Mobile Application Security

Testing and Code Review

19 Nov 2013 – Mobile and Smart Device Security 2013 – Boston, MA

Presented by:
Francis Brown &
Joe DeMesy
Bishop Fox
www.bishopfox.com

BISHOP FOX

Introductions

VERY QUICKLY…

- Hi, I'm Fran
- Partner at Bishop Fox
- You may remember me from such hacks as:
  - RFID Thief
  - Diggity Search Tool Suite
  - SharePoint Hacking

Introductions

VERY QUICKLY…

- Hi, I'm Joe
- Sr. Security Analyst at Bishop Fox
- I like Python, Linux, and cryptography
- Phones / embedded devices are pretty cool too

Today We're Covering

HACKING MOBILE APPS

- Attacks against mobile apps
- Real world examples
- Defense against the dark arts

Agenda

TECHNICAL BRIEF

- Breaking iOS Apps
  - Static Analysis
  - Dynamic Analysis
  - Counter-measures
- Breaking Android Apps
  - Static Analysis
  - Dynamic Analysis
  - Counter-measures

App Security Requirements

A FEW SCENARIOS

- Online finance
- Point-of-sale
- Streaming media and DRM
- Mobile Device Management (MDM)

The Golden Rule

OF APP SECURITY

Users are Evil

EVERY LAST ONE OF 'EM

- They have complete control
- Do not trust them
- Design apps & APIs accordingly

iOS Applications

STATIC ANALYSIS

iOS Perquisites

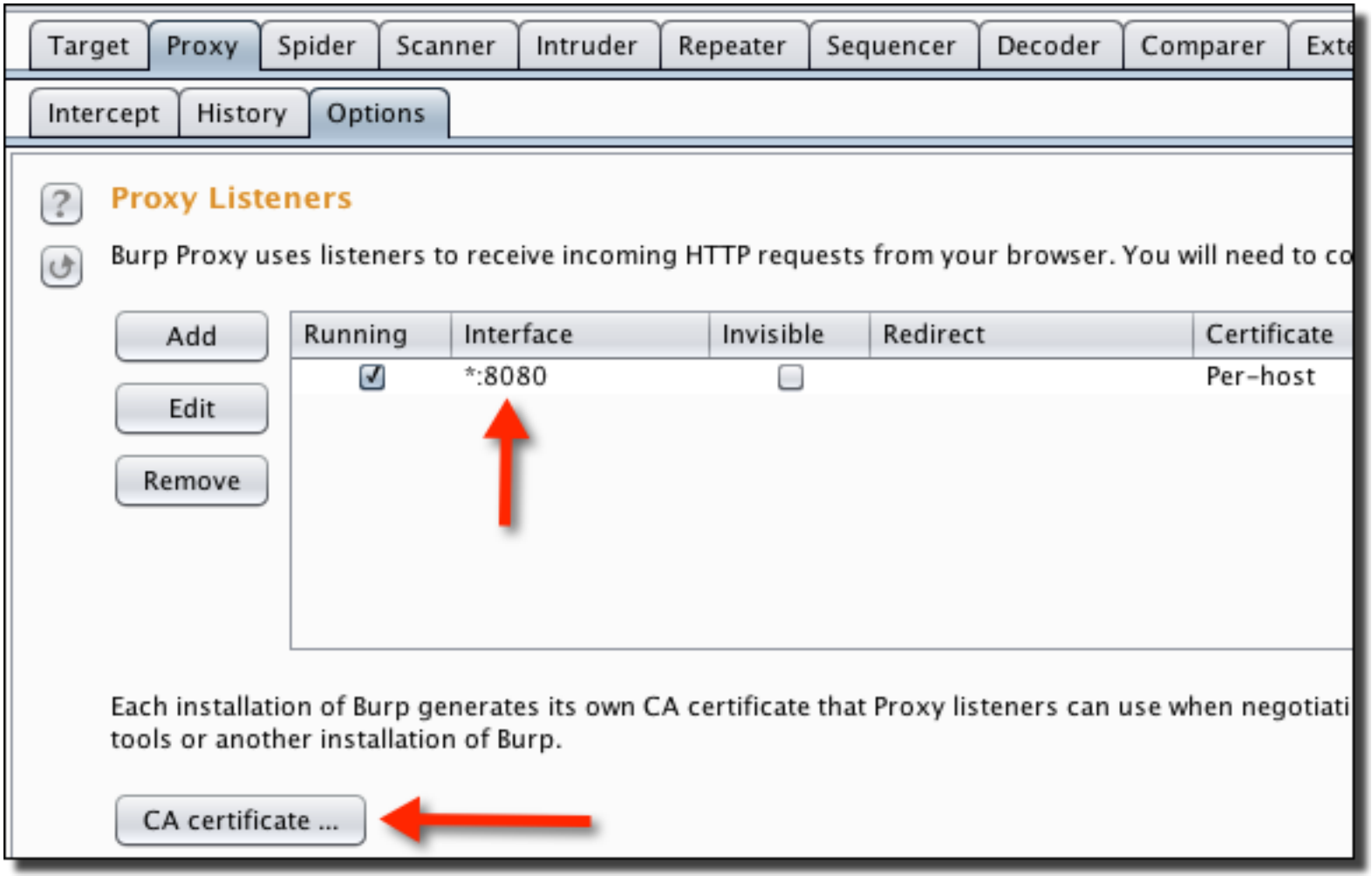
WHAT YOU NEED TO START

- Jailbroken iOS Device
  - SSH Access (scp)
- Mac & Xcode
- HTTP Proxy
  - Burp Suite Free/Pro ($300)
  - Zed Attack Proxy
- ARM Disassembler
  - Hopper ($50)
  - IDA Pro ($600+)

