RSA CONFERENCE 2014
FEBRUARY 24 – 28 | MOSCONE CENTER | SAN FRANCISCO

Share.
Learn.
Secure.
Capitalizing on
Collective Intelligence

Hacking iOS on the Run: Using Cycript

SESSION ID: HTA-R04A

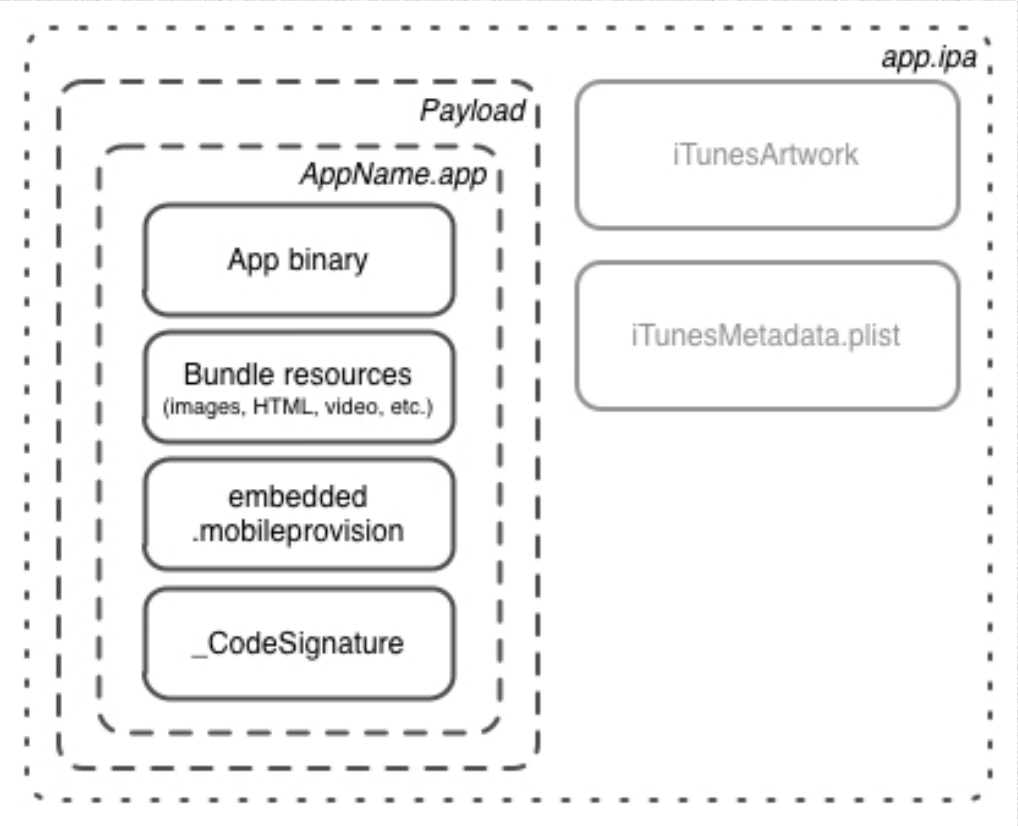
Sebastián Guerrero
Mobile Security Analyst
viaForensics
@0xroot

Agenda

- Analyzing binaries
- Encrypted binaries
- Abusing the Runtime with Cycript
- Securing the Runtime

