Mobile Threats Incident Handling

Yonas Leguesse
Disclaimer

References made herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by ENISA.

The references to material used are in the notes section.
Agenda

1 About us
2 Incident Handling Process
3 Case Study – Mobile Ransomware
About us

ENISA: European Union Agency For Network and Information Security

Operational Office in Athens

Seat in Heraklion
Positioning ENISA activities

Recommendations

Policy Implementation

Mobilising Communities

Hands on
Computer Security Incident Response Team (CSIRT) – (CERT)

“When an incidents occurs, the goal of the CSIRT is to control and minimize any damage, preserve evidence, provide quick and efficient recovery, prevent similar future events, and gain insight into threats against the organization”
National/governmental CSIRTs
the situation has changed...

ESTABLISHED IN 2005:

Finland
France
Germany
Hungary
The Netherlands
Norway
Sweden
United Kingdom

SITUATION IN 2015:

Armenia
Austria
Belgium
Bulgaria
Croatia
Czech Republic
Denmark
Estonia
Finland
France
Georgia
Germany
Greece
Hungary
Iceland
Ireland
Israel
Italy
Latvia
Lithuania
Luxembourg
Malta
Netherlands
Norway
Poland
Portugal
Romania
Slovakia
Slovenia
Spain
Sweden
Switzerland
Turkey
Ukraine
United Kingdom
EU Institutions

We are actively supporting a growing network of national/governmental CSIRTs

Tier 1: Good Practice Guides for CSIRTs

https://www.enisa.europa.eu/activities/cert/support
Tier 2: ENISA Training Resources

Over 30 training materials covering different topics like:

- **Setting up a CERT:**
  - Recruitment of staff
  - Developing infrastructure
  - Triage and Basic Incident Handling

- **Technical & operational:**
  - Advisories
  - Network & system forensics
  - Proactive detection of security incidents
  - APT
  - Mobile threats

- **Legal & cooperation:**
  - Assessing CERT communication channels
  - Cyber crime traces
  - Cooperation with law enforcement

https://www.enisa.europa.eu/activities/cert/training/training-resources/resources
Tier 3: Training for national / governmental CSIRTs

- ........
- Triage and Basic incident Handling
- Mobile threats incident handling
- ........
Incident Handling Process

Incident Handling Process

Good practice is to start with the simple model develop the procedure as you gain experience.

Considerations:

- Available resources
- Number of incidents
- Sensitivity of incidents
- .....
**Artifact analysis process chart**

<table>
<thead>
<tr>
<th>Triage</th>
<th>Automation needed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Artifact comes in (mail, URL, ticketing system, honeypot, autoscan).</td>
<td></td>
</tr>
<tr>
<td>Logging and storing the artifact (downloading, format conversion, automatic unpacking if possible, hashing).</td>
<td></td>
</tr>
<tr>
<td>Identifying the artifact (hash lookups, signature checks, artifact metadata, community shared information sources, IOC checks).</td>
<td></td>
</tr>
<tr>
<td>Artifact is submitted for automated analysis (sandbox analysis).</td>
<td></td>
</tr>
<tr>
<td>Analysis results are created, stored, updated and correlated.</td>
<td></td>
</tr>
<tr>
<td>Decision to proceed towards next step is taken and artifact is submitted for further analysis.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Manual analysis</th>
<th>Skilled analyser needed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Artifact is checked for obfuscation and deobfuscated if possible and necessary.</td>
<td></td>
</tr>
<tr>
<td>Artifact is analysed (reverse engineered) in debugger, or disassembler to identify timers, triggers, debugging and sandboxing evasion techniques. Based on findings custom changes may be implemented to automated analysis system, and decision to proceed towards next step is taken.</td>
<td></td>
</tr>
<tr>
<td>Modifying artifact code to reveal possible hidden functionality</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Communication and writing skills needed</th>
<th>Communication skills needed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Updating analysis results and indicators of compromise (as an optional step custom report can be created)</td>
<td></td>
</tr>
<tr>
<td>If possible initiating information sharing process (can be automated)</td>
<td></td>
</tr>
</tbody>
</table>

**Tools and Software**

- Debuggers: Ollydbg, Radare2, Immunity DBG, X64DBG, IDA Free
- Memory Dumpers: LordPE, OllyDump
- .Net deobfuscators: de4dot, ILSpy
- Packer Detection: Detect It Easy, Peld, Exeinfo PE, PEView, PE Tools

**MISP, CRITs**

- MISP, CRITs
- Virtualbox, Cuckoo, Volatility
- MISP, CRITs
Further develop a list of guidance or advice notes for an incident handlers.

Alternatively, develop a more advanced workflow with graphical representation of decision trees.
Incident Handling Workflow

Waves of particular types of incidents allows you to develop an effective workflow.
Case Study

- Based on real incident
- Names have been changed for this demo.
ACME Inc. IT Department

Hello IT
Have you tried turning it off and on again?
Incident Report

Incident Report
Director’s device is locked with Ransomware Message

- Files are Encrypted
- Device is Unusable
- Asking For payment of $10,000 in Bitcoins

Incident Report

Dear ACME Director

Your data has been Encrypted. If you want it back, send us $10,000 to Bitcoin wallet:
1F1tAaz5xxxxxxxxxxxxx
xxxxxxxxxxx
Registration and Triage

Registration

- Ticketing system
- Ticket ID
- Keywords
- Date
- ...