HTTP Proxy Setup

NETWORK ANALYSIS

Burp Suite Setup



Python –m SimpleHTTPServer

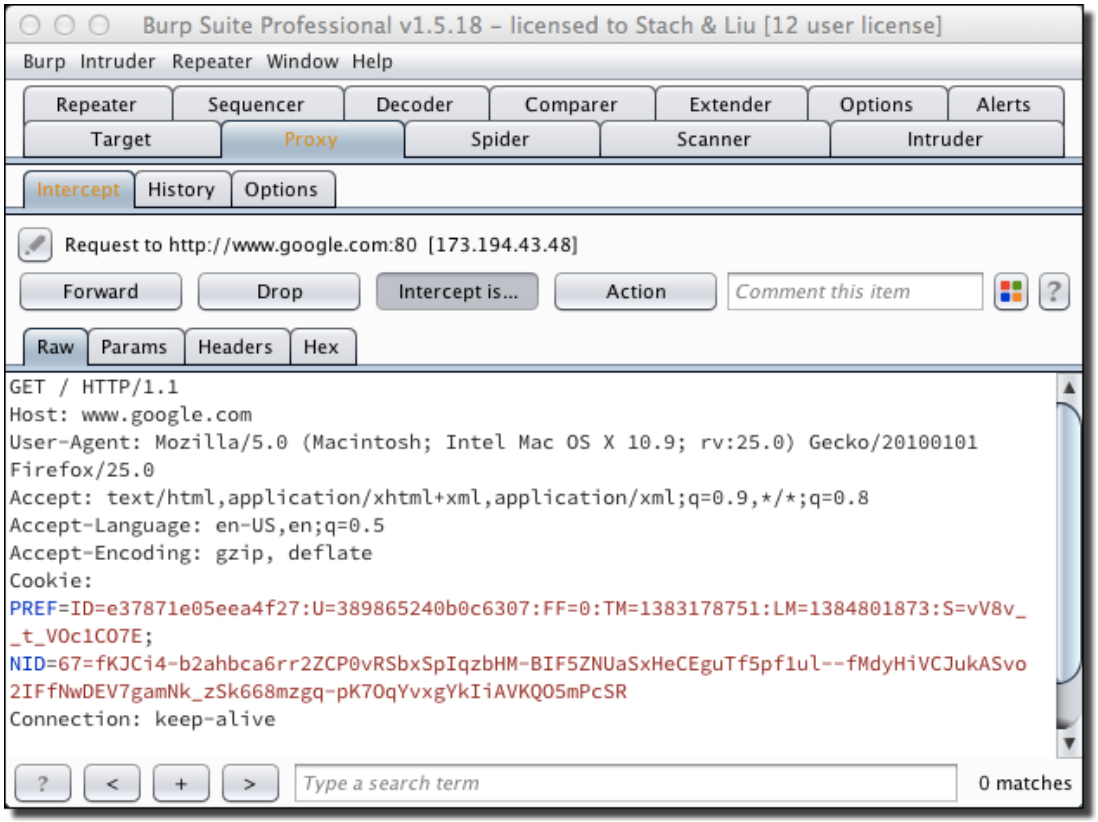
BISHOP FOX

Install You Own CA

WiFi Settings > Proxy

You're Done!

WELL SORT OF…



BISHOP FOX

AppStore Encryption

DECRYPTING BINARIES

Binary Encryption

GETTING PLAIN-TEXT BINS

- Encrypted Binaries
  - AppStore
    - Clutch
    - Rasticrac
- No Encryption
  - Provisioned Device
  - Test Flight, etc.

Clutch

AWESOME TOOL

- Decrypts iOS applications and repackages them
- Saves apps in:
  /var/root/Documents/Cracked
- Saves apps as .ipa files (they're just ZIPs)
- Use: clutch <app name>

AppStore Archive Structure

GETTING PLAIN-TEXT BINS

- Foobar.ipa
  - **iTunesMetadata.plist**
  - iTunesArtwork
  - Payload/
    - Foobar.app
      - **Foobar**
      - …

Bundle Identifier

ITUNES METADATA PLIST

```
<key>softwareSupportedDeviceIds</key>
<array>
    <integer>1</integer>
</array>
<key>softwareVersionBundleId</key>
<string>com.[redacted].agent</string>
<key>softwareVersionExternalIdentifier</key>
<integer>14817666</integer>
<key>softwareVersionExternalIdentifiers</key>
```

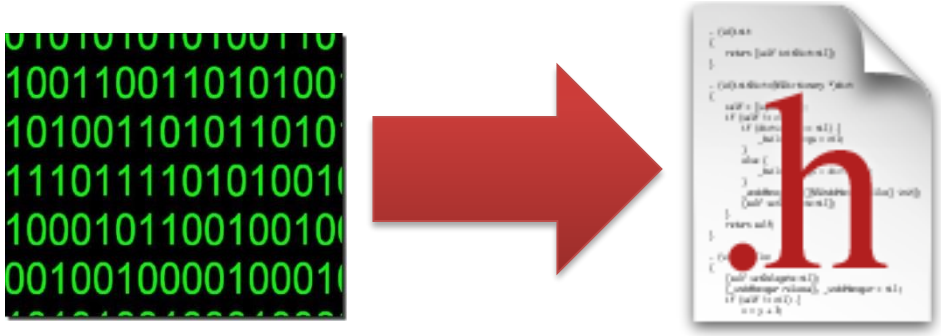
BISHOP FOX

Static Analysis

IOS APP SECURITY

Static Analysis

DUMPING CLASS INTERFACES



class-dump-z

ANOTHER AWESOME TOOL

- Dump class information from a Mach-O binary
- Shows Objective-C classes, methods, properties
- Useful for peering into iOS apps
- Great for searching for keywords

```
#import <XXUnknownSuperclass.h> // Unknown library

@class NSString;

@interface XXasymEncryptor : XXUnknownSuperclass {
        NSString* _publicKeyID;
        NSString* _privateKeyID;
}
@property(copy, nonatomic) NSString* privateKeyID;
@property(copy, nonatomic) NSString* publicKeyID;
-(id)privateQueryDict;
-(id)publicQueryDict;
-(void)decryptWithPrivateKey;
-(void)encryptWithPublicKey;
-(void)KeysPlease;
-(id)decryptData:(id)data;
-(id)encryptData:(id)data;
-(id)init;
@end
```

```
#import <XXUnknownSuperclass.h> // Unknown library

@class NSString;

@interface XXasymEncryptor : XXUnknownSuperclass {
        NSString* _publicKeyID;
        NSString* _privateKeyID;
}
@property(copy, nonatomic) NSString* privateKeyID;
@property(copy, nonatomic) NSString* publicKeyID;
-(id)privateQueryDict;
-(id)publicQueryDict;
-(void)decryptWithPrivateKey;
-(void)encryptWithPublicKey;
-(void)KeysPlease;
-(id)decryptData:(id)data;
-(id)encryptData:(id)data;
-(id)init;
@end
```

Jailbreak Detection

```
#import <Foundation/NSObject.h>

@interface FoobarUtil : NSObject {
}
+(id)getMACUID;
+(id)getMACBasedUID;
+(id)hashDataToHexString:(char*)hexString length:(int)length;
+(id)hashData:(id)data;
+(id)iTunesMetadataPlist;
+(BOOL)appIsCracked;
+(BOOL)deviceIsJailbroken;
+(int)deviceCPUFrequency;
```

