

Deconstructing Kony Android Applications

Kony Apps Past, Present, and Future

Agenda

JUNE 16 - 18, 2015
EXCEL LONDON | LONDON, UK
WWW.BLACKHAT.COM

- Introduction
- Examining and Exploring Kony Apps
- Static vs. Dynamic Analysis
- Conclusions and Takeaways

About Me...

JUNE 16 - 18, 2015
EXCEL LONDON | LONDON, UK
WWW.BLACKHAT.COM

- Security Consultant for NCC Group North America
- I do pentesting, pentest all the things!!
- Free time, what's that? I surf, take photos, play guitar, design and build random stuff, other stuff you don't care about
- I have a serious gadget problem... serious(feel free to send me free stuff!!)

About You...

JUNE 16 - 18, 2015
EXCEL LONDON | LONDON, UK
WWW.BLACKHAT.COM

- You are an App Dev and/or Security person
- You are interested in how enterprise application deployments work
- You had nothing better to do before lunch
- You work at Kony Studios :-P

About this talk...

JUNE 16 - 18, 2015
EXCEL LONDON | LONDON, UK
WWW.BLACKHAT.COM

- Originally, a memory dumping technique was used in conjunction with a Jolla phone.
- This worked pretty well to recover source code.
- Was it efficient? No, not even close. Did it work. Barely, but yes
- Is that what I'm going to show you today? Definitely not, we have new hotness to work on and worry about

More about this talk...

JUNE 16 - 18, 2015
EXCEL LONDON | LONDON, UK
WWW.BLACKHAT.COM

- The previous method mentioned was discussed at Ekoparty in October 2014. I hadn't looked at a Kony Application since. Upon acceptance to Black Hat, I began updating my research as the example applications I was using were a year or two old.

More about this talk...

JUNE 16 - 18, 2015
EXCEL LONDON | LONDON, UK
WWW.BLACKHAT.COM

- And the only reason I had even bothered to look at Kony was because of Jason Ross's blog post from way back. I.E. – the part 2 that never came
- Turns out, everything had changed. The Apps were no longer the Kony I knew(panic sets in).

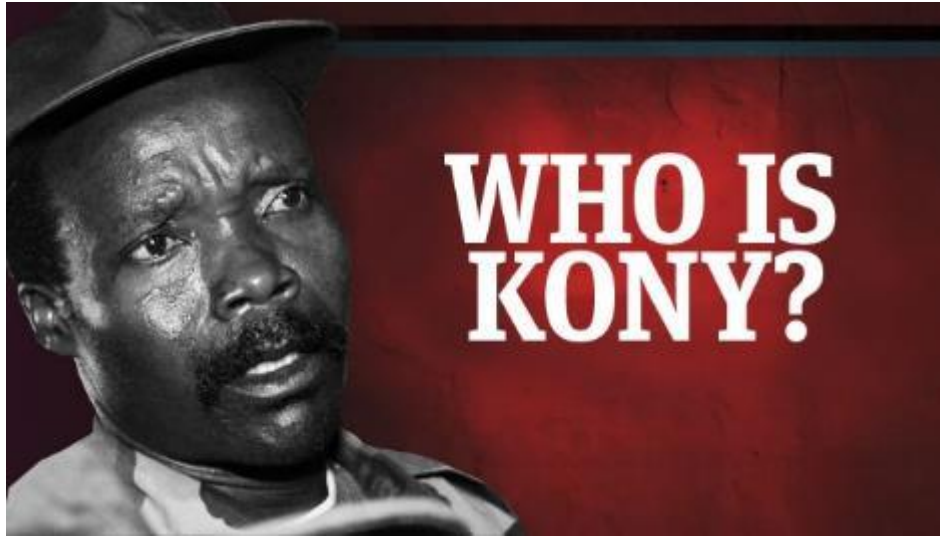
- After getting over the initial wave (and a couple subsequent waves of panic) I contemplated declining the speaker slot.
- The outline I proposed was no longer valid, and I wasn't even sure if I could find the time between full time consulting to redo the research given the new framework.

- Clearly, I ended up accepting the speaker slot. However, the contents of the talk have radically changed.
- The following talk contains the trials and tribulations of my efforts...

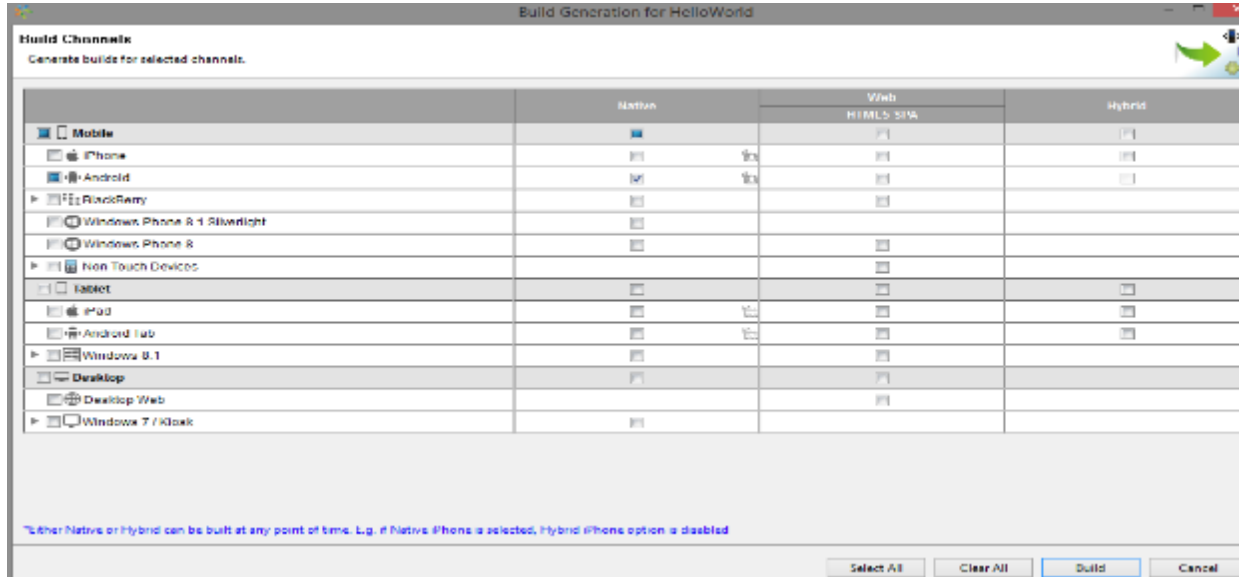
WTF is Kony?