Triage

A French medical term - describes a situation in which you have limited resources and have to decide on the priorities of your actions based on the severity of particular cases.

Is this incident for us?

Team member  Classification/Priority
John Smith      Malware(Ransomware)/High
### Incident Report

#### ACME Ransomware report

<table>
<thead>
<tr>
<th>Ticket ID</th>
<th>123456</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Severity</strong></td>
<td>High</td>
</tr>
<tr>
<td><strong>Assignee</strong></td>
<td>John Smith (Mobile Expert)</td>
</tr>
<tr>
<td><strong>Incident Type</strong></td>
<td>Malware - Ransomware</td>
</tr>
<tr>
<td><strong>Trigger</strong></td>
<td>ACME Director reported in person that his Mobile Device is blocked and prompted with a ransom request indicating that his data has been encrypted and will only be decrypted if the money is paid to Bitcoin Wallet: 1F1tAaz5xxxxxxxxxxxxxxxxxxxxxxxxx</td>
</tr>
</tbody>
</table>

---

![Image of ransomware email]

**Dear ACME Director**

Your data has been Encrypted. If you want it back, send us $10,000 to Bitcoin wallet: [Cryptocurrency Address】
Incident Resolving

Incident resolution

- Longest phase
- Leads to solution (hopefully 😊)
- It is a cycle
Data Analysis & Resolution Research

- Identify **stakeholders** with useful data/evidence
- Notify them
- Ask them for the data/evidence
Director remembers receiving this email before the incident.

Dear Director,

We have rolled out the following software for your company smartphone. Please proceed to downloading and installing the application:

http://www.falseacmesite.notcom/youreallyshouldntdownloadthis.apk

Best Regards,
Fake ACME Staff

Ask for Mobile Model, OS version, etc.
## Incident Report

### Data Analysis

<table>
<thead>
<tr>
<th>Method of infection</th>
<th>Director said that prior to the incident, he received an email suggesting that he installs new company software</th>
</tr>
</thead>
<tbody>
<tr>
<td>Device</td>
<td>Nexus 5</td>
</tr>
<tr>
<td>OS version</td>
<td>Android 4.4 (Kit Kat)</td>
</tr>
</tbody>
</table>
Resolution Research

- We will not pay ransom
- We have malware to analyse
  - understand the behaviour
  - reverse the operations
  - restore the device
- Isolate device
Static-Dynamic-Automated tools

Static (Code) analysis
Dynamic (Behavioural) analysis

Hybrid Automated tools
What Next?

1. Automated hybrid Analysis
2. Dynamic Analysis
3. Static Analysis
4. Eradication & Recovery
Automated Hybrid Analysis

1. Online tool: SandDroid

- Andrubis, SandDroid, TraceDroid, Mobile Sandbox ....
- Custom tools
- ....

Disclaimer: Use with caution, especially in targeted attacks!
## Automated Analysis

### General Information

<table>
<thead>
<tr>
<th>Description</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Analysis Start Time</td>
<td>2014-10-08 15:04:16</td>
</tr>
<tr>
<td>Analysis End Time</td>
<td>2014-10-08 15:05:43</td>
</tr>
<tr>
<td>File MD5</td>
<td>FD694CF5CA1DD4967AD6E8C67241114C</td>
</tr>
<tr>
<td>File Size</td>
<td>4.69 MB</td>
</tr>
<tr>
<td>File Name</td>
<td>FD694CF5CA1DD4967AD6E8C67241114C.apk</td>
</tr>
<tr>
<td>Package Name</td>
<td>org.simplelocker</td>
</tr>
<tr>
<td>Version Code</td>
<td>1</td>
</tr>
<tr>
<td>Version Name</td>
<td>1.0</td>
</tr>
<tr>
<td>Min SDK</td>
<td>9</td>
</tr>
<tr>
<td>Target SDK</td>
<td>17</td>
</tr>
<tr>
<td>Max SDK</td>
<td>N/A</td>
</tr>
<tr>
<td>Pcap File</td>
<td><img src="" alt="Pcap File" /></td>
</tr>
<tr>
<td>Logcat File</td>
<td><img src="" alt="Logcat File" /></td>
</tr>
</tbody>
</table>

### Risk Score

100

### Risky Behaviors
### Risky Behaviors

- Encrypt or Decrypt data
- Executes shell code
- Exist unused permissions
- Gets the unique device ID, IMEI for GSM and MEID for ESN or ESN for CDMA phones
- Utilizes Java reflection

