Advanced SCA Attacks & Business Perspective

PV204 Security Technologies

Łukasz Chmielewski

CRoCS,
Masaryk University,
chmiel@fi.muni.cz

Part of the content is reused with permission of Lejla Batina and Ileana Buhan, Radboud University Nijmegen, The Netherlands.

April 29th, 2024

Outline

1. Advanced Passive Attacks
   - Higher-Order Attacks
   - Profiled Attacks
2. Introductions/Recall on Fault Attacks
3. Recall on Differential Fault Analysis (RSA)
4. Real-world example: fault attacks on WolfSSL
5. Symmetric DFA
   - DFA on (Triple) DES
   - DFA on AES
6. Advanced Fault Injection Attacks (Selection)
   - Fault Injection And Side-Channel Analysis
   - Statistical Ineffective Fault Attacks
7. Practical Aspects of Fault Injection Attacks
8. The industry perspective
   - Evaluating Hardware attacks
9. `sec-certs`
10. Conclusions and Further Reading
    - Conclusions: SCA and FI
    - Conclusions: Business Perspective

Plan

1. Advanced Passive Attacks
   - Higher-Order Attacks
   - Profiled Attacks
2. Introductions/Recall on Fault Attacks
3. Recall on Differential Fault Analysis (RSA)
4. Real-world example: fault attacks on WolfSSL
5. Symmetric DFA
   - DFA on (Triple) DES
   - DFA on AES
6. Advanced Fault Injection Attacks (Selection)
   - Fault Injection And Side-Channel Analysis
   - Statistical Ineffective Fault Attacks
7. Practical Aspects of Fault Injection Attacks
8. The industry perspective
   - Evaluating Hardware attacks
9. `sec-certs`
10. Conclusions and Further Reading
    - Conclusions: SCA and FI
    - Conclusions: Business Perspective

CRoCS
Centre for Research on Cryptography and Security

Recall: Masking

Principle: randomizing intermediate values with a secret sharing scheme so DPA fails

Boolean masking: a dth-order secure Boolean masking scheme splits a sensitive value x into $d+1$ shares $(x_0, x_1, ..., x_d)$, as follows:

$$x = x_0 \oplus x_1 \oplus \cdots \oplus x_d$$

The number of traces required for a successful attack grows exponentially w.r.t. the security order d.

Probing-secure scheme. We refer to a scheme that uses certain families of shares as $t-$probing-secure iff any set of at most t intermediate variables is independent from the sensitive values.

Recall: Masking with 2 shares

- $X = X_1 \oplus X_2$
- The leakage $L(X) = HW(X_1, X_2)$ depends on two variables.
- It does not reveal any information on the value of X when a DPA is performed

x	x_1	x_2	$\mathcal{L}(x)$	$\text{Mean}(\mathcal{L}(x))$	$\text{Var}(\mathcal{L}(x))$
0	0	0	0	1	1
	1	1	2		
1	0	1	1	1	0
	1	0	1		

Recall: 2nd order attacks: the idea

The original paper of Kocher at al. states: ““Of particular importance are high-order DPA functions that combine multiple samples from within a trace.”

T. Messerges:

- $t = 1 : m = rand()$ (generate mask-byte)
- $t = 2 : x = p \oplus m$ (XOR mask with plaintext-byte)
- $t = 3 : y = x \oplus k$ (XOR masked plaintext with key-byte)

The point in the power trace where $t = 1$ is “subtracted” from the point in the power trace where $t = 3$. The joint distribution of these two power samples allows to derive the key-byte bit by bit.

The adversary calculates the means of the two sets S_0 and S_1 (depending on the plaintext bit), and then the DoM is used again.

Thomas S. Messerges. “Using Second-Order Power Analysis to Attack DPA Resistant Software”, CHES 2000, pages 238–251, Springer, 2000.

Recall: Approaches to 2nd order attacks

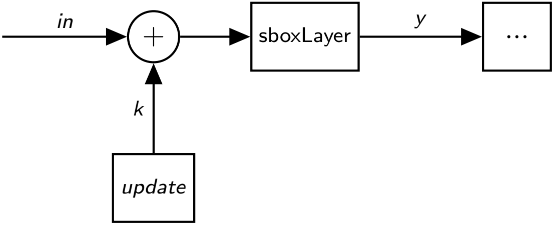
- It comes down to performing a pre-processing step and standard (1st-order) DPA-attacks
- Different options for the pre-processing function/step:
  - (absolute) difference the appropriate points in the power trace
  - multiplication the points
  - FFT
- Most of the approaches proven sound for the Hamming weight model and Gaussian noise

Recall: 2nd order DPA attack on a software impl.

The following relation holds:
HW $(a \oplus b) = |$HW(a) - HW(b)$|$.
Consequently, we can correctly predict $|$p(a)-p(b)$|$ with HW($a \oplus b$) and we do the attack as follows.

- Step 1: Fix an interval I for all power traces p_i. The interval is determined by an educated guess for the time frame in which two intermediate values $F_1(x_i) \oplus M_i$ and $F_2(x_i) \oplus M_i$ are computed.
  For each trace p_i we calculate a pre-processed trace that contains all values $|I_a - I_b|$
- Step 2: Perform standard (1st-order) DPA attack on the pre-processed power traces. With this attack, we guess a part of the key K to predict the value HW$(F_1(x_i) \oplus F_2(x_i))$. The value $|p(F_1(x_i) \oplus M_i) - p(F_2(x_i) \oplus M_i)|$ occurs in the pre-processed traces.

Template motivation example

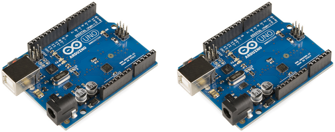


- Key k is refreshed before every encryption
- Does classical DPA work?
- No! It requires a constant key
- We rely on several assumptions (for attacks):
  1. We assume the leakage to be related to the Hamming weight (or distance)
  2. We look for leakage in the Sbox output only
  3. We assume the leakage to be univariate
- Ideally we would like to extract more leakage with minimal assumptions

Template Attack

- Attacking a well-protected device directly is hard
- We often do not get many traces with the same secret
- So we use an unprotected device of the same model

Figure: protected device (left), unprotected device (right)



- We **profile**, i.e. **template** the unprotected device
- We use the profile to break the protected device

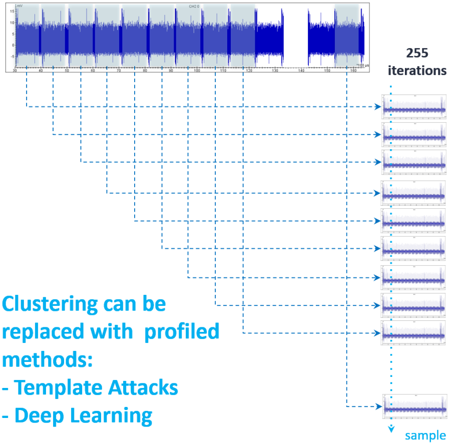
Template Attack Procedure

1. Choose a **model** that describes the power consumption
2. Profile the **unprotected device** to create the template (Template Building)
3. Use the template to break the **protected** device (Template Matching)

The same steps are always performed.

The **model** can be different.