JUNE 16 - 18, 2015
EXCEL LONDON | LONDON, UK
WWW.BLACKHAT.COM

- Not This Guy



- Write once deploy many Application IDE



WTF is Kony?(cont. 3)

JUNE 16 - 18, 2015
EXCEL LONDON | LONDON, UK
WWW.BLACKHAT.COM

- Write you application in HTML5 and/or JavaScript
- Because native code development is so 2012
- Also because “JavaScript is the language of the future” – OH someone, someplace, sometime

But where do I find Kony?

- Large companies(Kony ain't cheap)
 - Capital One
 - SunTrust Bank
 - Southwest Airlines
 - Scottrade
 - HSBC
- Take to Google ,Google Play Store, or 3rd Party App Stores
- Build Your Own Apps
 - They recently started offering a trial preview of their Studio IDE

Why you should care?

JUNE 16 - 18, 2015
EXCEL LONDON | LONDON, UK
WWW.BLACKHAT.COM

- Developers(at least good/decent ones) are expensive
 - Hiring a Dev, or a Dev Team for each platform can be prohibitively expensive
 - Additional overhead from managing multiple teams
 - More and more companies are starting to use multi-deployment framework models(Worklight/Cordova, Kony, Unity, Adobe AIR) – see Black Hat Asia talk by Grassi and Guerrero

Why you should care?

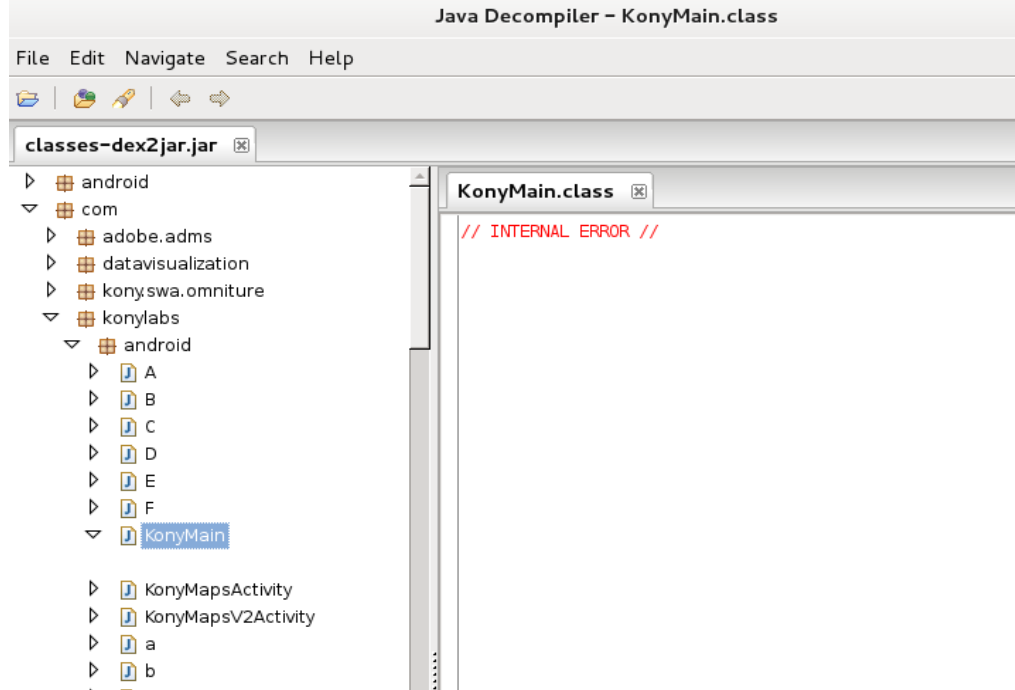
JUNE 16 - 18, 2015
EXCEL LONDON | LONDON, UK
WWW.BLACKHAT.COM

- While the individual steps for analyzing each framework will differ, the overall thought process will be similar.

- Analysis of Kony Apps using the "Standard Approach"
 - Extract APK file from device
 - Use Apktool to unzip
 - Dex2jar or baksmali the classes.dex file
 - Get absolutely nowhere... literally nowhere. WTF!?!?

Examining Kony Apps

JUNE 16 - 18, 2015
EXCEL LONDON | LONDON, UK
WWW.BLACKHAT.COM



- Analysis of Kony Apps using the "Standard Approach"
 - All paths in the App more or less call KonyMain
 - However this class is perpetually blank... What is going on here?
- Well, most of the application is a wrapper, this wrapper does the following:

- Older Kony (version < 6.0)
- APK loads
- Establish Dalvik Hooks
- Search for Lua Bytecode VM
- Run Bytecode VM

- The Bytecode VM **is** the App(There are a few serialized data “.kds” files lying around that are parsed, but the core code is the Bytecode)
- How do I know if I have “old Kony”? Look for “konyappluabytecode.o.mp3”

- New Kony (version > 6.0)
 - APK loads
 - Establish Dalvik Hooks
 - Search for libkonyJSVM.so library file
 - Runs the library file
 - Library file searches for startup.js and common-jslibs.kfm

Examining Kony

JUNE 16 - 18, 2015
EXCEL LONDON | LONDON, UK
WWW.BLACKHAT.COM

- Library unpacks the files in application memory space
- These files contain the application source code and necessary JS libraries
- How do I know if I have “new Kony”? Look for “libkonyJSVM.so, startup.js, and common-jslibs.kfm” in your APK file