### Malware Detected by VirusTotal

<table>
<thead>
<tr>
<th>Antivirus</th>
<th>Malware</th>
</tr>
</thead>
<tbody>
<tr>
<td>AVG</td>
<td>Android/Locker.A</td>
</tr>
<tr>
<td>Ad-Aware</td>
<td>Android.Trojan.SPlocker.A</td>
</tr>
<tr>
<td>Baidu-International</td>
<td></td>
</tr>
<tr>
<td>BitDefender</td>
<td>Android.Trojan.SPlocker.A</td>
</tr>
<tr>
<td>ESET-NOD32</td>
<td>Android/Simplocker.A</td>
</tr>
<tr>
<td>F-Secure</td>
<td>Trojan.Android/SPlocker.A</td>
</tr>
<tr>
<td>Fortinet</td>
<td>Android/Pletor.Atr</td>
</tr>
<tr>
<td>Kaspersky</td>
<td>HEUR.Trojan-Ransom.AndroidOS.Pletor.a</td>
</tr>
<tr>
<td>McAfee</td>
<td>ArtemisFD694CF5CA1D</td>
</tr>
<tr>
<td>Qihoo-360</td>
<td>Trojan.Generic</td>
</tr>
<tr>
<td>Symantec</td>
<td>Android.Simplocker</td>
</tr>
</tbody>
</table>
WARNING your phone is locked!
The device is locked for viewing and distribution.

Permissions

<table>
<thead>
<tr>
<th>Permission Name</th>
<th>Protection Level</th>
<th>Threat Level</th>
<th>Customized</th>
<th>Duplicated</th>
<th>Used</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>android.permission.ACCESS_NETWORK_STATE</td>
<td>normal</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Allows applications to access information about networks.</td>
</tr>
<tr>
<td>android.permission.INTERNET</td>
<td>dangerous</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Used for permissions that provide access to networking services. The or other related network operations. Allows applications to open network sockets.</td>
</tr>
<tr>
<td>android.permission.READ_EXTERNAL_STORAGE</td>
<td>normal</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Group of permissions that are related to SD card access. Allows an application to read from external storage. targetSdkVersion is 4 or higher.</td>
</tr>
<tr>
<td>android.permission.READ_PHONE_STATE</td>
<td>dangerous</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Allows read only access to phone state. targetSdkVersion is 4 or higher.</td>
</tr>
<tr>
<td>android.permission.RECEIVE_BOOT_COMPLETED</td>
<td>normal</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Allows an application to receive the to the user.</td>
</tr>
<tr>
<td>android.permission.WAKE_LOCK</td>
<td>normal</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Allows using PowerManager WakeLocks to keep processor from sleeping or screen from dimming</td>
</tr>
<tr>
<td>android.permission.WRITE_EXTERNAL_STORAGE</td>
<td>dangerous</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Allows an application to write to external storage. {android.content.Context#getExternalCacheDir}.</td>
</tr>
</tbody>
</table>

Activities

<table>
<thead>
<tr>
<th>Name</th>
<th>Main Activity</th>
<th>Exposed</th>
</tr>
</thead>
<tbody>
<tr>
<td>org.simplelocker.Main</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Activities

<table>
<thead>
<tr>
<th>Name</th>
<th>Main Activity</th>
<th>Exposed</th>
</tr>
</thead>
<tbody>
<tr>
<td>org.simplelocker.Main</td>
<td></td>
<td></td>
</tr>
<tr>
<td>· android.intent.action.MAIN</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Services

<table>
<thead>
<tr>
<th>Name</th>
<th>Exposed</th>
</tr>
</thead>
<tbody>
<tr>
<td>org.simplelocker.MainService</td>
<td></td>
</tr>
<tr>
<td>org.torproject.android.service.TorService</td>
<td></td>
</tr>
<tr>
<td>· org.torproject.android.service.ITorService</td>
<td></td>
</tr>
<tr>
<td>· org.torproject.android.service.TOR_SERVICE</td>
<td></td>
</tr>
</tbody>
</table>

### Broadcast Receivers

<table>
<thead>
<tr>
<th>Name</th>
<th>Dynamically Registered</th>
<th>Exposed</th>
</tr>
</thead>
<tbody>
<tr>
<td>android.support.v4.content.WakefulBroadcastReceiver</td>
<td></td>
<td></td>
</tr>
<tr>
<td>android.support.v4.media.TransportMediatorJellybeanMR2$3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>org.simplelocker.SDCardServiceStarter</td>
<td></td>
<td></td>
</tr>
<tr>
<td>· android.intent.action.ACTION_EXTERNAL_APPLICATIONS_AVAILABLE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>org.simplelocker.ServiceStarter</td>
<td></td>
<td></td>
</tr>
<tr>
<td>· android.intent.action.BOOT_COMPLETED</td>
<td></td>
<td></td>
</tr>
<tr>
<td>org.torproject.android.service.TorService2$2</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
WARNING your phone is locked!
The device is locked for viewing and distribution

Advertisement Modules
N/A

IP Distribution

Urls

<table>
<thead>
<tr>
<th>Country</th>
<th>Url</th>
<th>IP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Germany</td>
<td>N/A</td>
<td>212.112.245.170</td>
</tr>
<tr>
<td>N/A</td>
<td><a href="http://xeyocsu7fu2vjhs.onion/">http://xeyocsu7fu2vjhs.onion/</a></td>
<td>N/A</td>
</tr>
<tr>
<td>United States</td>
<td><a href="http://example.com/">http://example.com/</a></td>
<td>93.184.216.119</td>
</tr>
</tbody>
</table>