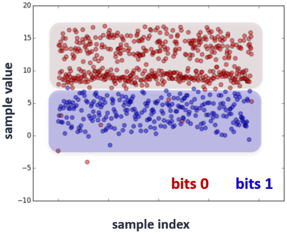
Example



Clustering can be replaced with profiled methods:

- **Template Attacks**
- **Deep Learning**

Apply clustering (**k-means**, for example) to the set of 255 samples:



[PC2015] G. Perin, Ł. Chmielewski, A Semi-Parametric Approach for Side-Channel Attacks on Protected RSA Implementations. Cardis 2015

Plan

1. Advanced Passive Attacks
   - Higher-Order Attacks
   - Profiled Attacks
2. Introductions/Recall on Fault Attacks
3. Recall on Differential Fault Analysis (RSA)
4. Real-world example: fault attacks on WolfSSL
5. Symmetric DFA
   - DFA on (Triple) DES
   - DFA on AES
6. Advanced Fault Injection Attacks (Selection)
   - Fault Injection And Side-Channel Analysis
   - Statistical Ineffective Fault Attacks
7. Practical Aspects of Fault Injection Attacks
8. The industry perspective
   - Evaluating Hardware attacks
9. `sec-certs`
10. Conclusions and Further Reading
    - Conclusions: SCA and FI
    - Conclusions: Business Perspective

Attack categories

- Side-channel attacks
  - use some physical (analog) characteristics
  - the target is running in normal conditions
- Faults: use abnormal conditions causing malfunctions in the system
- Micro-probing: accessing the chip surface directly in order to observe, learn and manipulate the device
- Reverse engineering

Types of implementation attacks

Active vs passive:

- Passive i.e. eavesdropping: the device operates within its specification
- Active i.e. tampering: the key is recovered by exploiting some abnormal behavior e.g. power glitches or laser pulses

Invasiveness:

- Non-invasive aka low-cost:
  - power/EM measurements
  - Coldboot attacks: data remanence in memories - cooling down is increasing the retention time
  - Rowhammer – is essentially a fault attack
- Semi-invasive: the device is de-packaged but no direct contact exists with the chip e.g. optical attacks
- Invasive aka expensive: the strongest type is bus probing

Methods

- Variation in supply voltage i.e. glitching
    - Can cause a processor skip instruction
    - Actively investigated by smartcard industry
    - So-called unloopers were used to activate the infinity loop in PayTV smartcards
- Variation in the external clock: may cause data misread or an instruction miss
- Change in temperature
    - The temperature threshold is defined for which the chip will work properly
    - Can cause changes in RAM content
- White light: photons induce faults
- X-rays and ion beams

Goals

- Insert computational fault
  - Null key
  - Wrong crypto result (Differential Fault Analysis - DFA)
- Change software decisions
  - Force approval of false PIN
  - Reverse life cycle state – PayTV and old phone cards
  - Enforce access rights
  - Break secure boot

Practical Fault Injection Aspects and what we concentrate on in this lecture

- Most common FI: voltage and EM (due to its price)
  - `https://github.com/newaetech/chipshouter-picoemp`
- Differential Fault Analysis (DFA)
  - We mention a few advanced recent methods that strongly relate to SCA
- Glitching decisions:
  - secure boot
  - obtaining memory dumps
  - enabling debug interfaces

DFA

- Bellcore attack in 1995
  - Differential faults on RSA-CRT signatures
  - Requires 1 correct and 1 wrong signature
- Attack on DES in 1997 by Biham and Shamir
- Special attacks on AES, ECC etc.
- Fault attacks on key transfer

DFA on cryptosystems

- Basic DFA scenario:
  - adversary obtains a pair of ciphertexts that are derived by encrypting the same plaintext (one is correct value and the other is faulty)
  - two encryptions are identical up to the point where the fault occurred
  - → two ciphertexts can be regarded as the outputs of a reduced-round iterated block cipher where the inputs are unknown but show a small (and possibly known) differential
- DFA on DES
  - the original attack of Biham and Shamir exploits computational errors occurring in the final rounds of the cipher
  - assumes that one bit of the right half of the DES internal state is flipped at a random position

RSA with CRT

Optimization of computing a signature giving about 4-fold speedup:
$n = p \cdot q \qquad$ Signature: $s = m^d \mod n$

Pre-computed values $d_p := d \mod (p-1) \qquad d_q := d \mod (q-1)$
$i_q := q^{-1} \mod p$

$s_p := m^{d_p} \mod p \qquad s_q := m^{d_q} \mod q$

Garner's method (1965) to recombine s_p and s_q:
$s = s_q + q \cdot (i_q(s_p - s_q) \mod p)$

Where to glitch?

Almost anywhere :-) computations of s_p and s_q.

If error is in s_p then the adversary can recover q as follows: $q = \gcd(n, s - \hat{s})$.

Plan

1. Advanced Passive Attacks
   - Higher-Order Attacks
   - Profiled Attacks
2. Introductions/Recall on Fault Attacks
3. Recall on Differential Fault Analysis (RSA)
4. Real-world example: fault attacks on WolfSSL
5. Symmetric DFA
   - DFA on (Triple) DES
   - DFA on AES
6. Advanced Fault Injection Attacks (Selection)
   - Fault Injection And Side-Channel Analysis
   - Statistical Ineffective Fault Attacks
7. Practical Aspects of Fault Injection Attacks
8. The industry perspective
   - Evaluating Hardware attacks
9. `sec-certs`
10. Conclusions and Further Reading
    - Conclusions: SCA and FI
    - Conclusions: Business Perspective

Ed25519

- Instance of EdDSA, which was proposed to “fix the unnecessary requirements on randomness” in ECDSA
- Does not depend on a “good” source of randomness, but instead derives a secret deterministically (hashing the msg and a long-term auxiliary key)
- Widely adopted by TLS1.3, Zcash, SSH, Tor, Signal, WolfSSL etc. (check “Things that use Ed25519”)
- Turns out to be easy to attacks in some real-world deployments i.e. WolfSSL

Niels Samwel, Lejla Batina, Guido Bertoni, Joan Daemen and Ruggero Susella: *Breaking Ed25519 in WolfSSL*, CTRSA2018.
Niels Samwel, Lejla Batina: *Practical Fault Injection on Deterministic Signatures: the Case of EdDSA*, Africacrypt 2018.

Ed25519

Algorithm 1 Ed25519 key setup and signature generation

```
Key setup.
Hash $k$ such that $H(k) = (h_0, h_1, \ldots, h_{2b-1}) = (a, b)$
$a = (h_0, \ldots, h_{b-1})$, Private scalar
$b = (h_b, \ldots, h_{2b-1})$, Auxiliary key
Compute public key: $A = aB$.
Signature generation.
Compute ephemeral private key: $r = H(b, M)$.
Compute ephemeral public key: $R = rB$.
Compute $h = H(R, A, M)$ and convert to integer.
Compute: $S = (r + ha) \mod l$.
Signature pair: $(R, S)$.
```

The Attack

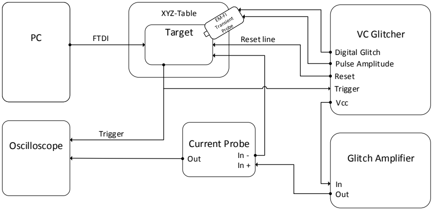
Two signatures, original (R, S) and faulty (R', S'):

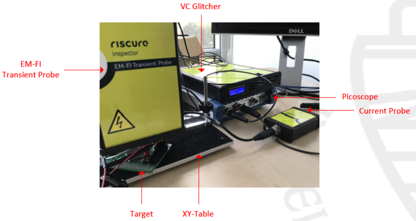
$$S = r + ha$$
$$S' = r + h'a$$

$$S - ha = S' - h'a$$

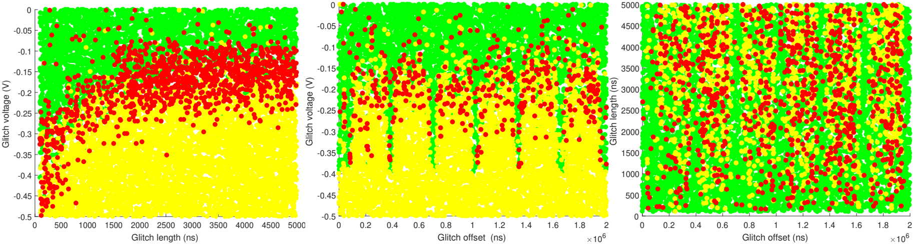
$$a = \frac{S - S'}{h - h'}$$

Setup


PC
FTDI
XYZ-Table
Target
EM-FI Transient Probe
Reset line
VC Glitcher
Digital Glitch
Pulse Amplitude
Reset
Trigger
Vcc
Oscilloscope
Trigger
Current Probe
Out
In -
In +
Glitch Amplifier
In
Out