Jailbreak Detection

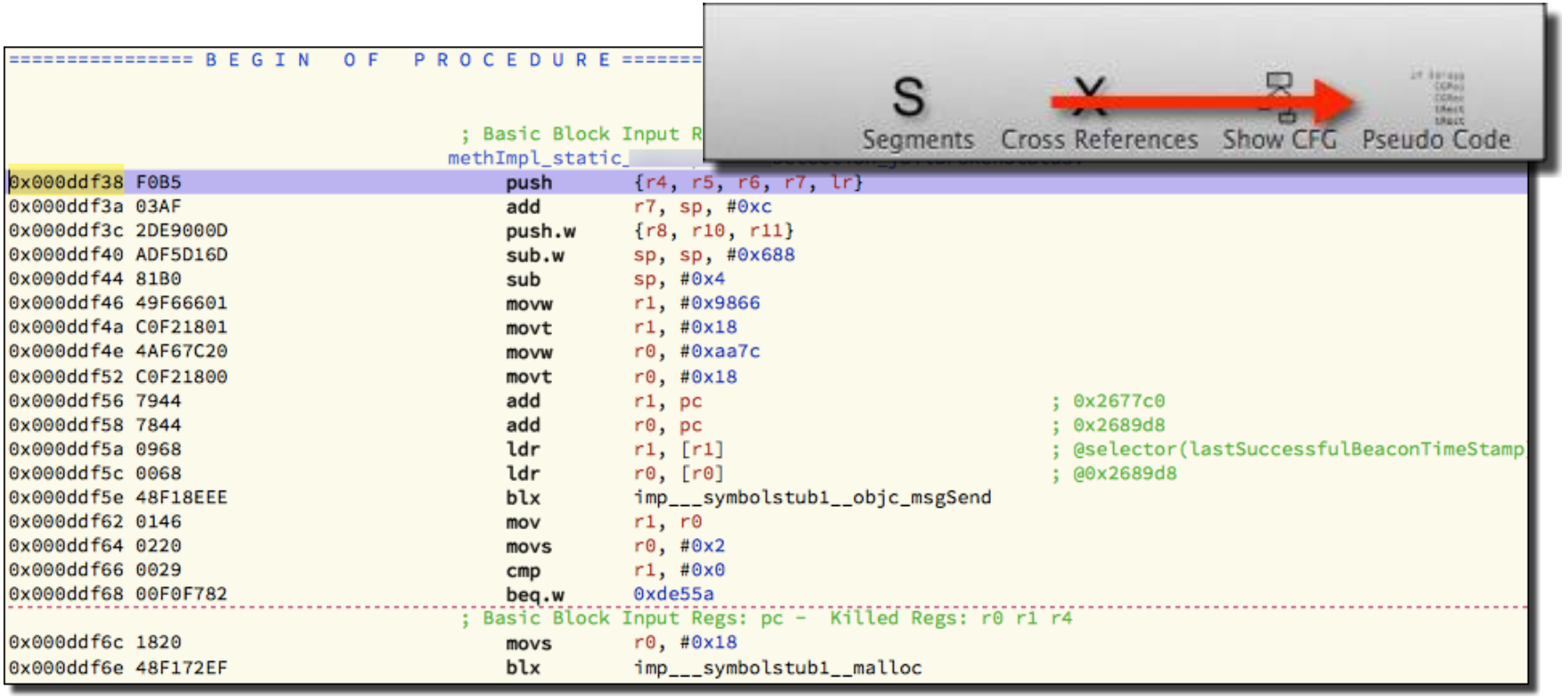
COMMON DETECTION METHODS

Methods

- Fork()
- Stat() / Lstat()
  - Cydia
  - /apt/
  - Etc

- dyld_count()
- dyld_get_image_name()

ARM Assembly



```
function methImpl_static_[redacted]Detection_jailBrokenStatus {
    r7 = &var_12;
    r13 = r13 - 0x688;
    r13 = r13 - 0x4;
    r0 = [*0x2689d8 lastSuccessfulBeaconTimeStamp];
    r1 = r0;
    r0 = 0x2;
    if (r1 == 0x0) goto loc_de55a;
    goto loc_ddf6c;

loc_de55a:
    Pop();
    Pop();
    Pop();
    return r0;

loc_ddf6c:
    r0 = malloc(0x18);
    do {
            *(int8_t *)(r4 + r0) = *(int8_t *)("\\xBA\\xD4\\xE5\\xE5
\xF1\\xFC\\xF4\\xBB\\xF4\\xE5\\xE5" + r0) ^ 0x95;
            r0 = r0 + 0x1;
    } while (r0 != 0x17);
    r0 = malloc(0xf);
    *(int8_t *)(r0 + 0xe) = 0x0;
```

```
function methImpl_static_[redacted]Detection_jailBrokenStatus {
    r7 = &var_12;
    r13 = r13 - 0x688;
    r13 = r13 - 0x4;
    r0 = [*0x2689d8 lastSuccessfulBeaconTimeStamp];
    r1 = r0;
    r0 = 0x2;
    if (r1 == 0x0) goto loc_de55a;
    goto loc_ddf6c;

loc_de55a:
    Pop();
    Pop();
    Pop();
    return r0;

loc_ddf6c:
    r0 = malloc(0x18);
    do {
            *(int8_t *)(r4 + r0) = *(int8_t *)("\\xBA\\xD4\\xE5\\xE5
\xF1\\xFC\\xF4\\xBB\\xF4\\xE5\\xE5" + r0) ^ 0x95;
            r0 = r0 + 0x1;
    } while (r0 != 0x17);
    r0 = malloc(0xf);
    *(int8_t *)(r0 + 0xe) = 0x0;
```

Pseudo Code

NO ASSEMBLY REQUIRED

```
    do {
                *(int8_t *)(r6 + r0) = *(int8_t *)("\\xBA\\xE3\\xF4\\xE7\\xBA
\xFA\\xF2" + r0) ^ 0x95;
                r0 = r0 + 0x1;
    } while (r0 != 0x12);
    r0 = stat(r4, &var_56);
    r8 = r0;
    r0 = stat(r5, &var_164);
    r10 = r0;
    r0 = stat(r6, &var_272);
    var_44 = r0;
    free(r4);
    free(r5);
    free(r6);
```

XOR is Not Obfuscation

```
    r0 = malloc(0x18);
    do {
            *(int8_t *)(r4 + r0) = *(int8_t *)("\\xBA\\xD4\\xE5\\xE5
\xF1\\xFC\\xF4\\xBB\\xF4\\xE5\\xE5" + r0) ^ 0x95;
            r0 = r0 + 0x1;
    } while (r0 != 0x17);
    r0 = malloc(0xf);
    *(int8_t *)(r0 + 0xe) = 0x0;
```

```
In [30]: out = ''

In [31]: for char in a:
    foo = int(repr(char)[3:-1], 16) ^ 0x95
    out += chr(foo)
   ....:

In [32]: print out
/Applications/Cydia.app
```

```
    Pop();
    Pop();
    Pop();
    return r0;

loc_ddf6c:
    r0 = malloc(0x18);
    do {
            *(int8_t *)(r4 + r0) = *(int8_t *)("\\
\xF9\\xFC\\xF6\\xF4\\xE1\\xFC\\xFA\\xFB\\xE6\\xBA\
\xF4\\xBB\\xF4\\xE5\\xE5" + r0) ^ 0x95;
```

