ABUSING WEB APIS THROUGH SCRIPTED ANDROID APPLICATIONS

DANIEL PECK

Principal Research Scientist at Barracuda Labs (barracudalabs.com)

Rapid prototyping
Malicious messaging
Larger studies of the current state of web security and infosec industry

@Ramblinpeck

@BarracudaLabs

Past Lives

Offensive, SCADA, Snort Jockey, Reverser

SESSION OUTLINE

Target Selection

Foundations

Exploration

Control

Hot social app that I want to ~~spam~~ be a part of

Great web interface, great api once we have a few hundred thousand accounts, but has some restrictions

APPROACH

People are too worried about “friction” to put many safeguard/throttling into mobile apps

Create our own client that mimics mobile app for api purposes.

ASSUMPTIONS AND HOPES

Twacebook has a well documented API thats protected using Oauth

We'll probably need to extract some keys

They probably use their published API in their apps

BUILD ON EXISTING TOOLS

INTERCEPTING APP COMMUNICATIONS

Need to MitM to be able to view tx/rx

Proxydroid

https://github.com/madeye/proxydroid

Run all/some of android traffic through our proxy

SSL

The developers aren't idiots

Create and add a cert to your testing device

```
$ adb pull /system/etc/security/cacerts.bks
$ keytool …
$ adb push cacerts.bks /system/etc/security
```

Gotchas

- Make sure you have the right version of bouncycastle
- otherwise things break in not-fun ways
- Different/easier procedures on Android 4.0+ devices

BURP PROXY

Invisible proxying, generates cert on demand

Look at dns requests/guess hostname (CN) for generated cert

since 1.4.12

INTERCEPTED TRAFFIC

```
POST /create_account HTTP/1.1
Content-Type: application/x-www-form-urlencoded
Content-Length: 296
Accept-Encoding: gzip,deflate
User-Agent: TwacebookAndroidApp(build 6294, v1.8.64)
Host: mobileapi.twacebook.com
Connection: Keep-Alive
Cache-Control: no-cache

auth_consumer_key=40iqOgCcXqfwwqoa02D7nQ
oauth_nonce=0437A32D733151CABA3A06A12243CD0A
oauth_signature_method=HMAC-SHA1
oauth_timestamp=1340141019
oauth_version=1.0
x_auth_mode=client_auth
```

```
x_auth_password=f00bar%24
x_auth_username=jimbo
oauth_signature=v%2FVnCJrssg9D07Zdy%2F8dPSapv8s%3D
```

OAUTH

Consumers requests a consumer key and consumer secret from provider

End users allow provider to grant a token and token secret to consumer to make requests on their behalf

Signs requests (HMAC-SHA1 usually) with consumer secret & token secret

MORE OAUTH

Users don't have to give their password to third party apps

Providers get to restrict apps accessing their api to only (honest) approved ones, essentially DRM

Designed and works well for server ← → server

Used extensively for mobile/desktop apps

ONTO KEY RETRIEVAL

DISASSEMBLY AND DECOMPILATION

Apktool **http://code.google.com/p/android-apktool/**

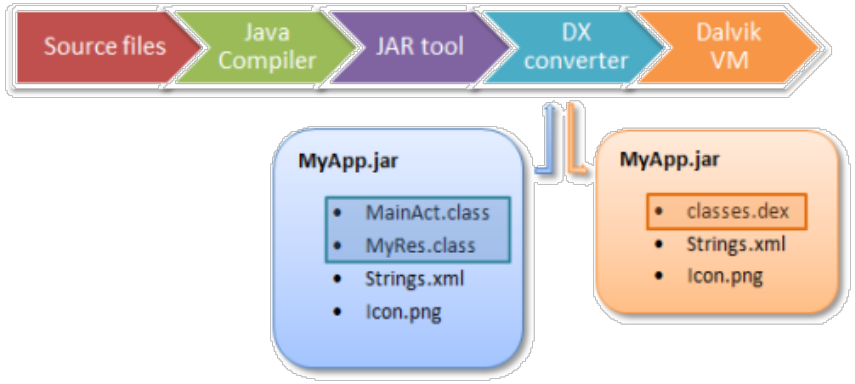
- Decodes apks
- Nice wrapper for smali/baksmali
- In theory should allow for some nice debugging..