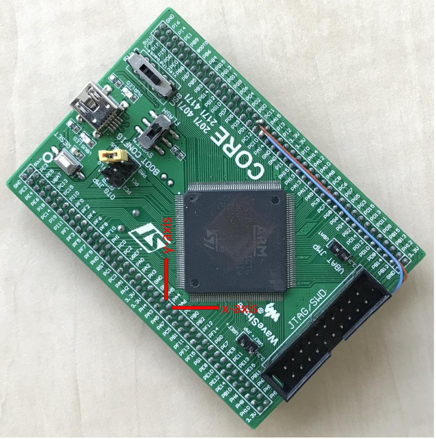

VC Glitcher
EM-FI
Transient Probe
Picoscope
Current Probe
Target
XY-Table

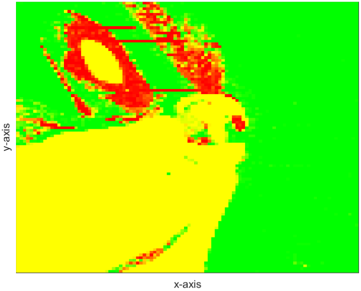
Results



Voltage fault injection results, Normal (green), Inconclusive (yellow), Successful (red).

Results


y-axis
x-axis
CORE 2071 4071 4171
JTAG/SWD
Waveshare


y-axis
x-axis

Conclusion

Two real physical side-channel attacks were actually performed against Ed25519

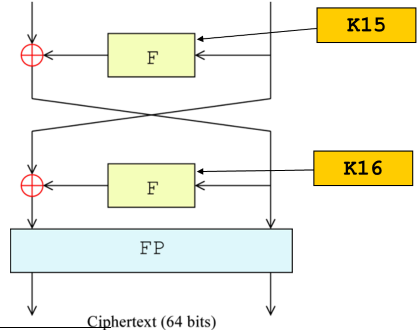
- Side-channel analysis of Ed25519 with 4 000 traces
- Fault injection on Ed25519 with 100% success rate for EM FI and 70% for voltage glitching out of 10 000 measurements
- For both attacks there exist inexpensive countermeasures

Plan

1. Advanced Passive Attacks
    - Higher-Order Attacks
    - Profiled Attacks
2. Introductions/Recall on Fault Attacks
3. Recall on Differential Fault Analysis (RSA)
4. Real-world example: fault attacks on WolfSSL
5. Symmetric DFA
    - DFA on (Triple) DES
    - DFA on AES
6. Advanced Fault Injection Attacks (Selection)
    - Fault Injection And Side-Channel Analysis
    - Statistical Ineffective Fault Attacks
7. Practical Aspects of Fault Injection Attacks
8. The industry perspective
    - Evaluating Hardware attacks
9. `sec-certs`
10. Conclusions and Further Reading
    - Conclusions: SCA and FI
    - Conclusions: Business Perspective

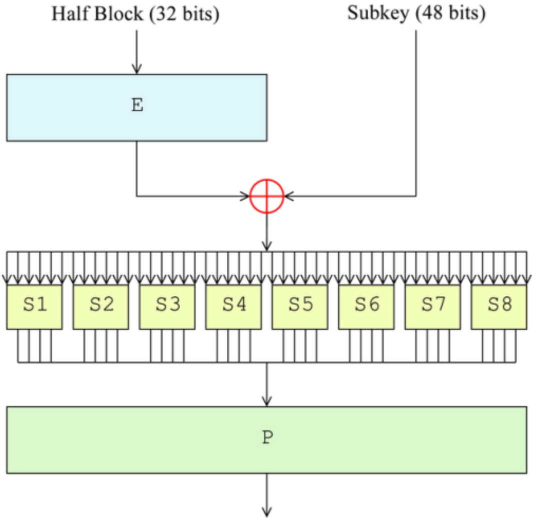
Final rounds of a DES encryption

- This part of the presentation is based on:
  - Eloi Sanfelix, Cristofaro Mune, and Job de Haas *Unboxing the White-Box Practical attacks against Obfuscated Ciphers*, Black Hat 2015.[1]
- Attack on Triple DES usually ≈ two (or three) attacks on DES
- Who remembers how DES works?
- How many rounds are there?
- How does DES finish?



[1] https://www.blackhat.com/docs/eu-15/materials/eu-15-Sanfelix-Unboxing-The-White-Box-Practical-Attacks-Against-Obfuscated-Ciphers-wp.pdf

The DES Feistel function, F



The Attack / Equations

- Recover one round key at a time, until the complete key can be computed.
- Recover the last round key (K_{16}) using a DFA attack by injecting faults during the execution of round 15.
- Using the correct and faulty output, we can write the following equations:

$$\begin{aligned} R_{16} &= F(R_{15}, K_{16}) \oplus L_{15}, \\ R'_{16} &= F(R'_{15}, K_{16}) \oplus L_{15}, \text{ where } K_{16} \text{ and } L_{15} \text{ are unknown.} \end{aligned}$$

- Combine the above equations to obtain the following:

$$R_{16} \oplus R'_{16} = F(R_{15}, K_{16}) \oplus F(R'_{15}, K_{16}), \text{ where only } K_{16} \text{ is unknown.}$$

- For each S-Box the following equation needs to be solved:

$$(P^{-1}(R_{16} \oplus R'_{16})))_i = S_i(E(R_{15}) \oplus K_{16,i}) \oplus S_i(E(R'_{15}) \oplus K_{16,i}),$$

where E and P represent the expansion and permutation in the F function.

The Attack: how to recover the key?

- Problems?
  - Typically a single equation results in a number of candidates for each affected sub-key (for each fault).
  - Sometimes when the faults are not injected as expected (by the attack) it is possible to discard a correct key.
- Counting strategy:
  - For each fault, compute the set of solutions to the equation and increase the count for the respective key-byte candidates.
  - When all faults are analyzed, the candidate with the highest count is selected as the correct candidate.
- Scores:

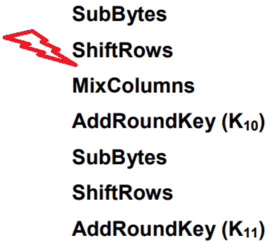
```
Processing round 16:
Best result S-Box 1:
0, sub key:  25 (0x19), value:  1.00000
1, sub key:  27 (0x1B), value:  0.43750
2, sub key:  17 (0x11), value:  0.38890
3, sub key:  24 (0x18), value:  0.33330
Best result S-Box 2:
...
```

The Attack: how to recover the key? cont'd

- When the last round key is known, the attack can be iterated to the previous round key. Why? The round key is 48-bits and the full key is 56-bits.
- Inject faults one round earlier and compute the output of the one but last round by using the recovered last round key.
- For 112-bit TDES: the same attack can be applied to the middle DES once the final DES broken.
- For 168-bit TDES (with three keys): the attack is iterated to the initial DES.

DFA High Level Attack

- Who remembers how AES works? How many rounds? How does AES-128 finish?