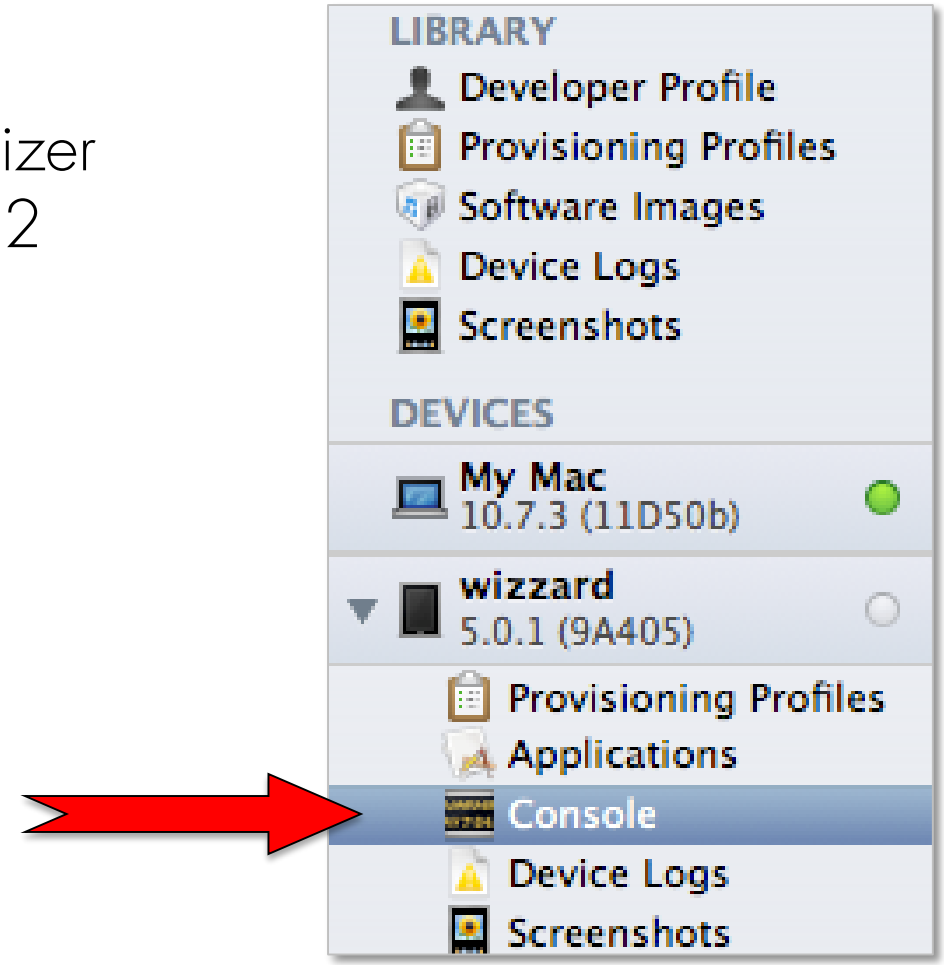
Dynamic Analysis

IOS APP SECURITY

iOS Device Logs

XCODE DEVICE CONSOLE

- Window > Organizer
  - Cmd + Shift + 2
- Real-time logs ☺



iOS Device Logs

WALL OF TEXT

```
Mobile Pro-iPad[23300] <Warning>: token success: c1bdb8fb722b718f53990c44a869eb1b1391695d32b5316848e7d3c47742c49c, 7
Mobile Pro-iPad[23300] <Warning>: setPushNotification returned:
Mobile Pro-iPad[23300] <Warning>: Copying persistent storage to local storage.
Mobile Pro-iPad[23300] <Warning>: Source File for Copy: /var/mobile/Applications/787961A9-FE78-4B00-B743-0DDB48BD9187/D
Mobile Pro-iPad[23300] <Warning>: Target File for Copy: /var/mobile/Applications/787961A9-FE78-4B00-B743-0DDB48BD9187/L
Mobile Pro-iPad[23300] <Warning>: Target successfully removed.
Mobile Pro-iPad[23300] <Warning>: Copy successful.
Mobile Pro-iPad[23300] <Warning>: setPushNotification returned:
Mobile Pro-iPad[23300] <Warning>: setIsNativeLoadingFinished(true) returned: true
Mobile Pro-iPad[23300] <Warning>: showApp returned:
```