JD-GUI **http://java.decompiler.free.fr/?q=jdgui**

- dex2jar first
- not compilable source, sometimes misleading, good for general idea

ABOUT ANDROID

Runs within a Dalvik application virtual machine



DALVIK

- Register based machine
- Optimized for low memory environments
- Runs dex files
  - Deduped
  - Dalvik instruction set instead of standard JVM
- Smali bytecode

SMALI

```
class public final Lcd;
super Ljava/lang/Object;
# static fields
.field public static final a:Lcd;

.method constructor
init
()V
.locals 2
const/4 v1, 0x0
const/4 v0, 0x0
invoke-direct {p0, v1, v0, v1}, Lcd;-
init
(Laa;ILjava/lang/String;)V
return-void
```

```
.end method
```

DECIPHERING SMALI

- Register based machine
  - Parameters are stored in p0...pX
  - Local registers v0...vY where
  - Last X local registers are identical to paramer registers
- Registers store 32-bit values
  - 64-bit values (J, long, and D, double primitives) are stored in 2 registers

PRIMITIVES

V: void - can only be used for return types

Z: boolean

B: byte

S: short

C: char

I: int

J: long (64 bits)

F: float

D: double (64 bits)

L: objects. You'll see in the form of “Lpackage/name

/ObjectName”

FUNCTION DECLARATIONS

```
method private static a #name and type
(
Lorg/apache/http/client/methods/HttpRequestBase; #p0
Laa; #p1
J #p2 + #p3
Ljava/lang/String; #p4
Ljava/lang/String; #p5
)Ljava/lang/String; #return type
```

VARIABLE ASSIGNMENT

```
const v2, 42
const-string v4, "this is a static string"
```

READABLE OPCODES

move-result vx

return-object vx

CALLING METHODS

invoke-direct parameters , methodtocall

```
invoke-direct {v4}, Ljava/lang/StringBuilder;->()V
```

invoke-static parameters , methodtocall

```
invoke-static {p1, p2}, Ljava/lang/Math;->min(II)I
```

http://pallergabor.uw.hu/androidblog/dalvik_opcodes.html

```
invoke-virtual {v0, v1}, Ljava/lang
/String;->getBytes(Ljava/lang/String;)[B
    move-result-object v0
    new-instance v1, Ljavax/crypto/spec/SecretKeySpec;
    const-string v2, "HmacSHA1"
    invoke-direct {v1, v0, v2}, Ljavax/crypto
/spec/SecretKeySpec;-><init>([BLjava/lang/String;)V
    invoke-static {v0}, Ljavax/crypto
/Mac;->getInstance(Ljava/lang/String;)Ljavax/crypto
/Mac;
...
    invoke-virtual {v0, v1}, Ljavax/crypto
/Mac;->init(Ljava/security/Key;)V
    const-string v1, "UTF8"
    invoke-virtual {p0, v1}, Ljava/lang
/String;->getBytes(Ljava/lang/String;)[B
    move-result-object v1
    invoke-virtual {v0, v1}, Ljavax/crypto
```

```
/Mac;->doFinal([B)[B
    move-result-object v0
```

AND FROM JD-GUI

```
    ...
    while (true)
    {
      try
      {
        str1 = "";
        SecretKeySpec localSecretKeySpec = new
SecretKeySpec((ch.a(paramString2) + "&" +
ch.a(str1)).getBytes("UTF8"), "HmacSHA1");
        Mac localMac = Mac.getInstance("HmacSHA1");
        localMac.init(localSecretKeySpec);
        String str3 = ch.a(new
String(cc.a(localMac.doFinal(paramString1.getBytes("UTF8"
"UTF8"));
        str2 = str3;
```

```
        return str2;
    }
...
```

LOOK SIMILAR?

