core.bytecodes.dvm</pre>	.DalvikCode	get_raw() (androguard.core.bytecodes.dvm.Instruction21c method), 109
method), 87	DRCPytage	deet_raw() (androguard.core.bytecodes.dvm.Instruction21h
method), 85	.рвовущесь	method), 110
<pre>get_raw() (androguard.core.bytecodes.d</pre>	vm.DCode	<pre>get_raw() (androguard.core.bytecodes.dvm.Instruction21s</pre>
	.DebugInfoI	tegret_raw() (androguard.core.bytecodes.dvm.Instruction21t method), 111
	.DebugInfoI	tegneEmptsw() (androguard.core.bytecodes.dvm.Instruction22b method), 111
	.EncodedAn	ngtettion:aw() (androguard.core.bytecodes.dvm.Instruction22c
method), 93	Engada JA	method), 111
	.ьпсоаеаАт	rayet_raw() (androguard.core.bytecodes.dvm.Instruction22cs
method), 94	EncodedA.	method), 112 ræy¥te <u>m</u> raw () (androguard.core.bytecodes.dvm.Instruction22s
method), 94	.ьпсоаваАт	терые <u>т</u> с aw () (anaroguara.core.bytecoaes.avm.tnstruction22s method), 112
	EncodedCa	memou), 112 t <u>cl<del>ull</del>an</u> dles () (androguard.core.bytecodes.dvm.Instruction22t
y = = _ + an ( ) (and o guard. core. by iccodes. aviil	coucucu	magaza <u>n</u> amen ( ) (mining minine or e. o y e e o u e s. u v n t. 111311 u e tront 2 t

method), 113	method), 125
<pre>get_raw() (androguard.core.bytecodes.dvm.Instruction22met_ra</pre>	w () (androguard.core.bytecodes.dvm.PackedSwitch
method), 113	method), 126
<pre>get_raw() (androguard.core.bytecodes.dvm.Instruction23yet_ra</pre>	w () (androguard.core.bytecodes.dvm.ParameterAnnotation
method), 114	method), 127
<pre>get_raw() (androguard.core.bytecodes.dvm.Instruction3@get_ra</pre>	
method), 114	method), 127
<pre>get_raw() (androguard.core.bytecodes.dvm.Instruction31get_ra</pre>	
method), 114	method), 128
<pre>get_raw() (androguard.core.bytecodes.dvm.Instruction31jet_ra</pre>	
method), 115	method), 129
get_raw() (androguard.core.bytecodes.dvm.Instruction315et_ra	
method), 115	method), 130
get_raw() (androguard.core.bytecodes.dvm.Instruction32get_ra	
method), 116	method), 130
get_raw() (androguard.core.bytecodes.dvm.Instruction35get_ra	
method), 116	method), 131
**	
get_raw() (androguard.core.bytecodes.dvm.Instruction35gmit_ra method), 117	
<i>"</i>	method), 131
get_raw() (androguard.core.bytecodes.dvm.Instruction35ymet_ra	
method), 117	method), 132
<pre>get_raw() (androguard.core.bytecodes.dvm.Instruction3rget_ra</pre>	
method), 117	method), 132
<pre>get_raw() (androguard.core.bytecodes.dvm.Instruction3rgmit_ra</pre>	
method), 118	method), 133
<pre>get_raw() (androguard.core.bytecodes.dvm.Instruction3rgmest_ra</pre>	
method), 118	guard.core.bytecodes.dvm.ClassManager
<pre>get_raw() (androguard.core.bytecodes.dvm.Instruction40sc</pre>	method), 84
	w_string() (andro-
<pre>get_raw() (androguard.core.bytecodes.dvm.Instruction41c</pre>	guard.core.bytecodes.dvm.Instruction21c
method), 119	method), 109
<pre>get_raw() (androguard.core.bytecodes.dvm.Instruction45get_ra</pre>	
method), 120	guard.core.bytecodes.dvm.Instruction31c
<pre>get_raw() (androguard.core.bytecodes.dvm.Instruction4rcc</pre>	method), 114
method), 120 get_re	al_descriptor() (andro-
<pre>get_raw() (androguard.core.bytecodes.dvm.Instruction51l</pre>	guard.core.bytecodes.dvm.MethodIdItem
method), 121	method), 125
<pre>get_raw() (androguard.core.bytecodes.dvm.Instruction52pet_re</pre>	ceivers() (andro-
method), 121	guard.core.bytecodes.apk.APK method),
<pre>get_raw() (androguard.core.bytecodes.dvm.Instruction5rc</pre>	72
<pre>method), 121</pre> get_re	f_kind() (andro-
<pre>get_raw() (androguard.core.bytecodes.dvm.MapItem</pre>	guard.core.bytecodes.dvm.Instruction method),
method), 122	106
<pre>get_raw() (androguard.core.bytecodes.dvm.MapList get_re</pre>	f_kind() (andro-
method), 123	guard.core.bytecodes.dvm.Instruction21c
<pre>get_raw() (androguard.core.bytecodes.dvm.MethodAnnotation</pre>	method), 109
- · · · · · · · · · · · · · · · · · · ·	f_kind() (andro-
<pre>get_raw() (androguard.core.bytecodes.dvm.MethodHIdItem</pre>	guard.core.bytecodes.dvm.Instruction22c
method), 124	method), 111
<pre>get_raw() (androguard.core.bytecodes.dvm.MethodIdItemget_re</pre>	
method), 125	guard.core.bytecodes.dvm.Instruction22cs
get_raw() (androguard.core.bytecodes.dvm.OdexDependencies	method), 112
	f_kind() (andro-
get_raw() (androguard.core.bytecodes.dvm.OdexHeaderItem	

method), 114	method), 88
<pre>get_ref_kind()</pre>	<pre>get_requested_aosp_permissions() (andro- guard.core.bytecodes.apk.APK method), 73</pre>
method), 116	<pre>get_requested_aosp_permissions_details()</pre>
<pre>get_ref_kind() (andro-</pre>	(androguard.core.bytecodes.apk.APK method),
guard.core.bytecodes.dvm.Instruction35mi	73
method), 117	<pre>get_requested_permissions() (andro-</pre>
<pre>get_ref_kind() (andro-</pre>	guard.core.bytecodes.apk.APK method),
guard.core.bytecodes.dvm.Instruction35ms	73
method), 117	<pre>get_requested_third_party_permissions()</pre>
<pre>get_ref_kind() (andro-</pre>	(androguard.core.bytecodes.apk.APK method),
guard.core.bytecodes.dvm.Instruction3rc	73
method), 117	<pre>get_res_configs() (andro-</pre>
<pre>get_ref_kind()</pre> (andro-	guard.core.bytecodes.axml.ARSCParser
guard.core.bytecodes.dvm.Instruction3rmi	method), 138
method), 118	get_res_id_by_key() (andro-
get_ref_kind() (andro-	guard.core.bytecodes.axml.ARSCParser
guard.core.bytecodes.dvm.Instruction3rms	method), 138
method), 118	get_res_value() (andro-
get_ref_kind() (andro-	guard.core.bytecodes.apk.APK method),
guard.core.bytecodes.dvm.Instruction40sc	73
method), 119	<pre>get_resolved_res_configs() (andro-</pre>
get_ref_kind() (andro-	guard.core.bytecodes.axml.ARSCParser
guard.core.bytecodes.dvm.Instruction41c	method), 138
method), 119	get_resolved_strings() (andro-
get_ref_kind() (andro-	guard.core.bytecodes.axml.ARSCParser
guard.core.bytecodes.dvm.Instruction52c	method), 138
method), 121	get_resource_bool() (andro-
<pre>get_ref_kind() (andro-</pre>	guard.core.bytecodes.axml.ARSCParser
guard.core.bytecodes.dvm.Instruction5rc	method), 138
method), 121	get_resource_color() (andro-
<pre>get_ref_off()</pre>	guard.core.bytecodes.axml.ARSCParser
guard.core.bytecodes.dvm.Instruction10t	method), 138
method), 107	get_resource_dimen() (andro-
<pre>get_ref_off()</pre>	guard.core.bytecodes.axml.ARSCParser
guard.core.bytecodes.dvm.Instruction20t	method), 138
method), 109	get_resource_id() (andro-
<pre>get_ref_off() (andro-</pre>	
guard.core.bytecodes.dvm.Instruction21t	method), 138
method), 111	get_resource_integer() (andro-
<pre>get_ref_off() (andro-</pre>	guard.core.bytecodes.axml.ARSCParser
guard.core.bytecodes.dvm.Instruction22t	method), 138
method), 113	<pre>get_resource_string() (andro-</pre>
get_ref_off() (andro-	guard.core.bytecodes.axml.ARSCParser
guard.core.bytecodes.dvm.Instruction30t	method), 138
method), 114	get_resource_style() (andro-
<pre>get_ref_off() (andro-</pre>	guard.core.bytecodes.axml.ARSCParser
guard.core.bytecodes.dvm.Instruction31t	method), 138
method), 115	get_resource_xml_name() (andro-
<pre>get_regex_strings() (andro-</pre>	guard.core.bytecodes.axml.ARSCParser
guard.core.bytecodes.dvm.DalvikVMFormat	method), 138
method), 92	get_return_type() (andro-
<pre>get_registers_size() (andro-</pre>	guard.core.bytecodes.dvm.ProtoIdItemInvalid
guard.core.bytecodes.dym.DalvikCode	method), 128

```
get_return_type_idx()
                                           (andro-
                                                             method), 96
        guard.core.bytecodes.dvm.ProtoIdItem
                                                    get size()(androguard.core.bytecodes.dvm.EncodedMethod
        method), 128
                                                             method), 100
                                                    get_size() (androguard.core.bytecodes.dvm.MapItem
get_return_type_idx_value()
                                           (andro-
        guard.core.bytecodes.dvm.ProtoIdItem
                                                             method), 122
        method), 128
                                                    get size() (androguard.core.bytecodes.dvm.TypeList
get rhs() (androguard.decompiler.dad.instruction.AssignExpressionethod), 133
        method), 160
                                                    get source()
                                                                                                (andro-
get rhs() (androguard.decompiler.dad.instruction.FillArrayExpressiand.core.bytecodes.dvm.ClassDefItem
        method), 162
                                                             method), 82
get_rhs() (androguard.decompiler.dad.instruction.IRForget_source()
                                                                                                (andro-
                                                             guard.core.bytecodes.dvm.EncodedMethod
        method), 162
get_rhs() (androguard.decompiler.dad.instruction.MoveExpressiomethod), 100
        method), 164
                                                                                                (andro-
                                                    get_source()
                     (in
                              module
                                            andro-
                                                             guard.decompiler.dad.decompile.DvClass
get_sbyte()
        guard.core.bytecodes.dvm), 134
                                                             method), 156
                                                                                                (andro-
get_services()
                                           (andro-
                                                    get_source()
        guard.core.bytecodes.apk.APK
                                          method),
                                                             guard.decompiler.dad.decompile.DvMethod
                                                             method), 157
get short string()
                                           (andro-
                                                    get source class()
                                                                                                (andro-
        guard.core.bytecodes.dvm.EncodedMethod
                                                             guard.decompiler.decompiler.DecompilerDAD
        method), 100
                                                             method), 177
get_shorty()
                                           (andro-
                                                    get_source_class()
                                                                                                (andro-
        guard.core.bytecodes.dvm.ProtoIdItemInvalid
                                                             guard.decompiler.decompiler.DecompilerDed
        method), 128
                                                             method), 177
get_shorty_idx()
                                           (andro-
                                                    get source class()
        guard.core.bytecodes.dvm.ProtoIdItem
                                                             guard.decompiler.decompiler.DecompilerDex2Fernflower
        method), 128
                                                             method), 178
get_shorty_idx_value()
                                                    get_source_class()
                                                                                                (andro-
                                           (andro-
        guard.core.bytecodes.dvm.ProtoIdItem
                                                             guard.decompiler.decompiler.DecompilerDex2Jad
        method), 128
                                                             method), 178
get_signature()
                                           (andro-
                                                    get_source_class()
                                                                                                (andro-
                                                             guard.decompiler.decompiler.DecompilerDex2WineJad
        guard.core.bytecodes.apk.APK
                                          method),
                                                             method), 178
get signature name()
                                           (andro-
                                                    get source class()
                                                                                                (andro-
        guard.core.bytecodes.apk.APK
                                                             guard.decompiler.decompiler.DecompilerJADX
                                          method),
        73
                                                             method), 178
get_signature_names()
                                           (andro-
                                                    get_source_class_ext()
                                                                                                (andro-
        guard.core.bytecodes.apk.APK
                                          method),
                                                             guard.decompiler.decompiler.DecompilerDAD
        73
                                                            method), 177
get_signatures()
                                           (andro-
                                                                                                (andro-
                                                    get source ext()
        guard.core.bytecodes.apk.APK
                                          method),
                                                             guard.core.bytecodes.dvm.ClassDefItem
                                                             method), 82
get_size() (androguard.core.bytecodes.dvm.DalvikCodeget_source_ext()
                                                                                                (andro-
                                                             guard.decompiler.dad.decompile.DvClass
        method), 88
get_size() (androguard.core.bytecodes.dvm.EncodedAnnotation method), 156
        method), 93
                                                    get source ext()
                                                                                                (andro-
get_size() (androguard.core.bytecodes.dvm.EncodedArray
                                                             guard.decompiler.dad.decompile.DvMethod
        method), 94
                                                             method), 157
qet_size() (androguard.core.bytecodes.dvm.EncodedCapeHH_midlerce_file_idx()
                                                                                                (andro-
                                                             guard.core.bytecodes.dvm.ClassDefItem
        method), 95
get_size() (androguard.core.bytecodes.dvm.EncodedCatchHandlerEtilsod), 82
        method), 95
                                                    get source method()
                                                                                                (andro-
                                                             guard.decompiler.decompiler.DecompilerDAD
get size()(androguard.core.bytecodes.dvm.EncodedField
```