Sensitive Files
## File Operations

<table>
<thead>
<tr>
<th>Operation</th>
<th>File Path</th>
<th>Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>read</td>
<td>/mnt/sdcard/screens-out/screen-001.png</td>
<td>t1xe51x901xf5Qlxe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d</td>
</tr>
<tr>
<td>read</td>
<td>/mnt/sdcard/screens-out/screen-001.png</td>
<td>7lxelxfbfx9d1x0elxe1fxbfx9d1xe1fxbfx9d31xe1fxbfx9d</td>
</tr>
<tr>
<td>read</td>
<td>/mnt/sdcard/screens-out/screen-002.png</td>
<td>1xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d</td>
</tr>
<tr>
<td>read</td>
<td>/mnt/sdcard/screens-out/screen-001.png</td>
<td>8lx071xe1fxbfx9d1xe1fxbfx9d61xe1fxbfx9d=R</td>
</tr>
<tr>
<td>read</td>
<td>/mnt/sdcard/screens-out/screen-002.png</td>
<td>^1xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d QJ</td>
</tr>
<tr>
<td>read</td>
<td>/mnt/sdcard/screens-out/screen-002.png</td>
<td>w#1xe1fxbfx9d1xe1fxbfx9d071xe1fxbfx9d+</td>
</tr>
<tr>
<td>read</td>
<td>/mnt/sdcard/screens-out/screen-001.png</td>
<td>xl152x1dxfxa21xe1fxbfx9d0a1xe1fxbfx9d</td>
</tr>
<tr>
<td>read</td>
<td>/mnt/sdcard/screens-out/screen-002.png</td>
<td>\1xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d</td>
</tr>
<tr>
<td>read</td>
<td>/mnt/sdcard/screens-out/screen-002.png</td>
<td>f1x101xe1fxbfx9d51xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d</td>
</tr>
<tr>
<td>read</td>
<td>/mnt/sdcard/screens-out/screen-001.png</td>
<td>1xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d</td>
</tr>
<tr>
<td>read</td>
<td>/mnt/sdcard/screens-out/screen-001.png</td>
<td>N1xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d</td>
</tr>
<tr>
<td>read</td>
<td>/mnt/sdcard/screens-out/screen-002.png</td>
<td>\1xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d</td>
</tr>
<tr>
<td>read</td>
<td>/mnt/sdcard/screens-out/screen-002.png</td>
<td>\1x19\1xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d</td>
</tr>
<tr>
<td>read</td>
<td>/mnt/sdcard/screens-out/screen-001.png</td>
<td>\001xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d</td>
</tr>
<tr>
<td>read</td>
<td>/mnt/sdcard/screens-out/screen-001.png</td>
<td>\1xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d</td>
</tr>
<tr>
<td>read</td>
<td>/mnt/sdcard/screens-out/screen-002.png</td>
<td>\1xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d</td>
</tr>
<tr>
<td>read</td>
<td>/mnt/sdcard/screens-out/screen-002.png</td>
<td>\1xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d</td>
</tr>
<tr>
<td>read</td>
<td>/mnt/sdcard/screens-out/screen-002.png</td>
<td>\1xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d</td>
</tr>
<tr>
<td>read</td>
<td>/mnt/sdcard/screens-out/screen-002.png</td>
<td>\1xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d</td>
</tr>
<tr>
<td>read</td>
<td>/mnt/sdcard/screens-out/screen-002.png</td>
<td>\1xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d1xe1fxbfx9d</td>
</tr>
</tbody>
</table>
WARNING your phone is locked!
The device is locked for viewing and distribution.
May Send SMS

Send SMS

Block SMS

Phone Call

Data Leakage

Sensitive APIs

- **API:** android.telephony.TelephonyManager;->getDeviceId
  - **Description:** Gets the unique device ID. IMEI for GSM and MEID for ESN or ESN for CDMA phones
  - **Caller Code:** Lorg/simplelocker/Utils;->getMEI(Landroid/content/Context;);Ljava/lang/String;
  - **Threat Level:** ""'
  - **Path Index:** 16

- **API:** Ljava/lang/Runtime;->exec
  - **Description:** Executes shell code
Sensitive APIs

- **API: Landroid/telephony/TelephonyManager;->getDeviceID**
  - Description: Gets the unique device ID, IMEI for GSM and MEID for ESN or ESN for CDMA phones
  - Caller Code: Lorg/simplelocker/Utils;->getIMEI(Landroid/content/Context;Ljava/lang/String;)
  - Threat Level: N/A
  - Path Index: 16

- **API: Ljava/lang/Runtime;->exec**
  - Description: Executes shell code
  - Caller Code: Linfo/guardianproject/onionkit/ui/TorServiceUtils;->doShellCommand(Ljava/lang/String;Ljava/lang/StringBuilder;Z Z)
  - Threat Level: N/A
  - Path Index: 24

- **API: Ljava/lang/Runtime;->exec**
  - Description: Executes shell code
  - Caller Code: Linfo/guardianproject/onionkit/ui/TorServiceUtils;->doShellCommand(Ljava/lang/String;Ljava/lang/StringBuilder;Z Z)
  - Threat Level: N/A
  - Path Index: 178