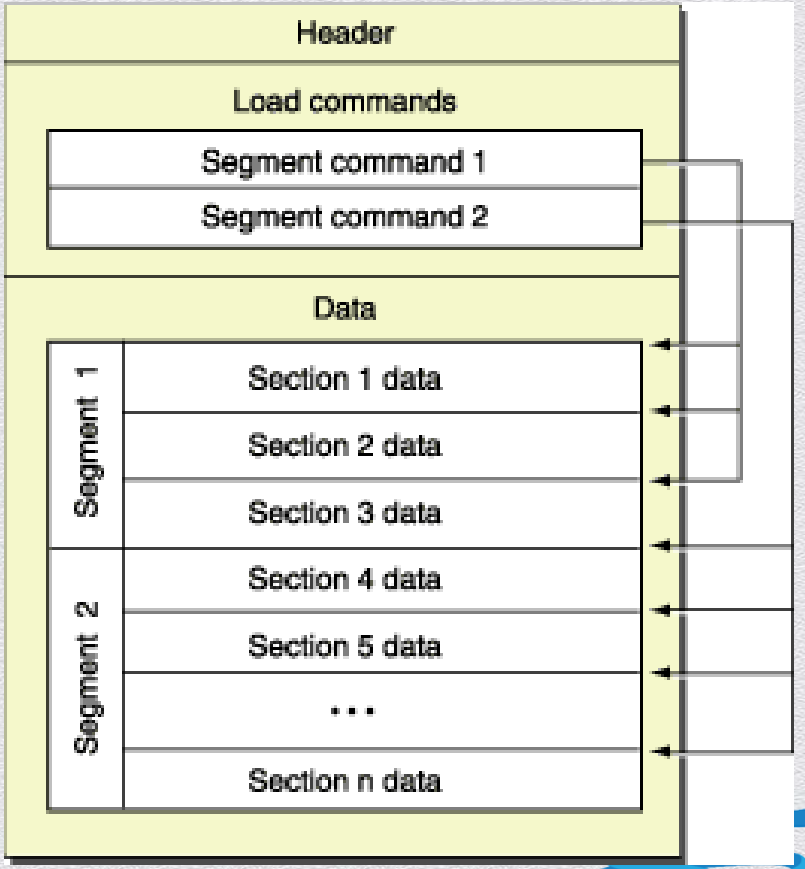
Analyzing binaries

iOS App Architecture



The Mach-O format

- Header
  - Target architecture
- Load commands
  - Location of symbol table
  - Shared libraries
- Data
  - Organized in segments



The Mach-O format

- Header section can be inspected using Otool utility

```
Sebass-iPhone:~ root# otool -h /var/mobile/Applications/AB44E74F-CB66-4F85-8089-E6DB49E6F330/Evernote.app/Evernote
/var/mobile/Applications/AB44E74F-CB66-4F85-8089-E6DB49E6F330/Evernote.app/Evernote:
Mach header
      magic cputype cpusubtype  caps    filetype ncmds sizeofcmds      flags
 0xfeedface      12          9  0x00           2    57       5864 0x00218085
```

- 'Load command' section can be analyzed too

```
Sebass-iPhone:~ root# otool -L /var/mobile/Applications/AB44E74F-CB66-4F85-8089-E6DB49E6F330/Evernote.app/Evernote
/var/mobile/Applications/AB44E74F-CB66-4F85-8089-E6DB49E6F330/Evernote.app/Evernote:
        /System/Library/Frameworks/CoreText.framework/CoreText (compatibility version 1.0.0, current version 1.0.0)
        /System/Library/Frameworks/CoreGraphics.framework/CoreGraphics (compatibility version 64.0.0, current version 600.0.0)
        /System/Library/Frameworks/Foundation.framework/Foundation (compatibility version 300.0.0, current version 1047.22.0)
        /System/Library/Frameworks/QuartzCore.framework/QuartzCore (compatibility version 1.2.0, current version 1.8.0)
        /System/Library/Frameworks/SystemConfiguration.framework/SystemConfiguration (compatibility version 1.0.0, current version 615.0.0)
        /System/Library/Frameworks/UIKit.framework/UIKit (compatibility version 1.0.0, current version 2903.23.0)
        /System/Library/Frameworks/Security.framework/Security (compatibility version 1.0.0, current version 1.0.0)
        /System/Library/Frameworks/Social.framework/Social (compatibility version 1.0.0, current version 87.0.0)
        /System/Library/Frameworks/Accounts.framework/Accounts (compatibility version 1.0.0, current version 113.0.0)
```

Introduction to class-dump-z

- Outputs the equivalent of an Objective-C header
  - Classes compiled into the program
  - Its associated methods
  - Instance variables and properties