Mobile Substrate

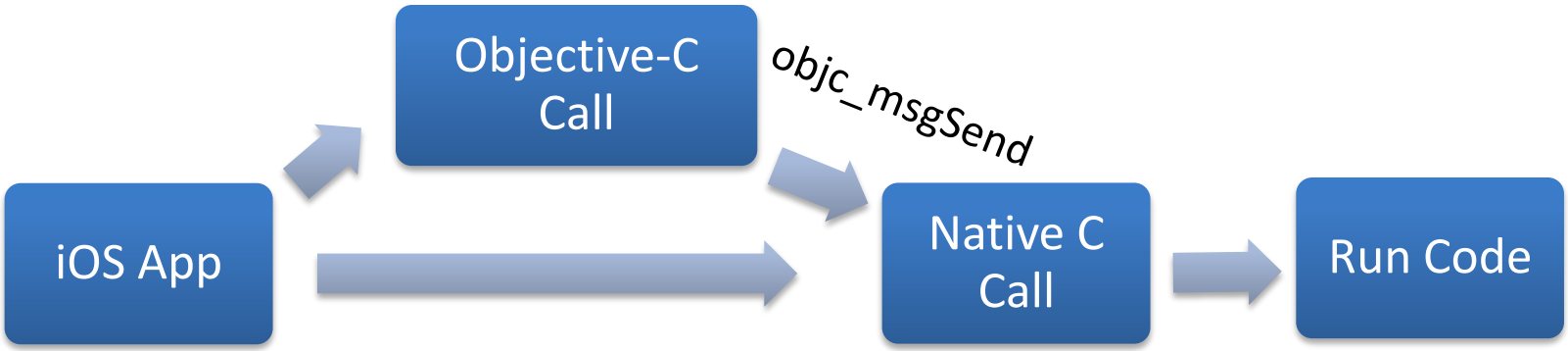
HOOKING MADE EASY

Mobile Substrate

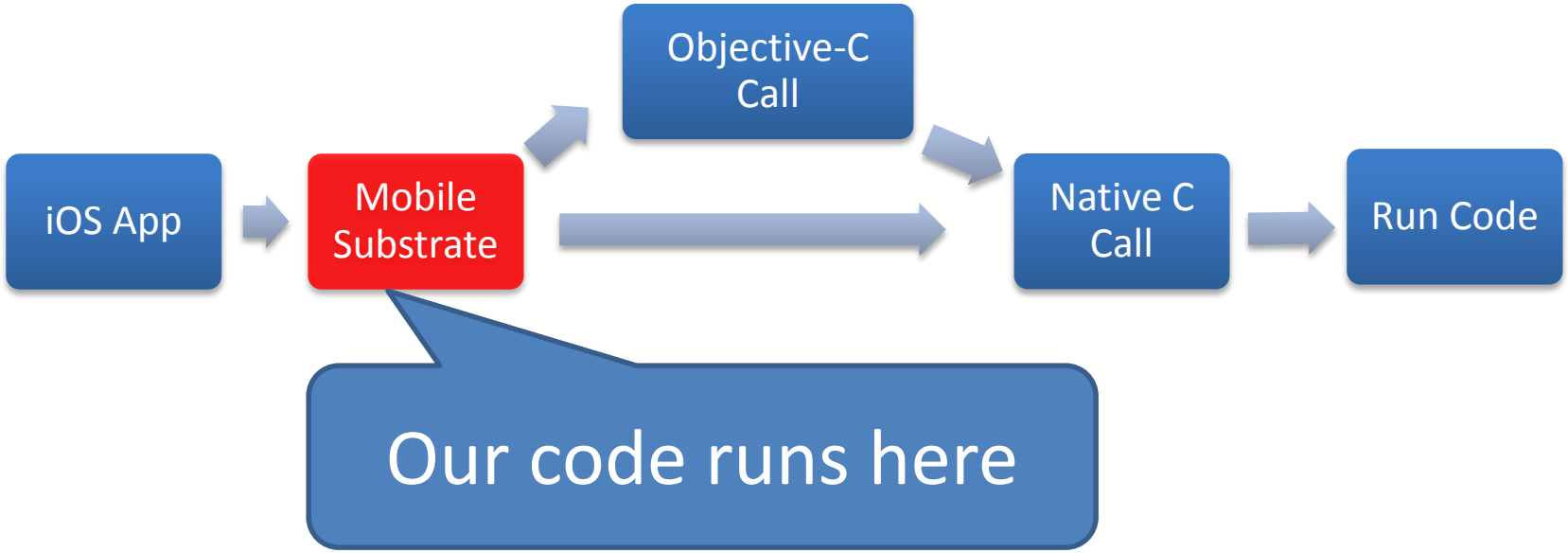
OBJ-C RUNTIME MANIPULATION

- Written by Jay Freeman "Saurik"
- Dynamic library injection framework
  - Cydia "Tweaks"
- Included with Cydia by default

Obj-C Message Passing



Obj-C Message Passing



Class Dump Example

IMPLEMENTING THE ATTACK

```
@class NSString;

@interface DeviceSecurity: {
        BOOL _jbstatus;
}
@property(assign, nonatomic) BOOL jbstatus;
+(BOOL)isJailbroken;
@end
```

Class Dump Example

IMPLEMENTING THE ATTACK

```
@class NSString;

@interface DeviceSecurity: {
        BOOL _jbstatus;
}
@property(assign, nonatomic) BOOL jbstatus;
+(BOOL)isJailbroken;
@end
```

Tweak Syntax

IMPLEMENTING THE ATTACK

```
#import “substrate.h”

%hook DeviceSecurity

-(BOOL) isJailbroken {

    %log;      // Logos built-in logging
    return NO; // Return FALSE

}

%end
```

Generating Function Hooks

LOG ALL THE THINGS

```
$ class-dump-z FoobarApp –H

$ ./ios-hooker.py --target Foobar.h –g –s –l

[*] Successfully parsed 1 of 1 file(s)
[*] Generated 120 function hook(s)
[*] Hooks written to: Tweak.xm (8954 bytes)

$ make package
```

Cycript

GIVES YOU SUPERPOWERS

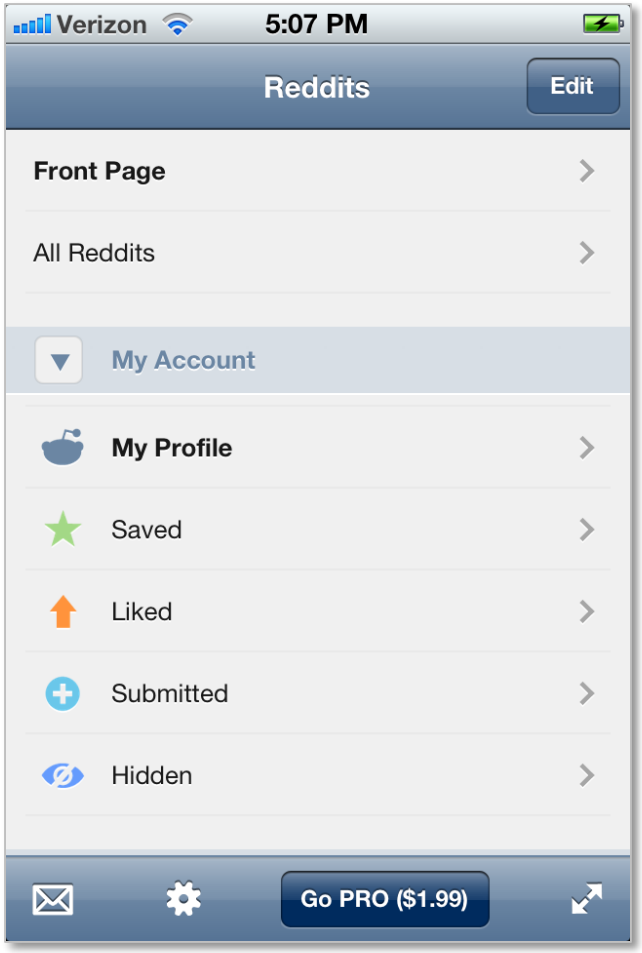
Cycript Magic

IOS HACKER'S BEST FRIEND

- JavaScript REPL
- JavaScript + Cycript language extensions
- Objective-C runtime is merged into the REPL
- Attach to running apps

Cycript

ALIEN BLUE APP



BISHOP FOX

Cycript

IOS HACKER'S BEST FRIEND

```
iphone:~root# cycript –p AlienBlue
cy#
cy# UIApp
@"<UIApplication: 0x8ba2c0>"
cy#
cy# UIApp.keyWindow.delegate
@"<CustomNavigationController: 0x836900>”
cy#
```

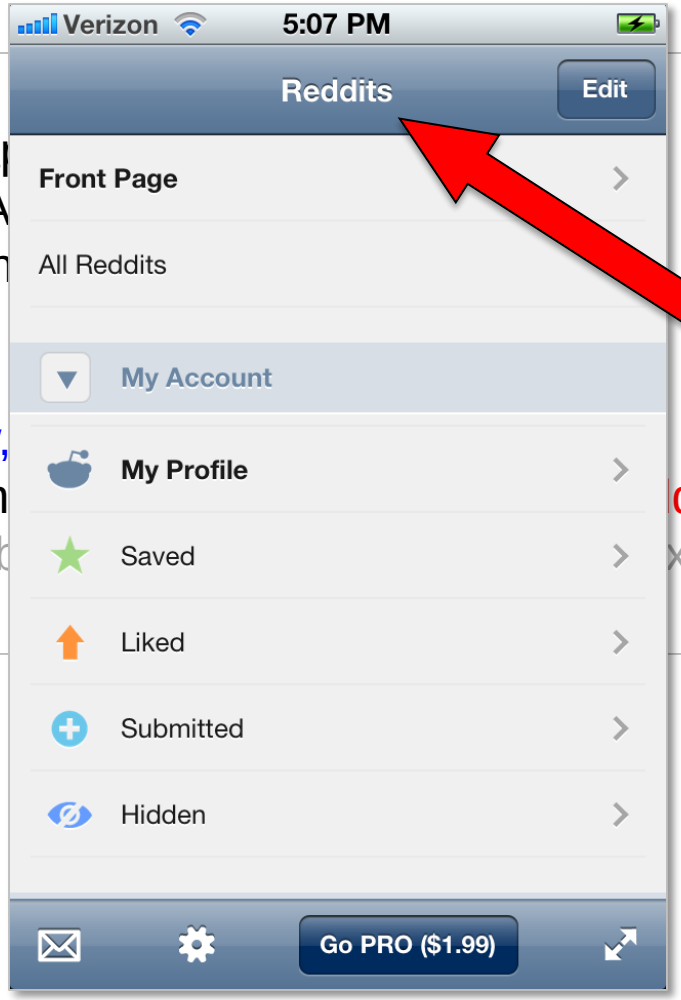
BISHOP FOX

Cycript Extended

IOS HACKER'S BEST FRIEND

```
$ ./slcycript AlienBlue
[+] Launching cycript wra
[+] Attaching to process A
[+] Importing JavaScript h

cy#
cy# ui(UIApp.keyWindow,
<UILabel: 0x82f0d0; fram                                    dits'; clipsToBounds =
YES; userInteractionEnab                                    x82f190>>
```

