Pawn Storm: Malware on iOS devices?

Axelle Apvrille - FortiGuard Labs, Fortinet

Insomni'hack, Geneva
March 2015
Who am I?

- Security Researcher at Fortinet, FortiGuard Labs
- Focus: (strange) malware not on desktops/laptops
- E.g mobile malware, Internet of Things...
- Twitter: @cryptax
Are there malware on iOS?
Are there malware on iOS?

Answer: Yes

but not many
They’re all for jailbroken phones, aren’t they?
They’re all for jailbroken phones, aren’t they?

No - but very rare
iOS/FindCall (2012)
Found (and removed) in Apple Store

- Spams all your contacts
- Sends your (email/skype/...) passwords and location in clear text
Other samples for non jailbroken iPhones

Adware/LBTM!iPhoneOS (2010)

Nicolas Seriot, CH - 2009 - PoC
PoC Jekyll malicious app on non jailbroken iPhones

Credits: Tielei Wang, Kangjie Lu, Long Lu, Simon Chung, and Wenke Lee - Georgia Tech

"Jekyll on iOS: When Benign Apps Become Evil", USENIX Security 2013
Does PawnStorm run on non jailbroken iPhones?

Yes and No ;)

- Version A: will work, but with limits.
- Version B: requires jailbreak.
What’s Pawn Storm?

The Operation

- Cyber espionage operation
- Discovered by **Trend Micro** in October 2014
- Targets military officials, government, defense industries

iOS malware

- **Feb 4, 2015**: Trend Micro discovers two iOS samples
- **Version A**: XAgent - hidden trojan spyware
- **Version B**: madcap.dylib - malicious Cydia Substrate extension
What happens when iOS/PawnStorm.A!tr is launched?

iOS App Launch Sequence
Default project template in Xcode 4.2 (without Storyboarding)

Credits: Ole Begemann
What happens when iOS/PawnStorm.A!tr is launched?

iOS App Launch Sequence
Default project template in Xcode 4.2
(without Storyboarding)

main()
calls

UIApplicationMain()
creates

UIApplication
create
delegate

XAppDelegate
creates

UIApplicationDelegate
application:didFinishLaunchingWithOptions:

RunLoop

Credits: Ole Begemann
What happens when iOS/PawnStorm.A!tr is launched?

iOS App Launch Sequence
Default project template in Xcode 4.2
(without Storyboarding)

main() 1 calls
UIApplicationMain() 2 creates
UIApplication 3 creates
XAppDelegate 4 loads
Info.plist

application:didFinishLaunchingWithOptions:

UIApplication delegate

RunLoop

XViewController

Credits: Ole Begemann
didFinishLaunchingWithOptions: Background Fetching

```c
// XAAppDelegate - (char)application:(id) didFinishLaunchingWithOptions:(id)
char cdecl XAAppDelegate_application_didFinishLaunchingWithOptions__ (struct XAAppDelegate *self,
{
  id v4; // STOC_4@1
  int v11; // ST04_4@1
  int v14; // [sp+14h] [bp-18h]@1
  int v15; // [sp+18h] [bp-16h]@1
  SEL v16; // [sp+1]
  struct XAAppDelegate

  v17 = self;
  v16 = a2;
  v15 = 0;
  v4 = a4;
  objc_storeStrong(&v15, a3);
  v14 = 0;
  objc_storeStrong(&v14, v4);
  __objc_msgSend(&OBJC_CLASS__UIApplication, "sharedApplication");
  _R0 = objc_retainAutoreleasedReturnValue();
  _R3 = UIApplicationBackgroundFetchIntervalMinimum;
  __asm { VLDR  
            D16, [R3] }
  v11 = _R0;
  __asm { VMDV            R2, R3, D16 }
  __objc_msgSend(_R0, "setMinimumBackgroundFetchInterval:"");
  objc_release(v11);
  objc_storeStrong(&v14, 0);
  objc_storeStrong(&v15, 0);
  return 1;
}
```
Background Fetching and Multi-Tasking

**Background Fetching in Info.plist**

```
<key>UIBackgroundModes</key>
<array>
  <string>fetch</string>
</array>
```

**Multi-tasking**

- applicationWillResignActive
- applicationDidEnterBackground
- applicationDidEnterForeground
- applicationDidBecomeActive
- ...