Encrypted binaries

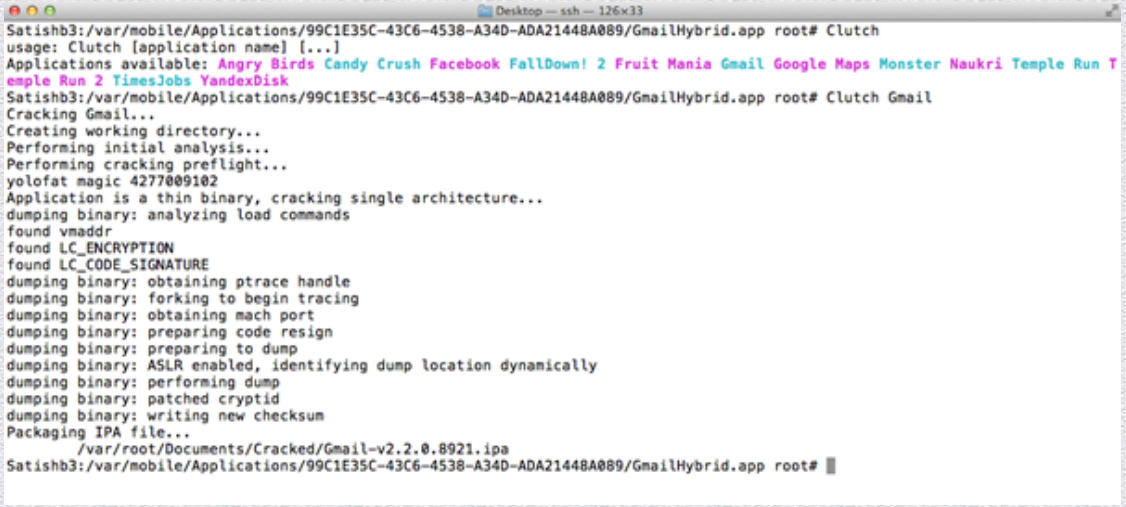
Encrypted binaries

- AppStore binaries are always encrypted
  - Similar to FairPlay DRM used on iTunes music
- Self distributed apps are not encrypted
- Loader decrypts the apps when loaded into memory
- Debugger can be used to dump the decrypted app from memory
- Manual process is tedious, there are tools available: Craculous, Clutch, Installous

Decrypting iOS Apps

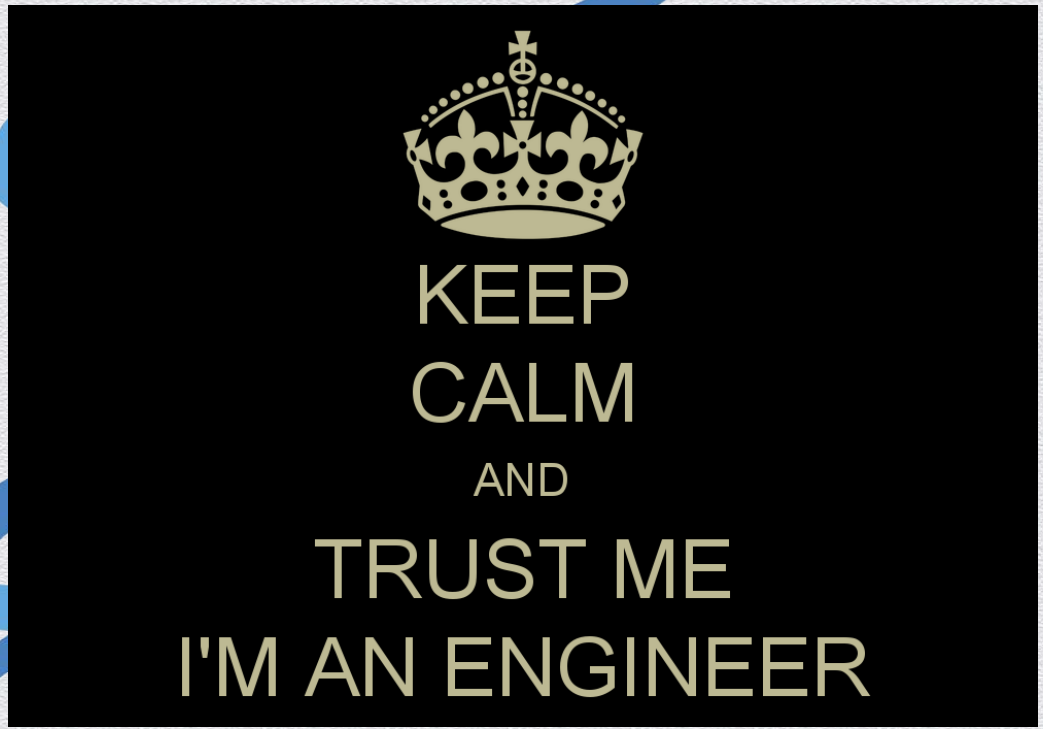
- Find the starting offset and the size of the encrypted data in the app binary.
- Find the memory loading address of the application (changes every time the app is compiled with PIE).
- Dump the decrypted portion of the application from memory using a debugger.
- Overwrite the application's encrypted area with the dumped binary data.
- Change the cycript value to 0.