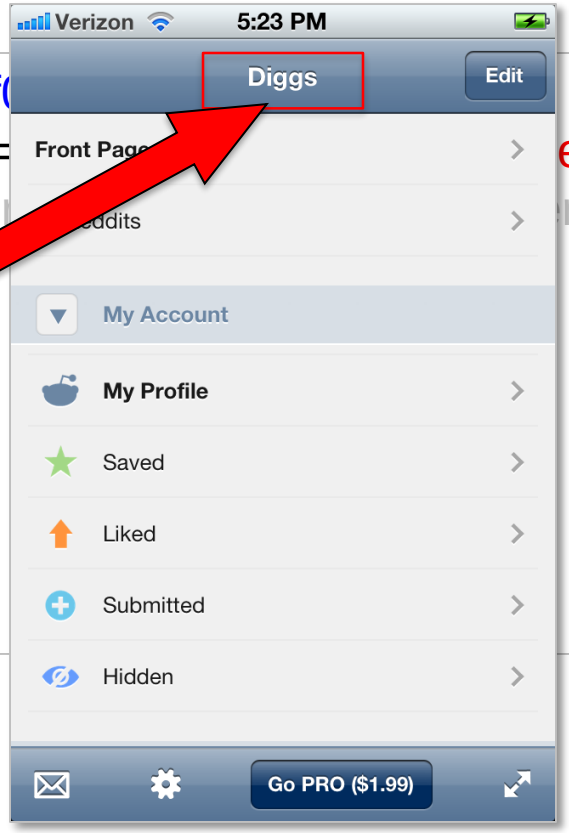


BISHOP FOX

Cycript Extended

IOS HACKER'S BEST FRIEND

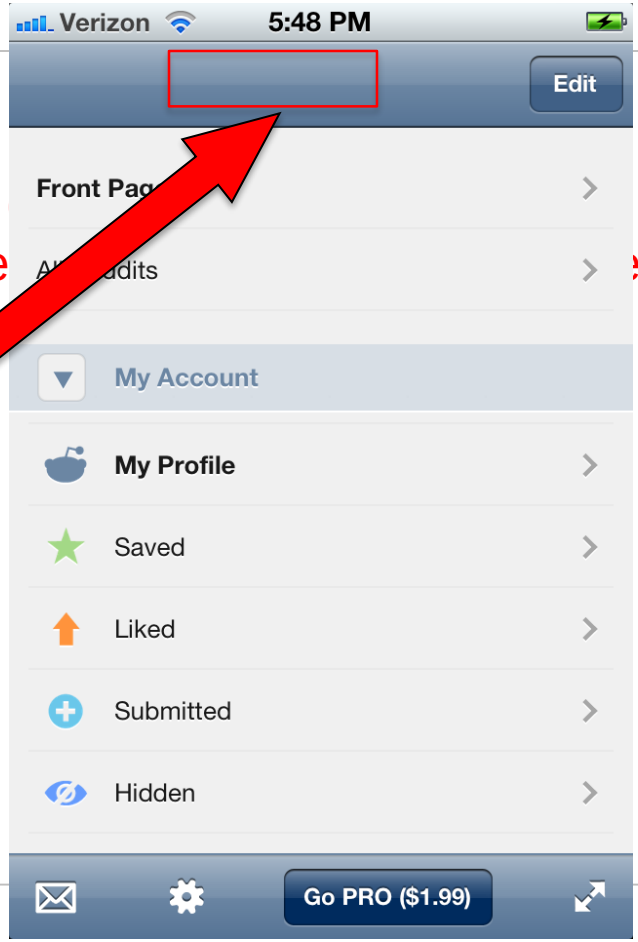
```
cy# label = new Instance(0x82f0
@"<UILabel: 0x82f0d0; frame = eddits';
clipsToBounds = YES; userInter r = <CALayer:
0x82f190>>"
cy#
cy# label.text
@"Reddits"
cy# label.text=@"Diggs"
@"Diggs"
cy#
```



Cycript Extended

IOS HACKER'S BEST FRIEND

```
cy# hexdump(label.text,64)
"
c8 02 20 3f 8c 07 00 06 05 44 69
12 48 65 6c 76 65 74 69 63 61 4e
6f 6c 64 00 00 00 00 00 00 00 00
c8 02 20 3f ad 07 00 01 b0 2b
"
cy#
cy# [ label setHidden: 1 ]
cy#
```



BISHOP FOX

MDM Security Policy

Cycript & iOS Mobile Substrate

DEMONSTRATION

Recommendations

REALISTIC CONSIDERATIONS

Defense Against Dark Arts

EXPECTO PATRONIS

- Mobile security can be defeated
- It comes down to context and difficulty
- For example…

iOS Recommendations

HARDENING YOUR APP

1. Assembly and/or C
2. Inline functions
3. Obj-C obfuscation
4. Certificate pinning
5. Change release

- Metaforic – Commercial
- AppMinder – BSD Licensed
  a) http://appminder.nesolabs.de/

Free ‘n Easy Obfuscation

DEFENSIVE SHELLCODE

```
#if !(TARGET_IPHONE_SIMULATOR)
  __attribute__ ((always_inline)) static void
MKJKqOBovPApokJQDajjRvdTEk (void)
{
        asm volatile ("b #4;pop {r0-r15};bx lr;mov r0, #63;mov r12, #256;asr r12,
#7;b #2;pop {r0-r15};sub r2, r2, r2;mov r0, r2;mov r0, r0;svc 0x80;mov r1, #1;b
#1;cmp r0, r10;cmp r0, r1;itt ne;movne r12, #1;swine 0x11;mov r3, r0;lsl r3, r3,
#3;add r3, r0;add r3, pc;bx r3;add sp, #120;ldmia sp, {r0-r15};bx lr;mov r3, r3;sub
r2, r2, r2;mov r0, r2;mov r12, #2;mov r2, r2;mov r0, #31;b #4;pop {r0-r15};bx lr;svc
0x80;sub r1, r1, r1;b #1;cmp r0, r10;mov r3, r1;b #2;push {r0-r12};add r3, r3, #1;b
#2;stmdb sp!, {r0-r12};cmp r0, r3;bne #1;b #4;mov r12, #1;swi 0x11;mov r2, #12;sub
r2, r2, r0;add r2, pc;bx r2;mov r0, #1;mov r12, #1;svc 0x80;" : : : "r0", "r1", "r2",
"r3", "r4", "r12", "cc", "memory");
}
#endif
```

STEP 2: Call the C function, where you would like to implement the jailbreak detection (e.g. in *didFinishLaunchingWithOptions:*):

```
#if !(TARGET_IPHONE_SIMULATOR)
MKJKqOBovPApokJQDajjRvdTEk ();
#endif
```

