Attacking encrypted USB keys the hard(ware) way

Jean-Michel Picod, Rémi Audebert, Sven Blumenstein, Elie Bursztein
with the help of many Googlers
13.6% of USB drives storing company data are lost or stolen.
Are encrypted USB drives really hacker proof?
How to audit encrypted drives
Show-case real attacks found while auditing
On-going research ahead
Logical view of a secure USB key

- USB plug
- Controller (crypto)
- Storage

Input
Key with controller and memory separated

Controller (crypto/USB)

Storage (eMMC)
Integrated key

Controller & storage

Fingerprint reader

https://g.co/research/protect
NIST certifications for encrypted USB keys
FIPS certifications

FIPS 140
Cryptographical security disclosure & validation process

FIPS 197
Fancy way to write AES ;)}
FIPS 140 scope in a nutshell

Cryptography

https://g.co/research/protect
One of the many FIPS certification fails
Audit methodology
Attacker type

**Serendipitous**
Opportunistic attacker with minimal resources

**Professional**
Attacker with resources, albeit limited

**State sponsored**
After specific data(keys) that are worth a large investment
Attack impact

- **Weaken security**
  Makes carrying attack easier

- **Single drive break**
  “Only” allowed to recover a single drive data

- **Full break**
  Recovery attack that affects all drives
Defensive manufacturing key goals

Mitigating hardware attacks
Slowdown initial analysis and prevent physical tampering & counterfeit

Preventing advanced attacks
Defend against very targeted attack: e.g TEMPEST, EVIL MAID
TEMPEST prevention via copper shielding
Mitigate component extraction with epoxy
Fake epoxy insecurity

Acetone

FAIL

https://g.co/research/protect
NIST evaluation fail

PLC. — (Underlying Steel Chassis) and Plus (Underlying Steel Chassis)

5 PHYSICAL SECURITY
The cryptographic boundary for the modules is defined as all components within the steel chassis only. All components are coated in epoxy and encased in the steel chassis. The rubber sleeve material which surrounds the steel chassis is not considered part of the cryptographic boundary, and has been excluded from the FIPS 140-2 requirements on the basis that it contributes nothing to the module’s security. The modules do not have removable doors or covers. They components with integrated circuit packaging that is production grade using standard passivation within the visible spectrum.
Etching
Etching failed
Leaving debug ports active
<table>
<thead>
<tr>
<th>Defensive manufacturing criteria</th>
</tr>
</thead>
<tbody>
<tr>
<td>Device is tamper evident</td>
</tr>
<tr>
<td>PCB is dipped in epoxy to make extraction harder</td>
</tr>
<tr>
<td>Hardware components marking &amp; serial numbers are erased</td>
</tr>
<tr>
<td>Firmware must be properly signed or read-only</td>
</tr>
<tr>
<td>PCB is shielded in copper to prevent TEMPEST attack</td>
</tr>
<tr>
<td>Device is counterfeiting resistant</td>
</tr>
<tr>
<td>Encryption key is wiped out upon tampering</td>
</tr>
</tbody>
</table>

- Serendipitous
- Professional
- State sponsored
- Weakness
- Single break
- Full break

[https://g.co/research/protect](https://g.co/research/protect)
Input mechanism key security goals

Identifying users accurately
Only valid users should be recognized

Securing credential enrollment
Prevent rogue identifiers (fingerprint/badges) to be enrolled
Pinpad
Badge
(RFID/NFC)
Fingerprint reader
Software

Software
Unlock command can be replayed

<table>
<thead>
<tr>
<th>Target</th>
<th>Fingerprint key 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Impact</td>
<td>Full break</td>
</tr>
<tr>
<td>Attacker</td>
<td>Professional</td>
</tr>
</tbody>
</table>

https://g.co/research/protect
Unlock command replay attack against software based key found in 2010
Vulnerable key
Vulnerable key internal
Identification command flow through serial line

USB plug → Controller (crypto) → Fingerprint manager

Storage → Sensor
Attack setup

Buspirate used for injection

USB Key

Serial line
Input can be cloned

<table>
<thead>
<tr>
<th>Target</th>
<th>HDD with badge</th>
</tr>
</thead>
<tbody>
<tr>
<td>Impact</td>
<td>Single drive break</td>
</tr>
<tr>
<td>Attacker</td>
<td>Professional</td>
</tr>
</tbody>
</table>
Vulnerable device
Vulnerable device internal