Clutch


Desktop — ssh — 126×33
Satishb3:/var/mobile/Applications/99C1E35C-43C6-4538-A34D-ADA21448A089/GmailHybrid.app root# Clutch
usage: Clutch [application name] [...]
Applications available: Angry Birds Candy Crush Facebook FallDown! 2 Fruit Mania Gmail Google Maps Monster Naukri Temple Run T
emple Run 2 TimesJobs YandexDisk
Satishb3:/var/mobile/Applications/99C1E35C-43C6-4538-A34D-ADA21448A089/GmailHybrid.app root# Clutch Gmail
Cracking Gmail...
Creating working directory...
Performing initial analysis...
Performing cracking preflight...
yolofat magic 4277009102
Application is a thin binary, cracking single architecture...
dumping binary: analyzing load commands
found vmaddr
found LC_ENCRYPTION
found LC_CODE_SIGNATURE
dumping binary: obtaining ptrace handle
dumping binary: forking to begin tracing
dumping binary: obtaining mach port
dumping binary: preparing code resign
dumping binary: preparing to dump
dumping binary: ASLR enabled, identifying dump location dynamically
dumping binary: performing dump
dumping binary: patched cryptid
dumping binary: writing new checksum
Packaging IPA file...
/var/root/Documents/Cracked/Gmail-v2.2.0.8921.ipa
Satishb3:/var/mobile/Applications/99C1E35C-43C6-4538-A34D-ADA21448A089/GmailHybrid.app root#