2 Answers active oldest **votes**

1

```
public String computeHmac(String baseString, String key)
    throws NoSuchAlgorithmException, InvalidKeyException, IllegalStateException, U
{
    Mac mac = Mac.getInstance("HmacSHA1");
    SecretKeySpec secret = new SecretKeySpec(key.getBytes(), mac.getAlgorithm());
    mac.init(secret);
    byte[] digest = mac.doFinal(baseString.getBytes());
    return Base64.encode(digest);
}
```

share | improve this answer

answered **Aug 14 '10 at 22:42**
Jaroslav Záruba
723 ●4 ●15

feedback

AGAIN, DUMB THING FIRST

Printf debugging

```
const-string v2, "SECRETKEY , v0"
invoke-static {v2, v0}, Landroid/util/Log;->d(Ljava
/lang/String;Ljava/lang/String;)I
invoke-virtual {v0, v1}, Ljava/lang
/String;->getBytes(Ljava/lang/String;)[B
move-result-object v0
new-instance v1, Ljavax/crypto/spec/SecretKeySpec;
const-string v2, "HmacSHA1"
invoke-direct {v1, v0, v2}, Ljavax/crypto
/spec/SecretKeySpec;-
init
  ([BLjava/lang/String;)V
```

Rebuild the apk and run it

```
$ apktool b twacebook.apk twacebook_new.apk
```

EXAMING THE LOGS

```
$ adb shell
$ adb logcat
…
"SECRETKEY , v0 -
I7PW5lgEkgMrqPOdxIj1o6llAbFdXHhVjFnvUsg1g"
```