method), 177			method), 84		
<pre>get_source_method()</pre>	(andro-	get_s	string_data_it	cem()	(andro-
guard.decompiler.decompiler.Decom	mpilerDed		guard.core.byted	odes.dvm.Dalvi	kVMFormat
method), 177			method), 92		
<pre>get_source_method()</pre>	(andro-	get_s	string_data_of	ff()	(andro-
guard.decompiler.decompiler.Decom method), 178	mpilerDex2Fe	ernflowe.	r guard.core.byted method), 130	odes.dvm.String	·IdItem
<pre>get_source_method()</pre>	(andro-	get_s	string_resourd	ces()	(andro-
guard.decompiler.decompiler.Decom method), 178	mpilerDex2Ja	ad	guard.core.byted method), 139	odes.axml.ARSC	<i>CParser</i>
<pre>get_source_method()</pre>		-	strings()		(andro-
guard.decompiler.decompiler.Decom method), 178	mpilerDex2W	Vine <b>J</b> ad	guard.core.anal <sub>3</sub> 51	ysis.analysis.And	ılysis method),
<pre>get_source_method()</pre>		get_s	strings()		(andro-
guard.decompiler.decompiler.Decom method), 179	mpilerJADX		guard.core.byted method), 92	odes.dvm.Dalvi	kVMFormat
<pre>get_special_ins()</pre>	(andro-	get_s	strings()	(androguard.:	session.Session
guard.core.analysis.analysis.DVME	BasicBlock		method), 183		
method), 56		get_s	strings_analys		(andro-
<pre>get_start()</pre>	(andro-		guard.core.anal	ysis.analysis.And	ılysis method),
guard.core.analysis.analysis.DVME	<i>sasicBlock</i>		52	()	( <b>1</b>
method), 57	(andro-	get_s	strings_resour		(andro-
<pre>get_start_addr()</pre>	method),		guard.core.byted method), 139	oaes.axmi.AKSC	raisei
131	memoa),	net s	superclass_idx	z ( )	(andro-
<pre>get_static_fields()</pre>	(andro-	900_0	guard.core.byted		,
guard.core.bytecodes.dvm.ClassDa			method), 82		,
method), 81		get_s	superclassname	÷()	(andro-
<pre>get_static_fields_size()</pre>	(andro-		guard.core.byted	odes.dvm.Class	DefItem
guard.core.bytecodes.dvm.ClassDa	taItem		method), 83		
method), 81		get_t	target_sdk_ver		(andro-
<pre>get_static_values_off()</pre>	(andro-		guard.core.byted	odes.apk.APK	method),
guard.core.bytecodes.dvm.ClassDe	fItem		73		
method), 82		get_t	targets()		(andro-
<pre>get_string()</pre>	(andro-		guard.core.bytec	odes.dvm.Packe	dSwitch
guard.core.bytecodes.axml.ARSCPa	arser		method), 126		( an dua
method), 139	(an dro	-	targets() <pre>guard.core.byted</pre>	and as down Snams	(andro-
<pre>get_string()</pre>			method), 129	oues.avm.spars	eswiich
method), 84	mager	net t	ranslated_kir	nd ()	(andro-
get_string()	(andro-	900_0	guard.core.byted		`
guard.core.bytecodes.dvm.Instruction	,		106		enen memeta),
method), 109		get t	ranslated_par	rameter name	es() (an-
<pre>get_string()</pre>	(andro-	· –	droguard.core.b		
guard.core.bytecodes.dvm.Instructi	on31c		method), 93		
method), 114		get_t	cries()		(andro-
<pre>get_string()</pre>	(andro-		guard.core.byted	odes.dvm.Dalvi	kCode
guard.core.bytecodes.dvm.TypeItem	n method),		method), 88		
132	, .	get_t	ries_size()		(andro-
<pre>get_string()</pre>	(andro-		guard.core.bytec	odes.dvm.Dalvi	kCode
guard.core.bytecodes.dvm.TypeList	method),	~~ + +	method), 88		( a.a. d
133 get_string_by_offset()	(andro	get_t	criple()  guard.core.byted	rodes dum Franc	(andro- ledMethod
guard.core.bytecodes.dvm.ClassMa	(andro- mager		method), 100	oues.avm.Encoc	шинени

get_tr	iple() guard.core.by method), 125	tecodes.dvi	n.MethodIdIt		get_ty	pe_list() guard.core.bytecodes.dv method), 84	vm.ClassManaz	(andro- ger
get_ty]	* *	uard.core.b	ytecodes.axm	l.ARSCRes	<i>T</i> y <b>p</b> ¢c_ty	pe_list_off() guard.core.bytecodes.dv	vm.TypeList	(andro- method),
get_ty]	pe() (androgu	uard.core.b	ytecodes.dvm	.ClassMan	-	133	71	, ·
act +12	method), 84 pe () (androgu	ard core b	vtacadas dym	FieldIdIte		pe_ref() guard.core.bytecodes.dv	m ClassMana	(andro-
	method), 103					method), 84	m.Ciassiriana <sub>z</sub>	-
get_ty <sub>]</sub>	pe () $(androgu$ $method)$ , $103$	ıard.core.b	ytecodes.dvm	.FieldIdIte	mdjævta <u>li</u> tdy	pe_size() (in guard.decompiler.dad.u	module til), 175	andro-
get_ty	pe() (androgu	ard.core.b	ytecodes.dvm	.MapItem	get_ty		LARGOR	(andro-
act +12	method), 122 pe () (androgu	uard core b	vtacadas dvm	TypeHIdIt	am	guard.core.bytecodes.ax method), 139	ml.ARSCParse	er
ger_ry	$p \in ()$ (anarogumethod), 131	iuru.core.o	yiecoues.uvm	.1 уреннан		ed_vars()		(andro-
aet tw	, ,	ard decom	niler dad inst	ruction Ar		ea_vars() E <b>gymedsilen</b> ompiler.dad.ir	struction Arro	`
get_ty	$p \in ()$ (anarogumethod), 159	ши.иесот	риет.иии.ты	гисион.Аг	таушендін.	<b>емрисызаен</b> отристаса: method), 159	isiruciion.Arra	yLenginExpression
get_ty	pe() (androgu	ard.decom	piler.dad.inst	ruction.Ar	rayeLto <u>a</u> deEs	* *		(andro-
	method), 159					guard.decompiler.dad.ir	struction.Arra	yLoadExpression
get_tyl	pe() (androgu	ıard.decom	piler.dad.inst	ruction.Ca	_			
	method), 160					ed_vars()		(andro-
get_ty <sub>}</sub>	pe () (androgu method), 161	ıard.decom	piler.dad.inst	ruction.Co	nstant	guard.decompiler.dad.ir method), 159	struction.Arra	yStoreInstruction
get_ty	pe() (androgu	ıard.decom	piler.dad.inst	ruction.Ins	tayece <u>E</u> хүзі			(andro-
	method), 162					guard.decompiler.dad.ir	struction.Assi	gnExpression
get_ty	pe() (androgu	ıard.decom	piler.dad.inst	ruction.Inv				
	method), 163					ed_vars()		(andro-
get_ty]	pe () (androgu method), 162	ıard.decom	piler.dad.inst	ruction.IR	Form	guard.decompiler.dad.ir method), 160	istruction.Bina	ryExpression
get_ty	pe() (androgu	ıard.decom	piler.dad.inst	ruction.Ne	wdjæst <u>an</u> ce			(andro-
	method), 165					guard.decompiler.dad.ir	istruction.Cast	Expression
get_ty]	pe()(androgu	ıard.decom	piler.dad.inst	ruction.Sta	_			
	method), 165		., , , , ,			ed_vars()		(andro-
get_ty		iard.decom	piler.dad.inst	ruction.Un	aryExpres	s <b>ġoua</b> rd.decompiler.dad.ir	istruction.Che	ckCastExpression
~a+ +	method), 166	in	module	an dro	~~+ 110	<i>method</i> ), 161 ed_vars()		(andro-
get_ty	guard.core.byi			anaro-	get_us	ea_vars() guard.decompiler.dad.ir	estruction Con	`
get_ty		in		andro-		method), 161	isiruciion.Com	illonulExpression
gcc_cyl	guard.decomp			anaro		ed_vars()		(andro-
aet tw	pe_configs		<i>i)</i> , 173	(andro-	gcc_as	guard.decompiler.dad.ir	struction Con	`
900_01	guard.core.byi		nl.ARSCPars	•		method), 161	isir werrom com	unionan Elipression
	<i>method</i> ), 139			•	aet us	ed_vars()		(andro-
aet tvi	pe_idx()			(andro-	900_00	guard.decompiler.dad.ir	struction.Con	`
J <u>—</u> - <u>1</u> 1	guard.core.by	tecodes.dvi	n.EncodedAn	•		method), 161		
	method), 94				get_us	ed_vars()		(andro-
get_ty	pe_idx()			(andro-	- <u> </u>	guard.decompiler.dad.ir	struction.FillA	ArrayExpression
	guard.core.by	tecodes.dvi	n.EncodedTy <sub>l</sub>	peAddrPair		method), 162		
	method), 101					ed_vars()		(andro-
get_ty	pe_idx()			(andro-		guard.decompiler.dad.ir	struction.Fille	dArrayExpression
	guard.core.by	tecodes.dvi	n.FieldIdItem			method), 162		
	method), 103				get_us	ed_vars()		(andro-
get_ty	pe_idx()			(andro-		guard.decompiler.dad.ir	struction.Insta	ınceExpression
	guard.core.by	tecodes.dvi	n.TypeItem	method),		method), 162		
	132				get_us	ed_vars()		(andro-

guard.decompiler.dad.instruction.Ins	stanceInstru	ction	guard.core.bytecodes.axml.ARSCI	ResTableEntry
method), 163			method), 140	
<pre>get_used_vars()</pre>		get_va		(andro-
guard.decompiler.dad.instruction.Inv method), 163	okeInstructi	on	guard.core.bytecodes.dvm.Annota. method), 77	tionElement
get_used_vars()	(andro-	get_va	lue()	(andro-
guard.decompiler.dad.instruction.Inv method), 164	okeStaticIns	struction	guard.core.bytecodes.dvm.DBGBy method), 85	rtecode
<pre>get_used_vars()</pre>	(andro-	get_va		(andro-
guard.decompiler.dad.instruction.IR		> <b>_</b>	guard.core.bytecodes.dvm.Encode	*
method), 162			method), 94	•
<pre>get_used_vars()</pre>	(andro-	get_va	lue()	(andro-
guard.decompiler.dad.instruction.Mo method), 164	oveException	ıExpressio	nguard.core.bytecodes.dvm.Encode method), 101	dValue
<pre>get_used_vars()</pre>	(andro-	get_va	lue_arg()	(andro-
guard.decompiler.dad.instruction.Mo			guard.core.bytecodes.dvm.Encode	dValue
method), 164			method), 101	
get_used_vars()	(andro-	get_va	lue_from_tag()	(andro-
guard.decompiler.dad.instruction.Ne method), 164	wArrayExpr	ession	guard.core.bytecodes.apk.APK 74	method),
<pre>get_used_vars()</pre>	(andro-	get_va	lue_type()	(andro-
guard.decompiler.dad.instruction.Ne method), 165	wInstance		guard.core.bytecodes.dvm.Encode method), 101	dValue
<pre>get_used_vars()</pre>	(andro-	get_va		(andro-
guard.decompiler.dad.instruction.No			guard.core.bytecodes.dvm.Encode	dArray
method), 165			method), 94	
<pre>get_used_vars()</pre>	(andro-	get_va	lues()	(andro-
guard.decompiler.dad.instruction.Re	fExpression		guard.core.bytecodes.dvm.Packed	Switch
method), 165			method), 126	
<pre>get_used_vars()</pre>		get_va		(andro-
guard.decompiler.dad.instruction.Re method), 165			guard.core.bytecodes.dvm.SparseS method), 129	
<pre>get_used_vars()</pre>			riables() (in module	
guard.decompiler.dad.instruction.Sta	iticInstructio		guard.decompiler.dad.opcode_ins	
method), 165	, .		rtual_methods()	(andro-
<pre>get_used_vars()</pre>	(andro-		guard.core.bytecodes.dvm.ClassD	ataItem
guard.decompiler.dad.instruction.Sw			method), 81	
		get_vi	rtual_methods_size()	(andro-
get_used_vars()	(andro-		guard.core.bytecodes.dvm.ClassD	атапет
guard.decompiler.dad.instruction.Un	ıary£xpressi		method), 81	(an dua
method), 166	(andro-	get_vi	sibility()	(andro-
<pre>get_used_vars()</pre>			guard.core.bytecodes.dvm.Annota. method), 77	uonnem
method), 166	riudie	aet vm	() (androguard.core.analysis.anal	vsis MethodAnalysis
get_uses_implied_permission_lis	+ () (an-	gcc_viii	method), 60	y515.111C1110t111ttai y51.
droguard.core.bytecodes.apk.APK	method),	aet. vm		(andro-
73	,,	5 · · · · ·	guard.core.analysis.analysis.Class	•
<pre>get_utf16_size()</pre>	(andro-		method), 54	,
guard.core.bytecodes.dvm.StringDat		get_vm	analysis()	(andro-
method), 130		_	guard.core.bytecodes.dvm.DalvikV	,
get_value()	(andro-		method), 92	
guard. core. analysis. analysis. String A	nalysis	get_xm	1 () (androguard.core.bytecodes.as	xml.AXMLPrinter
method), 61			method), 143	
<pre>get_value()</pre>	(andro-	get_xm	l_obj()	(andro-