- **API: Ljava/lang/Runtime;->exec**
  - Description: Executes shell code
  - Caller Code: Linfo/guardianproject/onionkit/ui/TorServiceUtils;->findProcessIdWithPS(Ljava/lang/String;)
  - Threat Level: N/A
  - Path Index: 16

- **API: Ljava/lang/Runtime;->exec**
  - Description: Executes shell code
  - Caller Code: Linfo/guardianproject/onionkit/ui/TorServiceUtils;->findProcessIdWithPidOf(Ljava/lang/String;)
  - Threat Level: N/A
  - Path Index: 52

- **API: Ljava/lang/Runtime;->exec**
  - Description: Executes shell code
  - Caller Code: Lorg/torproject/android/service/ExecShell;->executeCommand(Lorg/torproject/android/service/ExecShell$ISSLHELL_CMD;Ljava/util/ArrayList;)
  - Threat Level: N/A
  - Path Index: 26

- **API: Ljava/lang/Runtime;->exec**
  - Description: Executes shell code
  - Caller Code: Lorg/torproject/android/service/TorBinaryInstaller;->copyRawFile(Landroid/content/Context;Ljava/io/File;Ljava/lang/String;Z)
  - Threat Level: N/A
Permission Usage

- **Permission Name:** android.permission.ACCESS_NETWORK_STATE
  - **Used Type:** Api
  - **Caller Code:** android/support/v4/net/ConnectivityManagerCompat$GingerbreadConnectivityManagerCompatImpl;->isActiveNetworkMetered(android/net/ConnectivityManager;)
  - **Callee Code:** android/support/v4/net/ConnectivityManagerCompatGingerbread->isActiveNetworkMetered(android/net/ConnectivityManager;)
  - **Path Index:** 106

- **Permission Name:** android.permission.ACCESS_NETWORK_STATE
  - **Used Type:** Api
  - **Caller Code:** android/support/v4/net/ConnectivityManagerCompat$HoneycombMR2ConnectivityManagerCompatImpl;->isActiveNetworkMetered(android/net/ConnectivityManager;)
  - **Callee Code:** android/support/v4/net/ConnectivityManagerCompatHoneycombMR2->isActiveNetworkMetered(android/net/ConnectivityManager;)
  - **Path Index:** 0

- **Permission Name:** android.permission.ACCESS_NETWORK_STATE
  - **Used Type:** Api
  - **Caller Code:** android/support/v4/net/ConnectivityManagerCompat$JellyBeanConnectivityManagerCompatImpl;->isActiveNetworkMetered(android/net/ConnectivityManager;)
  - **Callee Code:** android/support/v4/net/ConnectivityManagerCompatJellyBean->isActiveNetworkMetered(android/net/ConnectivityManager;)
  - **Path Index:** 0

- **Permission Name:** android.permission.ACCESS_NETWORK_STATE
  - **Used Type:** Api
  - **Caller Code:** android/support/v4/net/ConnectivityManagerCompat;->isActiveNetworkMetered(android/net/ConnectivityManager;)
  - **Callee Code:** android/support/v4/net/ConnectivityManagerCompat$ConnectivityManagerCompatImpl;->isActiveNetworkMetered(android/net/ConnectivityManager;)
  - **Path Index:** 4

- **Permission Name:** android.permission.ACCESS_NETWORK_STATE
  - **Used Type:** Api
  - **Caller Code:** android/support/v4/net/ConnectivityManagerCompat;->getNetworkInfoFromBroadcast(android/net/ConnectivityManager;android/content/Intent;android/net/NetworkInfo;)
  - **Callee Code:** android/net/ConnectivityManager;->getNetworkInfo();android/net/NetworkInfo;
  - **Path Index:** 24

- **Permission Name:** android.permission.ACCESS_NETWORK_STATE
  - **Used Type:** Api
  - **Caller Code:** android/support/v4/net/ConnectivityManagerCompatGingerbread;->isActiveNetworkMetered(android/net/ConnectivityManager;)
  - **Callee Code:** android/net/ConnectivityManager;->getActiveNetworkInfo();android/net/NetworkInfo;
  - **Path Index:** 2
Dear ACME Director

Your data has been encrypted. If you want it back, send us $10,000 to Bitcoin wallet:
1F1tAaz5xxxxxxxxxxxxxx
xxxxxxxxxx

Developed by Botnet Research Team, Xi'an Jiaotong University

Contact me: mindmac.hu@gmail.com

Follow me: 🌐

Partners: VisualThreat, MobiSecLab
## Automated Analysis

<table>
<thead>
<tr>
<th><strong>Md5</strong></th>
<th>6D94CF5CA1DD4967xxxxxxxxxxxxxxxxxxxxxxx</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Risk rating</strong></td>
<td>High (100 score)</td>
</tr>
<tr>
<td><strong>Malware family</strong></td>
<td>Android/Locker.A – also virustotal result/AV</td>
</tr>
<tr>
<td><strong>Pcap file</strong></td>
<td>&lt;attach&gt;</td>
</tr>
<tr>
<td><strong>logcat file</strong></td>
<td>&lt;attach&gt;</td>
</tr>
</tbody>
</table>
| **Risky Behaviour** | • Encrypt Decrypt data  
|                 | • Executes shell code  
|                 | • Gets Device info (IMEI,...)         |
| **Dangerous Permissions** | • Internet  
|                 | • Read Phone State  
|                 | • Write External Storage               |
| **IP/URLs**      | • 212.112.245.170 – Germany – NA  
|                 | • 93.184.216.119 – USA – [http://example.com](http://example.com) – NOTE: could be used to test connection |
| **Other**        | • App probably uses TOR – (Tor service)  
|                 | • No calls or SMS                       |
|                 | • Many file reads/writes (writing read files with .enc extension...is it encrypting them?)  
|                 | • Interesting Broadcast ([WakefulBroadcastReceiver](https://github.com/eyaanlo/wakeful)) |
Dynamic Analysis