- Configuration
- EEPROM
- RFID controller
- RFID coil
- SATA port
Cloning equipment

Reprogrammable RFID tag

Proxmark3

https://g.co/research/protect
Badge is dumped
Using standard 125KHz reader
## Input audit criteria

<table>
<thead>
<tr>
<th>Requirement</th>
<th>Serendipitous</th>
<th>Professional</th>
<th>State sponsored</th>
<th>Weakness</th>
<th>Single drive break</th>
<th>Full break</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pin is not stored in software</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Unlock command can’t be replayed</td>
<td>Meow</td>
<td>Meow</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Keypad don’t show obvious sign of wear and tear</td>
<td>Meow</td>
<td>Meow</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Input (e.g. tag, fingerprint) can’t be cloned</td>
<td>Meow</td>
<td>Meow</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Debug ports are disabled</td>
<td>Meow</td>
<td>Meow</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Controller security key goals

Protect secrets
Pin hash and AES key must be non recoverable

Lock the drive when needed
Ensure that data is only accessed in safe circumstances

Destroy secrets when attacked
Make the drive unusable in case too many attempts are made

Firmware attestation
Ensure that the firmware used is really the one disclosed
USB communication interception

Key

Interceptor

monitoring PC

computer

https://g.co/research/protect
Interception in practice

- Cable to monitoring pc
- Interceptor
- USB Key
- Cable to laptop

Research at Google
https://g.co/research/protect
Password can be extracted

<table>
<thead>
<tr>
<th>Target</th>
<th>Fingerprint key 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Impact</td>
<td>Full break</td>
</tr>
<tr>
<td>Attacker</td>
<td>Serendipitous</td>
</tr>
</tbody>
</table>

Research at Google

https://g.co/research/protect
Vulnerable key
Key internals
**Input can be brute-forced**

<table>
<thead>
<tr>
<th>Target</th>
<th>HDD with badge</th>
</tr>
</thead>
<tbody>
<tr>
<td>Impact</td>
<td>Full break</td>
</tr>
<tr>
<td>Attacker</td>
<td>Serendipitous</td>
</tr>
</tbody>
</table>
Vulnerable badge
## Controller audit criteria

<table>
<thead>
<tr>
<th>Requirement</th>
<th>Pass</th>
<th>Fail</th>
</tr>
</thead>
<tbody>
<tr>
<td>Device is burned out after nth unsuccessful attempts</td>
<td><img src="https://example.com/pass.png" alt="Pass" /></td>
<td><img src="https://example.com/fail.png" alt="Fail" /></td>
</tr>
<tr>
<td>Password &amp; AES key can’t be requested</td>
<td><img src="https://example.com/pass.png" alt="Pass" /></td>
<td><img src="https://example.com/fail.png" alt="Fail" /></td>
</tr>
<tr>
<td>AES Key is regenerated upon drive reset</td>
<td><img src="https://example.com/pass.png" alt="Pass" /></td>
<td><img src="https://example.com/fail.png" alt="Fail" /></td>
</tr>
<tr>
<td>Device immediately lock itself when removed from USB port</td>
<td><img src="https://example.com/pass.png" alt="Pass" /></td>
<td><img src="https://example.com/fail.png" alt="Fail" /></td>
</tr>
<tr>
<td>Device lock itself after inactivity period</td>
<td><img src="https://example.com/pass.png" alt="Pass" /></td>
<td><img src="https://example.com/fail.png" alt="Fail" /></td>
</tr>
<tr>
<td>Device immediately lock itself after a USB reset</td>
<td><img src="https://example.com/pass.png" alt="Pass" /></td>
<td><img src="https://example.com/fail.png" alt="Fail" /></td>
</tr>
<tr>
<td>Encryption key is zeroed when the device is burnt</td>
<td><img src="https://example.com/pass.png" alt="Pass" /></td>
<td><img src="https://example.com/fail.png" alt="Fail" /></td>
</tr>
<tr>
<td>Data is zeroed out when the device is burnt</td>
<td><img src="https://example.com/pass.png" alt="Pass" /></td>
<td><img src="https://example.com/fail.png" alt="Fail" /></td>
</tr>
</tbody>
</table>
Cryptography key goals