guard.core.bytecodes.axml.AXMLPrinter	method), 102
method), 143	getString() (andro-
<pre>get_xref_from()</pre> (andro-	guard.core.bytecodes.axml.StringBlock
guard.core.analysis.analysis.ClassAnalysis	method), 143
method), 54	getStyle()(androguard.core.bytecodes.axml.StringBlock
<pre>get_xref_from()</pre> (andro-	method), 143
guard.core.analysis.analysis.MethodAnalysis	getText() (androguard.core.bytecodes.axml.AXMLParser
method), 60	method), 142
<pre>get_xref_from()</pre> (andro-	go() (androguard.core.analysis.auto.AndroAuto
guard.core.analysis.analysis.StringAnalysis	method), 62
method), 61	goto() (in module andro-
<pre>get_xref_read()</pre> (andro-	guard.decompiler.dad.opcode_ins), 170
guard.core.analysis.analysis.FieldAnalysis	goto16() (in module andro-
method), 58	guard.decompiler.dad.opcode_ins), 170
<pre>get_xref_to()</pre> (andro-	
guard.core.analysis.analysis.ClassAnalysis	guard.decompiler.dad.opcode_ins), 170
method), 55	Graph (class in androguard.decompiler.dad.graph), 157
<pre>get_xref_to()</pre> (andro-	- GREATER (androguard.decompiler.dad.opcode_ins.Op
guard.core.analysis.analysis.MethodAnalysis	attribute), 167
method), 60	Green (androguard.core.androconf.Color attribute), 145
<pre>get_xref_write()</pre> (andro-	- Grey (androguard.core.androconf.Color attribute), 145
guard.core.analysis.analysis.FieldAnalysis	<pre>group_variables() (in module andro-</pre>
method), 58	guard.decompiler.dad.dataflow), 156
<pre>getAttributeCount() (andro-</pre>	
guard.core.bytecodes.axml.AXMLParser	Н
method), 141	has_side_effect() (andro-
getAttributeName() (andro-	
guard.core.bytecodes.axml.AXMLParser	method), 159
method), 141	has_side_effect() (andro-
<pre>getAttributeNamespace() (andro-</pre>	
guard.core.bytecodes.axml.AXMLParser	method), 160
method), 142	has_side_effect() (andro-
getAttributeUri() (andro	
guard.core.bytecodes.axml.AXMLParser	method), 160
method), 142	has_side_effect() (andro-
getAttributeValue() (andro	
guard.core.bytecodes.axml.AXMLParser	method), 163
method), 142	has_side_effect() (andro-
getAttributeValueData() (andro-	
guard.core.bytecodes.axml.AXMLParser	method), 163
method), 142	has_side_effect() (andro-
<pre>getAttributeValueType() (andro-</pre>	guard.decompiler.dad.instruction.IRForm
guard.core.bytecodes.axml.AXMLParser	method), 162
method), 142	has_side_effect() (andro-
<pre>getName() (androguard.core.bytecodes.axml.AXMLPa</pre>	guard.decompiler.dad.instruction.MoveExceptionExpression
method), 142	method), 164
getPrefix() (andro-	has_side_effect() (andro-
guard.core.bytecodes.axml.AXMLParser	guard. decompiler. dad. instruction. Move Expression
method), 142	method), 164
gets() (androguard.core.analysis.analysis.BasicBlocks	S has_side_effect() (andro-
method), 52	guard. decompiler. dad. instruction. Move Result Expression
gets() (androguard.core.analysis.analysis.Exceptions	,,
method), 57	has_side_effect() (andro-
gets() (androguard.core.bytecodes.dvm.FieldHIdIten	guard.decompiler.dad.instruction.StaticInstruction

method), 165	158
	ro- implements() (andro
guard.core.bytecodes.axml.ARSCHeader property), 135	guard.core.analysis.analysis.ClassAnalysis property), 55
	ro- inc_ind() (androguard.decompiler.dad.writer.Write
guard.core.bytecodes.dvm), 104	method), 175
guara.core.bytecoues.avm), 101	init_print_colors() (in module andro
	guard.misc), 180
identify_structures() (in module and	
guard.decompiler.dad.control_flow), 155	attribute), 146
if_stmt() (in module and	To a base a Rose and it as a first first for the second se
guard.decompiler.dad.dast), 152	guard.decompiler.dad.instruction), 162
if_struct() (in module and	
guard.decompiler.dad.control_flow), 155	guard.decompiler.dad.instruction), 163
ifeq() (in module and	in the second of
guard.decompiler.dad.opcode_ins), 170	guard.decompiler.dad.opcode_ins), 171
ifeqz() (in module and	
guard.decompiler.dad.opcode_ins), 170	guard.core.bytecodes.dvm), 104
ifge() (in module and	
guard.decompiler.dad.opcode_ins), 170	guard.core.bytecodes.dvm), 106
ifgez() (in module and	$r_{O^{+}}$ Instruction10t (class in andro
guard.decompiler.dad.opcode_ins), 170	guard.core.bytecodes.dvm), 106
ifgt() (in module and	<sub>ro-</sub> Instruction10x (class in andro
guard.decompiler.dad.opcode_ins), 170	guard.core.bytecodes.dvm), 107
ifgtz() (in module and	$r_{\mathcal{O}}$ - Instruction11n ( $class$ in $andro$
guard.decompiler.dad.opcode_ins), 170	guard.core.bytecodes.dvm), 107
ifle() (in module and	$_{ro extsf{o} extsf{-}}$ Instruction11x ( $class$ in $andro$
guard.decompiler.dad.opcode_ins), 170	guard.core.bytecodes.dvm), 107
iflez() (in module and	$_{ro extsf{o} extsf{-}}$ Instruction12x ( $class$ in $andro$
guard.decompiler.dad.opcode_ins), 170	guard.core.bytecodes.dvm), 108
iflt() (in module and	<sub>ro-</sub> Instruction20bc (class in andro
guard.decompiler.dad.opcode_ins), 170	guard.core.bytecodes.dvm), 108
ifltz() (in module and	<sub>ro-</sub> Instruction20t (class in andro
guard.decompiler.dad.opcode_ins), 170	guard.core.bytecodes.dvm), 108
ifne() (in module and	
guard.decompiler.dad.opcode_ins), 170	guard.core.bytecodes.dvm), 109
ifnez() (in module and	<sub>ro-</sub> Instruction21h (class in andro
guard.decompiler.dad.opcode_ins), 170	guard.core.bytecodes.dvm), 109
iget() (in module and	
guard.decompiler.dad.opcode_ins), 170	guard.core.bytecodes.dvm), 110
igetboolean() (in module and	
guard.decompiler.dad.opcode_ins), 170	guard.core.bytecodes.dvm), 110
igetbyte() (in module and	
guard.decompiler.dad.opcode_ins), 170	guard.core.bytecodes.dvm), 111
igetchar() (in module and	
<pre>guard.decompiler.dad.opcode_ins), 170</pre>	guard.core.bytecodes.dvm), 111
igetobject() (in module and	
guard.decompiler.dad.opcode_ins), 170	guard.core.bytecodes.dvm), 112
igetshort() (in module and	
guard.decompiler.dad.opcode_ins), 170	guard.core.bytecodes.dvm), 112
igetwide() (in module and	
guard.decompiler.dad.opcode_ins), 170	guard.core.bytecodes.dvm), 112
immediate_dominators() (and	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
guard.decompiler.dad.graph.Graph method	d), guard.core.bytecodes.dvm), 113

•	inttofloat() (in module andro-
guard.core.bytecodes.dvm), 113	guard.decompiler.dad.opcode_ins), 171
Instruction30t (class in andro-	inttolong() (in module andro-
guard.core.bytecodes.dvm), 114	guard.decompiler.dad.opcode_ins), 171
Instruction31c (class in andro-	inttoshort() (in module andro-
guard.core.bytecodes.dvm), 114	guard.decompiler.dad.opcode_ins), 171
Instruction31i (class in andro-	InvalidInstruction, 122
guard.core.bytecodes.dvm), 115	InvalidResourceError, 146
Instruction31t (class in andro-	INVOKE_DIRECT (andro-
guard.core.bytecodes.dvm), 115	guard.core.analysis.analysis.REF_TYPE
Instruction32x (class in andro-	attribute), 60
guard.core.bytecodes.dvm), 115	INVOKE_DIRECT_RANGE (andro-
Instruction35c (class in andro-	guard.core.analysis.analysis.REF_TYPE
guard.core.bytecodes.dvm), 116	attribute), 60
Instruction35mi (class in andro-	INVOKE_INTERFACE (andro-
guard.core.bytecodes.dvm), 116	guard.core.analysis.analysis.REF_TYPE
Instruction35ms (class in andro-	attribute), 60
guard.core.bytecodes.dvm), 117	INVOKE_INTERFACE_RANGE (andro-
Instruction3rc (class in andro-	guard.core.analysis.analysis.REF_TYPE
guard.core.bytecodes.dvm), 117	attribute), 60
Instruction3rmi (class in andro-	INVOKE_STATIC (andro-
guard.core.bytecodes.dvm), 118	guard.core.analysis.analysis.REF_TYPE
Instruction3rms (class in andro-	attribute), 60
guard.core.bytecodes.dvm), 118	INVOKE_STATIC_RANGE (andro-
Instruction40sc (class in andro-	guard.core.analysis.analysis.REF_TYPE
guard.core.bytecodes.dvm), 119	attribute), 60
Instruction41c (class in andro-	INVOKE_SUPER (andro-
guard.core.bytecodes.dvm), 119	guard.core.analysis.analysis.REF_TYPE
Instruction45cc (class in andro-	attribute), 60
guard.core.bytecodes.dvm), 119	INVOKE SUPER RANGE (andro-
Instruction4rcc (class in andro-	guard.core.analysis.analysis.REF_TYPE
guard.core.bytecodes.dvm), 120	attribute), 60
Instruction511 (class in andro-	INVOKE_VIRTUAL (andro-
guard.core.bytecodes.dvm), 120	guard.core.analysis.analysis.REF_TYPE
Instruction52c (class in andro-	attribute), 60
guard.core.bytecodes.dvm), 121	INVOKE_VIRTUAL_RANGE (andro-
•	
× ·	guard.core.analysis.analysis.REF_TYPE
guard.core.bytecodes.dvm), 121	attribute), 61
interpolate_tuple() (in module andro-	invokedirect() (in module andro-
guard.core.androconf), 146	guard.decompiler.dad.opcode_ins), 171
Interval (class in androguard.decompiler.dad.node),	InvokeDirectInstruction (class in andro-
166	guard.decompiler.dad.instruction), 163
intervals() (in module andro-	invokedirectrange() (in module andro-
guard.decompiler.dad.control_flow), 155	guard.decompiler.dad.opcode_ins), 171
INTSHL (androguard.decompiler.dad.opcode_ins.Op at-	InvokeInstruction (class in andro-
tribute), 167	guard.decompiler.dad.instruction), 163
INTSHR (androguard.decompiler.dad.opcode_ins.Op at-	invokeinterface() (in module andro-
tribute), 167	guard.decompiler.dad.opcode_ins), 171
inttobyte() (in module andro-	invokeinterfacerange() (in module andro-
guard.decompiler.dad.opcode_ins), 171	guard.decompiler.dad.opcode_ins), 171
inttochar() (in module andro-	InvokeRangeInstruction (class in andro-
guard.decompiler.dad.opcode_ins), 171	guard.decompiler.dad.instruction), 163
inttodouble() (in module andro-	invokestatic() (in module andro-
guard.decompiler.dad.opcode_ins), 171	guard.decompiler.dad.opcode_ins), 171