“Standard Approach” Failures

JUNE 15 - 18, 2015
EXCEL LONDON | LONDON, UK
WWW.BLACKHAT.COM

- Seems legit... Y U no Work!?
 - As previously state, the source code is contained in either bytecode format, or being run in the Android shared object library.
 - Can't simply “jd-gui” your way to victory
 - This makes static and dynamic analysis a pain

“Standard Approach” Failures

- Using static decompilers or other OSS tools that just didn't quite work(errors, crashes, stack traces 4 dayzzzz)
- Previous attempts to bypass this involved, hooking calls into the “Main” class, monitoring the inputs and monitoring the outputs

“Standard Approach” Failures

JUNE 15 - 18, 2015
EXCEL LONDON | LONDON, UK
WWW.BLACKHAT.COM

- Massive device memory dumping, fixing up, and reassembling images to get source code in memory.
- Previously OSS decompilers namely unluac.jar didn't quite work right, it has been updated and now works splendidly.
 - Still do love playing with my Jolla Phone though
- Other tomfoolery and shenanigans that just didn't work well

Expanding the “Standard Approach”

JUNE 15 - 16 2015
300,000 LONDON, UK
WWW.BLACKHATCONF.COM

- The “Standard Approach” still works-ish... However we need to add to it
 - Extract APK file from device
 - Use Apktool to unzip
 - Dex2jar or baksmali the classes.dex file
 - Choose your own adventure:

Expanding the “Standard Approach”

JUNE 15 - 16 2015
JULY 15 - 16 2015
LONDON, UK
LONDON, UK
blackhat.com

- Memory Dump your way to source code ← Why would you do this now? I mean other than to play with a shiny Jolla device?
- If Kony version < 6.0, then: unluac.jar and go home, you're done now. ← Quick and easy
- If Kony version >= 6.0 then: see the rest of this talk

Kony Past, Present, Future

- What exactly has changed?
- Previous versions of the Kony Studio IDE used the compiled Lua bytecode.
 - My guess is that this was expensive to maintain and/or crashed often with newer android versions.

Kony Past, Present, Future

JUNE 16 - 18, 2015
EXCEL, LONDON | LONDON, UK
WWW.BLACKHAT.COM

- Lua bytecode support was *deprecated* starting with version 6.0 released December 14, 2014
 - I use that word loosely, looking through the IDE, there still seems to be plenty of legacy support.
- Lua Bytecode = old and busted: libkonyjsvm.so = new hotness

Kony Past, Present, Future

- Within the libkonyjsvm.so category there are two camps:
 - Present: Applications you'll probably find in the Play store now.
 - Future: Applications that will probably be in the Play store soon

Kony Past, Present, Future

JUNE 16 - 18, 2015
EXCEL LONDON | LONDON, UK
WWW.BLACKHAT.COM

- So what's the difference?
- Sometime after version 6.0 they updated the framework and the Android shared object file to encrypt the source code.
- I'm not sure when they did this, changelogs on the site seem to be sparse or I can't find them

Kony Past, Present, Future

- The version I'm currently working with is Kony Studio version 6.3
- So how does this change our analysis?

Kony Past, Present, Future

JUNE 16 - 18, 2015
EXCEL LONDON | LONDON, UK
WWW.BLACKHAT.COM

- Again, it's a choose your own adventure.
 - Extract APK file from device
 - Use Apktool to unzip
 - Dex2jar or baksmali the classes.dex file
 - Choose your own adventure:
 - If startup.js = zip then: unzip all the things, you are done, go home
 - If startup.js = data then: see the rest of this talk

- Yes, that's right, with the early version of the JSVM, they actually made things easier for us.

```
root@kali: ~/Southwest-apktool/assets/js
```

```
File Edit View Search Terminal Help
```

```
root@kali:~/Southwest-apktool/assets/js# file *  
startup.js: Zip archive data, at least v1.0 to extract  
root@kali:~/Southwest-apktool/assets/js#
```

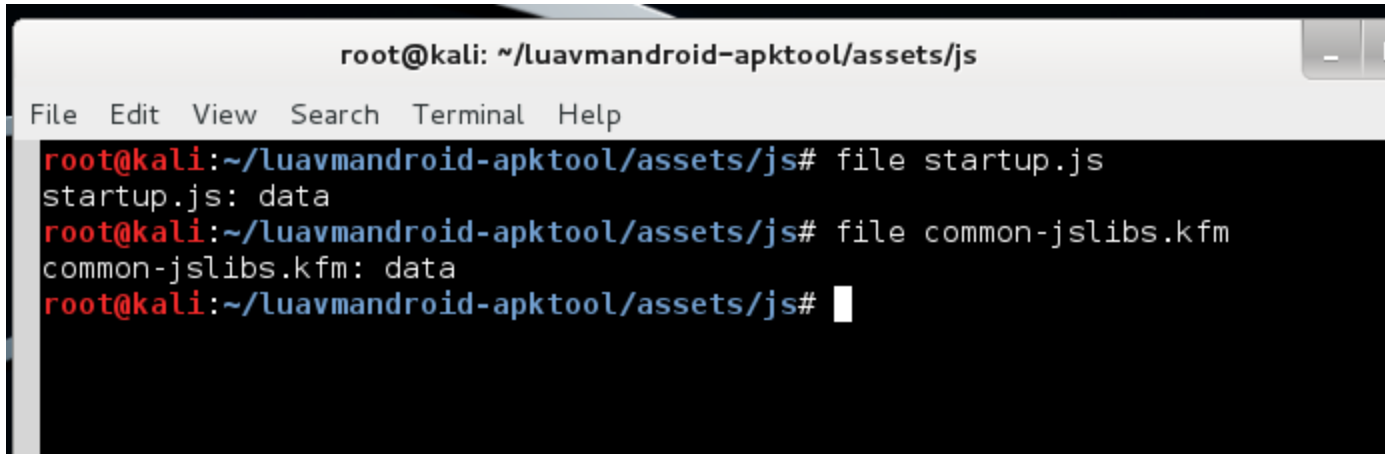
Kony Present

JUNE 16 - 18, 2015
EXCEL LONDON | LONDON, UK
WWW.BLACKHAT.COM

- That was easy, why can't they all be that easy?
- Plug the source into whatever text reader you choose and find bugs galore
- Debugging the App is slightly more complicated
debugging solely using ADB will not really cut it.