RSA CONFERENCE 2014
FEBRUARY 24 – 28 | MOSCONE CENTER | SAN FRANCISCO

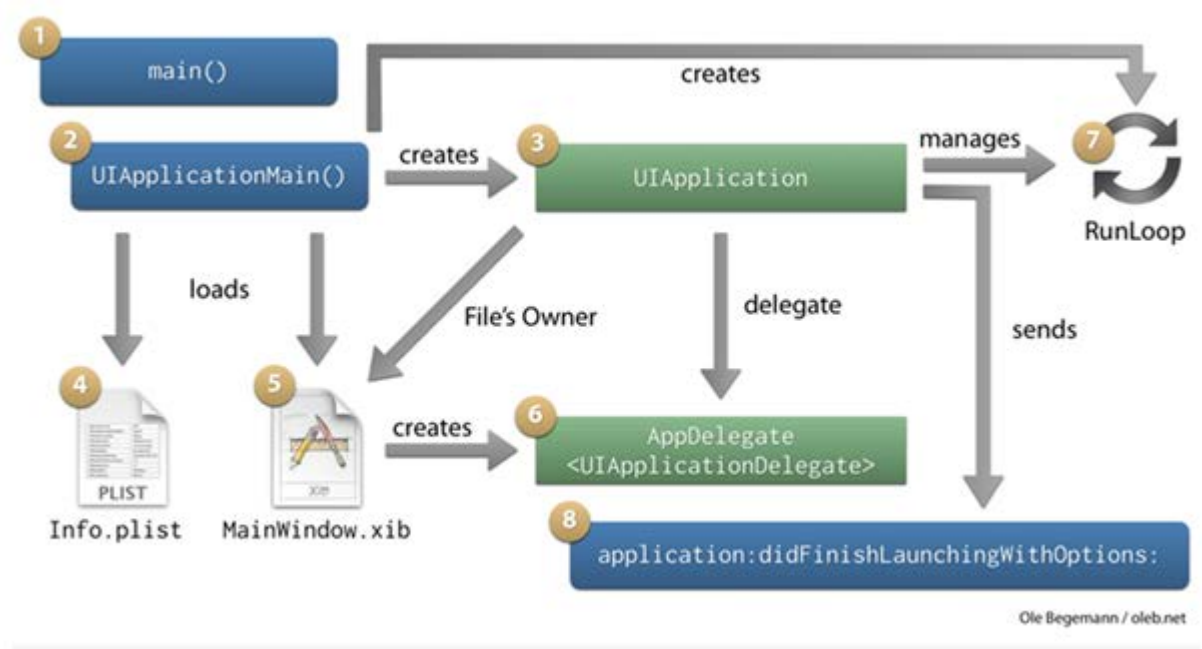
Abusing the runtime with Cycript


KEEP
CALM
AND
TRUST ME
I'M AN ENGINEER

Cycript

- Combination of JavaScript and Objective-C interpreter
- App runtime can be easily modified using Cycript
- Can be hooked to a running process
- Gives access to all classes and instance variables within the app
- Used for runtime analysis
  - Bypass security locks / Authentication Bypass attacks
  - Access sensitive information from memory
  - Accessing restricted areas of the applications

iOS App Execution Flow



Breaking simple locks

- Create object for the class and directly access the instance variables and invoke methods

DEMO

Trawling for data

- Instance variables – Provides a simple way to display an object's instance variable

```
function tryPrintIvars(a){ var x={}; for(i in *a){ try{ x[i] = (*a)[i]; } catch(e){} } return x; }
```

```
cy# *a
{message:"hasProperty callback returned true for a property that doesn't exist.",name:"ReferenceError"}
cy# tryPrintIvars(a)
{isa:"SBWaveView",_layer:"<CALayer: 0x2a5160>",_tapInfo:null,_gestureInfo:null,_gestureRecognizers:...
```

Trawling for data

- Methods– List methods as well as memory locations of their respective implementations

```
function printMethods(className) {
  var count = new new Type("I");
  var methods = class_copyMethodList(objc_getClass(className), count);
  var methodsArray = [];
  for(var i = 0; i < *count; i++) {
    var method = methods[i];
    methodsArray.push({selector:method_getName(method), implementation:method_getImplementation(method)});
  }
  free(methods);
  free(count);
  return methodsArray;
}
```