InvokeStaticInstruction (class in andro-	method), 163
guard.decompiler.dad.instruction), 163	is_call() (androguard.decompiler.dad.instruction.IRForm
invokestaticrange() (in module andro-	method), 162
guard.decompiler.dad.opcode_ins), 171	$is\_call()$ (and roguard. decompiler. dad. instruction. Move Expression
invokesuper() (in module andro-	method), 164
guard.decompiler.dad.opcode_ins), 171	is_class_present() (andro-
invokesuperrange() (in module andro-	guard.core.analysis.analysis.Analysis method),
guard.decompiler.dad.opcode_ins), 171	52
invokevirtual() (in module andro-	is_complex() (andro-
guard.decompiler.dad.opcode_ins), 171	guard.core.by tecodes.axml. ARSCR es Table Entry
invokevirtualrange() (in module andro-	method), 141
guard.decompiler.dad.opcode_ins), 171	$\verb is_cond()  (and roguard. decompiler. dad. instruction. Conditional Expression and the con$
iput() (in module andro-	method), 161
guard.decompiler.dad.opcode_ins), 171	$\verb is_cond()  (and roguard. decompiler. dad. instruction. Conditional ZExpression and all the conditional ZExpression and the$
iputboolean() (in module andro-	method), 161
guard.decompiler.dad.opcode_ins), 171	is_cond() (androguard.decompiler.dad.instruction.IRForm
iputbyte() (in module andro-	method), 162
guard.decompiler.dad.opcode_ins), 171	<pre>is_cond() (androguard.decompiler.dad.node.NodeType</pre>
iputchar() (in module andro-	property), 167
guard.decompiler.dad.opcode_ins), 171	<pre>is_const() (androguard.decompiler.dad.instruction.BaseClass</pre>
iputobject() (in module andro-	method), 160
guard.decompiler.dad.opcode_ins), 171	<pre>is_const() (androguard.decompiler.dad.instruction.CastExpression</pre>
iputshort() (in module andro-	method), 161
guard.decompiler.dad.opcode_ins), 171	<pre>is_const() (androguard.decompiler.dad.instruction.CheckCastExpressi</pre>
iputwide() (in module andro-	method), 161
guard.decompiler.dad.opcode_ins), 171	<pre>is_const() (androguard.decompiler.dad.instruction.Constant</pre>
IRForm (class in andro-	method), 161
guard.decompiler.dad.instruction), 162	<pre>is_const() (androguard.decompiler.dad.instruction.IRForm</pre>
is_android() (in module andro-	method), 162
guard.core.androconf), 146	<pre>is_const() (androguard.decompiler.dad.instruction.Param</pre>
is_android_api() (andro-	method), 165
guard.core.analysis.analysis.ClassAnalysis	is_default() (andro-
method), 55	guard.core.bytecodes.axml.ARSCResTableConfig
is_android_api() (andro-	method), 140
guard.core. analysis. analysis. Method Analysis	is_endless() (andro-
method), 60	guard.decompiler.dad.node.LoopType prop-
is_android_raw() (in module andro-	erty), 167
guard.core.androconf), 146	is_external() (andro-
is_androidtv() (andro-	guard.core.analysis.analysis.ClassAnalysis
guard.core.bytecodes.apk.APK method),	method), 55
74	is_external() (andro-
is_ascii_obfuscation() (in module andro-	guard.core.analysis.analysis.MethodAnalysis
guard.core.analysis.analysis), 61	method), 60
is_ascii_problem() (in module andro-	<pre>is_ident() (androguard.decompiler.dad.instruction.IRForm</pre>
guard.core.androconf), 146	method), 162
is_cached_instructions() (andro-	<pre>is_ident() (androguard.decompiler.dad.instruction.Variable</pre>
guard.core.bytecodes.dvm.DCode method),	method), 166
86	is_leanback() (andro-
is_cached_instructions() (andro-	guard.core.bytecodes.apk.APK method),
guard.core.bytecodes.dvm.EncodedMethod	74
method), 100	is_multidex() (andro-
is_call()(androguard.decompiler.dad.instruction.Ass	
method), 160	74
is_call()( <i>androguard.decompiler.dad.instruction.Inve</i>	okalm <u>s</u> townetionritten() (andro-

guard.core.analysis.analysis.StringAna method), 61		is_tag	_matched() guard.core.bytecodes.apk.APK	(andro- method),
<pre>is_packed()</pre>	(andro- er	is_thr	75 ow() (androguard.decompiler.dad.n property), 167	ode.NodeType
is_posttest() guard.decompiler.dad.node.LoopType	(andro- prop-		id() (androguard.core.bytecodes.as method), 142	
<pre>erty), 167 is_pretest()</pre>	(andro-	is_val	id() (androguard.core.bytecodes.ax method), 143	cml.AXMLPrinter
<pre>guard.decompiler.dad.node.LoopType erty), 167</pre>	prop-	is_val	id_APK() guard.core.bytecodes.apk.APK	(andro- method),
<pre>is_propagable()       guard.decompiler.dad.instruction.Assig</pre>	(andro- nExpressi	<i>o</i> ins_wea		nl.ARSCResTableEntry
method), 160	, ,		method), 141	
<pre>is_propagable()           guard.decompiler.dad.instruction.FillAn           method), 162</pre>		is_wea ssion	rable() guard.core.bytecodes.apk.APK 75	(andro- method),
is_propagable()	(andro-	isOpen	() (androguard.session.Session meth	hod), 183
guard.decompiler.dad.instruction.IRFormethod), 162	rm	J		
<pre>is_propagable()           guard.decompiler.dad.instruction.Move           method), 164</pre>	(andro- ResultExp	JADXDe Presign()	compilerError, 179 (androguard.core.mutf8.MUTF8St method), 144	ring class
is_propagable() guard.decompiler.dad.instruction.NewA	(andro- ArrayExpre	JSONWr ession		andro-
<pre>method), 164 is_propagable()</pre>	(andro-	jump_s	tmt () (in module guard.decompiler.dad.dast), 152	andro-
guard.decompiler.dad.instruction.RefExmethod), 165	xpression	L		
<pre>is_public()      guard.core.bytecodes.axml.ARSCResTa</pre>	(andro- bleEntry	last()	(androguard.decompiler.dad.graph.Comethod), 157	GenInvokeRetName
<pre>method), 141 is_reference()</pre>	(andro-		(androguard.core.bytecodes.dvm.Ins	
guard.core.bytecodes.axml.ARSCResStr method), 140	ringPoolR	<i>ef</i> length	(androguard.core.bytecodes.dvm.Ins attribute), 106	truction00x
<pre>is_return()       guard.decompiler.dad.node.NodeType       erty), 167</pre>	(andro- prop-		(androguard.core.bytecodes.dvm.Insattribute), 107	
is_signed() (androguard.core.bytecodes.a method), 74	ıpk.APK	_	(androguard.core.bytecodes.dvm.Ins attribute), 107	
is_signed_v1()	(andro-	length	(androguard.core.bytecodes.dvm.Ins attribute), 107	truction11n
74	nethod),	length	(androguard.core.bytecodes.dvm.Ins attribute), 108	truction11x
<pre>is_signed_v2()     guard.core.bytecodes.apk.APK</pre>	(andro- nethod),		(androguard.core.bytecodes.dvm.Insattribute), 108	
is_signed_v3()	(andro- nethod),	_	(androguard.core.bytecodes.dvm.Ins attribute), 108	
75			(androguard.core.bytecodes.dvm.Insattribute), 109	
is_stmt() (androguard.decompiler.dad.node.l property), 167		length	(androguard.core.bytecodes.dvm.Insattribute), 109	truction21c
<pre>is_switch()      guard.decompiler.dad.node.NodeType      erty), 167</pre>	(andro- prop-	length	(androguard.core.bytecodes.dvm.Ins attribute), 110	truction21h

length (androguard.core.bytecodes.dvm.Instruction21s	(andro- method),
attribute), 111 LEQUAL (androguard.decompiler.dad.opco	de_ins.Op at-
length (androguard.core.bytecodes.dvm.Instruction22b tribute), 167	
attribute), 111 LinearSweepAlgorithm (class	in andro-
length (androguard.core.bytecodes.dvm.Instruction22c guard.core.bytecodes.dvm), 122	( J
attribute), 112	(andro- VMFormat
attribute), 112 guard.core.bytecodes.avm.histraction22cs guard.core.bytecodes.avm.batvik	VIVITOTHIAI
length (androguard.core.bytecodes.dvm.Instruction22s literal() (in module	andro-
attribute), 112 guard.decompiler.dad.dast), 152	
length (androguard.core.bytecodes.dvm.Instruction22t literal_bool() (in module attribute), 113 guard.decompiler.dad.dast), 152	e andro-
length(androguard.core.bytecodes.dvm.Instruction22x literal_class() (in modul	e andro-
attribute), 113 guard.decompiler.dad.dast), 152	1
length (androguard.core.bytecodes.dvm.Instruction23x literal_double() (in modu attribute), 114 guard.decompiler.dad.dast), 152	le andro-
length (androguard.core.bytecodes.dvm.Instruction30t literal_float() (in modul	'e andro-
attribute), 114 guard.decompiler.dad.dast), 152	
<pre>length(androguard.core.bytecodes.dvm.Instruction31c literal_hex_int() (in mode</pre>	ıle andro-
attribute), 115 guard.decompiler.dad.dast), 152	
<pre>length (androguard.core.bytecodes.dvm.Instruction31i literal_int() (in module</pre>	andro-
attribute), 115  guard.decompiler.dad.dast), 152	andua
length (androguard.core.bytecodes.dvm.Instruction31t literal_long() (in module attribute), 115 guard.decompiler.dad.dast), 152	e andro-
length (androguard.core.bytecodes.dvm.Instruction32x literal_null() (in module	e andro-
attribute), 116 guard.decompiler.dad.dast), 152	
length(androguard.core.bytecodes.dvm.Instruction35c literal_string() (in modu	le andro-
attribute), 116 guard.decompiler.dad.dast), 152	
length (androguard.core.bytecodes.dvm.Instruction35mi load() (androguard.core.bytecodes.dvm. attribute), 117 method), 97	
length (androguard.core.bytecodes.dvm.Instruction35ms load() (androguard.core.bytecodes.dvm.Instructio	
length (androguard.core.bytecodes.dvm.Instruction3rc Load() (in module androguard.session), 1	
attribute), 118 load_api_specific_resource_mo length (androguard.core.bytecodes.dvm.Instruction3rmi module androguard.core.androco	
attribute), 118 load_array_exp() (in modu	
length (androguard.core.bytecodes.dvm.Instruction3rms guard.decompiler.dad.opcode_instruction3rms length (androguard.core.bytecodes.dvm.Instruction3rms guard.decompiler.dad.opcode_instruction3rms length (androguard.core.bytecodes.dvm.Instruction3rms length (androguard.core.bytecodes.dvm.Instruction3rms) length (androguard.core.bytecodes.dvm.Instruction3r	
attribute), 119 load_permission_mappings() (in n	
$\verb length  (and roguard.core.bytecodes.dvm.Instruction 40sc \\   guard.core.api\_specific\_resource. \\  $	s), 66
attribute), 119 load_permissions() (in mode	
length (androguard.core.bytecodes.dvm.Instruction41c guard.core.api_specific_resource.	
attribute), 119 local() (in module androguard.decomplength (androguard.core.bytecodes.dvm.Instruction45cc 152	uer.aaa.aasi),
attribute), 120 local_decl_stmt() (in modi	ıle andro-
length (androguard.core.bytecodes.dvm.Instruction4rcc guard.decompiler.dad.dast), 152	
attribute), 120 LONGSHL (androguard.decompiler.dad.o	pcode_ins.Op
length (androguard.core.bytecodes.dvm.Instruction51l attribute), 167	
attribute), 121 LONGSHR (androguard.decompiler.dad.o	pcode_ins.Op
length (androguard.core.bytecodes.dvm.Instruction52c attribute), 167 attribute), 121 longtodouble() (in modulo	e andro-
	e unaro-
<pre>length (androguard.core.bytecodes.dvm.Instruction5rc</pre>	s), 171