BISHOP FOX

Android Applications

STATIC ANALYSIS

Google Play Store

APK PACKAGES

Android Packages

EASILY ACQUIRED

- APKs are signed, not encrypted
- APK Extractor
- Direct Download

Android Perquisites

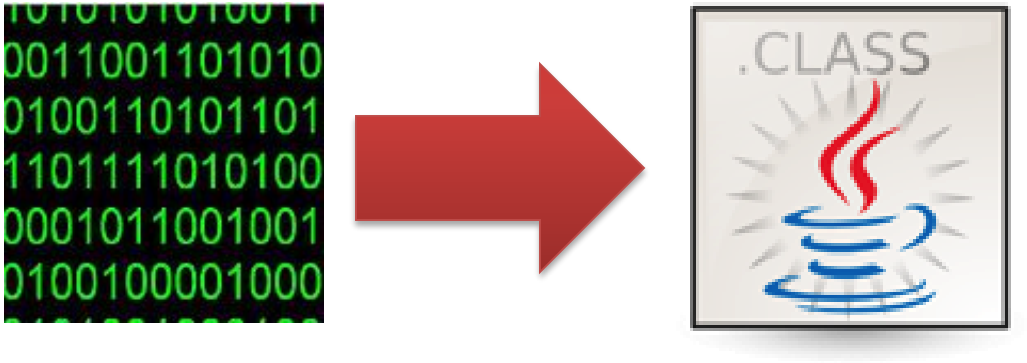
WHAT YOU NEED TO START

- Root'd Device
  - Cydia Substrate
- ADT Eclipse Bundle
  - Procyon
  - Dex2jar
  - Substrate Plug-in
- HTTP Proxy
  - Burp Suite Free/Pro ($300)
  - Zed Attack Proxy

Decompile the Bytecode

BYTECODE REFLECTION

dex2jar <App>.apk



Dex2jar & Procyon

MORE THAN JUST INTERFACES

```
$ dex2jar Foobar.apk

dex2jar foobar.apk -> Foobar-dex2jar.jar

$ procyon –jar Foobar-dex2jar.jar –o src/
Decompiling com/foobar/Parser...
Decompiling com/foobar/XMLWriter...
…
```

Decompiled Java Code

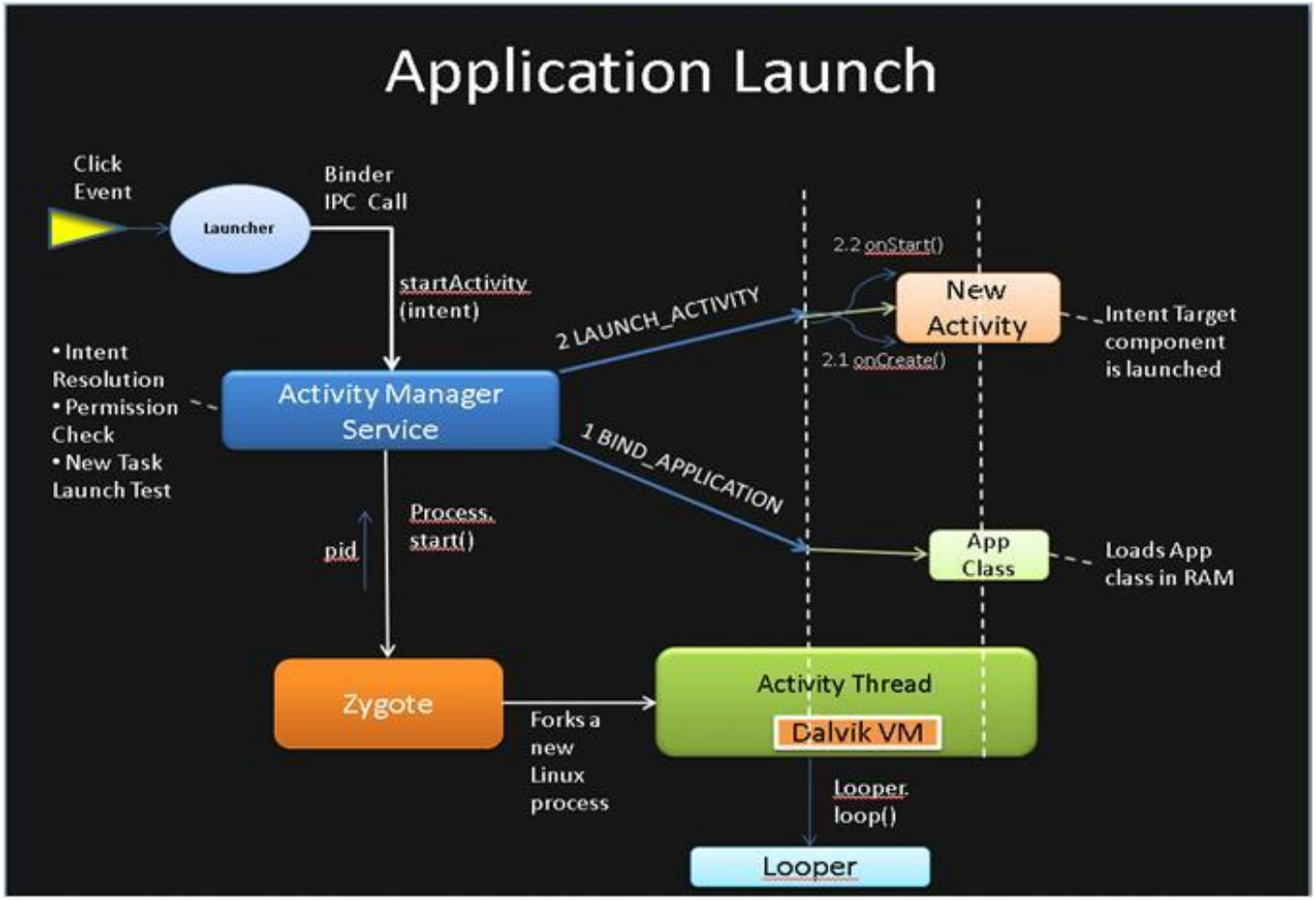
MORE THAN JUST INTERFACES

```
public MainActivity() {
    super();
    this.d = a.a();
    this.l = new IntentFilter("                                    .ACTION_DOWNLOA
    this.m = new IntentFilter("                                    .CHECK_THEME_UP
    this.n = new BroadcastReceiver() {
        public void onReceive(final Context context, final Intent intent)
                                                    f.c.a(n.confirm_restart_title, n.
                public void run() {
                    MainActivity.this.x();
                }
            }).show(MainActivity.this.getSupportFragmentManager(), "confi
        }
    };
```

Dynamic Analysis

ANDROID RUNTIME

Android Zygote



Cydia Substrate

SUBSTRATE FOR ANDROID

Class Hook Example

IMPLEMENTING THE ATTACK

Class to hook

```
/* Get the class we want to hook */
Class _class = Class.forName("android.net.http.AndroidHttpClient");

/* Get the method we want to hook, in this case it's the constructor */
Method execMethod = _class.getMethod(
        "execute", HttpUriRequest.class, HttpContext.class);
```