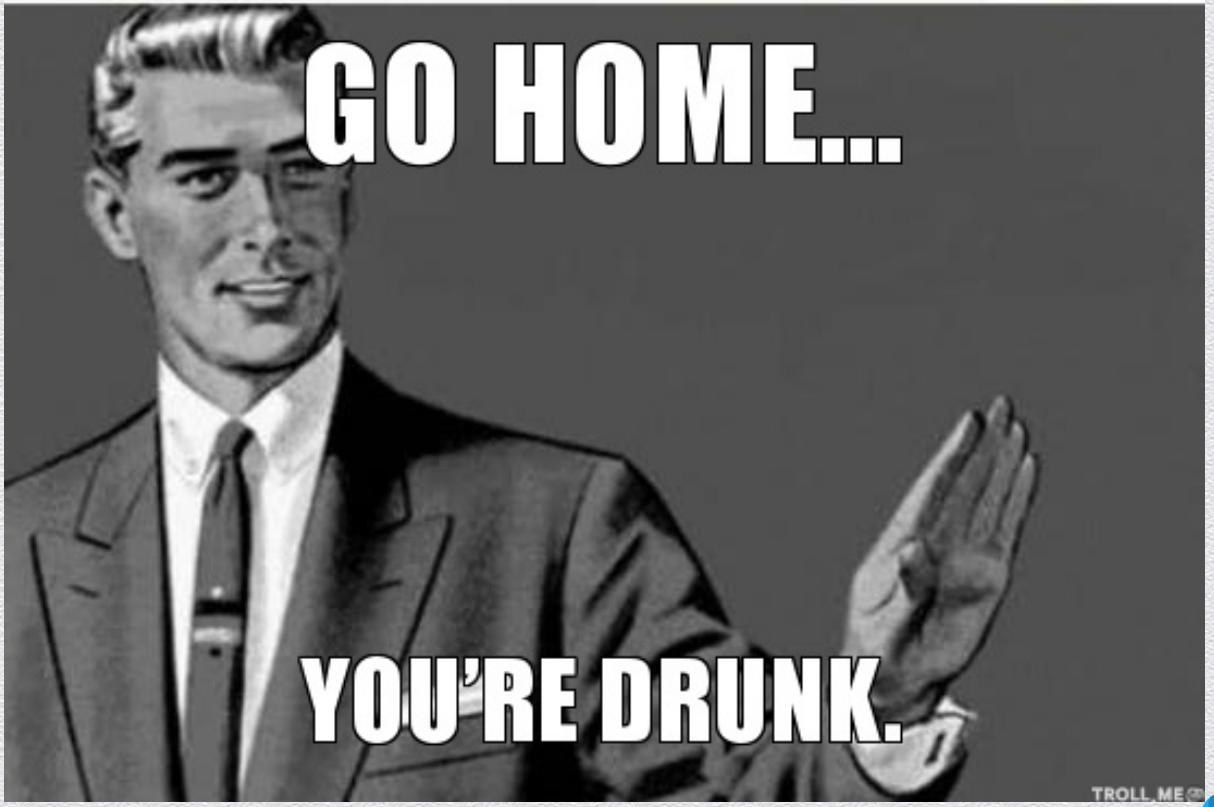
```
cy# printMethods("MailboxPrefsTableCell")
[{selector:@selector(layoutSubviews),implementation:0x302bf2e9},{selector:@selector(setCurrentMailbox:),implementation:0x302bee0d},...
cy#
```

Trawling for data

- Classes – A complete listing of classes can be dumped by referencing Cycript's built-in ObjectiveC object
  - cy# ObjectiveC.classes

Evernote Demo

- Activate premium features.
- Retrieve the PIN access code.
- Disable PIN access code.


GO HOME...
YOU'RE DRUNK.
TROLL.ME

More serious implications

- Fun applications aren't the only programs suffering from terrible security holes in their applications.
  - Financial and enterprise applications are just as bad.
  - Personal data vaults
  - Payment processing applications
  - Electronic banking
  - …

Securing the Runtime

Securing the Runtime

- Tamper response
- Process trace checking
- Blocking debuggers
- Runtime Class integrity checks
- Complicating disassembly

Summary

- Mobile devices are a hostile environment
- Is important to protect your apps
- Identify the common app vulnerabilities and remediate them

References

- https://viaforensics.com/blog/
- https://viaforensics.com/resources/reports/best-practices-ios-android-secure-mobile-development/
- http://www.cycript.org/
- http://resources.infosecinstitute.com/ios-application-security-part-8-method-swizzling-using-cycript/
- http://resources.infosecinstitute.com/ios-application-security-part-4-runtime-analysis-using-cycript-yahoo-weather-app/

Q&A | Contact | Feedback

- Thanks for listening…


@0xroot


github/0xroot


sguerrero@viaforensics.com