1. Droidbox

- ANANAS 😊
- Adb Logcat
- Tcpdump
- Custom tools
- ....
DroidBox is developed to offer dynamic analysis of Android applications. The following information is described in the results, generated when analysis is complete:

- Hashes for the analyzed package
- Incoming/outgoing network data
- File read and write operations
- Started services and loaded classes through DexClassLoader
- Information leaks via the network, file and SMS
- Circumvented permissions
- Cryptographic operations performed using Android API
- Listing broadcast receivers
- Sent SMS and phone calls

Additionally, two graphs are generated visualizing the behavior of the package. One showing the temporal order of the operations and the other one being a treemap that can be used to check similarity between analyzed packages.

https://github.com/pjlantz/droidbox
Dynamic Analysis - Droidbox

Start emulator with droidbox

- ./startemu.sh <Emulator Name>
Add dummy pictures to the emulator:
Api Demos > Content > Storage > External Storage
Install application on emulator with droidbox

- `./droidbox.sh <application name>`
Dynamic Analysis - Droidbox

Droidbox pushes, installs and runs the application
Dynamic Analysis - Droidbox

Droidbox generates logs as the application runs.

Here you can perform various operations to generate logs.

Notice counter is 474
Dynamic Analysis - Droidbox

Notice counter is 9501 (probably all the file read and writes)

Press ctrl + c to show logs and generate log file
Dynamic Analysis - Droidbox

Dear ACME Director

ACME Inc.

Your data has been encrypted. If you want it back, send us $10,000 to Bitcoin wallet: 1F1tAaz5xxxxxxxxxxxx xxxxxxxxxx

[Image of Android device and command lines]
Dynamic Analysis - Droidbox

Logs show file write of DemoPicture.jpg.enc
Dynamic Analysis - Droidbox

Droidbox Generated a .json log file and 2 images (behaviorgraph and tree)
Dynamic Analysis - Droidbox
Open json log with json viewer (Mozilla extension above)
Dynamic Analysis - Droidbox

The logs clearly shows read and write operations
## Dynamic analysis

<table>
<thead>
<tr>
<th>Pre-Infection</th>
<th>We created an SD card with dummy images based on previous analysis. We assume that these will be encrypted</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Post-Infection Behaviour</strong></td>
<td>Screen keeps prompting message</td>
</tr>
<tr>
<td><strong>Other</strong></td>
<td>• Confirmed DemoPicture.jpg was encrypted</td>
</tr>
<tr>
<td></td>
<td>• Behaviour graph and Tree confirms large number of read/writes</td>
</tr>
<tr>
<td></td>
<td>• Output confirms read/writes with .enc extension</td>
</tr>
<tr>
<td></td>
<td>• Network connections listed in JSON output</td>
</tr>
</tbody>
</table>
Static Analysis

1. Apktool
2. Dex2Jar – JD-GUI

- Androguard
- Enjarify
- Jeb, JAD,…
- Custom tools
  - command-line fu techniques ;)
- …..
Manifest
AesCrypt

```java
public class AesCrypt {
    private final Cipher cipher;
    private final SecretKeySpec key;
    private AlgorithmParameterSpec spec;

    public AesCrypt(String paramString) throws Exception {
        MessageDigest localMessageDigest = MessageDigest.getInstance("SHA-256");
        localMessageDigest.update(paramString.getBytes("UTF-8"));
        byte[] arrayOfByte = new byte[32];
        System.arraycopy(localMessageDigest.digest(), 0, arrayOfByte, 0, arrayOfByte.length);
        this.cipher = Cipher.getInstance("AES/CBC/PKCS7Padding");
        this.key = new SecretKeySpec(arrayOfByte, "AES");
        this.spec = getIV();
    }

    public void decrypt(String paramString1, String paramString2) throws Exception {
        FileInputStream localFileInputStream = new FileInputStream(paramString1);
        FileOutputStream localFileOutputStream = new FileOutputStream(paramString2);
        this.cipher.init2(this.key, this.spec);
        CipherInputStream localCipherInputStream = new CipherInputStream(localFileInputStream, this.cipher);
        byte[] arrayOfByte = new byte[8];
        while (true)
        {
            int i = localCipherInputStream.read(arrayOfByte);
            if (i == -1)
            {
                localFileOutputStream.flush();
                localFileOutputStream.close();
                localCipherInputStream.close();
                return;
            }
            localFileOutputStream.write(arrayOfByte, 0, i);
        }
    }
}
```