SO WE CAN REVERSE AROUND API CALLS EASILY, BUT CAN WE REVERSE CUSTOM CODE?

```
.method public final a([BIILjava/io/OutputStream;)I
    .locals 9

    const/4 v8, 0x0

    rem-int/lit8 v0, p3, 0x3

    sub-int v1, p3, v0

    move v2, v8

    :goto_0
    add-int/lit8 v3, v1, 0x0

    if-ge v2, v3, :cond_0

    aget-byte v3, p1, v2
```

```
    and-int/lit16 v3, v3, 0xff

    add-int/lit8 v4, v2, 0x1

    aget-byte v4, p1, v4
and-int/lit16 v4, v4, 0xff

    add-int/lit8 v5, v2, 0x2

    aget-byte v5, p1, v5

    and-int/lit16 v5, v5, 0xff

    iget-object v6, p0, Ll;->a:[B

    ushr-int/lit8 v7, v3, 0x2

    and-int/lit8 v7, v7, 0x3f

    aget-byte v6, v6, v7
```

```
    invoke-virtual {p4, v6}, Ljava/io
/OutputStream;->write(I)V

    iget-object v6, p0, Ll;->a:[B

    shl-int/lit8 v3, v3, 0x4

    ushr-int/lit8 v7, v4, 0x4

    or-int/2addr v3, v7

    and-int/lit8 v3, v3, 0x3f

    aget-byte v3, v6, v3

    invoke-virtual {p4, v3}, Ljava/io
/OutputStream;->write(I)V

    iget-object v3, p0, Ll;->a:[B
```

```
    shl-int/lit8 v4, v4, 0x2

    ushr-int/lit8 v6, v5, 0x6

    or-int/2addr v4, v6

    and-int/lit8 v4, v4, 0x3f

    aget-byte v3, v3, v4

    invoke-virtual {p4, v3}, Ljava/io
/OutputStream;->write(I)V

    iget-object v3, p0, Ll;->a:[B

    and-int/lit8 v4, v5, 0x3f

    aget-byte v3, v3, v4
```

```
    invoke-virtual {p4, v3}, Ljava/io
/OutputStream;->write(I)V

    add-int/lit8 v2, v2, 0x3

    goto :goto_0

    :cond_0
    packed-switch v0, :pswitch_data_0

    :goto_1
    :pswitch_0
    div-int/lit8 v1, v1, 0x3

    mul-int/lit8 v1, v1, 0x4

    if-nez v0, :cond_1

    move v0, v8
```

```
:goto_2
add-int/2addr v0, v1

return v0

:pswitch_1
add-int/lit8 v2, v1, 0x0

aget-byte v2, p1, v2

and-int/lit16 v2, v2, 0xff

ushr-int/lit8 v3, v2, 0x2

and-int/lit8 v3, v3, 0x3f

shl-int/lit8 v2, v2, 0x4

and-int/lit8 v2, v2, 0x3f
```

```
    iget-object v4, p0, Ll;->a:[B

    aget-byte v3, v4, v3

    invoke-virtual {p4, v3}, Ljava/io
/OutputStream;->write(I)V

    iget-object v3, p0, Ll;->a:[B

    aget-byte v2, v3, v2

    invoke-virtual {p4, v2}, Ljava/io
/OutputStream;->write(I)V

    iget-byte v2, p0, Ll;->b:B

    invoke-virtual {p4, v2}, Ljava/io
/OutputStream;->write(I)V

    iget-byte v2, p0, Ll;->b:B
```

```
    invoke-virtual {p4, v2}, Ljava/io
/OutputStream;->write(I)V

    goto :goto_1

    :pswitch_2
    add-int/lit8 v2, v1, 0x0

    aget-byte v2, p1, v2

    and-int/lit16 v2, v2, 0xff

    add-int/lit8 v3, v1, 0x0

    add-int/lit8 v3, v3, 0x1

    aget-byte v3, p1, v3

    and-int/lit16 v3, v3, 0xff
```

```
ushr-int/lit8 v4, v2, 0x2

and-int/lit8 v4, v4, 0x3f

shl-int/lit8 v2, v2, 0x4

ushr-int/lit8 v5, v3, 0x4

or-int/2addr v2, v5

and-int/lit8 v2, v2, 0x3f

shl-int/lit8 v3, v3, 0x2

and-int/lit8 v3, v3, 0x3f

iget-object v5, p0, Ll;->a:[B

aget-byte v4, v5, v4
```

```
    invoke-virtual {p4, v4}, Ljava/io
/OutputStream;->write(I)V

    iget-object v4, p0, Ll;->a:[B

    aget-byte v2, v4, v2

    invoke-virtual {p4, v2}, Ljava/io
/OutputStream;->write(I)V

    iget-object v2, p0, Ll;->a:[B

    aget-byte v2, v2, v3

    invoke-virtual {p4, v2}, Ljava/io
/OutputStream;->write(I)V

    iget-byte v2, p0, Ll;->b:B
```

```
    invoke-virtual {p4, v2}, Ljava/io
/OutputStream;->write(I)V

    goto :goto_1

    :cond_1
    const/4 v0, 0x4

    goto :goto_2

    :pswitch_data_0
    .packed-switch 0x0
        :pswitch_0
        :pswitch_1
        :pswitch_2
    .end packed-switch
.end method
```

```
  public final int a(byte[] paramArrayOfByte, int
paramInt1, int paramInt2, OutputStream
paramOutputStream)
  {
    int i = paramInt2 % 3;
    int j = paramInt2 - i;
    for (int k = 0; k < j + 0; k += 3)
    {
      int i9 = 0xFF & paramArrayOfByte[k];
      int i10 = 0xFF & paramArrayOfByte[(k + 1)];
      int i11 = 0xFF & paramArrayOfByte[(k + 2)];
      paramOutputStream.write(this.a[(0x3F & i9 >>>
2)]);
      paramOutputStream.write(this.a[(0x3F & (i9 << 4
| i10 >>> 4))]);
      paramOutputStream.write(this.a[(0x3F & (i10 << 2
| i11 >>> 6))]);
      paramOutputStream.write(this.a[(i11 & 0x3F)]);
```

```
}
int i4;
switch (i)
{
case 0:
default:
  i4 = 4 * (j / 3);
  if (i != 0)
    break;
case 1:
case 2:
}
for (int i5 = 0; ; i5 = 4)
{
  return i5 + i4;
  int i6 = 0xFF & paramArrayOfByte[(j + 0)];
  int i7 = 0x3F & i6 >>> 2;
  int i8 = 0x3F & i6 << 4;
  paramOutputStream.write(this.a[i7]);
  paramOutputStream.write(this.a[i8]);
```

```
        paramOutputStream.write(this.b);
        paramOutputStream.write(this.b);
        break;
        int m = 0xFF & paramArrayOfByte[(j + 0)];
        int n = 0xFF & paramArrayOfByte[(1 + (j + 0))];
        int i1 = 0x3F & m >>> 2;
        int i2 = 0x3F & (m << 4 | n >>> 4);
        int i3 = 0x3F & n << 2;
        paramOutputStream.write(this.a[i1]);
        paramOutputStream.write(this.a[i2]);
        paramOutputStream.write(this.a[i3]);
        paramOutputStream.write(this.b);
        break;
      }
    }
```