---

**Insomni’Hack 2015 - A. Apvrille**
viewDidLoad in XAVViewController

Next method called viewDidLoad:

1. Instantiate **XA_HTTP_Chanel**: calls **getAgentID**. Retrieves a UUID.
   
   ```
   NSUUID *vendorIdentifier = [[UIDevice currentDevice] identifierForVendor];
   uuid_t uuid;
   [vendorIdentifier getUUIDBytes:uuid];
   ```

2. Creates a thread: **modulesThread**
   
   ```
   _thread = [[NSThread alloc] initWithTarget:self selector:@selector(modulesThread:) object:nil];
   ```

   modulesThread calls **cycleLoop** of **XAIInfolphone**
It’s a big switch

0: Get Info Device
1: Start Record
2: Get Audio File
3: Get Contact List
4: Current Location
5: Get Installed Apps
6: Wifi Status
7: Get All Pictures From Lib.
8: List a given directory
9: Get a given file
10: Get process list
11: Get SMS
Example: Get All Pictures from Photo Library

```
case 7:
    v265 = *(_DWORD *)(v591 + 400);
    v743 = -1;
    v389 = -1;
    v388 = (int)_objc_msgSend;
    v387 = &OBJC_IVAR__PhotoLibrary_http_param;
    v386 = (void)(__fastcall *)(int, int, int)_objc_msgSend;
    v385 = (int)se1Ref_dequeue_
      &OBJC_IVAR__XA_HTTP_Chanel_out_array;
    _objc_msgSend(v265, "getAllPicturePhotoLibrary");
    v266 = *((_DWORD *)(*(v591 + 400) + &v387) + &v384);
    v267 = *((_DWORD *)(*(v591 + 400) + &v387) + &v384);
    v268 = *((_DWORD *)(v385);
    v743 = v389;
    v386(v266, v268, v267);
    v743 = v389;
    v269 = objc_retainAutoreleasedReturnValue();
    objc_release(v269);
    break;
```

```
Getting Pictures from the Photo Library

From disassembly

```c
alasset_obj = &OBJC_CLASS___ALAssetsLibrary;
...
v18 = (void*alassetslib->library;
...
objc_msgSend(v18,
    "enumerateGroupsWithTypes:usingBlock:failureBlock:");
```

User authorization is not requested

```c
if ([ALAssetsLibrary authorizationStatus]) {
    //Library Access code goes here
}
```

nowhere to be seen in the malware’s code
Get GPS coordinates

Since iOS 8, an additional requestAlwaysAuthorization must be requested

Are location services enabled?

GPS accuracy is set to "best"

Report location for any movement (no distance filter)

Format GPS coordinates

Since iOS 8, an additional requestAlwaysAuthorization must be requested
Get Installed Applications

To get outside the sandbox → jailbreak

installation.plist file

.plist is expected to be found at ..../ from sandbox directory
Get Installed Applications - Objective C version

Pseudo Objective C code

```objective-c
static NSString *const cacheFileName = @"com.apple.mobile.installation.plist";
NSString *relativeCachePath = [@"Library"
    stringByAppendingPathComponent:
    @"Caches"] stringByAppendingPathComponent:
    cacheFileName];
path = [@NSHomeDirectory() stringByAppendingPathComponent:
    @"../.."] stringByAppendingPathComponent:
    relativeCachePath];
```
nsfileman_obj = &OBJC_CLASS___NSFileManager;
defaultMan = "defaultManager";
v105 = objc_msgSend(nsfileman_obj, defaultman);
...
v104 = objc_msgSend((void*)v202,
    "contentsOfDirectoryAtPath:error",
    path,
    &error);
...
v84 = objc_msgSend(&OBJC_CLASS___NSString,
    "stringWithFormat:",
    CFSTR("<table><caption color=blue> Directory: %@
            </caption></table>"),
    path);

Sandboxing limits to /private/var/mobile/Applications/THEAPP
Pseudo decompiled code in XAIErrorInfo `getInfoDevice`