SubBytes	SubBytes
ShiftRows	ShiftRows
MixColumns	MixColumns
AddRoundKey (K_{10})	AddRoundKey (K_{10})
SubBytes	SubBytes
ShiftRows	ShiftRows
AddRoundKey (K_{11})	AddRoundKey (K_{11})

- This DFA on AES is also based on aforementioned Black Hat 2015 paper based on the attack from
  - P. Dusart, G. Letournex and O. Vivolo *Differential fault analysis on AES*, Springer, 2003.[2]
- The attack is more complex than for DES so refer to the papers for details or ask me.

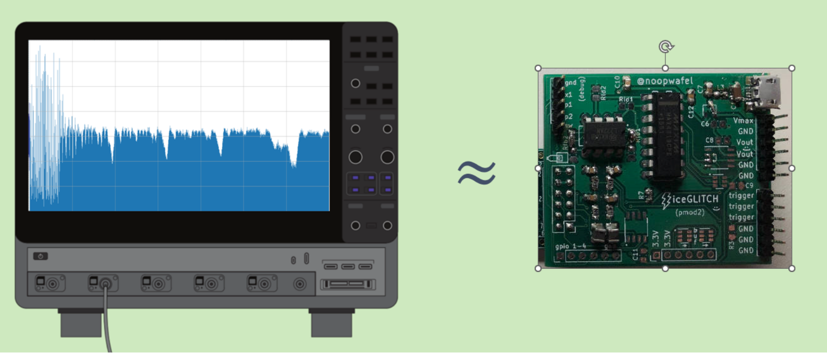
[2] https://eprint.iacr.org/2003/010.pdf

Plan

1. Advanced Passive Attacks
   - Higher-Order Attacks
   - Profiled Attacks
2. Introductions/Recall on Fault Attacks
3. Recall on Differential Fault Analysis (RSA)
4. Real-world example: fault attacks on WolfSSL
5. Symmetric DFA
   - DFA on (Triple) DES
   - DFA on AES
6. Advanced Fault Injection Attacks (Selection)
   - Fault Injection And Side-Channel Analysis
   - Statistical Ineffective Fault Attacks
7. Practical Aspects of Fault Injection Attacks
8. The industry perspective
   - Evaluating Hardware attacks
9. `sec-certs`
10. Conclusions and Further Reading
    - Conclusions: SCA and FI
    - Conclusions: Business Perspective

Fault Injection as an Oscilloscope

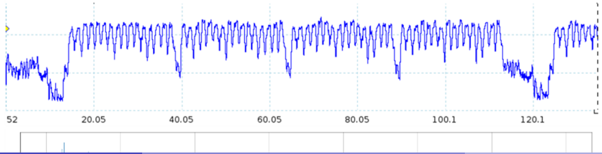
- This part of the presentation is based on:
  - Albert Spruyt, Alyssa Milburn, and Łukasz Chmielewski *Fault Injection as an Oscilloscope: Fault Correlation Analysis*, CHES 2021.[3]



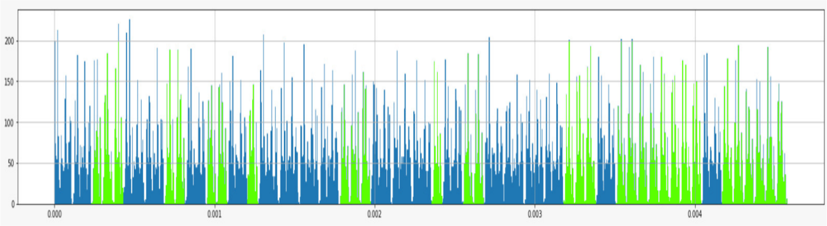
[3] https://tches.iacr.org/index.php/TCHES/article/view/8732/8332

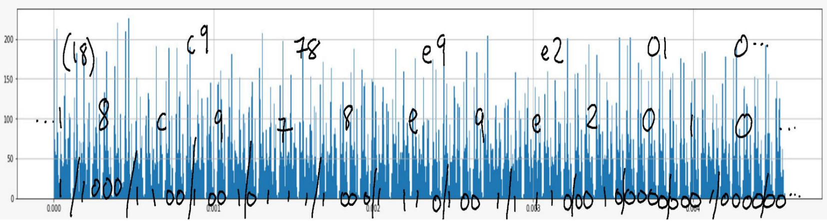
Constructing 'probability traces' from faults

- A fault probability is dependent on the data being processed by a device.
    - From: Yang Li, Kazuo Sakiyama, Shigeto Gomisawa, Toshinori Fukunaga, Junko Takahashi, and Kazuo Ohta *Fault sensitivity analysis*, CHES 2010. This paper shows how to exploit this leakage to recover the AES key.
    - That is true but operation leakage would also be visible!
    - A voltage FI device can be transformed into a 1-bit sampling oscilloscope!
- Let's see one AES power trace.
- Approximately 15k FI attempts per point in time for one AES FI barchart trace.

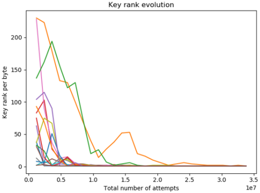


Simple Fault Analysis on RSA

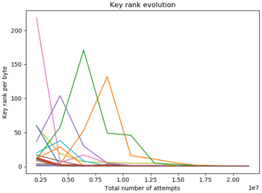




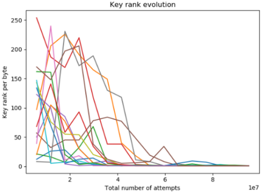
Fault Correlation Analysis on AES (1)



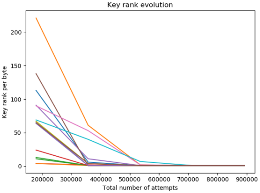
RISC-V simple AES: mutes



RISC-V simple AES: successes



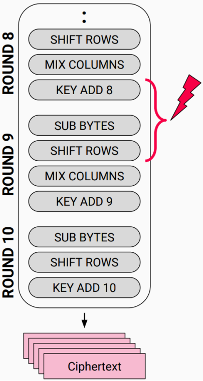
ARM simple AES: mutes



ARM simple AES: successes

Statistical Ineffective Fault Attacks (SIFA)

- Presented in:
    - C. Dobraunig, M. Eichlseder, T. Korak, S. Mangard, F. Mendel, and R. Primas *SIFA: Exploiting Ineffective Fault Inductions onSymmetric Cryptography*[4]
- SIFA combines older attacks
    - Ineffective Fault Attacks (Christophe Clavier, *Secret external encodings do not prevent transient fault analysis*, CHES 2010)
        - + Exploits only correct ciphertexts
        - − Requires precise faults with known effect
    - Statistical Fault Analysis (T. Fuhr, É. Jaulmes, V. Lomné, and A. Thillard. *Fault attacks on AES with faulty ciphertexts only*, FDTC 2013)
        - + Any fault, even if effect is unknown
        - − Mitigated by detection/infection
- SIFA exploits
    - only correct ciphertexts
    - any fault, even if effect is unknown
    - works against masked implementations
- For me see the paper or ask me.



[4] https://tches.iacr.org/index.php/TCHES/article/view/7286/6463

Plan

1. Advanced Passive Attacks
   - Higher-Order Attacks
   - Profiled Attacks
2. Introductions/Recall on Fault Attacks
3. Recall on Differential Fault Analysis (RSA)
4. Real-world example: fault attacks on WolfSSL
5. Symmetric DFA
   - DFA on (Triple) DES
   - DFA on AES
6. Advanced Fault Injection Attacks (Selection)
   - Fault Injection And Side-Channel Analysis
   - Statistical Ineffective Fault Attacks
7. Practical Aspects of Fault Injection Attacks
8. The industry perspective
   - Evaluating Hardware attacks
9. `sec-certs`
10. Conclusions and Further Reading
    - Conclusions: SCA and FI
    - Conclusions: Business Perspective

What is fault injection used for?