guard.decompiler.dad.opcode_ins), 171	
<pre>longtoint() (in module andro- guard.decompiler.dad.opcode_ins), 171</pre>	MethodAnalysis (class in andro-
<pre>loop_follow() (in module andro-</pre>	guard.core.analysis.analysis), 59
guard.decompiler.dad.control_flow), 155	
loop_stmt() (in module andro-	
guard.decompiler.dad.dast), 152	MethodClassAnalysis (class in andro-
loop_struct() (in module andro-	
<pre>guard.decompiler.dad.control_flow), 155</pre>	MethodFilter (class in andro-
<pre>loop_type() (in module andro-</pre>	guard.decompiler.decompiler), 179
guard.decompiler.dad.control_flow), 155	
LoopBlock (class in andro-	
guard.decompiler.dad.basic_blocks), 154	MethodIdItem (class in andro-
	guard.core.bytecodes.dvm), 124
	•
166	MethodIdItemInvalid (class in andro-
LOWER (androguard.decompiler.dad.opcode_ins.Op at-	guard.core.bytecodes.dvm), 125
tribute), 168	MOD (androguard.decompiler.dad.opcode_ins.Op at-
lstrip() (androguard.core.mutf8.MUTF8String	tribute), 168
method), 144	monitorenter() (in module andro-
	guard.decompiler.dad.opcode_ins), 171
M	MonitorEnterExpression (class in andro-
main() (in module andro-	
guard.decompiler.dad.decompile), 157	
` ` ` `	guard.decompiler.dad.opcode_ins), 171
guard.core.androconf), 146	MonitorExitExpression (class in andro-
make_node() (in module andro-	guard.decompiler.dad.instruction), 164
guard.decompiler.dad.graph), 159	move() (in module andro-
MakeProperties (class in andro-	guard.decompiler.dad.opcode_ins), 171
guard.decompiler.dad.node), 167	move16() (in module andro-
MapItem (class in androguard.core.bytecodes.dvm), 122	guard.decompiler.dad.opcode_ins), 171
	moveexception() (in module andro-
MapList (class in androguard.core.bytecodes.dvm), 122	
mark_loop() (in module andro-	
guard.decompiler.dad.control_flow), 155	MoveExceptionExpression (class in andro-
<pre>mark_loop_rec() (in module andro-</pre>	
<pre>guard.decompiler.dad.control_flow), 155</pre>	MoveExpression (class in andro-
merge_inner() (in module andro-	guard.decompiler.dad.instruction), 164
guard.decompiler.dad.util), 175	movefrom16() (in module andro-
method2dot() (in module androguard.core.bytecode),	guard.decompiler.dad.opcode_ins), 171
150	moveobject() (in module andro-
method2format() (in module andro-	guard.decompiler.dad.opcode_ins), 171
•	moveobject16() (in module andro-
guard.core.bytecode), 150	· · · · · · · · · · · · · · · · · · ·
<pre>method2jpg() (in module androguard.core.bytecode),</pre>	guard.decompiler.dad.opcode_ins), 172
150	moveobjectfrom16() (in module andro-
method2json() (in module andro-	guard.decompiler.dad.opcode_ins), 172
guard.core.bytecode), 150	moveresult() (in module andro-
method2json_direct() (in module andro-	guard.decompiler.dad.opcode_ins), 172
guard.core.bytecode), 151	MoveResultExpression (class in andro-
method2json_undirect() (in module andro-	guard.decompiler.dad.instruction), 164
- · · · · · · · · · · · · · · · · · · ·	moveresultobject() (in module andro-
guard.core.bytecode), 151	guard.decompiler.dad.opcode_ins), 172
method2png() (in module androguard.core.bytecode),	Summaccompiler.aua.opcode_ms), 1/2
151	morrorocall traido () (in modulo andre
	moveresultwide() (in module andro-
method_idx_diff (andro-	guard.decompiler.dad.opcode_ins), 172
method_idx_diff (andro- guard.core.bytecodes.dvm.EncodedMethod attribute), 100	· · · · · · · · · · · · · · · · · · ·

movew	ide16()						(in		andro-	
		_	opcode_ins), 1					.opcode_ins),		
movew	idefrom16(						(in		andro-	
			opcode_ins), 1					.opcode_ins),		
MUL (	(androguard.de	_	lad.opcode_ins	s.Op at-	negint					
	tribute), 168							.opcode_ins),		
muldo	uble()				neglor		(in		andro-	
			$pcode\_ins), 1$					.opcode_ins),		
muldo	uble2addr(	) (in	module	andro-	NEQUAI	□ (androgua	rd.decompil	er.dad.opcode	_ins.Op at-	
			$ppcode\_ins), 1$			tribute), 1				
mulfl	oat()	(in	module	andro-	new()	(androguara	l.decompilei	:dad.graph.Ge	enInvokeRetName	e
	guard.decon	ıpiler.dad.d	opcode_ins), 1	72		method), 1	157			
mulfl	oat2addr()	(in	module	andro-	new_zi	p()	(androguara	l.core.bytecod	es.apk.APK	
	guard.decon	ipiler.dad.d	opcode_ins), 1'	72		method), 7	75			
mulin <sup>.</sup>	t() (	in	module	andro-	newarr	ray()	(in	module	andro-	
	guard.decon	ipiler.dad.d	opcode_ins), 1	72		guard.dec	ompiler.dad	.opcode_ins),	172	
mulin <sup>.</sup>	t2addr()	(in	module	andro-	NewArr	rayExpres	ssion	(class in	andro-	
	guard.decon	piler.dad.c	opcode_ins), 1	72		guard.dec	ompiler.dad	instruction), 1	164	
mulin <sup>.</sup>	tlit16()				NewIns	stance	(class	in	andro-	
	guard.decon	piler.dad.c	opcode_ins), 1	72		guard.dec	ompiler.dad	instruction), 1	165	
mulin <sup>.</sup>	tlit8()	_	_		newins	•	•		andro-	
			opcode_ins), 1					.opcode_ins),	172	
mullo	ng()	_	_		Node (a	-	_	bytecode), 14		
	_		opcode_ins), 1				-	ompiler.dad.no		
mullo	ng2addr()	_	_				-	ırd.decompilei		
	_		opcode_ins), 1			167	0	1	,,	
MUTF8	String(class				nop()	(i	n	module	andro-	
	2 (	0	,	,,	1 17	,		.opcode_ins),		
N					NopExp	-	_	s in	andro-	
name (	) (androguard.	core analy	sis analysis Cl	accAnalysis				instruction), 1		
manic (	property), 55		sis.anaiysis.Ci	assi mai ysis		-	_	droconf.Color		
name (	) (androguard.		sis analysis Fi	eldAnalysis		145		J	,,	
manne (	property), 59		sis.anaiysis.1 ii	eiuz irmi ysis	NOT (	androguard	.decompiler.	dad.opcode_ii	ns.Ov at-	
nama (	) (androguard.		sis analysis M	ethod Analy		tribute), 1			v P	
manne (	property), 60		sis.unui ysis.ivi	етоилниту.	notint	: ()		module	andro-	
nama (	) (androguara		codes avml AY	MI Parsor				opcode_ins),		
manne (	property), 14		oues.uxmi.AX	WILI UISEI	not lor	-	(in	-	andro-	
2222	property), 15 pace()	† <i>L</i>		(andro				.opcode_ins),		
IIailles <sub>]</sub>		vtacadas a	xml.AXMLPar		nsmap	() (androgu	ard.core.bv1	ecodes.axml.A	XMLParser	
	property), 14	-	xmi.AxiviLi ar	ser	monag.	property),	•			
NEG (	androguard.de		lad oncode in	s.Op at-	num()(			:dad.basic_blo	ocks.TrvBlock	
NEG (	tribute), 168	_	idd.opcode_ins	в.Ор ш-	11 ( ) (	property),	_		ye.us.17, j210 e.u	
200()	(androguard.d		dad basic bloo	eks CondRle	<i>a</i> umber				(andro-	
neg ()			aaa.basic_bioc	ks.Conabic	ускания с т	vuard.dec	ompiler.dad	.basic_blocks.	`	
202()	method), 153 (androguard.d		dad basia bloc	ka Canditia		method), 1		ouste_oreens.	BusicBioch	
neg ()		-	aaa.vasic_vioc	ks.Conaino		ins()			(andro-	
()	method), 153		dad basis blos	dra I o am Dla			ompiler dad	graph.Graph	method),	
neg ()	(androguard.d	_	aaa.vasic_vioc	кѕ.ьоорыо	СК	158	ompiler.aaa	. 8 гарн. Өгарн	memoa),	
()	method), 154		1 11 . 11	1 01 .0.	ו ומיי					
neg()	(androguard.d		aaa.vasic_bloc	ks.SnortCii	снявіоск	5				
	method), 154		11:	. C 1	 	:. + a b	20 ()	n madula	and wa	
neg()	(androguard.d	_	aaa.instruction	i.Condition	а <b>нхр</b> те\$ <del>5</del>				andro-	
	method), 16		11:	. C 1	<b>P</b> VAD D -	~	e.bytecode),		andro-	
neg()	(androguard.d method), 16	_	uaa.instruction	i.Conaitiona	u <b>szexpr</b> es		e.bytecodes.	elass in	anaro-	
	memon in					zwara.com	$\dots$	WY11014 14J		

guard.core.bytecodes.dvm), 125	o- parse_v2_signing_block() (andro- guard.core.bytecodes.apk.APK method),
off_to_pos() (andr guard.core.bytecodes.dvm.DCode method	o- 75 l), parse_v2_v3_signature() (andro-
86	guard.core.bytecodes.apk.APK method),
OffObj (class in androguard.core.bytecodes.dvm), 12	
OP (androguard.core.bytecodes.dvm.Instruction of tribute), 105	nt- parse_v3_signing_block() (andro- guard.core.bytecodes.apk.APK method),
Op (class in androguard.decompiler.dad.opcode_in	· · · · · · · · · · · · · · · · · · ·
167	peek() (androguard.core.bytecode.BuffHandle nethod), 147
tribute), 168	permission_api_name() (andro-
order_cases() (andr guard.decompiler.dad.basic_blocks.SwitchBl	o- guard.core.analysis.analysis.ExternalMethod
method), 154	place_declarations() (in module andro-
orint() (in module and	
guard.decompiler.dad.opcode_ins), 172	pop() (androguard.core.analysis.analysis.BasicBlocks
orint2addr() (in module and guard.decompiler.dad.opcode_ins), 172	
orintlit16() (in module and	
guard.decompiler.dad.opcode_ins), 172	158
	o- preds() (androguard.decompiler.dad.graph.Graph
guard.decompiler.dad.opcode_ins), 172	method), 158
orlong() (in module and guard.decompiler.dad.opcode_ins), 172	o- PrettyShow() (in module androguard.core.bytecode), 149
orlong2addr() (in module andr guard.decompiler.dad.opcode_ins), 172	o- PrettyShowEx() (in module andro- guard.core.bytecode), 149
P	<pre>print_classes_hierarchy() (andro- guard.core.bytecodes.dvm.DalvikVMFormat</pre>
PackageContext (class in and	T D 00
guard.core.bytecodes.axml), 143 PackedSwitch (class in and	process() (androguard.decompile.dad.decompile.DvClass
guard.core.bytecodes.dvm), 125	process() (androguard.decompiler.dad.decompile.DvMachin
packedswitch() (in module and guard.decompiler.dad.opcode_ins), 172	process() (androguard.decompiler.dad.decompile.DvMethod
packer()(androguard.core.bytecodes.dvm.ClassMan	nager method), 157
property), 85	process_and_show() (andro- n), guard.decompiler.dad.decompile.DvMachine
Param ( <i>ciass in anaroguara.aecompuer.aaa.instructio</i> i 165	method), 157
ParameterAnnotation (class in and	
guard.core.bytecodes.dvm), 126	guard.decompiler.dad.decompile.DvClass
parenthesis() (in module and	
guard.decompiler.dad.dast), 152	ProtoHIdItem (class in andro-
parse() (androguard.core.bytecodes.dvm.MapItemethod), 122	ProtoIdItem (class in andro-
<pre>parse_descriptor() (in module and</pre>	o- guard.core.bytecodes.dvm), 127 ProtoIdItemInvalid (class in andro-
parse_id() (androguard.core.bytecodes.axml.ARSC static method), 139	Parser guard.core.bytecodes.dvm), 128 Purple (androguard.core.androconf.Color attribute),
parse_lxml_dom()	1.45
guard.core.bytecodes.apk), 76	push() (androguard.core.analysis.analysis.BasicBlocks
parse_signatures_or_digests() (and	
guard.core.bytecodes.apk.APK method), 75	push () (androguard.core.analysis.analysis.DVMBasicBlock method), 57

