Enhancing Mobile Malware: an Android RAT Case Study

BSIDES VIENNA 2014
November 22
About

Marco Lancini
Security Consultant, CEFRIEL
@lancinimarco

Roberto Puricelli
Security Consultant, CEFRIEL
@robywankenoby
Introduction
Demonstrate how it is possible to **easily create powerful malware**, combining public available attack toolkits and exploits of known vulnerabilities

**GOAL**

Given the source code of a mobile RAT, it is possible to **extend its features**, adapting and modifying its behavior (hiding malicious features, adding exploits)

**HOW**

**AndroRAT++**, a proof-of-concept mobile malware, embedded in a legitimate application, that enhances the features of a well-known RAT application

**POC**
Mobile malware evolution
Mobile malware is a (relatively) new trend
• Actually almost 10 years of samples

DroidDream

- Infected 60 different legitimate apps in the Android Market
- Breached the Android security sandbox, installed additional software, and stole data
- Created a botnet
• A.k.a. **Eurograbber**
• Widespread in Europe
• Bypass 2FA (SMS OTP)
• **36M €** stolen
Why Android is the most targeted platform?

• Wide-spread
• “Open” philosophy
• Lacks of controls

Social engineering plays a big role in the exploit

- By installing a trojan app that perform unauthorized operations
- The malware is “embedded in the app”

Renowned for not making controls over published applications
Used to spread malicious applications disguised as famous ones
What can an attacker do?

**Malicious Activity**
- Add new features
- Edit configurations
- Install new apps
- Launch DDoS attacks
- Click fraud

**Surveillance**
- SMS
- Call logs
- Audio
- Camera
- Location

**Impersonation**
- SMS redirection
- Send emails
- Post to social media

**Data Theft**
- Stored files
- Account details
- Contacts
- Call logs
- Phone number
- IMEI

**Financial**
- Send premium rate SMS
- Steal transaction auth numbers (TANs)
- Extortion via ransomware
- Fake antivirus

How to build a powerful malware?
What’s new in Android Malware?

**HIGHLIGHTS THIS QUARTER**

**ON ANDROID**
With 99% of the new threats that emerged in Q1 2014 designed to run on the Android operating system (OS), it’s not surprising the most interesting mobile malware technical developments involved this platform. Here are a few noteworthy advances seen in Android malware in the last few months:

**WINDOWS TROJAN HOPS ON ANDROID**
A banking-trojan named Drebin that targets Windows PCs also tries to install a mobile banking-trojan on any Android devices connected via USB to the infected machine. Depending on the variant, we detect the mobile banking-trojan used as Trojan-Spy.Android/Simfor.H or Trojan.Android/Cepew.A or .B.

**FIRST CRYPTOMINER**
Trojan.Android/CoinMiner.A is distributed in a repackaged application. When installed, it essentially hijacks the device to silently mine virtual currency (such as Litecoin) for the malware author. Apart from any data changes incurred, the constant use of the device’s hardware may also affect its battery life and eventual lifespan.

**FIRST BOOTKIT**
Trojan.Android/Okbook.A is believed to be Android’s first bootkit, or malware that affects the earliest stages of the device’s bootup routine, making it extremely difficult to detect or remove. The malware is thought to have spread in modified firmware updates, with most infections reportedly seen in China.

**PILEUP EXPLOIT**
Researchers reported vulnerabilities in the Android OS that could allow an installed malware to silently upgrade its permissions during a system update, and named an exploit of this loophole Pileup (as in, “privilege escalation through updating”).

**DENDROID TOOLKIT**
Backdoor.Android/Dendroid.A is a toolkit for creating Remote Access Trojans (RAT) that allow an attacker to create trojans that can remotely access an infected device’s audio and video functions. It also creates trojans that can evade Google Play Store security.

Remote Access Trojan? Interesting, let’s Google it…
Remote Access Trojan

I’m feeling lucky...
• First result gave us a possible trojan name

AndroRAT
• Open source proof of concept
• Powerful features
• “Easy like Sunday Morning”!!!!

Ok, we just need to find the code…
• Let’s try GitHub
AndroRAT Source Code

Still lucky…

- Lots of different working versions
AndroRAT

How it works

- **Java “server”** application
- **Android service** on the phone

The application itself is not so attractive

- We can embed it into another one, it’s easy
- A game, or another app could be effective for our target

If we could just **exploit the certificate validation** in Android..
Injection of malicious code

2004

Cabinet
First worm affecting Symbian Series 60 phones. Spreads from phone to phone by using Bluetooth OBEX push protocol.