- IDA and/or GDB up!!
 - Start app in “Wait for debugger mode”
 - Attach gdb or IDA to the process ← so you can analyze the shared object library
 - Attach ADB to the application ← so you can debug the actual application.
 - Profit? IDK WTF

- However, if they use the newer library with encryption... well then we have ourselves a process...



```
root@kali: ~/luavmandroid-apktool/assets/js
File Edit View Search Terminal Help
root@kali:~/luavmandroid-apktool/assets/js# file startup.js
startup.js: data
root@kali:~/luavmandroid-apktool/assets/js# file common-jslibs.kfm
common-jslibs.kfm: data
root@kali:~/luavmandroid-apktool/assets/js#
```

Kony Future

JUNE 16 - 18, 2015
EXCEL LONDON | LONDON, UK
WWW.BLACKHAT.COM

- WTF is data means?
 - Strings, hexdumping, etc... pretty useless if figuring out what this file was
 - Analysis of the files determined that they were not compressed data or serialized data(/dev/tty0 blog post on the subject was great, link at bottom)

Kony Future

JUNE 16 - 18, 2015
EXCEL LONDON | LONDON, UK
WWW.BLACKHAT.COM

- Tried debugging/dynamic analysis, that didn't really seem to help(in gaining source code access)
- What is one to do?
 - Time to put my "Dev" hat on and download the IDE
- Started with the Application Build process and worked my way backward(BTW, I have never "HelloWorld'd" so hard in my life, must have built this app 200+ times)

Console Progress Auto Preview Console

Kony Studio Console

```
[mkdir] Created dir: C:\Users\Chris\KonySampleApps\temp\HelloWorld\build\luaandroid\minification
[mkdir] Created dir: C:\Users\Chris\KonySampleApps\temp\HelloWorld\build\luaandroid\minification\nongenerated
[mkdir] Created dir: C:\Users\Chris\KonySampleApps\temp\HelloWorld\build\luaandroid\minification\generated
[mkdir] Created dir: C:\Users\Chris\KonySampleApps\temp\HelloWorld\build\luaandroid\minification\startup
[copy] Copying 4 files to C:\Users\Chris\KonySampleApps\temp\HelloWorld\build\luaandroid\minification\generated
[copy] Copying 5 files to C:\Users\Chris\KonySampleApps\temp\HelloWorld\build\luaandroid\minification\nongenerated
[copy] Copying 1 file to C:\Users\Chris\KonySampleApps\temp\HelloWorld\build\luaandroid\minification\startup
[mkdir] Created dir: C:\Users\Chris\KonySampleApps\temp\HelloWorld\build\luaandroid\dist\HelloWorld\assets\js
[taskdef] Could not load definitions from resource net/sf/antcontrib/antlib.xml. It could not be found.
```

zip-common-jslibs

```
[zip] Building zip: C:\Users\Chris\KonySampleApps\temp\HelloWorld\build\luaandroid\dist\HelloWorld\assets\js\common-jslibs.kfm
[taskdef] Could not load definitions from resource net/sf/antcontrib/antlib.xml. It could not be found.
```

zip-workerthreads:

```
[taskdef] Could not load definitions from resource net/sf/antcontrib/antlib.xml. It could not be found.
```

gen-zip-files:

```
[taskdef] Could not load definitions from resource net/sf/antcontrib/antlib.xml. It could not be found.
```

gen-js-zip-files:

```
[zip] Building zip: C:\Users\Chris\KonySampleApps\temp\HelloWorld\build\luaandroid\dist\HelloWorld\assets\js\startup.js
[taskdef] Could not load definitions from resource net/sf/antcontrib/antlib.xml. It could not be found.
```

encrypt-zip-files:

```
[echo] Encrypting C:\Users\Chris\KonySampleApps\temp\HelloWorld\build\luaandroid\dist\HelloWorld\assets\js\common-jslibs.kfm ...
```

Kony Future

JUNE 16 - 18, 2015
EXCEL LONDON | LONDON, UK
WWW.BLACKHAT.COM

- However, if they use the newer library with encryption... well then we have ourselves a process...
 - Start with the build process...
 - We need to break it down and analyze what's being done on the simplest Apps, so we can move on to more complex Apps

- Build Process Breakdown
 - Copy build process script and configuration details to project folders
 - Escape JavaScript source code
 - If enabled, “encode” JavaScript source
 - Generate zip files of JavaScript source

- Build Process Breakdown(cont.)
 - Encrypt zip files
 - Apply Proguard if enabled(I didn't enable this, but that should not affect the JavaScript source code at all)
 - Package contents up and generate an APK file

Kony Future

JUNE 16 - 18, 2015
EXCEL LONDON | LONDON, UK
WWW.BLACKHAT.COM

- Analysis of Kony Apps using the "Standard Approach"
 - Start by hacking apart the build process.
 - Definitely turn on debugging output at the build console
 - Saw that they were using the file "build-jssource.xml" for the source code operations(escape, encode, zip, etc.)