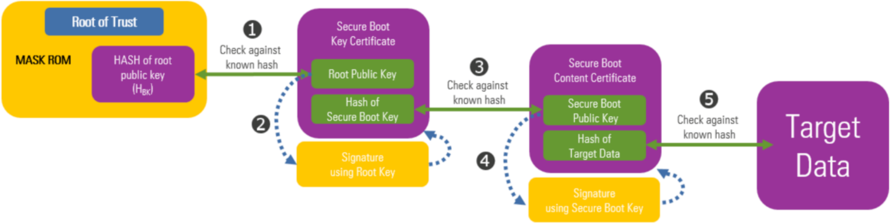
- Enabling JTAG
    - glitching check?
    - glitching registers?
    - changing setting during booting?
- glitching secure boot?
    - Side Channel Analysis Characterization +
    - Fault Injection
- Common: Voltage / EMFI
- More rare: Laser
    - Setting registers

What is fault injection used for? cont'd

- glitching communication for memory dumps
- wild jungle jump (ARM)
- FISim / other simulatiors
- DFA, RSA, SIFA, ...
- TEE security mechanisms?
- attacking (protected) secure ROMs
    - chain of trust, runtime control
- Settings: DDR/Clock, do we need FI?
- what to attack in boot process?

What to attack in boot process?

Automotive example[5]:



[5] Example taken from `https://www.renesas.com/us/en/blogs/achieving-root-trust-secure-boot-automotive-rh850-and-r-car-devices-part-3`

Comparison of various FIs - what we want

- Generic solutions with little assumptions
  - No ciphertext knowledge
- Attacks on generic target
  - No clock control
- Non-profiled
- How much are we willing to pay for generic attacks?
  All that flexibility comes at a price of large number of faults.

Plan

1. Advanced Passive Attacks
   - Higher-Order Attacks
   - Profiled Attacks
2. Introductions/Recall on Fault Attacks
3. Recall on Differential Fault Analysis (RSA)
4. Real-world example: fault attacks on WolfSSL
5. Symmetric DFA
   - DFA on (Triple) DES
   - DFA on AES
6. Advanced Fault Injection Attacks (Selection)
   - Fault Injection And Side-Channel Analysis
   - Statistical Ineffective Fault Attacks
7. Practical Aspects of Fault Injection Attacks
8. The industry perspective
   - Evaluating Hardware attacks
9. sec-certs
10. Conclusions and Further Reading
    - Conclusions: SCA and FI
    - Conclusions: Business Perspective

Why invest in a security evaluation?

Motivation for purchasing security evaluation services:

1. Have to, or the product cannot be sold;
2. Protect against potential future damage;
3. Competitive advantage;
4. Produce secure devices for the safety of their customers;

Security Certification

- Evidence that a products meets a set of given security requirements;
- Regulate access to certain markets: payment, content protection, government, etc
- Different security evaluation standards are available:
  - industry
  - product type: IC, OS, application
  - security requirements
  - geographical location
- Cost-effective:
  - recognition of certificates
  - pre-defined security requirements
  - pre-defined evaluation methodology
- Vendor liability

Short history:

Certification schemes timeline:

- (1994) Common Criteria: France, Germany, the Netherlands, UK
- (1994) FIPS 140-1, USA
- (1999) EMVco - payment industry
- (2001)FIPS 140-2
- (2019) FIPS 140-3
- (2019) Security Evaluation Standard for IoT Platforms (SESIP)
- ...

Common Criteria

- (1994) France, Germany, the Netherlands, UK
- (2022)
  - Certificate Authorizing Members: Australia, Canada, France, Germany, India, Italy, Japan, Malaysia, Netherlands, New Zeeland, Norway, Korea, Singapore, Spain, Sweden, Turkey, USA.
  - Certificate Consuming Members: Austria, Czech Republic, Denmark, Ethiopia, Finland, Greece,Hungary, Indonesia,Israel, Pakistan, Poland, Qatar, Slovak Republic, UK.
- Separation of the role of the certifier(national schemes) and evaluator(accredited commercial laboratories).
- The sponsor of the evaluation is the vendor.

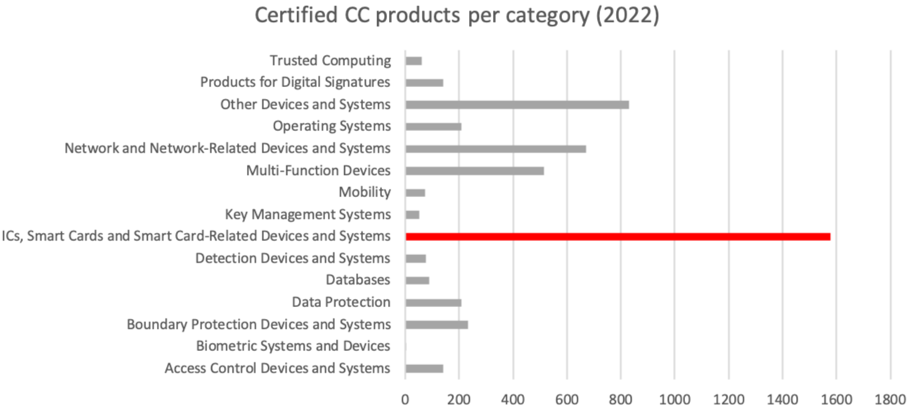
Objectives of CC evaluation

Stated [6] objective of CC evaluations:

1. to ensure that evaluations of Information Technology (IT) products and protection profiles are performed to high and consistent standards, and are seen to contribute significantly to confidence in the security of those products and profiles;
2. to improve the availability of evaluated, security-enhanced IT products and protection profiles;
3. to eliminate the burden of duplicating evaluations of IT products and protection profiles;
4. to continuously improve the efficiency and cost-effectiveness of the evaluation and certification/validation process for IT products and protection profiles.

[6] Arrangement on the Recognition of Common Criteria Certificates In the field of Information Technology Security, May 2000, https://www.commoncriteriaportal.org/files/operatingprocedures/cc-recarrange.pdf

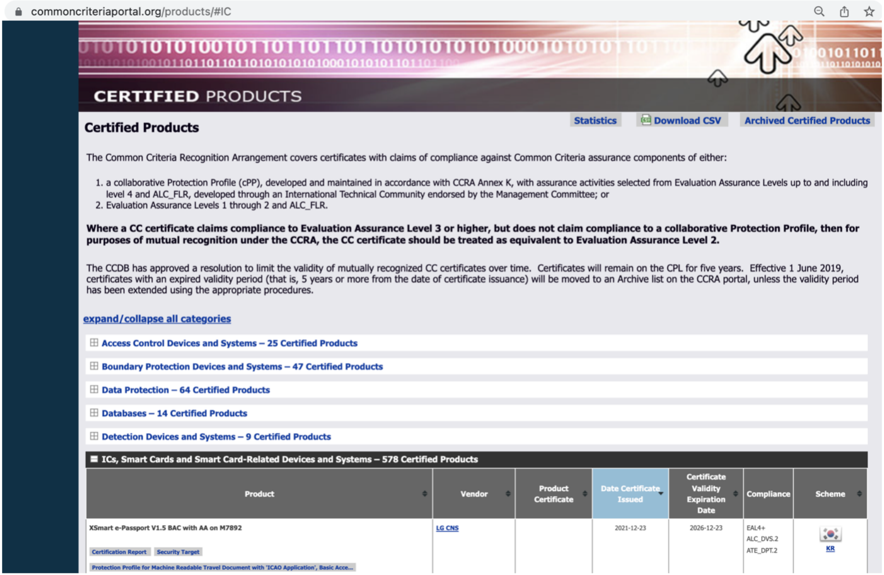
Which products?