Method to hook

BISHOP FOX

Class Hook Example

IMPLEMENTING THE ATTACK

```
/* Our method alteration code */
MS.hookMethod(_class, execMethod, new MS.MethodAlteration() {
    public Object invoked(Object _this, Object... args) throws Throwable
    {
        /* Cast arguments into useful types */
        HttpUriRequest request = (HttpUriRequest) args[0];
        HttpContext context = null;
        if (args[1] != null) {
            context = (HttpContext) args[1];
        }

        /* Log pertinent information */
        Log.d("HttpInterceptor", "[" + request.getMethod() + "] -> " + r

        /* Call the original method */
        return invoke(_this, request, context);
    }
});
```

Class Hook Example

IMPLEMENTING THE ATTACK

```
/* Our method alteration code */
MS.hookMethod(_class, execMethod, new MS.MethodAlteration() {
    public Object invoke(Object _this, Object... args) throws Throwable
    {
        /* Cast arguments into useful types */
        HttpUriRequest request = (HttpUriRequest) args[0];
        HttpContext context = null;
        if (args[1] != null) {
            context = (HttpContext) args[1];
        }

        /* Log pertinent information */
        Log.d("HttpInterceptor", "[" + request.getMethod() + "] -> " + r

        /* Call the original method */
        return invoke(_this, request, context);
    }
});
```

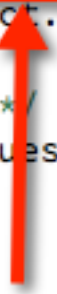
Class Hook Example

IMPLEMENTING THE ATTACK

```
/* Our method alteration code */
MS.hookMethod(_class, execMethod, new MS.MethodAlteration() {
    public Object invoked(Object _this, Object... args) throws Throwable
    {
        /* Cast arguments into useful types */
        HttpUriRequest request = (HttpUriRequest) args[0];
        HttpContext context = null;
        if (args[1] != null) {
            context = (HttpContext) args[1];
        }

        /* Log pertinent information */
        Log.d("HttpInterceptor", "[" + request.getMethod() + "] -> " + r

        /* Call the original method */
        return invoke(_this, request, context);
    }
});
```



BISHOP FOX

Class Hook Example

IMPLEMENTING THE ATTACK

```
/* Our method alteration code */
MS.hookMethod(_class, execMethod, new MS.MethodAlteration() {
    public Object invoked(Object _this, Object... args) throws Throwable
    {
        /* Cast arguments into useful types */
        HttpUriRequest request = (HttpUriRequest) args[0];
        HttpContext context = null;
        if (args[1] != null) {
            context = (HttpContext) args[1];
        }

        /* Log pertinent information */
        Log.d("HttpInterceptor", "[" + request.getMethod() + "] -> " + r

        /* Call the original method */
        return invoke(_this, request, context);
    }
});
```

Class Hook Example

IMPLEMENTING THE ATTACK

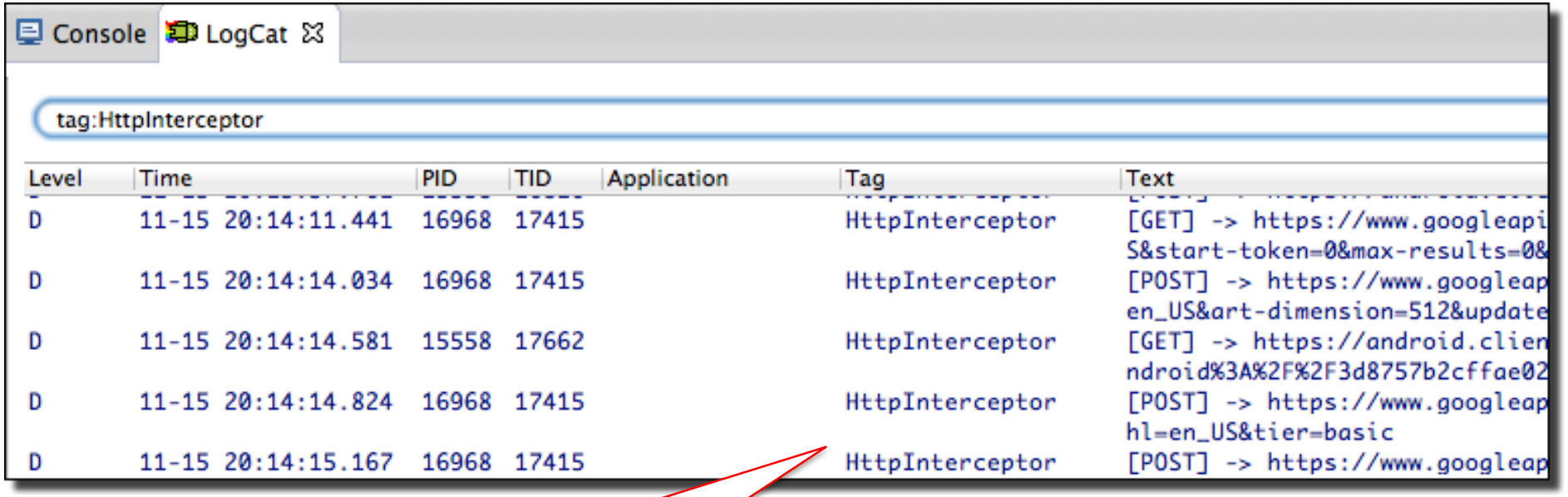
```
/* Our method alteration code */
MS.hookMethod(_class, execMethod, new MS.MethodAlteration() {
    public Object invoked(Object _this, Object... args) throws Throwable
    {
        /* Cast arguments into useful types */
        HttpUriRequest request = (HttpUriRequest) args[0];
        HttpContext context = null;
        if (args[1] != null) {
            context = (HttpContext) args[1];
        }

        /* Log pertinent information */
        Log.d("HttpInterceptor", "[" + request.getMethod() + "] -> " + r

        /* Call the original method */
        return invoke(_this, request, context);
    }
});
```

ADT's Logcat

ANOTHER BEAUTIFUL WALL OF TEXT



Filter by tag

BISHOP FOX

Android Substrate

DEMONSTRATION

Recommendations

REALISTIC CONSIDERATIONS

Android Recommendations

HARDENING YOUR APP

1. NDK + assembly and/or C
2. Inline functions
3. Avoid kernel calls if possible
4. Java bytecode obfuscation
5. Certificate pinning

- Java Obfuscator

Conclusions

ON MOBILE SECURITY

Management Not Security

BYOD VERTICAL CLOUD INTEGRATION

- MDM is Mobile Device *Management*
- Client-side enforcement
- Devices lie

Mobile Security

IN A NUTSHELL

- Do **NOT** make security decisions on a mobile device
- But if you must…

Client-side Enforcement

IF YOU MUST…

- Your architecture is probably broken
- Fix that instead
- But if the business model dictates that you must…
  - Perhaps the revenue model depends on it
  - Perhaps you have to integrate with legacy code
  - Perhaps there's some other crazy reason for on-device security

Questions & Thank You

@bishopfox

http://github.com/moloch--

Bishop Fox – see for more info:
http://www.bishopfox.com/

BISHOP FOX

Image Sources

- Binary Image - http://foter.com/f/photo/6949896929/4797170191/

BISHOP FOX