ENTER JVM BASED LANGUAGES

- JRuby
- Jython
- Clojure
- Scala

Built on the high performance multiplatform JVM

Huge number of Java libraries available

JAVA LIBRARIES WITH JRUBY

And thankfully, dex are just another kind of jar

```
$ unzip twacebook.apk
$ d2j-dex2jar.sh classes.dex -o twacebook.jar
```

```
require 'java'
require './jars/twacebook.jar'
require './jars/android.jar'

java_import 'cc' do |clasname|
  "Obfuscater"
end

obs_arr = Obfuscater.a(byte_arr)
signature = String.from_java_bytes(obs_arr)
```

ITTERATING UP

```
require 'java'
require './jars/twacebook.jar'
require './jars/android.jar'

java_import 'ab' do |clasname|
    "User"
end

java_import 'cc' do |clasname|
    "ApiFactory"
end

social_bot = ApiFactory.register_new_user(<name>,
<email>)
```

```
social_bot.post_update("Posting from a JRUBY")
```

```
jarfile = ARGV[0]
require 'java'
require jarfile
require './jars/android.jar'
require 'pry'
Pry.config.pager = false

jf = java.util.jar.JarFile.new(jarfile)

jf.entries.each do |entry|
        name = entry.getName
        name.gsub!(/.class$/, '')

        #ruby hates lowercase classnames
        begin
        java_import name do |classname|
                @cap_name = name.capitalize
        end
```

```
        rescue NameError => e
                puts "got an error, wtf? #{e}"
        end

        puts "#{name} Instance Methods"
        class_const = Kernel.const_get(@cap_name)
        methods =
Pry::Method.all_from_class(class_const).select {
|method|
            method.visibility == :public &&
method.owner == class_const && !["==", "__jsend!",
"equals", "__jcreate!"].include?(method.name)
        }

        puts methods.map(&:signature).join("\n")
end
```

DYNAMIC, AND EVEN FUN, STYLE OF REVERSING

Acquire Apk

Find interesting function

Load class into JVM session

Call functions in arbitrary ways

Itterate

KNOWN ISSUES

Native code methods (eg: extensions written in C and compiled for system) don't import and you can't load classes using them.

Going to be some incompatibilities between full JVM and Dalvik, but you'll seldom run into that.

People might confuse you for someone who likes Java

TOWARDS AN ADVERTORIAL WORLD

SEEDS

Skull Security torrent dump from Facebook (2010)

- 100 Million unique names
- Male/Female
- Unique ID number assigned by Facebook

US Census Data

- Associate last name with likelyhood of ethnicity
- Ethnicity can map (with weights) to their region
- Region randomly maps to area code/city/etc for profile

SEEDS (CONT)

Facebook

Gather public profile images from UIDs

Interests

Twitter suggests categories on account creation. Randomly select a few so your bot knows what they're interested in, food, movies, books, tv, etc.
Twitter public stream, 1% of traffic, more than your bots could ever need
Chattiness level, people post at different rates, your bots

should too. Some once a day, some a couple of times an hour, some once a week.

Time zone rand(-3..3) hours before/after dawn and midnight in their location. They'll only tweet then.

BUILDING

Stacey Jackson

Evergreen, Colorado

sassystacey303

Likes book, movies

Moderate poster

Friends easily, about 15% of the users that Twacebook suggests

MONITIZATION: EARNING REPORT OF SELLING FAKE FOLLOWERS BUSINESS

1k Followers sell for ~ $11 USD

Average person who purchases fake followers gets around 50k

Around 50 people selling followers through eBay at any given time

THANK YOU APPSECUSA

Questions/Thoughts?

dpeck@barracuda.com
peck@pragmatic.io
@barracudalabs
@ramblinpeck

Huge Thank You to Everyone Who Has Contributed to Tools Mentioned