The graph includes archived products (expired certificates).[7]

[7] data from: https://www.commoncriteriaportal.org/products/stats/

CC certificates are public



commoncriteriaportal.org/products/#IC

CERTIFIED PRODUCTS

Certified Products

The Common Criteria Recognition Arrangement covers certificates with claims of compliance against Common Criteria assurance components of either:

1. a collaborative Protection Profile (cPP), developed and maintained in accordance with CCRA Annex K, with assurance activities selected from Evaluation Assurance Levels up to and including level 4 and ALC_FLR, developed through an International Technical Community endorsed by the Management Committee; or
2. Evaluation Assurance Levels 1 through 2 and ALC_FLR.

Where a CC certificate claims compliance to Evaluation Assurance Level 3 or higher, but does not claim compliance to a collaborative Protection Profile, then for purposes of mutual recognition under the CCRA, the CC certificate should be treated as equivalent to Evaluation Assurance Level 2.

The CCDB has approved a resolution to limit the validity of mutually recognized CC certificates over time. Certificates will remain on the CPL for five years. Effective 1 June 2019, certificates with an expired validity period (that is, 5 years or more from the date of certificate issuance) will be moved to an Archive list on the CCRA portal, unless the validity period has been extended using the appropriate procedures.

expand/collapse all categories

⊞ Access Control Devices and Systems – 25 Certified Products

⊞ Boundary Protection Devices and Systems – 47 Certified Products

⊞ Data Protection – 64 Certified Products

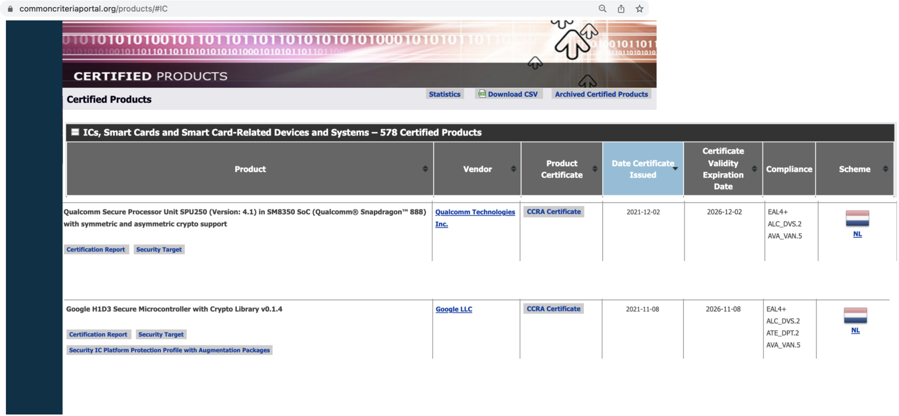
⊞ Databases – 14 Certified Products

⊞ Detection Devices and Systems – 9 Certified Products

⊟ ICs, Smart Cards and Smart Card-Related Devices and Systems – 578 Certified Products

Product	Vendor	Product Certificate	Date Certificate Issued	Certificate Validity Expiration Date	Compliance	Scheme
XSmart e-Passport V1.5 BAC with AA on M7892 Certification Report Security Target Protection Profile for Machine Readable Travel Document with 'ICAO Application', Basic Acce...	LG CNS		2021-12-23	2026-12-23	EAL4+ ALC_DVS.2 ATE_DPT.2	KR

Recent evaluations



commoncriteriaportal.org/products/#IC
CERTIFIED PRODUCTS

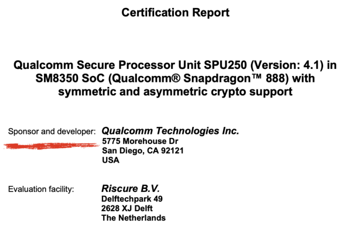
Certified Products

Statistics
Download CSV
Archived Certified Products

ICs, Smart Cards and Smart Card-Related Devices and Systems – 578 Certified Products

Product	Vendor	Product Certificate	Date Certificate Issued	Certificate Validity Expiration Date	Compliance	Scheme
Qualcomm Secure Processor Unit SPU250 (Version: 4.1) in SM8350 SoC (Qualcomm® Snapdragon™ 888) with symmetric and asymmetric crypto support Certification Report Security Target	Qualcomm Technologies Inc.	CCRA Certificate	2021-12-02	2026-12-02	EAL4+ ALC_DVS.2 AVA_VAN.5	NL
Google H1D3 Secure Microcontroller with Crypto Library v0.1.4 Certification Report Security Target Security IC Platform Protection Profile with Augmentation Packages	Google LLC	CCRA Certificate	2021-11-08	2026-11-08	EAL4+ ALC_DVS.2 ATE_DPT.2 AVA_VAN.5	NL

Inside a CC certificate



Certification Report

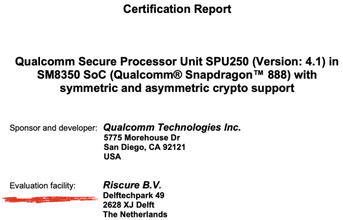
Qualcomm Secure Processor Unit SPU250 (Version: 4.1) in SM8350 SoC (Qualcomm® Snapdragon™ 888) with symmetric and asymmetric crypto support

Sponsor and developer: ***Qualcomm Technologies Inc.***
5775 Morehouse Dr
San Diego, CA 92121
USA

Evaluation facility: ***Riscure B.V.***
Delftechpark 49
2628 XJ Delft
The Netherlands

Qualcomm Secure Processor Unit SPU250 (Version: 4.1) in SM8350 SoC (Qualcomm® Snapdragon™ 888) with symmetric and asymmetric crypto support, 2 December 2021, `https://www.commoncriteriaportal.org/files/epfiles/NSCIB-CC-0227918-CR.pdf`

Inside a CC certificate



Certification Report

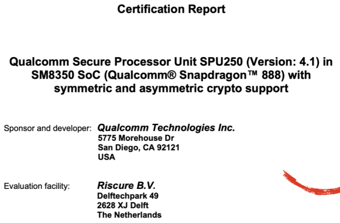
Qualcomm Secure Processor Unit SPU250 (Version: 4.1) in SM8350 SoC (Qualcomm® Snapdragon™ 888) with symmetric and asymmetric crypto support

Sponsor and developer: ***Qualcomm Technologies Inc.***
5775 Morehouse Dr
San Diego, CA 92121
USA

Evaluation facility: ***Riscure B.V.***
Delftechpark 49
2628 XJ Delft
The Netherlands

Qualcomm Secure Processor Unit SPU250 (Version: 4.1) in SM8350 SoC (Qualcomm® Snapdragon™ 888) with symmetric and asymmetric crypto support, 2 December 2021, `https://www.commoncriteriaportal.org/files/epfiles/NSCIB-CC-0227918-CR.pdf`

Inside a CC certificate

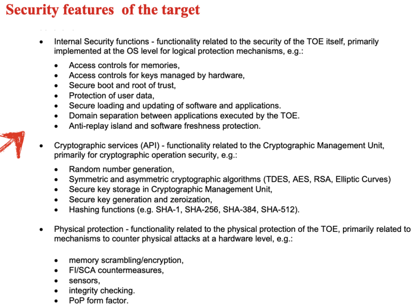


Certification Report

Qualcomm Secure Processor Unit SPU250 (Version: 4.1) in SM8350 SoC (Qualcomm® Snapdragon™ 888) with symmetric and asymmetric crypto support

Sponsor and developer: ***Qualcomm Technologies Inc.***
5775 Morehouse Dr
San Diego, CA 92121
USA