- This file is basically an ANT configuration script
- So I added to the file

```
Console Progress Auto Preview Console
Kony Studio Console
[zip] Building zip: C:\Users\Chris\KonySampleApps\temp\HelloWorld\build\luaanc
[taskdef] Could not load definitions from resource net/sf/antcontrib/antlib.xml. I

encrypt-zip-files:
[echo] Encrypting C:\Users\Chris\KonySampleApps\temp\HelloWorld\build\luaandroi
[echo] WEEDON was here
[copy] Copying 1 file to C:\Users\Chris\testing
[echo] WEEDON-DEBUG C:\Users\Chris\KonySampleApps\temp\HelloWorld\build\luaandroi
[move] Moving 1 file to C:\Users\Chris\KonySampleApps\temp\HelloWorld\build\luaandroi
[echo] Encrypting C:\Users\Chris\KonySampleApps\temp\HelloWorld\build\luaandroi
[echo] WEEDON was here
[copy] Copying 1 file to C:\Users\Chris\testing
[echo] WEEDON-DEBUG C:\Users\Chris\KonySampleApps\temp\HelloWorld\build\luaandroi
[move] Moving 1 file to C:\Users\Chris\KonySampleApps\temp\HelloWorld\build\luaandroi
[echo] WEEDON-DEBUG: Print Variables App-dir: C:\Users\Chris\KonySampleApps\temp\HelloWorld\build\luaandroi
[taskdef] Could not load definitions from resource net/sf/antcontrib/antlib.xml. I

copy-PaaS-resources:
```

- The mysterious case of kony_loadfile.exe
 - I went through almost every file in the build process, some things stuck out... such as “yuicompressor.jar” or pictured below “EncodeScriptTask.jar”

Kony Future

JUNE 16 - 18, 2015
EXCEL LONDON | LONDON, UK
WWW.BLACKHAT.COM

```
Java Decompiler - EncodeScriptTask.class
File Edit Navigate Search Help
KonyLuaVM.jar codegen.jar EncodeScriptTask.jar
META-INF
com.konylabs.android.anttask
  EncodeScriptTask
    EncodeScriptTask
      MAX_STRING_LEN: int
      appSource: String
      propertyName: String
      decrypt(String, InputStream, OutputStream)
      doCopy(InputStream, OutputStream)
      encrypt(String, InputStream, OutputStream)
      encryptOrDecrypt(String, int, InputStream, OutputStream)
      execute(): void
      setAppSource(String): void

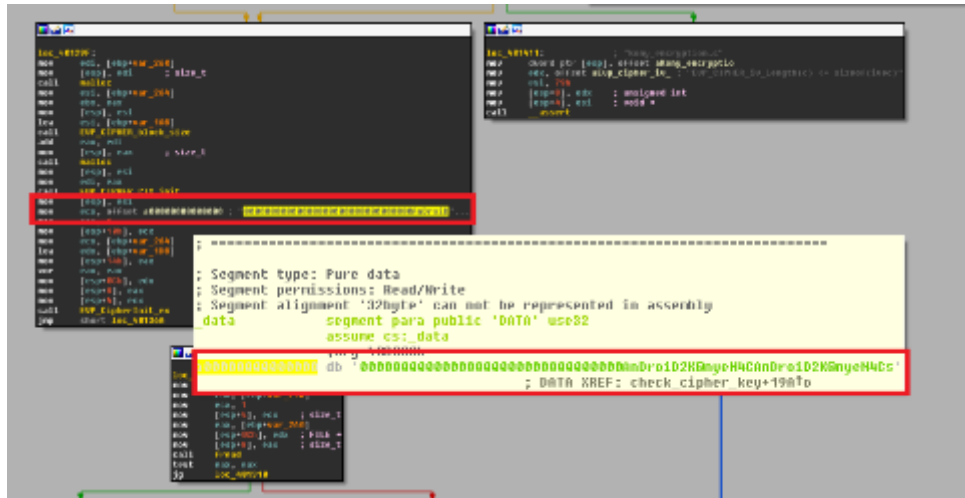
EncodeScriptTask.class
{
    localFileNotFoundException2.printStackTrace();
}
localFile = new File(this.appSource.replace("startup.js", "startup_encrypt.js"));
if (!localFile.exists())
{
    try
    {
        localFile.createNewFile();
    }
    catch (IOException localIOException)
    {
        localIOException.printStackTrace();
    }
    localFileOutputStream = new FileOutputStream(localFile, false);
    try
    {
        encrypt("android123", localFileInputStream, localFileOutputStream);
    }
    catch (Throwable localThrowable)
    {
        localThrowable.printStackTrace();
    }
}
catch (FileNotFoundException localFileNotFoundException1)
{
    localFileNotFoundException1.printStackTrace();
}
finally
{
}
}

public static void encrypt(String paramString, InputStream paramInputStream, OutputStream paramOutputStream)
throws Throwable
{
    encryptOrDecrypt(paramString, 1, paramInputStream, paramOutputStream);
}
```

Kony Future

JUNE 16 - 18, 2015
EXCEL LONDON | LONDON, UK
WWW.BLACKHAT.COM

- The mysterious case of kony_loadfile.exe
 - However, DES encryption with a silly, simple key aside, kony_loadfile.exe stuck out the most, as it was kony_loadfile being called during the encrypt build process.



Kony Future

JUNE 16 - 18, 2015
EXCEL LONDON | LONDON, UK
WWW.BLACKHAT.COM

- Okay, so we know that this thing is encrypting the files, we know that the key that's hardcoded into the binary.
 - But do we really? ... really...?
 - So we search for a decryption process in the Android APK
 - After exhaustively searching the decompiled APK code, the only thing left to search was the shared object library.
BINGO!!

Kony Future

JUNE 16 - 18, 2015
EXCEL LONDON | LONDON, UK
WWW.BLACKHAT.COM

IDA - C:\Users\Chris\Desktop\Kony\Library Files\libkonyjsvm.idb (libkonyjsvm.so)

Help

Remote ARM Linux/Android debugger

External symbol

IDA View-A Strings window Hex View-1 Structures Enums

Address	Disassembly	Comment
.rodata:003563D4	DCB 0	
.rodata:003563D5	DCB 0	
.rodata:003563D6	DCB 0	
.rodata:003563D7	DCB 0	
.rodata:003563D8	DCB 0x41	; DATA XREF: eod+54to
.rodata:003563D9	DCB 0x6E	; eod+56to ...
.rodata:003563DA	DCB 0x44	
.rodata:003563DB	DCB 0x72	
.rodata:003563DC	DCB 0x6F	
.rodata:003563DD	DCB 0x69	
.rodata:003563DE	DCB 0x44	
.rodata:003563DF	DCB 0x32	
.rodata:003563E0	DCB 0x4B	
.rodata:003563E1	DCB 0x40	
.rodata:003563E2	DCB 0x6E	
.rodata:003563E3	DCB 0x79	
.rodata:003563E4	DCB 0x65	
.rodata:003563E5	DCB 0x4E	
.rodata:003563E6	DCB 0x34	
.rodata:003563E7	DCB 0x43	
.rodata:003563E8	DCB 0x41	
.rodata:003563E9	DCB 0x6E	
.rodata:003563EA	DCB 0x44	
.rodata:003563EB	DCB 0x72	
.rodata:003563EC	DCB 0x6F	
.rodata:003563ED	DCB 0x69	
.rodata:003563EE	DCB 0x44	