Data is properly encrypted (duh!)
State of the art encryption with proper initialization and settings

Encryption keys are truly random
Each drive have a unique key that is not predictable
Cryptography audit on hardware complexity

Black box (literally!)
Algorithms are baked in the silicon

Most tests are too expensive in current setting
Audit can only catch the most blatant errors
Use outdated crypto

<table>
<thead>
<tr>
<th>Target</th>
<th>Password protected key</th>
</tr>
</thead>
<tbody>
<tr>
<td>Impact</td>
<td>Single drive break</td>
</tr>
<tr>
<td>Attacker</td>
<td>State sponsored</td>
</tr>
</tbody>
</table>
A few examples we came across

**RSA-512 used to communicate on USB port**
Open the door to factoring attacks

**RC4 to encrypt files**
Broken cipher
## Encryption audit criteria

<table>
<thead>
<tr>
<th>Criteria</th>
<th>Secure?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Encryption key is unique per device</td>
<td>🗝️</td>
</tr>
<tr>
<td>There is no recovery/master key</td>
<td>🔒</td>
</tr>
<tr>
<td>Encryption key derivation use PBKDF2 algorithm with enough round</td>
<td>🗝️</td>
</tr>
<tr>
<td>Encryption key is randomly generated</td>
<td>🗝️</td>
</tr>
<tr>
<td>A secure random generator is used</td>
<td>🗝️</td>
</tr>
<tr>
<td><strong>Data is encrypted with AES or newer standard</strong></td>
<td>🔒</td>
</tr>
<tr>
<td>IV are properly randomized</td>
<td>🗝️</td>
</tr>
<tr>
<td>Encryption chaining algorithm is secure</td>
<td>🗝️</td>
</tr>
</tbody>
</table>

[Research at Google](https://g.co/research/protect)
Storage goals

Full encryption
Everything including data, configuration, software should be encrypted

Integrity
Storage tampering should be detected
Extracting memory content - not that easy ^^

1. Chip removal
2. Memory dump
3. Error correction (BCH ECC)
4. Undo XOR scrambling
5. Interleave blocks
6. Undo file translation layer
7. Strip away metadata
8. Decrypt file system
CD-ROM partition can be backdoored

<table>
<thead>
<tr>
<th>Target</th>
<th>Password protected key</th>
</tr>
</thead>
<tbody>
<tr>
<td>Impact</td>
<td>Single drive break</td>
</tr>
<tr>
<td>Attacker</td>
<td>State sponsored</td>
</tr>
</tbody>
</table>

https://g.co/research/protect
After removing the metal enclosure
Epoxy removed
Chip-off
Memory dump
Splitting areas based on visual patterns
Patterns after unscrambling

Data  Metadata  ECC  Data
After XOR unscrambling

<table>
<thead>
<tr>
<th>00000000</th>
<th>00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F</th>
</tr>
</thead>
<tbody>
<tr>
<td>00000001</td>
<td>42 74 50 72 61 6D 43 64 00 00 00 00 00 00 00 00</td>
</tr>
<tr>
<td>00000002</td>
<td>10 10 08 00 00 00 00 00 00 00 00 00 00 00 00 00</td>
</tr>
<tr>
<td>00000003</td>
<td>00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00</td>
</tr>
<tr>
<td>00000004</td>
<td>2C 48 04 1A 05 00 00 00 00 00 00 00 00 00 00 00</td>
</tr>
<tr>
<td>00000005</td>
<td>04 1A 01 00 33 08 00 00 00 00 00 00 00 00 00 00</td>
</tr>
<tr>
<td>00000006</td>
<td>0D 13 80 00 00 00 00 00 00 00 00 00 00 00 00 00</td>
</tr>
<tr>
<td>00000007</td>
<td>18 08 04 01 00 00 00 00 00 00 00 00 00 00 00 00</td>
</tr>
<tr>
<td>00000008</td>
<td>34 43 30 00 01 02 03 06 07 0A 05 0E 0F 12 13 16</td>
</tr>
<tr>
<td>00000009</td>
<td>17 1A 1B 1E 1F 22 23 26 27 2A 2B 2E 2F 32 33 36</td>
</tr>
<tr>
<td>0000000A</td>
<td>37 3A 3B 3E 3F 42 43 46 47 4A 4B 4E 4F 52 53 56</td>
</tr>
<tr>
<td>0000000B</td>
<td>57 5A 5B 5E 5F 62 63 66 67 6A 6B 6E 6F 72 73 76</td>
</tr>
<tr>
<td>0000000C</td>
<td>00 FF 00 00 00 00 00 00 00 00 00 00 00 00 00 00</td>
</tr>
<tr>
<td>0000000D</td>
<td>00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00</td>
</tr>
<tr>
<td>0000000E</td>
<td>00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00</td>
</tr>
<tr>
<td>0000000F</td>
<td>00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00</td>
</tr>
</tbody>
</table>