Evaluation facility: ***Riscure B.V.***
Delftechpark 49
2628 XJ Delft
The Netherlands

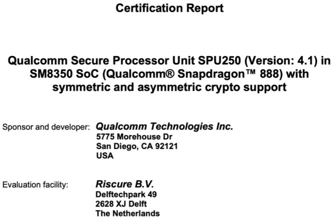


Security features of the target

- Internal Security functions - functionality related to the security of the TOE itself, primarily implemented at the OS level for logical protection mechanisms, e.g.:
  - Access controls for memories,
  - Access controls for keys managed by hardware,
  - Secure boot and root of trust,
  - Protection of user data,
  - Secure loading and updating of software and applications.
  - Domain separation between applications executed by the TOE.
  - Anti-replay island and software freshness protection.
- Cryptographic services (API) - functionality related to the Cryptographic Management Unit, primarily for cryptographic operation security, e.g.:
  - Random number generation,
  - Symmetric and asymmetric cryptographic algorithms (TDES, AES, RSA, Elliptic Curves)
  - Secure key storage in Cryptographic Management Unit,
  - Secure key generation and zeroization,
  - Hashing functions (e.g. SHA-1, SHA-256, SHA-384, SHA-512).
- Physical protection - functionality related to the physical protection of the TOE, primarily related to mechanisms to counter physical attacks at a hardware level, e.g.:
  - memory scrambling/encryption,
  - FI/SCA countermeasures,
  - sensors,
  - integrity checking.
  - PoP form factor.

Qualcomm Secure Processor Unit SPU250 (Version: 4.1) in SM8350 SoC (Qualcomm® Snapdragon™ 888) with symmetric and asymmetric crypto support, 2 December 2021, https://www.commoncriteriaportal.org/files/epfiles/NSCIB-CC-0227918-CR.pdf

Inside a CC certificate

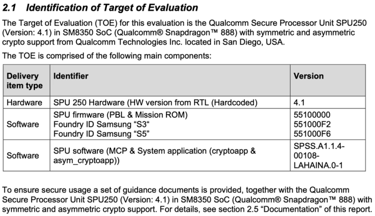


Certification Report

Qualcomm Secure Processor Unit SPU250 (Version: 4.1) in SM8350 SoC (Qualcomm® Snapdragon™ 888) with symmetric and asymmetric crypto support

Sponsor and developer: ***Qualcomm Technologies Inc.***
5775 Morehouse Dr
San Diego, CA 92121
USA

Evaluation facility: ***Riscure B.V.***
Delftechpark 49
2628 XJ Delft
The Netherlands



2.1 Identification of Target of Evaluation

The Target of Evaluation (TOE) for this evaluation is the Qualcomm Secure Processor Unit SPU250 (Version: 4.1) in SM8350 SoC (Qualcomm® Snapdragon™ 888) with symmetric and asymmetric crypto support from Qualcomm Technologies Inc. located in San Diego, USA.

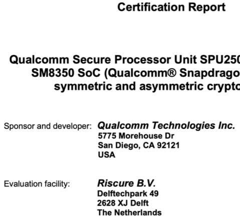
The TOE is comprised of the following main components:

Delivery item type	Identifier	Version
Hardware	SPU 250 Hardware (HW version from RTL (Hardcoded)	4.1
Software	SPU firmware (PBL & Mission ROM) Foundry ID Samsung "S3" Foundry ID Samsung "S5"	55100000 551000F2 551000F6
Software	SPU software (MCP & System application (cryptoapp & asym_cryptoapp))	SPSS.A1.1.4-00108-LAHAINA.0-1

To ensure secure usage a set of guidance documents is provided, together with the Qualcomm Secure Processor Unit SPU250 (Version: 4.1) in SM8350 SoC (Qualcomm® Snapdragon™ 888) with symmetric and asymmetric crypto support. For details, see section 2.5 "Documentation" of this report.

Qualcomm Secure Processor Unit SPU250 (Version: 4.1) in SM8350 SoC (Qualcomm® Snapdragon™ 888) with symmetric and asymmetric crypto support, 2 December 2021, `https://www.commoncriteriaportal.org/files/epfiles/NSCIB-CC-0227918-CR.pdf`

Inside a CC certificate

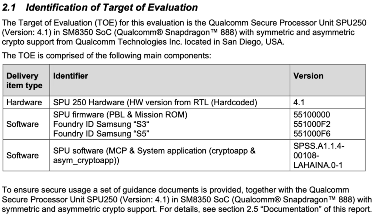


Certification Report

Qualcomm Secure Processor Unit SPU250 (Version: 4.1) in SM8350 SoC (Qualcomm® Snapdragon™ 888) with symmetric and asymmetric crypto support

Sponsor and developer: ***Qualcomm Technologies Inc.***
5775 Morehouse Dr
San Diego, CA 92121
USA

Evaluation facility: ***Riscure B.V.***
Delftechpark 49
2628 XJ Delft
The Netherlands



2.1 Identification of Target of Evaluation

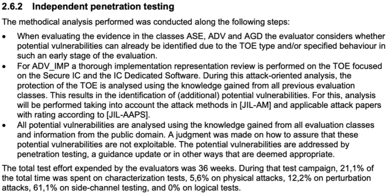
The Target of Evaluation (TOE) for this evaluation is the Qualcomm Secure Processor Unit SPU250 (Version: 4.1) in SM8350 SoC (Qualcomm® Snapdragon™ 888) with symmetric and asymmetric crypto support from Qualcomm Technologies Inc. located in San Diego, USA.

The TOE is comprised of the following main components:

Delivery item type	Identifier	Version
Hardware	SPU 250 Hardware (HW version from RTL (Hardcoded)	4.1
Software	SPU firmware (PBL & Mission ROM) Foundry ID Samsung "S3" Foundry ID Samsung "S5"	55100000 551000F2 551000F6
Software	SPU software (MCP & System application (cryptoapp & asym_cryptoapp))	SPSS.A1.1.4-00108-LAHAINA.0-1

To ensure secure usage a set of guidance documents is provided, together with the Qualcomm Secure Processor Unit SPU250 (Version: 4.1) in SM8350 SoC (Qualcomm® Snapdragon™ 888) with symmetric and asymmetric crypto support. For details, see section 2.5 "Documentation" of this report.

Details about the evalution



2.6.2 Independent penetration testing

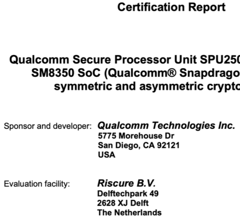
The methodical analysis performed was conducted along the following steps:

- When evaluating the evidence in the classes ASE, ADV and AGD the evaluator considers whether potential vulnerabilities can already be identified due to the TOE type and/or specified behaviour in such an early stage of the evaluation.
- For ADV_IMP a thorough implementation representation review is performed on the TOE focused on the Secure IC and the IC Dedicated Software. During this attack-oriented analysis, the protection of the TOE is analysed using the knowledge gained from all previous evaluation classes. This results in the identification of (additional) potential vulnerabilities. For this, analysis will be performed taking into account the attack methods in [JIL-AM] and applicable attack papers with rating according to [JIL-AAPS].
- All potential vulnerabilities are analysed using the knowledge gained from all evaluation classes and information from the public domain. A judgment was made on how to assure that these potential vulnerabilities are not exploitable. The potential vulnerabilities are addressed by penetration testing, a guidance update or in other ways that are deemed appropriate.

The total test effort expended by the evaluators was 36 weeks. During that test campaign, 21,1% of the total time was spent on characterization tests, 5,6% on physical attacks, 12,2% on perturbation attacks, 61,1% on side-channel testing, and 0% on logical tests.