<pre>put_ate_value()</pre>	reload() (androguard.core.bytecodes.dvm.FieldIdItem
method), 136	reload() (androguard.core.bytecodes.dvm.MethodHIdItem
put_item_value() (andro-	method), 124
	eResalvari () (androguard.core.bytecodes.dvm.MethodIdItem
method), 136	method), 125
D	remdouble() (in module andro-
R	guard.decompiler.dad.opcode_ins), 172
<pre>reach_def_analysis() (in module andro- guard.decompiler.dad.dataflow), 156</pre>	remdouble2addr() (in module andro- guard.decompiler.dad.opcode_ins), 172
read() (androguard.core.bytecode.BuffHandle	remfloat() (in module andro-
method), 147	guard.decompiler.dad.opcode_ins), 172
read() (in module androguard.util), 184	remfloat2addr() (in module andro-
<pre>read_at() (androguard.core.bytecode.BuffHandle</pre>	guard.decompiler.dad.opcode_ins), 173
method), 147	remint() (in module andro-
read_b() (androguard.core.bytecode.BuffHandle	guard.decompiler.dad.opcode_ins), 173
method), 148	remint2addr() (in module andro-
read_null_terminated_string() (in module	guard.decompiler.dad.opcode_ins), 173
androguard.core.bytecodes.dvm), 134	remintlit16() (in module andro-
read_uint32_le() (andro-	<pre>guard.decompiler.dad.opcode_ins), 173 remintlit8() (in module andro-</pre>
guard.core.bytecodes.apk.APK method),	remintlit8() (in module andro- guard.decompiler.dad.opcode_ins), 173
76	remlong() (in module andro-
readat() (androguard.core.bytecode.BuffHandle	guard.decompiler.dad.opcode_ins), 173
method), 148 readNullString() (andro-	remlong2addr() (in module andro-
guard.core.bytecode.BuffHandle method),	guard.decompiler.dad.opcode_ins), 173
147	remove_colors() (in module andro-
readsleb128() (in module andro-	guard.core.androconf), 146
guard.core.bytecodes.dvm), 134	remove_defined_var() (andro- guard.decompiler.dad.instruction.AssignExpression
readuleb128() (in module andro- guard.core.bytecodes.dvm), 134	method), 160
readuleb128p1() (in module andro-	remove_defined_var() (andro-
guard.core.bytecodes.dvm), 134	guard.decompiler.dad.instruction.IRForm
Red (androguard.core.androconf.Color attribute), 145	method), 162
REF_CLASS_USAGE (andro-	remove_ins() (andro-
guard.core.analysis.analysis.REF_TYPE	guard.decompiler.dad.basic_blocks.BasicBlock
attribute), 61	<pre>method), 153 remove_ins() (andro-</pre>
REF_NEW_INSTANCE (andro-	guard.decompiler.dad.graph.Graph method),
guard.core.analysis.analysis.REF_TYPE attribute), 61	158
REF_TYPE (class in androguard.core.analysis.analysis),	remove_node() (andro-
60	guard.decompiler.dad.graph.Graph method),
RefExpression (class in andro-	158
guard.decompiler.dad.instruction), 165	replace() (androguard.core.mutf8.MUTF8String
register_propagation() (in module andro-	method), 144
guard.decompiler.dad.dataflow), 156	replace() (androguard.decompiler.dad.instruction.ArrayLengthExpress
reload() (androguard.core.bytecodes.dvm.ClassDefItem	* * * * * * * * * * * * * * * * * * *
method), 83	replace() (androguard.decompiler.dad.instruction.ArrayLoadExpressio
reload() (androguard.core.bytecodes.dvm.DebugInfoIte	* * * * * * * * * * * * * * * * * * *
method), 93	replace() (androguard.decompiler.dad.instruction.ArrayStoreInstructio
reload() (androguard.core.bytecodes.dvm.EncodedField	d method), 159
method), 97	$\verb"replace" () (and roguard. decompiler. dad. instruction. Assign Expression$
reload() (androguard.core.bytecodes.dvm.EncodedMeth	
method), 100	$\verb"replace" () \textit{ (and roguard. decompiler. dad. instruction. Binary Expression}$

method), 160			method), 159	
replace() (androguard.decompiler.dad.instr	ruction.Chec	ktGqsiExqp		(andro-
method), 161				ruction.ArrayStoreInstruction
replace() (androguard.decompiler.dad.instr method), 161			p <b>neskiod)</b> , 159 e var()	(andro-
replace() (androguard.decompiler.dad.instr		-		*
method), 161	исноп.Сопа	monuizz	method), 160	ruction.AssignExpression
replace() (androguard.decompiler.dad.instr	ruction.FillA	nrayExpre	es <u>s</u> ioar()	(andro-
method), 162			guard.decompiler.dad.inst	ruction.BinaryExpression
replace()(androguard.decompiler.dad.instr	ruction.Filled	dArrayEx <sub>l</sub>	pmessliod), 160	
method), 162		_	e_var()	(andro-
replace() (androguard.decompiler.dad.instr	ruction.Insta	nceExpre		ruction.CheckCastExpression
method), 162			method), 161	
replace() (androguard.decompiler.dad.instr	ruction.Insta	nce pistra		(andro-
method), 163			guard.decompiler.dad.inst	ruction.ConditionalExpression
replace() (androguard.decompiler.dad.instr	ruction.Invok	eInstruct	ionnethod), 161	
method), 163		-	e_var()	(andro-
replace() (androguard.decompiler.dad.instr method), 162	ruction.IRFo	rm	guard.decompiler.dad.inst method), 161	ruction.ConditionalZExpression
replace() (androguard.decompiler.dad.instr	ruction.Move	Fæmre soin	**	(andro-
method), 164		23/2 2004		ruction.FillArrayExpression
replace() (androguard.decompiler.dad.instr	ruction.New/	ArrayExp	= =	
method), 164			e_var()	(andro-
replace() (androguard.decompiler.dad.instr		_		`
method), 165			method), 162	
replace() (androguard.decompiler.dad.instr	ruction.RefE	voeeskieen		(andro-
method), 165	v	1 1		ruction.InstanceExpression
replace() (androguard.decompiler.dad.instr	ruction.Retur	nInstruct	= =	•
method), 165		replac	e_var()	(andro-
replace()(androguard.decompiler.dad.instr	ruction.Static	Expressi	o <b>g</b> uard.decompiler.dad.inst	ruction.InstanceInstruction
method), 165			method), 163	
replace()(androguard.decompiler.dad.instr	ruction.Statio	Hestruatio	<b>⊕</b> n_var()	(andro-
method), 166			guard.decompiler.dad.inst	ruction.InvokeInstruction
replace()(androguard.decompiler.dad.instr	ruction.Switc	hExpress	i <b>on</b> ethod), 163	
method), 166			e_var()	(andro-
replace() (androguard.decompiler.dad.instr	ruction.Unar	yExpressi	= =	ruction.IRForm
method), 166	(andro	man 1 a a	method), 162	(andro-
replace_lhs() guard.decompiler.dad.instruction.Ass	(andro-		e_var() guard.decompiler.dad.inst	*
method), 160	signExpressio	m	method), 164	ruction.MoveExpression
	(andro	roplas	* *	(andro-
replace_lhs()		гертас	e_var()	*
guard.decompiler.dad.instruction.IRI method), 162	rorm		method), 164	ruction.NewArrayExpression
replace_lhs()	(andro-	replac	e_var()	(andro-
guard.decompiler.dad.instruction.Mo	veException	Expressio	nguard.decompiler.dad.inst	ruction.RefExpression
method), 164			method), 165	
replace_lhs()	(andro-	replac	e_var()	(andro-
guard.decompiler.dad.instruction.Mo	veExpression	ı	guard.decompiler.dad.inst	ruction.ReturnInstruction
method), 164			method), 165	
replace_var()	(andro-	replac	e_var()	(andro-
guard.decompiler.dad.instruction.Arr	rayLengthExp	oression	guard.decompiler.dad.inst	ruction.StaticInstruction
method), 159	- •		method), 166	
replace_var()	(andro-	replac	e_var()	(andro-
guard.decompiler.dad.instruction.Arr	rayLoadExpr	ession	guard.decompiler.dad.inst	ruction.SwitchExpression

method), 166	set_code_idx()	(andro-
replace_var() (and	dro- guard.core.bytecodes.dvm.EncodedMeth	od
guard.decompiler.dad.instruction.UnaryExp		
method), 166		(andro-
$\verb"reset" () \textit{ (and roguard.session.Session method)}, 183$	•	r
$\verb resolve ()  (and roguard.core.by tecodes.axml. ARSC   and roguard.core.by tecodes.axml. Archive.by tecodes.axml. Archive.b$		
method), 136	<del>-</del>	(andro-
ResParserError, 143	guard.core.bytecodes.dvm.DalvikVMFor	mat
,	dro- method), 92	/ 1
3 · · · · · · · · · · · · · · · · · · ·		(andro-
_ ,	dro- guard.core.analysis.analysis.DVMBasic	Віоск
guard.decompiler.dad.dast), 152 ReturnBlock (class in and	<pre>method), 57 dro- set_fathers()</pre>	(andro-
guard.decompiler.dad.basic_blocks), 154	guard.core.analysis.analysis.DVMBasica	,
	dro- method), 57	Diock
guard.decompiler.dad.instruction), 165		(andro-
returnobject() (in module and		,
guard.decompiler.dad.opcode_ins), 173	method), 85	,
		(andro-
guard.decompiler.dad.opcode_ins), 173	guard.core.bytecodes.dvm.ClassManage	•
returnwide() (in module and		
guard.decompiler.dad.opcode_ins), 173		(andro-
rrmdir() (in module androguard.core.androconf),		•
rsplit() (androguard.core.mutf8.MUTF8St		
method), 144		(andro-
rsubint() (in module and	dro- guard.core.bytecodes.dvm.ClassManage	r
guard.decompiler.dad.opcode_ins), 173	method), 85	
rsubintlit8() (in module and	<pre>dro- set_idx() (androguard.core.bytecode.Buff.</pre>	Handle
guard.decompiler.dad.opcode_ins), 173	method), 148	
	eachDefet_idx()(androguard.core.bytecodes.dvm.Da	lvikCode
method), 155	method), $88$	
RunDecompiler() (in module androguard.misc),		DCode
S	method), 86	
		(andro-
save() (androguard.core.bytecode.BuffHan	ndle guard.core.bytecodes.dvm.EncodedField	l
method), 148	method), 97	DC ada
save() (androguard.core.bytecodes.dvm.DalvikOde	exVMFormatinsn() (androguard.core.bytecodes.dvm.	DCoae
memou); ee	method), 86	(andro-
save() (androguard.core.bytecodes.dvm.DalvikVMI	10mai —	ethod),
method), 92	86	emou),
save() (androguard.session.Session method), 183 Save() (in module androguard.session), 181		(andro-
· · · · · · · · · · · · · · · · · · ·	dro- guard.core.bytecodes.dvm.EncodedMeth	,
guard.core.androconf), 146	method), 100	
Session (class in androguard.session), 181	set_item()(androguard.core.bytecodes.dvm.M	<b>l</b> apItem
set_buff() (androguard.core.bytecode.BuffHar	J. D. 100	•
method), 148		(andro-
	dro- guard.core.bytecodes.axml.PackageCont	text
guard.decompiler.dad.basic_blocks.BasicBl	J D 140	
method), 153	set_name()(androguard.core.bytecodes.dvm.C	:lassDefItem
	dro- method), 83	
guard.core.analysis.analysis.DVMBasicBloc method), 57	ock set_name()(androguard.core.bytecodes.dvm.E method), 97	ncodedField
numuij, 57	set_name() (androguard.core.bytecodes.dvm.E	ncodedMethod

method), 100		set_op	tions()	(in	module	andro-	
<del>-</del>	(andro-		guard.core.c	and roconf),	146		
guard.core.analysis.analysis.DVMBasic	Block	set_st	atic_fiel	lds()		(andro-	
method), 57			guard.core.l	bytecodes.dv	m.ClassDate	aItem	
set_off()(androguard.core.bytecodes.dvm.An	notation	Item	method), 81				
method), 77		set_to	() (androgue	ard.decomp	iler.dad.grapi	h.GenInvokeRet	Name
set_off()(androguard.core.bytecodes.dvm.An	notations	Directory	Itauthod), 15	7			
method), 80		set_ty	pe()(andro	guard.decoi	mpiler.dad.in	struction.IRFor	m
set_off()(androguard.core.bytecodes.dvm.An	notationS	SetItem	method), 16	52			
method), 78		set_va				(andro-	
set_off()(androguard.core.bytecodes.dvm.An	notation	SetRefList	guard.core.c	analysis.ana	lysis.StringA	nalysis	
method), 79			method), 61				
set_off()(androguard.core.bytecodes.dvm.Cl	assDataI	<i>tesn</i> et_vm				(andro-	
method), 81			-		vm.DalvikVM	lFormat	
set_off() (androguard.core.bytecodes.dvm.Cl	assHDefl	tem	method), 92				
method), 83		sget()	(in		nodule	andro-	
set_off() (androguard.core.bytecodes.dvm.Co	odeItem		-	-	pcode_ins), 1		
method), 85			olean()			andro-	
set_off()(androguard.core.bytecodes.dvm.Do	ılvikCode		-	-	pcode_ins), 1		
method), 88		sgetby		(in	module	andro-	
set_off()(androguard.core.bytecodes.dvm.De	ebugInfol		-	-	_		
method), 93	7 74	sgetch		(in	module	andro-	
set_off()(androguard.core.bytecodes.dvm.En	codedAri	•	~		•		
method), 94	1 10		ject()		module	andro-	
set_off()(androguard.core.bytecodes.dvm.En	codedCa		-	-	_		
method), 95	and alCa	sgetsh		(in	module	andro-	
set_off() (androguard.core.bytecodes.dvm.En	coaeaCa			-	_		
method), 96	ald A at.	sgetwi		•	module	andro-	
set_off() (androguard.core.bytecodes.dvm.Fig	гиАппон		-	-	pcode_ins), 1 modulo		
method), 102	ما اللاللال	shlint		•	module	andro-	
<pre>set_off() (androguard.core.bytecodes.dvm.Fid method), 102</pre>	гиппане		<i>guara.aecor</i> 2addr()	приетаца.о <sub>ј</sub> (in	pcode_ins), 1 module	andro-	
set_off() (androguard.core.bytecodes.dvm.He	ador <b>I</b> tom				pcode_ins), 1		
method), 104	шетиет		lit8()		module module	andro-	
set_off() (androguard.core.bytecodes.dvm.N	ManI ist	SHITTHE			pcode_ins), 1		
method), 123	парыя	shllon			module	andro-	
set_off() (androguard.core.bytecodes.dvm.Me	othodAnn						
method), 123					module		
set_off() (androguard.core.bytecodes.dvm.Me			=		pcode_ins), 1		
method), 124			-		(in modu		
set_off()(androguard.core.bytecodes.dvm.Pa	rameterA				*		
method), 127			ircuitBlo	_	lass in	andro-	
set_off()(androguard.core.bytecodes.dvm.Pr	otoHIdIte			,	asic_blocks),	. 154	
method), 127			-	•		DVMBasicBlock	
set_off()(androguard.core.bytecodes.dvm.Str	ingDatal		method), 57		•		
method), 130	Ü		(androguard	l.core.analys	sis.analysis.N	1ethodAnalysis	
set_off()(androguard.core.bytecodes.dvm.Str	ingIdIten		method), 60	•	•	•	
method), 130		show()	(androguare	d.core.bytece	odes.apk.API	K method),	
set_off() (androguard.core.bytecodes.dvm.	TryItem		76	•	-		
method), 131	•	show()	(androgua	ırd.core.byte	codes.axml.S	StringBlock	
set_off()(androguard.core.bytecodes.dvm.Ty	peHIdItei	n	method), 14			-	
method), 131					des.dvm.Ann	notationElement	
set_off() (androguard.core.bytecodes.dvm.T	TypeList		method), 77				
method), 133		show()	(androguard	l.core.byteco	des.dvm.Ann	otationItem	