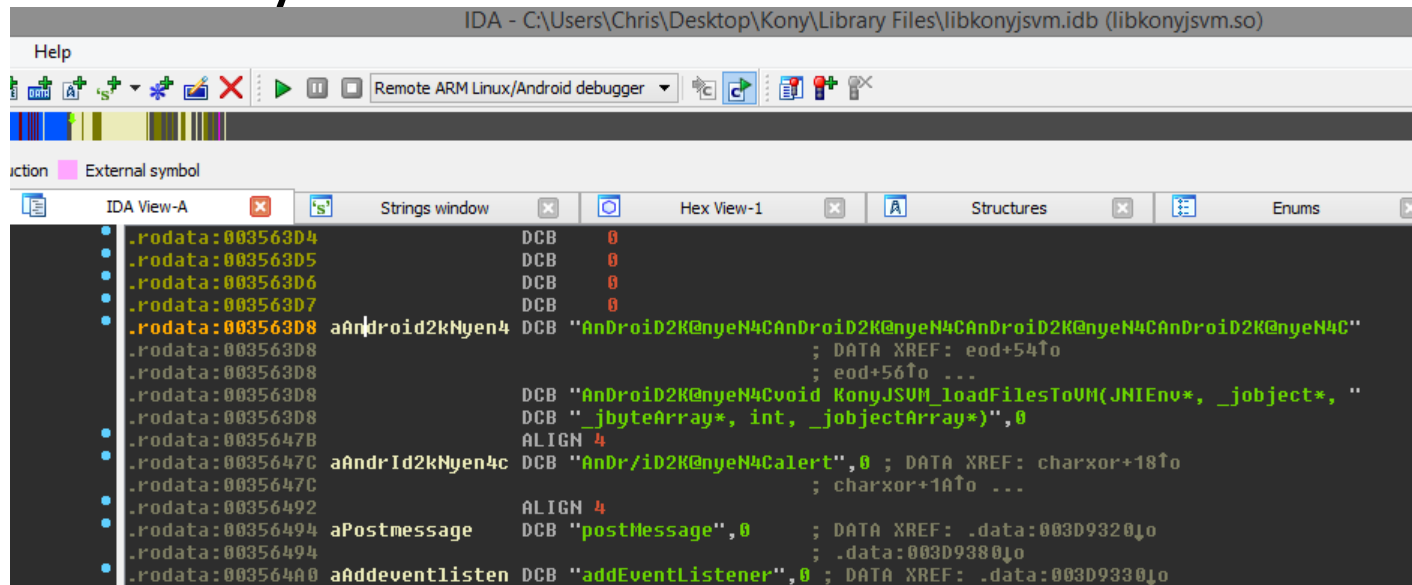
Please confirm

Directly convert to string?

Yes No

☐ Don't display this message again (for this database only)

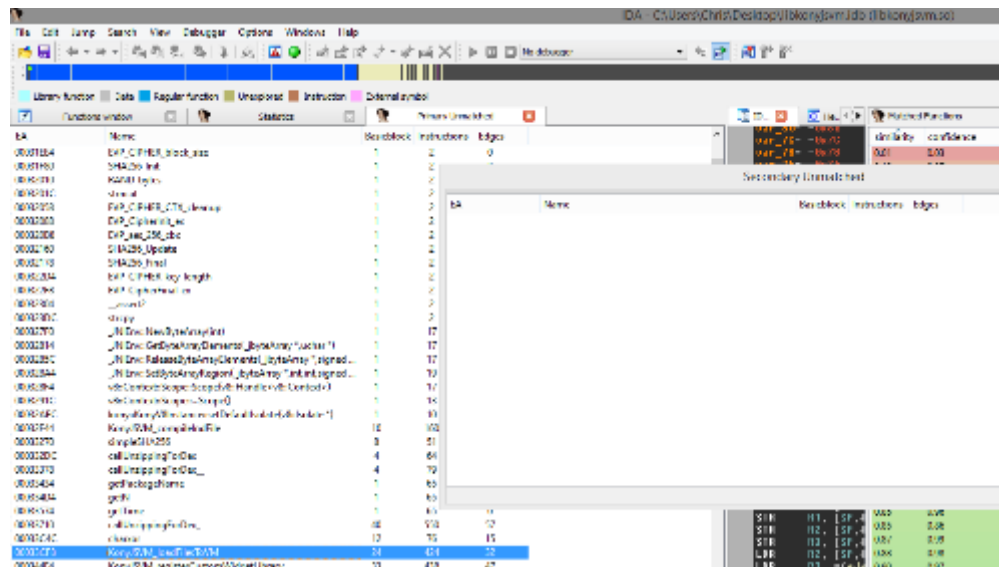
- Oh there you are!!



Kony Future

JUNE 16 - 18, 2015
EXCEL LONDON | LONDON, UK
WWW.BLACKHAT.COM

- So we should look at the new library file vs the old library file for good measure.
- Pull an older Kony App from the Play store and reverse.
- Use my good friend BinDiff or whatever binary differential tool or your choosing to get the same task done.



BinDiff of newer libkonyjsvm.so file vs. older libkonyjsvm.so file

Kony Future

JUNE 16 - 18, 2015
EXCEL LONDON | LONDON, UK
WWW.BLACKHAT.COM

- Okay, so no secondary unmatched functions.
- Everything in the old library stayed the same, they just added to it.
- Looking through what they added, primary unmatched functions look to be only the new crypto functionality. This is good, streamlines what we need to look at.