```c
telephony_obj = objc_msgSend(
    &OBJC_CLASS___CTTelephonyNetworkInfo,
    "alloc");
...
v1223 = objc_msgSend(telephony_obj, "init");
...
subscriberProvider = "subscriberCellularProvider";
...
v1449 = objc_msgSend(v1153, subscriberProvider);
...
v1448 = objc_msgSend((void*)v9, "mobileNetworkCode");
...
v1447 = objc_msgSend(v12, "mobileCountryCode");
```
Get Phone Number

Later in XAIInfophsne getInfoDevice

Phone number is read from /private/var/wireless/Library/Preferences/com.apple.commcenter.plist

Out of sandbox → Requires jailbreak

Get the "PhoneNumber" key
Hiding the icon

The application icon does not appear on the home screen:

```xml
<key>SBApTags</key>
<array>
  <string>hidden</string>
</array>
```

Limitations

- Known not to work on iOS 8
- Hidden tag is easy to detect → Apple bans it from Apple Store
### Features of XAgent that require jailbreak

<table>
<thead>
<tr>
<th>Requires jailbreak</th>
</tr>
</thead>
<tbody>
<tr>
<td>▶ Read <strong>SMS</strong> database</td>
</tr>
<tr>
<td>▶ Read com.apple.commcenter.plist for <strong>phone number</strong></td>
</tr>
<tr>
<td>▶ <strong>Hiding</strong> icon</td>
</tr>
</tbody>
</table>

### Limited without jailbreak

<table>
<thead>
<tr>
<th>Limited without jailbreak</th>
</tr>
</thead>
<tbody>
<tr>
<td>▶ List content of <strong>directory</strong></td>
</tr>
<tr>
<td>▶ Retrieve <strong>file</strong></td>
</tr>
</tbody>
</table>
Features of XAgent that require user authorization

Malware does not ask these authorizations

Will not work (unless granted from elsewhere)?

- **Get Photos** from library. Should request `requestRecordPermission`
- **Geolocation**. Authorization needed for iOS 8.
- **Record voice**. Should request `requestRecordPermission`

Malware asks for these authorizations

**Read lists of contacts.** Code requires authorization via `ABAddressBookRequestAccessWithCompletion`
Features of XAgent which should work fine

- Get **model**, name, systemName, systemVersion, localizedModel via UIDevice
- Test for existence of **jailbreak** via /private/var/lib/apt
- List **running process** via call to Unix command `sysctl`
- **WifiStatus** via calls to Reachability API
- Background fetching of C&C URLs
- Take **screenshots**?? (not called)
Communication with C&C

iPhone infected
With Pawn Storm

hx xp://198.27XXXXXX/watch/?ai=<BASE 64 of RC4 data>

close/?
search/?
find/?
...
text=
from=
utm=

50 bytes key
Who wrote Pawn Storm?

Indications

- **BuildMachineOSBuild** 13E28: 10 possibilities: iMac, MacBook Pro, MacBook Air...
- /Users/mac/Desktop/work/IOS_PROJECT
- XAgent-azeuuhvvhelifolbyqbjqwuwimdho?
- **DTXcodeBuild** 5B1008: Xcode 5.1.1
- **Implementation**. Intended for jailbroken iOS 7.x?
- **Many typos**: XA_HTTP_Channel, runningProcecces, generateUrlParametrs
- **Grammar**: ”Host not exist”…
Am I safe from iOS/PawnStorm?

YES ... probably!