2009

Ikee and Duh
Worms affecting jailbroken iPhones using Cydia app distribution system due to a hardcoded password in sshd.

2010

FakePlayer
First malware for Android makes money by sending SMS messages to premium line numbers in Russia.

2011

DroidDream
First large attack to Google Play market. Over 50 apps containing a root exploit published to Android Market.

2012

Zitmo
Popular Windows bot and banking malware Zeus improved with its Android component designed to steal banking mTANs.

2013

Masterkey
A vulnerability in Android discovered exploiting certificate validation in Android which allows malware to disguise as a legitimate app.

2014

DownAPK
Windows based malware uses Android debugging bridge to install fake banking app to Android devices connected to the infected PC.

If we could just **exploit the certificate validation** in Android..
Android Master Key Vulnerability

- Allows to: "**modify APK code without breaking an application's cryptographic signature**, to turn any legitimate application into a malicious Trojan, completely unnoticed by the app store, the phone, or the end user"
- Android can be tricked into **believing the app is unchanged** even if it has been
- Corrected with Android 4.4

• Let’s embed our RAT into a benign application

• The purpose here is to **simulate** the attack, not to do it for real.
  • **AndroRAT** has been injected into a *fake* application of BSides
    • Not available in any store 😊
  • New features were added (**AndroRAT**++)
DEMO Scenario
Scenario

1. *Installation* of a malicious APK
Scenario

1. *Installation* of a malicious APK
2. *Remote control* of the phone

Androrat + some configurations
Scenario

1. **Installation** of a malicious APK
2. **Remote control** of the phone
3. Leverage the *botnet*

Add some coding
• Bulk actions allow to execute a command **on all the controlled devices**

• If the attacker compromises a large number of devices, a **botnet** is created

• The resources of infected devices could be used to carry out attacks on third-party services
1. **Installation** of a malicious APK
2. **Remote control** of the phone
3. Leverage the *botnet* (DoS attacks)
4. Privilege **escalation**

We need more… root power!
…but how? Let’s find an easy way
Privilege escalation

I’m feeling lucky (AGAIN!!!!)...
- First result gave us an application that can easily root an Android phone

**Framaroot**
- Not open source, but we can get the APK from XDA
- One-click root
- Works from Android 2.0 to 4.2…good enough!

• We can also embed the exploits used by Framaroot within the RAT application....

• The embedded version is "silent"

• The attacker can root the devices remotely

```java
p = Runtime.getRuntime().exec("su");
DataOutputStream os = new DataOutputStream(p.getOutputStream());

os.writeBytes(cmd + "\n");
os.flush();

os.writeBytes("exit\n");
os.flush();

exitCode = p.waitFor();
```

We can now execute system commands from within our code
Scenario

1. **Installation** of a malicious APK
2. **Remote control** of the phone
3. Leverage the **botnet** (DoS attacks)
4. Privilege **escalation**
5. **Exfiltration** of sensitive data

Add some more code…
Scenario

1. Installation of a malicious APK
2. Remote control of the phone
3. Leverage the botnet (DoS attacks)
4. Privilege escalation
5. Exfiltration of sensitive data
6. Silent installation of new applications

Still some code…
Which application to install?

I just have to choose the application…
- The purpose is always to make money

FIRST CRYPTOMINER

Trojan:Android/CoinMiner.A is distributed in a repackaged application. When installed, it essentially hijacks the device to silently mine virtual currency (such as Litecoin) for the malware author. Apart from any data charges incurred, the constant use of the device’s hardware may also affect its battery life and eventual lifespan.

```java
DataOutputStream os = new DataOutputStream(p.getOutputStream());
os.writeBytes("su -c pm install -r " + appname + "\n");
os.flush();
```
1. *Installation* of a malicious APK
2. *Remote control* of the phone
3. Leverage the *botnet* (DoS attacks)
4. Privilege *escalation*
5. *Exfiltration* of sensitive data
6. *Silent installation* of new applications
7. *Interception* of communications
ProxyDroid

- Used to set the proxy (HTTP/SOCKS4/SOCKS5) on Android devices
- The app has been modified
  - The GUI has been stripped entirely
  - When launched, sets the proxy and exit
  - The app is installed and run automatically
Conclusions
Maybe it’s just a bit of luck, but we demonstrated that it’s easy to create a powerful Android-based malware…

What we did

Take an app
Add malware
Make it bad
THANK YOU!

Marco Lancini  
Security Consultant, CEFRIEL  
@lancinimarco

Roberto Puricelli  
Security Consultant, CEFRIEL  
@robywankenoby