	method), 77	method), 103
show()	$(and roguard. core. by tecodes. dvm. Annotation Off Item \verb how  ) \\ method), 78$	(androguard.core.bytecodes.dvm.FieldIdItemInvalid method), 103
show()	(androguard.core.bytecodes.dvm.AnnotationsDirectoryHemmethod), 80	(androguard.core.bytecodes.dvm.FillArrayData method), 104
show()	$(and roguard. core. by tecodes. dvm. Annotation Set Item {\tt how}~()~\\method), 78$	(androguard.core.bytecodes.dvm.HeaderItem method), 104
show()	$(and roguard. core. by tecodes. dvm. Annotation Set \textit{Refstraw} \ () \\ method), 79$	(androguard.core.bytecodes.dvm.Instruction method), 106
show()	(androguard.core.bytecodes.dvm.AnnotationSetRef&istw () method), 79	(androguard.core.bytecodes.dvm.MapItem method), 122
show()	(androguard.core.bytecodes.dvm.ClassDataItem show() method), 81	(androguard.core.bytecodes.dvm.MapList method), 123
show()	(androguard.core.bytecodes.dvm.ClassDefItem show() method), 83	(androguard.core.bytecodes.dvm.MethodAnnotation method), 123
show()	(and roguard. core. by tecodes. dvm. Class HDefItem show() method), 83	(androguard.core.bytecodes.dvm.MethodHIdItem method), 124
show()	(androguard.core.bytecodes.dvm.CodeItem show() method), 85	(androguard.core.bytecodes.dvm.MethodIdItem method), 125
show()	(androguard.core.bytecodes.dvm.DalvikCode show() method), 88	(androguard.core.bytecodes.dvm.MethodIdItemInvalid method), 125
show()	$(and roguard.core.by tecodes.dvm.Dalvik VMF or mat \verb show   () \\ method), 93$	(androguard.core.bytecodes.dvm.OdexHeaderItem method), 125
show()	(androguard.core.bytecodes.dvm.DBGBytecode show() method), 85	(androguard.core.bytecodes.dvm.PackedSwitch method), 126
show()	(androguard.core.bytecodes.dvm.DCode show() method), 86	(androguard.core.bytecodes.dvm.ParameterAnnotation method), 127
show()	(and roguard. core. by tecodes. dvm. Debug Info Item show () method), 93	(androguard.core.bytecodes.dvm.ProtoHIdItem method), 127
show()	$(and roguard. core. by tecodes. dvm. Debug Info I tem Erapty w~()~\\method),~93$	(androguard.core.bytecodes.dvm.ProtoIdItem method), 128
show()	$(and roguard.core.by tecodes.dvm. Encoded Annotati \textbf{\o} \textbf{h} \circ \textbf{w} \ () \\ method), 94$	(androguard.core.bytecodes.dvm.ProtoIdItemInvalid method), 128
show()	(androguard.core.bytecodes.dvm.EncodedArray show() method), 94	(androguard.core.bytecodes.dvm.SparseSwitch method), 129
	$(and roguard.core.by tecodes.dvm. Encoded Array Ite \textit{Bt} how () \\ method), 94$	method), 130
show()	$(and roguard.core.by tecodes.dvm. Encoded Catch \textit{Havidbear}\ () \\ method), 95$	(androguard.core.bytecodes.dvm.StringIdItem method), 130
show()	(androguard.core.bytecodes.dvm.EncodedCatchHandlenI(i)s method), 96	t (androguard.core.bytecodes.dvm.TypeHIdItem method), 131
show()	(androguard.core.bytecodes.dvm.EncodedField show() method), 97	(androguard.core.bytecodes.dvm.TypeIdItem method), 132
show()	$(and roguard.core.by tecodes.dvm. Encoded Method \verb  show ()  \\ method), 100$	(androguard.core.bytecodes.dvm.TypeItem method), 132
show()	$(and roguard. core. by tecodes. dvm. Encoded Type Adds \textit{Pain}~() \\ method),~101$	(androguard.core.bytecodes.dvm.TypeList method), 133
show()	(androguard.core.bytecodes.dvm.EncodedValue show() method), 101 show_b	(androguard.session.Session method), 183 uff() (andro-
show()	(androguard.core.bytecodes.dvm.FieldAnnotation method), 102	guard.core.analysis.analysis.ExceptionAnalysis method), 57
show()	(androguard.core.bytecodes.dvm.FieldHIdItem show_b method), 102	uff() (andro- guard.core.bytecodes.dvm.FillArrayData
show()	(androguard.core.bytecodes.dvm.FieldIdItem	method), 104

show_buff() (andro-	space() (androguard.decompiler.dad.writer.Writer
guard.core.bytecodes.dvm.Instruction method),	method), 175
106	SparseSwitch (class in andro-
show_buff() (andro-	guard.core.bytecodes.dvm), 128
guard.core.bytecodes.dvm.PackedSwitch	sparseswitch() (in module andro-
method), 126	guard.decompiler.dad.opcode_ins), 173
show_buff() (andro-	split() (androguard.core.mutf8.MUTF8String
guard.core.bytecodes.dvm.SparseSwitch	method), 145
method), 129	split_if_nodes() (in module andro-
show_Certificate() (in module andro-	guard.decompiler.dad.graph), 159
guard.core.bytecodes.apk), 76	split_variables() (in module andro-
show_info() (andro-	guard.decompiler.dad.dataflow), 156
guard.core.bytecodes.dvm.EncodedMethod	sput() (in module andro-
method), 100	guard.decompiler.dad.opcode_ins), 173
show_logging() (in module andro-	sputboolean() (in module andro-
guard.core.androconf), 147	guard.decompiler.dad.opcode_ins), 173
show_notes() (andro-	sputbyte() (in module andro-
guard.core.bytecodes.dvm.EncodedMethod	guard.decompiler.dad.opcode_ins), 173
method), 100	sputchar() (in module andro-
show_source() (andro-	guard.decompiler.dad.opcode_ins), 173
guard.decompiler.dad.decompile.DvClass	sputobject() (in module andro-
method), 156	guard.decompiler.dad.opcode_ins), 173
show_source() (andro-	sputshort() (in module andro-
guard.decompiler.dad.decompile.DvMachine	guard.decompiler.dad.opcode_ins), 174
method), 157	sputwide() (in module andro-
show_source() (andro-	guard.decompiler.dad.opcode_ins), 174
guard.decompiler.dad.decompile.DvMethod	startswith() (androguard.core.mutf8.MUTF8String
method), 157	method), 145
show_xrefs() (andro-	statement_block() (in module andro-
guard.core.analysis.analysis.MethodAnalysis	guard.decompiler.dad.dast), 152 StatementBlock (class in andro-
<pre>method), 60 shrint() (in module andro-</pre>	StatementBlock (class in andro- guard.decompiler.dad.basic_blocks), 154
shrint() (in module andro- guard.decompiler.dad.opcode_ins), 173	static_operand_instruction() (in module an-
shrint2addr() (in module andro-	droguard.core.bytecodes.dvm), 135
guard.decompiler.dad.opcode_ins), 173	StaticExpression (class in andro-
shrintlit8() (in module andro-	guard.decompiler.dad.instruction), 165
guard.decompiler.dad.opcode_ins), 173	
shrlong() (in module andro-	
guard.decompiler.dad.opcode_ins), 173	store_array_inst() (in module andro-
shrlong2addr() (in module andro-	guard.decompiler.dad.opcode_ins), 174
guard.decompiler.dad.opcode_ins), 173	str_ext() (androguard.decompiler.dad.writer.Writer
sign_apk() (in module androguard.misc), 180	method), 175
simplify() (in module andro-	string() (in module andro-
guard.decompiler.dad.graph), 159	guard.decompiler.dad.writer), 177
SIZE (androguard.core.bytecodes.axml.ARSCHeader at-	StringAnalysis (class in andro-
tribute), 135	guard.core.analysis.analysis), 61
size() (androguard.core.bytecode.BuffHandle	StringBlock (class in andro-
method), 148	guard.core.bytecodes.axml), 143
size() (androguard.core.bytecodes.axml.ARSCHeader	StringDataItem (class in andro-
property), 135	guard.core.bytecodes.dvm), 129
source() (androguard.core.bytecodes.dvm.ClassDefIten	
method), 83	guard.core.bytecodes.dvm), 130
source() (androguard.core.bytecodes.dvm.EncodedMeth	
method), 100	tribute), 168