BtPramCd.
odol:________________________
...3.________________________
...E________________________
4CE________________________
"#/a*/+.236_________________
7:>>?BCFGJKNORSV_________________
WZ[^_bcfgjknorsv_________________

https://g.co/research/protect
### ISO9660 partition

<table>
<thead>
<tr>
<th>Address</th>
<th>Data</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0070003A30</td>
<td>20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 4D 4B</td>
<td>MK</td>
</tr>
<tr>
<td>0070003A40</td>
<td>49 53 4F 46 53 20 49 53 4F 39 36 36 30 2F 48 46</td>
<td>ISOFS ISO9660/HF</td>
</tr>
<tr>
<td>0070003A50</td>
<td>53 2F 55 44 46 20 46 49 4C 45 53 59 53 54 45 4D</td>
<td>S/UDF FILESYSTEM</td>
</tr>
<tr>
<td>0070003A60</td>
<td>20 42 55 49 4C 44 45 52 20 26 20 43 44 52 45 43</td>
<td>BUILDER &amp; CDREC</td>
</tr>
<tr>
<td>0070003A70</td>
<td>4F 52 44 20 43 44 2F 44 56 44 2F 42 6C 75 52 61</td>
<td>ORD CD/DVD/BluRay</td>
</tr>
<tr>
<td>0070003A80</td>
<td>79 20 43 52 45 41 54 4F 52 20 28 43 29 20 31 39</td>
<td>CREATOR (C) 19</td>
</tr>
<tr>
<td>0070003A90</td>
<td>39 33 20 45 2E 59 4F 55 4E 47 44 41 4C 45 20 28</td>
<td>93 E.YOUNGDALE (C) 1997 J.PEARSON/J.SCHILLING</td>
</tr>
<tr>
<td>0070003AA0</td>
<td>43 29 20 31 39 39 37 20 4A 2E 50 45 41 52 53 4F</td>
<td></td>
</tr>
<tr>
<td>0070003AB0</td>
<td>4E 2F 4A 2E 53 43 48 49 4C 4C 49 4E 47 20 20 20</td>
<td></td>
</tr>
<tr>
<td>0070003AC0</td>
<td>20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20</td>
<td></td>
</tr>
<tr>
<td>0070003AD0</td>
<td>20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20</td>
<td></td>
</tr>
<tr>
<td>0070003AE0</td>
<td>20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20</td>
<td></td>
</tr>
<tr>
<td>0070003AF0</td>
<td>20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20</td>
<td></td>
</tr>
<tr>
<td>0070003B00</td>
<td>20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20</td>
<td></td>
</tr>
<tr>
<td>0070003B10</td>
<td>20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20</td>
<td></td>
</tr>
<tr>
<td>0070003B20</td>
<td>20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20</td>
<td></td>
</tr>
<tr>
<td>0070003B30</td>
<td>35 31 30 32 32 31 31 33 30 39 31 39 0C 32 30</td>
<td>201</td>
</tr>
<tr>
<td>0070003B40</td>
<td>31 35 31 30 32 32 31 31 33 33 39 31 39 0C 30</td>
<td>5102211130919.20</td>
</tr>
<tr>
<td>0070003B50</td>
<td>30 30 30 30 30 30 30 30 30 30 30 30 30 30 30 30 30 30 30 30</td>
<td>15102211130919.0</td>
</tr>
<tr>
<td>0070003B60</td>
<td>32 30 31 35 31 30 32 32 31 31 33 33 30 39 31 39</td>
<td>00000000000000.0</td>
</tr>
</tbody>
</table>