Kony Future

JUNE 16 - 18, 2015
EXCEL LONDON | LONDON, UK
WWW.BLACKHAT.COM

- So we DEFINITELY know we have the key!
 - Start your python... many lines of failure later
 - Wait WTF? Why isn't this working?
- Seems as though someone at Kony knows hardcoding keys is a bad idea

Kony Future

JUNE 16 - 18, 2015
EXCEL LONDON | LONDON, UK
WWW.BLACKHAT.COM

- So the hardcoded key seems to go through some “magic”
- After some really long nights, copious amounts of coffee, and some help from friends(Thanks to Dominic Wang and Jay Smith!!) We isolated the “magic”

- [illegible]

- And because that's probably impossible to see...
breakdown below:
 - Take the command line arguments from the kony_loadfile.exe call and do some charxor voodoo
 - Take the output of the charxor voodoo and the hardcoded key and feed the into SHA256
 - Take the output of SHA256 and the IV and feed into AES-256-CBC

Kony Future

JUNE 16 - 18, 2015
EXCEL LONDON | LONDON, UK
WWW.BLACKHAT.COM

- So I asked myself “Self, how am I to divine such command line parameters without the IDE and building the App myself?”
 - This is AES, they must be passing these values to the Android application some how.
 - Not Immediately clear to me, so I turn to the build process

Kony Future

JUNE 16 - 18, 2015
EXCEL LONDON | LONDON, UK
WWW.BLACKHAT.COM

- Add more debugging output to your build process. And echo the parameters back to yourself in the build console. The Parameters are:
 - App ID = HelloWorld ← easy to get
 - Package Name = com.kony.HelloWorld ← also easy
 - Timestamp of the App build = 20150530095920 ← where is this located?

Kony Future

JUNE 16 - 18, 2015
EXCEL LONDON | LONDON, UK
WWW.BLACKHAT.COM

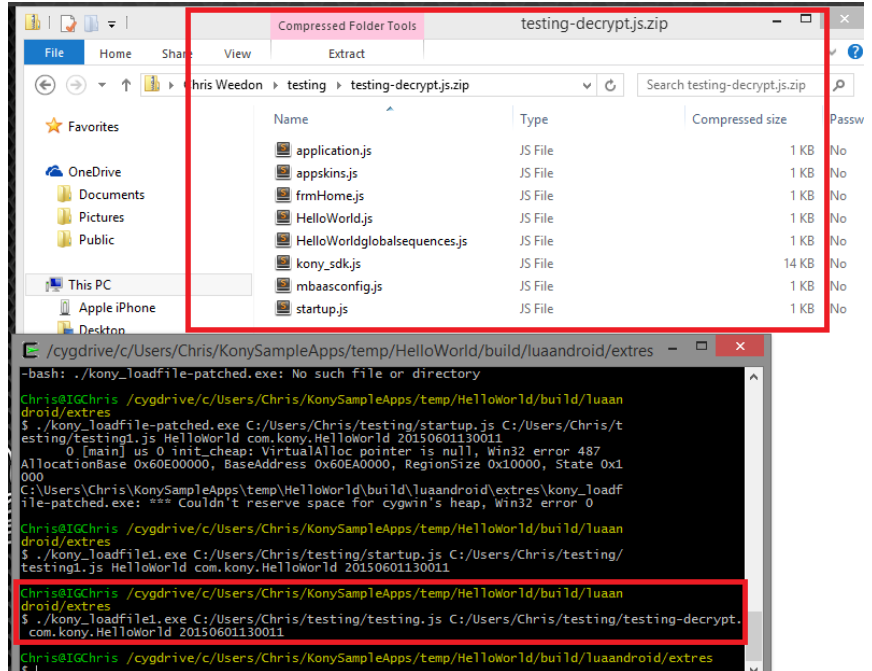
- Everything you need is right before you
 - Turns out, the timestamp is written into the APK in a file called `application.properties` located at `$yourapkname/assets`
 - So in the APK, we have all the VARs we need.
 - To the python!!

Kony Future

JUNE 16 - 18, 2015
EXCEL LONDON | LONDON, UK
WWW.BLACKHAT.COM

- Eh... no time. They already gave us a binary, we already spent the time to reverse most of it.
 - Binary Patching FTW
 - Note: when calling EVP_AES256, one of the parameters you pass in is “1” for encrypt, or “0” for decrypt.
- So by patching one byte in this binary, we turn their encrypter into a decrypter!!

- Using the patched binary



Kony Future

JUNE 16 - 18, 2015
EXCEL LONDON | LONDON, UK
WWW.BLACKHAT.COM

- Boom goes the dynamite, source code!!
- But... What if you really, really want to debug this new hotness?
- Why would you want to do that? Honestly if you're asking yourself "How do I debug these apps?" You already know why ;-)

Kony Future Beyond

JUNE 16 - 18, 2015
EXCEL LONDON | LONDON, UK
WWW.BLACKHAT.COM

- Various way to debug, previously describe method of ADB + GDB and/or IDA still works... but wait, there's more!
- Back to the IDE we go...