AES/CBC/PKCS7Padding
Constants

```java
package org.simplelocker;

import java.util.Arrays;

public class Constants {
    public static final String ADMIN_URL = "http://xyyocst7fu2vzhxs.onion/";
    public static final int CHECK_MAIN_WINDOW_TIME_SECONDS = 1;
    public static final String CIPHER_PASSWORD = "jqmllasf074hr";
    public static final String CLIENT_NUMBER = "19";
    public static final String DEBUG_TAG = "DEBUGGING";
    public static final String DISABLE_LOCKER = "DISABLE_LOCKER";
    public static final List<String> EXTENSIONS_TO_ENCRYPT = Arrays.asList("jpeg", "jpg", "png", "bmp", "gif", "pdf", "doc", "docx");
    public static final String FILES_WAS_ENCRYPTED = "FILES_WAS_ENCRYPTED";
    public static final int MONEYPACK_DIGITS_NUMBER = 14;
    public static final int PAYSAFECARD_DIGITS_NUMBER = 16;
    public static final int POLLING_TIME_MINUTES = 3;
    public static final String PREFERENCES_NAME = "AppPrefs";
    public static final int UKASH_DIGITS_NUMBER = 19;
}
```
Utils

```java
private String getIMEI(Context paramContext) {
    String str = getIMEI(paramContext);
    if (str != null)
        return str.substring(0, Math.min(str.length(), 10));
    return "";
}
```

getIMEI Method
AesCrypt

Decryption Method

```java
public void decrypt(String paramString1, String paramString2) throws Exception {
    FileInputStream localFileStream = new FileInputStream(paramString1);
    FileOutputOutputStream localFileOutputOutputStream = new FileOutputOutputStream(paramString2);
    CipherInputStream localCipherInputStream = new CipherInputStream(localFileStream, this.cipher);
    byte[] arrayOfByte = new byte[8];
    while (true) {
        int i = localCipherInputStream.read(arrayOfByte);
        if (i == -1) {
            localFileOutputStream.flush();
            localFileOutputStream.close();
            localCipherOutputStream.close();
            return;
        }
        localFileOutputStream.write(arrayOfByte, 0, i);
    }
}
```
## Incident Report

### Static analysis

<table>
<thead>
<tr>
<th>Manifest info</th>
<th>Permissions</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Activity: .Main (Main Launcher)</td>
</tr>
<tr>
<td></td>
<td>Broadcast receiver: SD CardServiceStarted</td>
</tr>
<tr>
<td></td>
<td>Broadcast receiver: ServiceStarter</td>
</tr>
<tr>
<td></td>
<td>Service: MainService</td>
</tr>
<tr>
<td></td>
<td>Service: org.torproject.android.service.TorService</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Encryption algorithm being used</th>
<th>AES/CBC/PKCS7Padding</th>
</tr>
</thead>
<tbody>
<tr>
<td>Encryption key</td>
<td>jndlasf074hr</td>
</tr>
<tr>
<td>Files types that are being encrypted</td>
<td>jpeg, jpg, png, bmp, gif, pdf, doc, docx...</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Other</th>
</tr>
</thead>
<tbody>
<tr>
<td>We can see methods retrieving device info</td>
</tr>
<tr>
<td>o Util methods: getME(), getModel(), getOS()...</td>
</tr>
<tr>
<td>o Constants class contains some useful data</td>
</tr>
<tr>
<td>o url matches the one found in Automated analysis</td>
</tr>
<tr>
<td>o encryption key found again</td>
</tr>
<tr>
<td>Encryption and Decryption Methods are available</td>
</tr>
</tbody>
</table>

### Code Snippet

```java
public void decrypt(String paramString1, String paramString2) throws Exception {
    FileInputStream localFileInputStream = new FileInputStream(paramString1);
    FileOutputStream localFileOutputStream = new FileOutputStream(paramString2);
    this.cipher.init(?, this.key, this.spec);
    CipherInputStream localCipherInputStream = new CipherInputStream(localFileInputStream, this.cipher);
    byte[] arrayByte = new byte[8];
    while (true) {
        int i = localCipherInputStream.read(arrayByte);
        if (i == -1) {
            localFileOutputStream.flush();
            localFileOutputStream.close();
            localCipherInputStream.close();
            return;
        }
        localFileOutputStream.write(arrayByte, 0, i);
    }
}
```
Propose actions to the different parties involved
Stay on top of it, monitor, remind, ...
1. We will try to uninstall the ransomware
2. We will then use the encryption information gathered to recover the data!
3. The above will be tested on an emulator (or test device) first (Try Emulate Similar Device- Nexus 5 Running Android 4.4)
Install Ransomware

Dear ACME Director

Your data has been encrypted. If you want it back, send us $10,000 to Bitcoin wallet:
1F1tAaz5xxxxxxxxxxxxx
xxxxxxxxxxxx
Uninstall Ransomware

- **adb shell**
  In android shell:
- **cd data/data**
  Find the package name (org.simplelocker)
  (alternatively see running processes)
Uninstall Ransomware

- `adb uninstall org.simplelocker`
Action Proposed

• Uninstallation works!
• For decryption we create an app that reverses the action of the Ransomware:
  • Parses all .enc files
  • Decrypts using decryption method & key
• We will work on a copy of the files just in case the decryption fails
• Again Test on Emulator (or Test Device)
Action Proposed

decryption method and encryption keys discovered earlier
The app parses all files looking for .enc extension.
Action Proposed