Research at Google

https://g.co/research/protect
Main binary to unlock the key
Backdoor:

1. Patch executable
2. Apply XOR scrambler
3. Recompute ECC
4. Rewrite memory chip
5. Solder chip back
6. Rewrap the key
Sometime a SDcard reader will do
Where are the secret stored?
Interception platform overview

Custom C eMMC firmware

JTAG Terminal

FPGA logic analyzer

Researcher’s workstation

Altera Stratix IV FPGA

Nios II soft core CPU
Running at 150MHz

JTAG UART

16 channels logic analyzer

Custom verilog synchronous eMMC core

Control and status registers

INT

CLK

CMD

DATA

USB Key eMMC
Interception platform simplified overview :)
Interception platform

Logic analyzer

FPGA emulating memory

Key without memory

https://g.co/research/protect
Key wiring details

- Custom PCB
- Individual 0.02" wires
- Key without memory
Analysing eMMC at the logic level

Decoded overlay
Digital channels
Analog channels
Measurement panel
Emulating an eMMC chip with a FPGA
Analyzer output

[-] eMMC firmware started
[-] eMMC system ready for commands!
[-] CMD 0, H2D, verdict: ACCEPT, arg: 0xfffffffffa, ctr: 0
[-] CMD 0, H2D, verdict: ACCEPT, arg: 0xfffffffffa, ctr: 1
[-] CMD 0, H2D, verdict: ACCEPT, arg: 0xf0f0f0f0f0, ctr: 2
[-] CMD 0, H2D, verdict: ACCEPT, arg: 0x0, ctr: 3
[-] CMD 1, H2D, verdict: ACCEPT, arg: 0x401c0000, ctr: 4
[-] CMD 2, H2D, verdict: ACCEPT, arg: 0x0, ctr: 5
[-] CMD 3, H2D, verdict: ACCEPT, arg: 0x10000, ctr: 6
[-] CMD 7, H2D, verdict: ACCEPT, arg: 0x10000, ctr: 7
[-] CMD 13, H2D, verdict: ACCEPT, arg: 0x10000, ctr: 8
AES key can be recovered?

<table>
<thead>
<tr>
<th>Target</th>
<th>?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Impact</td>
<td>?</td>
</tr>
<tr>
<td>Attacker</td>
<td>?</td>
</tr>
</tbody>
</table>
Looking forward to collaborate on this
### Storage audit criteria

<table>
<thead>
<tr>
<th>Data is encrypted :)</th>
<th>![Lock Icon]</th>
<th>![Unlock Icon]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Should not be stored in memory chip</td>
<td>![Lock Icon]</td>
<td>![Unlock Icon]</td>
</tr>
<tr>
<td>- Pin in clear</td>
<td>![Lock Icon]</td>
<td>![Unlock Icon]</td>
</tr>
<tr>
<td>- AES key</td>
<td>![Lock Icon]</td>
<td>![Unlock Icon]</td>
</tr>
<tr>
<td>- User pin in hashed form</td>
<td>![Lock Icon]</td>
<td>![Unlock Icon]</td>
</tr>
<tr>
<td>- Firmware signing key</td>
<td>![Lock Icon]</td>
<td>![Unlock Icon]</td>
</tr>
<tr>
<td>Tool (CD) partition integrity must be verified</td>
<td>![Lock Icon]</td>
<td>![Unlock Icon]</td>
</tr>
</tbody>
</table>

- Serendipitous
- Professional
- State sponsored
- Weakness
- Single drive break
- Full break

[Research at Google](https://g.co/research/protect)
Takeaways

Certification is important (e.g. FIPS-140)
Ensure cryptographic info are (somewhat) disclosed - helps with audit

Current certification is not enough!
Many areas are not covered by current certification as demonstrated

The security of various drives greatly varies
Not all manufacturers or models for a given manufacturers provide the same level of security
Next steps

Use secure encrypted keys
Don’t get breached because sensitive data was on a lost USB key

Demand transparency from manufacturers
Key security audit shouldn’t be hard - full spec must be disclosed

Let’s build a better audit and certification process
It is our community responsibility to create a sound methodology
Questions?
Thank you

g.co/research/protect