- Included in the IDE are two versions of libkonyjsvm.so
 - libkonyjsvm_release.so
 - libkonyjsvm_debug.so

armeabi			
KonySampleApps ▶ temp ▶ HelloWorld ▶ build ▶ luaandroid ▶ extlibs ▶ armeabi			
Name	Date modified	Type	Size
libKLChartWidgetNeXt.so	6/1/2015 4:40 PM	SO File	1,390 KB
libkonyjsvm_debug.so	6/1/2015 4:40 PM	SO File	3,959 KB
libkonyjsvm_release.so	6/1/2015 4:40 PM	SO File	3,943 KB

Kony Future Beyond

JUNE 16 - 18, 2015
EXCEL LONDON | LONDON, UK
WWW.BLACKHAT.COM

- They GAVE us a debug version!!
 - But really what are the differences between these two?
 - File size is, negligibly different...
 - We can't really just plug and play can we?
 - Let's look at the files using BinDiff... next slide

Kony Future

JUNE 16 - 18, 2015
EXCEL LONDON | LONDON, UK
WWW.BLACKHAT.COM

luatabrcandroid\extlibs\armeabi\libkonyjsvm_release.idb (libkonyjsvm_release.so)

similarity	name	EA	Name	EA	Name
1.00	basicBlock matches (l			00032200	sleep
1.00	basicBlock matches (r			000323B0	__android_log_print
1.00	basicBlocks primary (l			00032ACC	__ZN7_JNIEnv23CallStaticBooleanMethodEP7_jclassP10_j...
1.00	basicBlocks primary (l			00032B0C	__ZN7_JNIEnv20CallStaticVoidMethodEP7_jclassP10_jmet...
1.00	basicBlocks secondary			00032D78	__ZN7_JavaVM19AttachCurrentThreadEPP7_JNIEnvPv
1.00	basicBlocks secondary			00032D98	__ZN7_JavaVM19DetachCurrentThreadEv
1.00	flowGraph edge matc			00032DE4	__ZN2v814HeapStatistics15total_heap_sizeEv
1.00	flowGraph edge matc			00032DF4	__ZN2v814HeapStatistics26total_heap_size_executableEv
1.00	flowGraph edges prim			00032E04	__ZN2v814HeapStatistics14used_heap_sizeEv
1.00	flowGraph edges prim			00032E14	__ZN2v814HeapStatistics15heap_size_limitEv
1.00	flowGraph edges seco			00033058	sub_33058_8903
1.00	flowGraph edges seco			000380EC	KonyJSVM_MessageCallback
1.00	function matches (lib			00038198	KonyJSVM_FatalErrorCallback
1.00	function matches (no			00038220	KonyJSVM_GCPrologueCallback
1.00	functions primary (lib			0003829C	KonyJSVM_GCPrologueCallback
1.00	functions primary (no			000384F0	__ZN2v810PersistentINS_7ContextEE3NewENS_6HandleIS1...
1.00	functions secondary (0003898C	sub_3898C_8905
1.00	functions secondary (000389B8	sub_389B8_8906
1.00	instruction matches (l			00044DA0	__ZN7_JNIEnv17ExceptionDescribeEv
1.00	instruction matches (r			003E5EAC	__imp_sleep
1.00	instructions primary (003E5F38	__imp__android_log_print
1.00	instructions primary (
1.00	instructions secondary				

Kony Future

JUNE 16 - 18, 2015
EXCEL LONDON | LONDON, UK
WWW.BLACKHAT.COM

- Seems pretty legit. Differences seem to only apply to the debug functionality that's added. For printing out log info etc.
- Possible to rename and replace the debug shared object library with the release one. Repackage, resign, re-run, and debug.

Kony Future Beyond

JUNE 16 - 18, 2015
EXCEL LONDON | LONDON, UK
WWW.BLACKHAT.COM

- Winning debug setup combination. Allows debugging of native functions, ADB + JNI/JDWP + GDB and/or IDA + highly verbose ADB logcat
- Combine this with a user-debug build of Android for your particular device

Kony Future Beyond

JUNE 16 - 18, 2015
EXCEL LONDON | LONDON, UK
WWW.BLACKHAT.COM

- If you stuck with me, or even if you didn't and you grabbed the slides and ran. At this point you should know:
 - Differences between 3 major revisions of the Kony Application Framework

- How to analyze, decompile, and debug:
 - Kony Lua Bytecode VMs
 - Kony applications using Android Shared library file without encryption
 - Kony applications using Android Shared library file with encryption

- Misc. things you now know:
 - Kony is pretty regularly updating their framework
 - These methods and techniques can change overnight with a new framework update [?]
 - You'll have to keep track of their library files by pulling them from Apps yourself if you don't grab the IDE

- Other Notes and Misc.:
- If not already done, will be working on a python script to decrypt files statically in place.
- Will try to keep this research updated as framework updates occur.
- Look for a step by step blog post coming in the near future.

- Notes on process: There are options for additionally protecting with Proguard, JavaScript source “encoding”, and FIPS encryption. The methods and techniques were not investigated. Maybe next time... or a future blog post coming, who knows.

- Sources and links:
 - <https://www.nccgroup.trust/us/about-us/newsroom-and-events/blog/2013/june/kony-2013-a-different-kind-of-android-reversing/>
 - <http://www.kony.com/products/development#trial>
 - <http://developer.kony.com/twiki/pub/Portal/Releases/>
 - <https://www.blackhat.com/asia-15/briefings.html#the-nightmare-behind-the-cross-platform-mobile-apps-dream>
 - <http://www.devttys0.com/2013/06/encryption-vs-compression-part-2/#more-1596>
 - <http://sourceforge.net/projects/unluac/?source=directory>

Thanks

JUNE 16 - 18, 2015
EXCEL LONDON | LONDON, UK
WWW.BLACKHAT.COM

- Thanks to my company, managers for paying me to hack stuff
- To Black Hat for having me speak
- Jason Ross for starting me down this dark path with his killer blog post
- Friends and coworkers for giving me a hand or a kick when I needed it
- To you all for listening!! Hope this stuff is helpful.

Thanks

JUNE 16 - 18, 2015
EXCEL LONDON | LONDON, UK
WWW.BLACKHAT.COM

- Reach out, We're hiring!
- Also, if you have questions, comments, hate mail feel free to send those too.
- EMAIL: chris@intrepidusgroup.com
- Twitter: [crweedon](https://twitter.com/crweedon)



black hat[®]

MOBILE SECURITY SUMMIT
LONDON 2015



JUNE 16 - 18, 2015
EXCEL LONDON | LONDON, UK
WWW.BLACKHAT.COM

- Click to edit Master text styles
 - Second level
 - Third level
 - Fourth level
 - » Fifth level