subdoul	ble() guard.decon		module opcode_ins), 17		TypeItem (class in androguard.core.bytecodes.dvm), 132
	ble2addr(	(in	module opcode_ins), 17-	andro-	TypeList (class in androguard.core.bytecodes.dvm),
			module		typen() (in module androguard.decompiler.dad.dast),
Subito			opcode_ins), 17		153
suhflo:	at2addr()	-		andro-	
			opcode_ins), 17		U
	()	_	_		unary_postfix() (in module andro-
SUDIIIC		•	opcode_ins), 17		guard.decompiler.dad.dast), 153
suhint'			module		unary_prefix() (in module andro-
			opcode_ins), 17		guard.decompiler.dad.dast), 153
	g ( )	_	_		UnaryExpression (class in andro-
	_		opcode_ins), 17		guard.decompiler.dad.instruction), 166
	g2addr()			andro-	update_attribute_with() (andro-
			opcode_ins), 17		guard.decompiler.dad.basic_blocks.CondBlock
	-	_	ompiler.dad.grap		
	method), 15	-			update_attribute_with() (andro-
			module	andro-	guard.decompiler.dad.basic_blocks.LoopBlock
	guard.decon				method), 154
				andro-	update_attribute_with() (andro-
			control_flow), 1:		guard.decompiler.dad.basic_blocks.SwitchBlock
Switchl	Block	_	in		method), 154
	guard.decon	npiler.dad.	basic_blocks), 1	54	update_attribute_with() (andro-
Switchl	Expressio	on (ci	lass in	andro-	guard.decompiler.dad.node.Node method),
	guard.decon	npiler.dad.	instruction), 166	5	167
_					update_chain() (in module andro-
Τ					guard.decompiler.dad.dataflow), 156
tell()	(and	droguard.c	ore.bytecode.Bu	ffHandle	update_dom() (in module andro-
	method), 14	8			<pre>guard.decompiler.dad.control_flow), 155</pre>
text()	(androguare	d.core.byte	codes.axml.AXN	<i>ALParser</i>	ushrint() (in module andro-
	property), 1				guard.decompiler.dad.opcode_ins), 174
ThisPa		`	in		ushrint2addr() (in module andro-
			instruction), 166	5	guard.decompiler.dad.opcode_ins), 174
throw(	(i	in	module	andro-	ushrintlit8() (in module andro-
			opcode_ins), 17	4	guard.decompiler.dad.opcode_ins), 174
throw_:	stmt()	(in			ushrlong() (in module andro-
	guard.decon	npiler.dad.	dast), 153		guard.decompiler.dad.opcode_ins), 174
ThrowB.		(class	in		ushrlong2addr() (in module andro-
	-	-	basic_blocks), 1	54	guard.decompiler.dad.opcode_ins), 174
ThrowE	xpressior			andro-	V
	-	_	instruction), 166		•
		_	d.core.bytecode)	_	value() (androguard.decompiler.dad.instruction.Variable
try_str		(in	module	andro-	method), 166
	guard.decor	_		7	<pre>var_decl() (in module andro-</pre>
TryBlo		(class	in	andro-	guard.decompiler.dad.dast), 153
	-	_	basic_blocks), 1		Variable (class in andro-
_		_	core.bytecodes.d		guard.decompiler.dad.instruction), 166
	_		codes.axml.ARS	СHeader	version() (androguard.core.bytecodes.dvm.DalvikVMForma
	property), 1		i	an dua	property), 93
ТуреНІ		(class bytacodas a	in hvm) 131	andro-	visit() (androguard.decompiler.dad.basic_blocks.CatchBlock
TypeId	guard.core.l	oyiecoaes.c (class	ivm), 151 in	andro-	method), 153
	euard.core.l			anaro-	visit() (androguard.decompiler.dad.basic_blocks.CondBlock method) 153

```
visit () (androguard.decompiler.dad.basic_blocks.Conditionsit () (androguard.decompiler.dad.instruction.MoveExceptionExpressi
              method), 154
                                                                                                     method), 164
visit() (androguard.decompiler.dad.basic_blocks.LoopBtockit() (androguard.decompiler.dad.instruction.MoveExpression
              method), 154
                                                                                                     method), 164
visit() (androguard.decompiler.dad.basic_blocks.ReturnBlockt() (androguard.decompiler.dad.instruction.MoveResultExpression
              method), 154
                                                                                                     method), 164
visit() (androguard.decompiler.dad.basic blocks.StatementBlock) (androguard.decompiler.dad.instruction.NewArrayExpression
              method), 154
                                                                                                     method), 164
visit() (androguard.decompiler.dad.basic_blocks.SwitchBlockt() (androguard.decompiler.dad.instruction.NewInstance
              method), 154
                                                                                                     method), 165
\verb|visit(|)| (and roguard. decompiler. dad. basic\_blocks. Throw \textit{Block} t(|)| (and roguard. decompiler. dad. instruction. No pExpression) | (and roguard. decompiler. dad. basic\_blocks. Throw \textit{Block} t(|)) | (and roguard. decompiler. dad. basic\_blocks. Throw \textit{Block} t(|)) | (and roguard. decompiler. dad. basic\_blocks. Throw \textit{Block} t(|)) | (and roguard. decompiler. dad. basic\_blocks. Throw \textit{Block} t(|)) | (and roguard. decompiler. dad. basic\_blocks. Throw \textit{Block} t(|)) | (and roguard. decompiler. dad. basic\_blocks. Throw \textit{Block} t(|)) | (and roguard. decompiler. dad. basic\_blocks. Throw \textit{Block} t(|)) | (and roguard. decompiler. dad. basic\_blocks. Throw \textit{Block} t(|)) | (and roguard. decompiler. dad. basic\_blocks. Throw \textit{Block} t(|)) | (and roguard. decompiler. dad. basic\_blocks. Throw \textit{Block} t(|)) | (and roguard. decompiler. dad. basic\_blocks. Throw \textit{Block} t(|)) | (and roguard. decompiler. dad. basic\_blocks. Throw \textit{Block} t(|)) | (and roguard. decompiler. dad. basic\_blocks. Throw \textit{Block} t(|)) | (and roguard. decompiler. dad. basic\_blocks. Throw \textit{Block} t(|)) | (and roguard. decompiler. dad. basic\_blocks. Throw \textit{Block} t(|)) | (and roguard. decompiler. dad. basic\_blocks. Throw \textit{Block} t(|)) | (and roguard. decompiler. dad. basic\_blocks. Throw \textit{Block} t(|)) | (and roguard. decompiler. dad. basic\_blocks. Throw \textit{Block} t(|)) | (and roguard. decompiler. dad. basic\_blocks. data bas
                                                                                                     method), 165
              method), 154
visit() (androguard.decompiler.dad.basic_blocks.TryBlocksit() (androguard.decompiler.dad.instruction.Param
              method), 154
                                                                                                     method), 165
visit () (androguard.decompiler.dad.instruction.ArrayLengthExp(r)s(siandroguard.decompiler.dad.instruction.ReturnInstruction
              method), 159
                                                                                                     method), 165
visit () (androguard.decompiler.dad.instruction.ArrayLoadExpression
              method), 159
                                                                                                     method), 165
visit () (androguard.decompiler.dad.instruction.ArrayStorelsestructionadroguard.decompiler.dad.instruction.StaticInstruction
              method), 160
                                                                                                     method), 166
visit () (androguard.decompiler.dad.instruction.AssignExpression) (androguard.decompiler.dad.instruction.SwitchExpression
              method), 160
                                                                                                     method), 166
visit() (androguard.decompiler.dad.instruction.BaseClassisit() (androguard.decompiler.dad.instruction.ThisParam
              method), 160
                                                                                                     method), 166
visit() (androguard.decompiler.dad.instruction.BinaryCompExp(res(xiandroguard.decompiler.dad.instruction.ThrowExpression
              method), 160
                                                                                                     method), 166
visit () (androguard.decompiler.dad.instruction.BinaryExpression) (androguard.decompiler.dad.instruction.UnaryExpression
              method), 160
                                                                                                     method), 166
visit () (androguard.decompiler.dad.instruction.CastExpression () (androguard.decompiler.dad.instruction.Variable
              method), 161
                                                                                                     method), 166
visit() (androguard.decompiler.dad.instruction.CheckCastExpression.qth()
                                                                                                                                                               (andro-
              method), 161
                                                                                                     guard.decompiler.dad.writer.Writer
                                                                                                                                                             method),
visit () (androguard.decompiler.dad.instruction.ConditionalExpresh76n
              method), 161
                                                                                       visit_aload()
                                                                                                                                                               (andro-
visit () (androguard.decompiler.dad.instruction.ConditionalZExpressiond.decompiler.dad.writer.Writer
                                                                                                                                                             method),
              method), 161
visit()(androguard.decompiler.dad.instruction.Constantvisit arr data()
                                                                                                                                  (in
                                                                                                                                              module
                                                                                                                                                                andro-
              method), 161
                                                                                                     guard.decompiler.dad.dast), 153
visit() (androguard.decompiler.dad.instruction.FillArrayExpressionsign()
                                                                                                                                                               (andro-
              method), 162
                                                                                                     guard.decompiler.dad.writer.Writer
                                                                                                                                                             method),
visit () (androguard.decompiler.dad.instruction.FilledArrayExpressiton
              method), 162
                                                                                       visit astore()
                                                                                                                                                               (andro-
visit () (androguard.decompiler.dad.instruction.InstanceExpressionuard.decompiler.dad.writer.Writer
                                                                                                                                                             method),
              method), 163
                                                                                                     175
visit () (androguard.decompiler.dad.instruction.InstanceInistructionase_class()
                                                                                                                                                               (andro-
              method), 163
                                                                                                     guard.decompiler.dad.writer.Writer
                                                                                                                                                             method),
visit () (androguard.decompiler.dad.instruction.InvokeInstruction 175
                                                                                                                                                               (andro-
              method), 163
                                                                                       visit_binary_expression()
visit() (androguard.decompiler.dad.instruction.IRForm
                                                                                                     guard.decompiler.dad.writer.Writer
                                                                                                                                                             method),
              method), 162
visit () (androguard.decompiler.dad.instruction.MonitorEnteriExpression()
                                                                                                                                                               (andro-
              method), 164
                                                                                                     guard.decompiler.dad.writer.Writer
                                                                                                                                                             method),
visit() (androguard.decompiler.dad.instruction.MonitorExitExpresslδn
```

visit catch node()

(andro-

*method*), 164

guard.decompiler.dad.writer.Writer 175		guard.decompiler.dad.dast.JSONWri	(andro- ter
visit_check_cast()  guard.decompiler.dad.writer.Writer  175	(andro- method),	<pre>method), 151 visit_ins()     guard.decompiler.dad.writer.Writer</pre>	(andro- method),
visit_cond()  guard.decompiler.dad.basic_blocks.C  method), 153	(andro- CondBlock	176 visit_ins() (in module guard.decompiler.dad.dast), 153	andro-
visit_cond() guard.decompiler.dad.basic_blocks.L method), 154		visit_invoke()  guard.decompiler.dad.writer.Writer  176	(andro- method),
visit_cond() guard.decompiler.dad.basic_blocks.S method), 154		visit_loop_node()	(andro- ter
visit_cond_expression()  guard.decompiler.dad.writer.Writer  175		visit_loop_node()  guard.decompiler.dad.writer.Writer  176	(andro- method),
visit_cond_node()  guard.decompiler.dad.dast.JSONWri  method), 151		visit_monitor_enter()  guard.decompiler.dad.writer.Writer  176	(andro- method),
visit_cond_node()  guard.decompiler.dad.writer.Writer  175	(andro- method),	visit_monitor_exit()  guard.decompiler.dad.writer.Writer  176	(andro- method),
visit_condz_expression()  guard.decompiler.dad.writer.Writer  175		visit_move()  guard.decompiler.dad.writer.Writer  176	(andro- method),
visit_constant()  guard.decompiler.dad.writer.Writer  175	(andro- method),	visit_move_exception()  guard.decompiler.dad.writer.Writer  176	(andro- method),
visit_decl() guard.decompiler.dad.instruction.Var method), 166		visit_move_result()  guard.decompiler.dad.writer.Writer  176	(andro- method),
visit_decl()  guard.decompiler.dad.writer.Writer  175	(andro- method),	visit_new()  guard.decompiler.dad.writer.Writer  176	(andro- method),
<pre>visit_decl() (in module</pre>	andro- (andro-	visit_new_array()  guard.decompiler.dad.writer.Writer  176	(andro- method),
guard.decompiler.dad.basic_blocks.C method), 153	CatchBlock	visit_node() guard.decompiler.dad.dast.JSONWri	(andro- ter
<pre>visit_expr() (in module</pre>	andro- (andro-	<pre>method), 152 visit_node()     guard.decompiler.dad.writer.Writer</pre>	(andro- method),
guard.decompiler.dad.writer.Writer 175	method),	176 visit_nop()	(andro-
visit_filled_new_array()  guard.decompiler.dad.writer.Writer  175	(andro- method),	<pre>guard.decompiler.dad.writer.Writer 176 visit_param()</pre>	method), (andro-
visit_get_instance()     guard.decompiler.dad.writer.Writer	(andro- method),	guard.decompiler.dad.writer.Writer 176	method),
176 visit_get_static() guard.decompiler.dad.writer.Writer	(andro- method),	visit_put_instance()  guard.decompiler.dad.writer.Writer  176	(andro- method),
176	,,	visit_put_static()	(andro-

guard.decompiler.dad.writer.Writer 176	method),	guard.decompiler.dad.writer.Writer method), 176
visit_return()  guard.decompiler.dad.writer.Writer  176	(andro- method),	visit_variable() (andro- guard.decompiler.dad.writer.Writer method), 176
visit_return_node()	(andro-	vm2json() (in module androguard.core.bytecode), 151
guard.decompiler.dad.dast.JSONWri method), 152	`	W
visit_return_node()  guard.decompiler.dad.writer.Writer  176  visit_return_void()	(andro- method), (andro-	while_block_struct() (in module andro- guard.decompiler.dad.control_flow), 155 write() (androguard.decompiler.dad.writer.Writer method), 176
guard.decompiler.dad.writer.Writer 176	method),	write_ext() (andro-guard.decompiler.dad.writer.Writer method),
visit_short_circuit_condition()	(andro- method),	176 write_ind() (androguard.decompiler.dad.writer.Writer method),
visit_statement_node()	(andro-	176
guard.decompiler.dad.dast.JSONWri	ter	write_ind_visit_end() (andro-
<pre>method), 152 visit_statement_node()</pre>	(andro-	guard.decompiler.dad.writer.Writer method), 176
guard.decompiler.dad.writer.Writer 176	method),	<pre>write_ind_visit_end_ext() (andro- guard.decompiler.dad.writer.Writer method),</pre>
visit_super()  guard.decompiler.dad.writer.Writer  176	(andro- method),	176 write_inplace_if_possible() (androguard.decompiler.dad.writer.Writer method),
visit_switch()  guard.decompiler.dad.writer.Writer	(andro- method),	177 write_inplace_if_possible() (in module an-
176 visit_switch_node()	(andro-	droguard.decompiler.dad.dast), 153 write_method() (andro-
guard.decompiler.dad.dast.JSONWri method), 152	•	guard.decompiler.dad.writer.Writer method), 177
visit_switch_node()  guard.decompiler.dad.writer.Writer  176	(andro- method),	Writer (class in androguard.decompiler.dad.writer), 175
visit_this()  guard.decompiler.dad.writer.Writer  176	(andro- method),	writesleb128() (in module andro- guard.core.bytecodes.dvm), 135 writeuleb128() (in module andro-
visit throw()	(andro-	guard.core.bytecodes.dvm), 135
guard.decompiler.dad.writer.Writer	method),	XOD (androquard decompiler dad opcode in s On at
<pre>visit_throw_node()</pre>	(andro- iter	XOR (androguard.decompiler.dad.opcode_ins.Op at- tribute), 168 xorint() (in module andro-
visit_throw_node()  guard.decompiler.dad.writer.Writer  176	(andro- method),	guard.decompiler.dad.opcode_ins), 174 xorint2addr() (in module andro- guard.decompiler.dad.opcode_ins), 174
visit_try_node()  guard.decompiler.dad.dast.JSONWri  method), 152	(andro- iter	<pre>xorintlit16() (in module andro-</pre>
visit_try_node() guard.decompiler.dad.writer.Writer	(andro- method),	<pre>guard.decompiler.dad.opcode_ins), 174 xorlong() (in module andro-</pre>
guara.aecompuer.aaa.wruer.wruer 176 visit_unary_expression()	(andro-	guard.decompiler.dad.opcode_ins), 174 xorlong2addr() (in module andro-
	(2	guard.decompiler.dad.opcode_ins), 174



 ${\tt Yellow} \ \ \textit{(and roguard.core.and roconf. Color \ attribute)}, \\ 145$