Install app on Emulator

Notice encrypted file
Action Proposed

The app created a copy of the file and decrypted it.
DemoPicture is decrypted 😊

Note: simply renaming the .enc file or using another decryption key does not work
Seek approval before performing potentially dangerous tasks
Eradication & Recovery

Ensure that all files are back to normal and there are no traces of malware on the device.
Eradication & Recovery
Incident Report

Recovery

- Uninstall simplelocker using adb uninstall org.simplelocker
- This removes the app, now we need to decrypt files
- Using the information in this report, we know that we have to
  - parse all files in sd card that have a .enc extension
  - decrypt files using decrypt method discovered above with the encryption key
    jndlasf074hr
  - It is probably best to create a copy rather than replacing the encrypted files.
- We can make an app that does these operations

```java
public void decrypt(String paramString1, String paramString2)
    throws Exception
{
    FileInputStream localFileInputStream = new FileInputStream(paramString1);
    FileOutputStream localFileOutputStream = new FileOutputStream(paramString2);
    this.cipher.init(2, this.key, this.spec);
    CipherInputStream localCipherInputStream = new CipherInputStream(localFileInputStream, this.cipher);
    byte[] arrayByte = new byte[8];
    while (true)
    {
        int i = localCipherInputStream.read(arrayByte);
        if (i == -1)
        {
            localFileOutputStream.flush();
            localFileOutputStream.close();
            localCipherInputStream.close();
            return;
        }
        localFileOutputStream.write(arrayByte, 0, i);
    }
}
```
Incident Report

- Several tools offer removal and decryption
- EG: Avast Ransomware Removal

  How to use avast! Ransomware Removal?

  Once installed, it removes the malware from your device and decrypts all files which the malware has encrypted.

  Necessary steps:

  2. Login to the Google Play with the same user information you use to login to your phone.
  3. Search for the avast! Ransomware Removal application (it may be this app you are looking at).
  4. Click on the “Install” button, and the app will be installed on your device in a minute.
  5. After the app is installed on your phone, click the app name in the notification bar.
  6. The app will start and provide you with further instructions.
  7. Uninstall the app at the end so you can install it again in the future if necessary.
Closing Incident

Closing Incident

Finalise report

• **Who?**
  - an attack target (very often a reporter of the incident);
  - important parties involved in resolving the incident, who are usually ISPs/ICPs, other CSIRTs, and LEAs, Contractors;

• **What?**
  - a short description of the incident (including information about your classification of the incident);
  - the results of your work – whether the incident was resolved or not;
  - your main findings and recommendations.
Closing Incident

Incident Duration: 1 day!
Improvement Proposals

Improvement Proposals

• Need for a BYOD Policy?
• MDM or similar tools?
• Mobile backup?
• No untrusted apps allowed!
• We need to train our incident handlers
• Prevent similar event by organizing awareness raising campaigns
• Update to Latest OS, update software ......
• Attack was targeted. We should forward info to law enforcement
• Should we forward info to others? CSIRTs, ....
• ......
Improvement Proposals

• Incident Handling Process
• Mobile Threats Incident Handling Workflow
  • Update Workflow
  • Include tools (incl. recovery tool)
• Automate Automate Automate....
  • Develop scripts for repetitive tasks
  • Purchase commercial tools
  • Reduce SLA

P.S: this is probably a good time to ask for more resources, training, etc.
Ideal Scenario

Automated Tool

Behavior

- Encrypted Files:
  ....
- Screen Lock
  ....

Encryption Key: abc123

Decryption Method:

decrypt(key, file) {
  ...
}

1 year later
ACME Executive!
Search Ticketing System

ra
ransomware Android
Recovery

- Uninstall simplelocker using adb uninstall org.simplelocker
- This removes the app, now we need to decrypt files
- Using the information in this report, we know that we have to
  - parse all files in sd card that have a .enc extension
  - decrypt files using decrypt method discovered above with the encryption key jndlasf074hr
  - It is probably best to create a copy rather than replacing the encrypted files.
- We can make an app that does these operations

```java
public void decrypt(String paramString1, String paramString2)
    throws Exception
{
    FileInputStream localFileInputStream = new FileInputStream(paramString1);
    OutputStream localFileOutputStream = new FileOutputStream(paramString2);
    CipherInputStream localCipherInputStream = new CipherInputStream(localFileInputStream, this.cipher.init(2, this.key, this.spec));
    byte[] arrayOfByte = new byte[8];
    while (true)
    {
        int i = localCipherInputStream.read(arrayOfByte);
        if (i == -1)
        {
            localFileOutputStream.flush();
            localFileOutputStream.close();
            localCipherInputStream.close();
            return;
        }
    }
    localFileOutputStream.write(arrayOfByte, 0, i);
}
```
Requirements

Improve process
Increase automation
Improve tools

Reduce response & resolution time
Questions
Thank you

PO Box 1309, 710 01 Heraklion, Greece
Tel: +30 28 14 40 9710
info@enisa.europa.eu
www.enisa.europa.eu