```python
def safeFromPawnStormA():
    if (military official or defense contractor): # etc
        if (iOS >= 8):
            print "Do you have an XAgent icon?"
        elif (iOS >= 7.1):
            if (jailbroken iPhone):
                print "Check your iPhone"
            else: # only parts of XAgent can work
                print "Probably safe, check to be sure"
        else:
            print "Probably ok"
    else:
        print "You’re not targeted, so probably safe"
```

Insomni’Hack 2015 - A. Apvrille 29/32
PawnStorm: so, what’s all this fuss? (and FUD?)

It’s a **TARGETED** malware!
Not for the *masses*

Quoting myself from Fortinet’s blog

”*it is unlikely you’ll be affected because the malware probably wasn’t distributed massively, but only to targeted victims*”
”*it is very unlikely the malware could have been on the Apple Store*”

Not my fault if anything concerning iOS gets more attention in the press, is it? ;P
PawnStorm: so, what’s all this fuss? (and FUD?)

It’s a **TARGETED** malware!
Not for the *masses*
*Its importance* depends … on the **target**!

Quoting myself from Fortinet’s blog

”*it is unlikely you’ll be affected because the malware probably wasn’t distributed massively, but only to targeted victims*

”*it is very unlikely the malware could have been on the Apple Store*”

Not my fault if anything concerning iOS gets more attention in the press, is it? ;P
Quick recap

Am I infected with Pawn Storm?

- Probably not
- Was Pawn Storm on the Apple Store? No
- What version is it for? iOS 7.1 and +
- Does it run on non jailbroken iPhones? Yes but with limits
- ... And on jailbroken iPhones? Yes !!!
- Can I spot it? On iOS 8, yes, otherwise difficult
- Who coded it? We don’t know
- Is iOS safe from malware? No !!!
- Is Android less safe than iOS? Perhaps. Difficult question
Am I infected with Pawn Storm? **Probably not**
Quick recap

Am I infected with Pawn Storm? Probably not

Was Pawn Storm on the Apple Store?

... And on jailbroken iPhones? Yes !!!

Can I spot it? On iOS 8, yes, otherwise difficult

Who coded it? We don’t know

Is iOS safe from malware? No !!!

Is Android less safe than iOS? Perhaps. Difficult question
Quick recap

Am I infected with Pawn Storm? **Probably not**
Was Pawn Storm on the Apple Store? **No**
Am I infected with Pawn Storm? **Probably not**

Was Pawn Storm on the Apple Store? **No**

What version is it for? **iOS 7.1 and +**

Does it run on non jailbroken iPhones? **Yes but with limits**

... And on jailbroken iPhones? **Yes !!!**

Can I spot it? **On iOS 8, yes, otherwise difficult**

Who coded it? **We don’t know**

Is iOS safe from malware? **No !!!**

Is Android less safe than iOS? **Perhaps. Difficult question**
Quick recap

Am I infected with Pawn Storm? Probably not

Was Pawn Storm on the Apple Store? No

What version is it for? iOS 7.1 and +

... And on jailbroken iPhones? Yes !!!

Can I spot it? On iOS 8, yes, otherwise difficult

Who coded it? We don’t know

Is iOS safe from malware? No !!!

Is Android less safe than iOS? Perhaps. Difficult question
Quick recap

Am I infected with Pawn Storm? Probably not
Was Pawn Storm on the Apple Store? No
What version is it for? iOS 7.1 and +
Does it run on non jailbroken iPhones?
Quick recap

Am I infected with Pawn Storm? **Probably not**
Was Pawn Storm on the Apple Store? **No**
What version is it for? **iOS 7.1 and +**
Does it *run* on non jailbroken iPhones? **Yes but with limits**
Quick recap

Am I infected with Pawn Storm? **Probably not**
Was Pawn Storm on the Apple Store? **No**
What version is it for? **iOS 7.1 and +**
Does it *run* on non jailbroken iPhones? **Yes but with limits**

... And on jailbroken iPhones?
Quick recap

Am I infected with Pawn Storm?  Probably not
Was Pawn Storm on the Apple Store?  No
What version is it for?  iOS 7.1 and +
Does it *run* on non jailbroken iPhones?  Yes but with limits
... And on jailbroken iPhones?  Yes !!!
Am I infected with Pawn Storm? Probably not
Was Pawn Storm on the Apple Store? No
What version is it for? iOS 7.1 and +
Does it run on non jailbroken iPhones? Yes but with limits
... And on jailbroken iPhones? Yes !!!
Can I spot it?
Am I infected with Pawn Storm? Probably not
Was Pawn Storm on the Apple Store? No
What version is it for? iOS 7.1 and +
Does it run on non jailbroken iPhones? Yes but with limits
... And on jailbroken iPhones? Yes !!!
Can I spot it? On iOS 8, yes, otherwise difficult
Quick recap

Am I infected with Pawn Storm? Probably not
Was Pawn Storm on the Apple Store? No
What version is it for? iOS 7.1 and +
Does it run on non jailbroken iPhones? Yes but with limits
... And on jailbroken iPhones? Yes !!!
Can I spot it? On iOS 8, yes, otherwise difficult
Who coded it?
Quick recap

Am I infected with Pawn Storm? Probably not
Was Pawn Storm on the Apple Store? No
What version is it for? iOS 7.1 and +
Does it run on non jailbroken iPhones? Yes but with limits
... And on jailbroken iPhones? Yes !!!
Can I spot it? On iOS 8, yes, otherwise difficult
Who coded it? We don’t know
Am I infected with Pawn Storm? Probably not
Was Pawn Storm on the Apple Store? No
What version is it for? iOS 7.1 and +
Does it run on non jailbroken iPhones? Yes but with limits
... And on jailbroken iPhones? Yes !!!
Can I spot it? On iOS 8, yes, otherwise difficult
Who coded it? We don’t know
Is iOS safe from malware?
Quick recap

Am I infected with Pawn Storm? Probably not
Was Pawn Storm on the Apple Store? No
What version is it for? iOS 7.1 and +
Does it run on non jailbroken iPhones? Yes but with limits
... And on jailbroken iPhones? Yes !!!
Can I spot it? On iOS 8, yes, otherwise difficult
Who coded it? We don’t know
Is iOS safe from malware? No !!!
Am I infected with Pawn Storm? Probably not
Was Pawn Storm on the Apple Store? No
What version is it for? iOS 7.1 and +
Does it run on non jailbroken iPhones? Yes but with limits
... And on jailbroken iPhones? Yes !!!
Can I spot it? On iOS 8, yes, otherwise difficult
Who coded it? We don’t know
Is iOS safe from malware? No !!!
Is Android less safe than iOS?
Am I infected with Pawn Storm? Probably not

Was Pawn Storm on the Apple Store? No

What version is it for? iOS 7.1 and +

Does it run on non jailbroken iPhones? Yes but with limits

... And on jailbroken iPhones? Yes !!!

Can I spot it? On iOS 8, yes, otherwise difficult

Who coded it? We don’t know

Is iOS safe from malware? No !!!

Is Android less safe than iOS? Perhaps. Difficult question
Thank You!

Contact info

@cryptax or aapvrille (at) fortinet (dot) com

References and interesting links

▶ Blog post from Trend Micro
▶ Blog post on Fortinet
▶ Wang et al, Jekyll on iOS, USENIX Security 2013
▶ C. Livitt, Rethinking & Repackaging iOS Apps: Part 1, Feb 2015
▶ Zheng et al, Enpublic Apps: Security Threats Using iOS Enterprise and Developer Certificates, ASIA CCS 2015

Thanks to: Claud Xiao, Ruchna Nigam, Nicolas Seriot, Trend Micro

PowerPoint? No way! This is Lobster