Qualcomm Secure Processor Unit SPU250 (Version: 4.1) in SM8350 SoC (Qualcomm® Snapdragon™ 888) with symmetric and asymmetric crypto support, 2 December 2021, https://www.commoncriteriaportal.org/files/epfiles/NSCIB-CC-0227918-CR.pdf

Inside a CC certificate

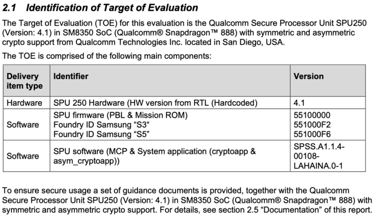


Certification Report

Qualcomm Secure Processor Unit SPU250 (Version: 4.1) in SM8350 SoC (Qualcomm® Snapdragon™ 888) with symmetric and asymmetric crypto support

Sponsor and developer: ***Qualcomm Technologies Inc.***
5775 Morehouse Dr
San Diego, CA 92121
USA

Evaluation facility: ***Riscure B.V.***
Delftechpark 49
2628 XJ Delft
The Netherlands



2.1 Identification of Target of Evaluation

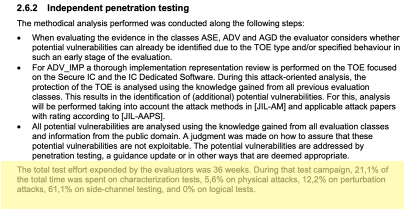
The Target of Evaluation (TOE) for this evaluation is the Qualcomm Secure Processor Unit SPU250 (Version: 4.1) in SM8350 SoC (Qualcomm® Snapdragon™ 888) with symmetric and asymmetric crypto support from Qualcomm Technologies Inc. located in San Diego, USA.

The TOE is comprised of the following main components:

Delivery item type	Identifier	Version
Hardware	SPU 250 Hardware (HW version from RTL (Hardcoded)	4.1
Software	SPU firmware (PBL & Mission ROM) Foundry ID Samsung “S3“ Foundry ID Samsung “S5”	55100000 551000F2 551000F6
Software	SPU software (MCP & System application (cryptoapp & asym_cryptoapp))	SPSS.A1.1.4-00108-LAHAINA.0-1

To ensure secure usage a set of guidance documents is provided, together with the Qualcomm Secure Processor Unit SPU250 (Version: 4.1) in SM8350 SoC (Qualcomm® Snapdragon™ 888) with symmetric and asymmetric crypto support. For details, see section 2.5 “Documentation” of this report.

Details about the evalution



2.6.2 Independent penetration testing

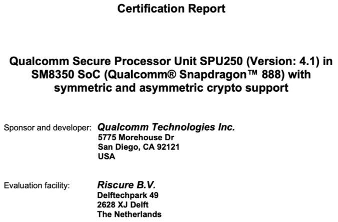
The methodical analysis performed was conducted along the following steps:

- When evaluating the evidence in the classes ASE, ADV and AGD the evaluator considers whether potential vulnerabilities can already be identified due to the TOE type and/or specified behaviour in such an early stage of the evaluation.
- For ADV_IMP a thorough implementation representation review is performed on the TOE focused on the Secure IC and the IC Dedicated Software. During this attack-oriented analysis, the protection of the TOE is analysed using the knowledge gained from all previous evaluation classes. This results in the identification of (additional) potential vulnerabilities. For this, analysis will be performed taking into account the attack methods in [JIL-AM] and applicable attack papers with rating according to [JIL-AAPS].
- All potential vulnerabilities are analysed using the knowledge gained from all evaluation classes and information from the public domain. A judgment was made on how to assure that these potential vulnerabilities are not exploitable. The potential vulnerabilities are addressed by penetration testing, a guidance update or in other ways that are deemed appropriate.

The total test effort expended by the evaluators was 36 weeks. During that test campaign, 21,1% of the total time was spent on characterization tests, 5,6% on physical attacks, 12,2% on perturbation attacks, 61,1% on side-channel testing, and 0% on logical tests.

Qualcomm Secure Processor Unit SPU250 (Version: 4.1) in SM8350 SoC (Qualcomm® Snapdragon™ 888) with symmetric and asymmetric crypto support, 2 December 2021, `https://www.commoncriteriaportal.org/files/epfiles/NSCIB-CC-0227918-CR.pdf`

Inside a CC certificate

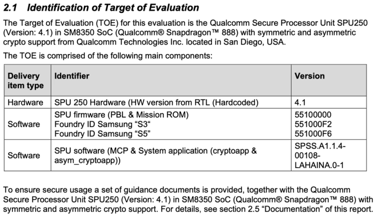


Certification Report

Qualcomm Secure Processor Unit SPU250 (Version: 4.1) in SM8350 SoC (Qualcomm® Snapdragon™ 888) with symmetric and asymmetric crypto support

Sponsor and developer: ***Qualcomm Technologies Inc.***
5775 Morehouse Dr
San Diego, CA 92121
USA

Evaluation facility: ***Riscure B.V.***
Delftechpark 49
2628 XJ Delft
The Netherlands



2.1 Identification of Target of Evaluation

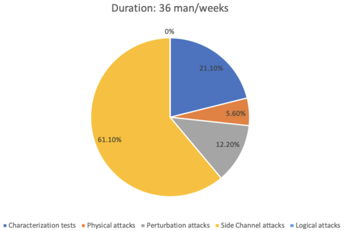
The Target of Evaluation (TOE) for this evaluation is the Qualcomm Secure Processor Unit SPU250 (Version: 4.1) in SM8350 SoC (Qualcomm® Snapdragon™ 888) with symmetric and asymmetric crypto support from Qualcomm Technologies Inc. located in San Diego, USA.

The TOE is comprised of the following main components:

Delivery item type	**Identifier**	**Version**
Hardware	SPU 250 Hardware (HW version from RTL (Hardcoded)	4.1
Software	SPU firmware (PBL & Mission ROM) Foundry ID Samsung “S3“ Foundry ID Samsung “S5”	55100000 551000F2 551000F6
Software	SPU software (MCP & System application (cryptoapp & asym_cryptoapp))	SPSS.A1.1.4-00108-LAHAINA.0-1

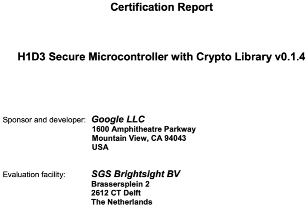
To ensure secure usage a set of guidance documents is provided, together with the Qualcomm Secure Processor Unit SPU250 (Version: 4.1) in SM8350 SoC (Qualcomm® Snapdragon™ 888) with symmetric and asymmetric crypto support. For details, see section 2.5 “Documentation” of this report.

Details about the evalution



Qualcomm Secure Processor Unit SPU250 (Version: 4.1) in SM8350 SoC (Qualcomm® Snapdragon™ 888) with symmetric and asymmetric crypto support, 2 December 2021, `https://www.commoncriteriaportal.org/files/epfiles/NSCIB-CC-0227918-CR.pdf`

Inside a CC certificate



Certification Report

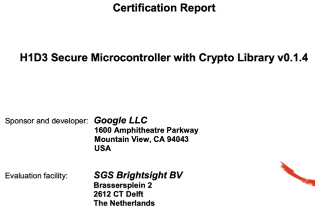
H1D3 Secure Microcontroller with Crypto Library v0.1.4

Sponsor and developer: ***Google LLC***
1600 Amphitheatre Parkway
Mountain View, CA 94043
USA

Evaluation facility: ***SGS Brightsight BV***
Brassersplein 2
2612 CT Delft
The Netherlands

Certification Report H1D3 Secure Microcontroller with Crypto Library v0.1.4, 12 November 2021, `https://www.commoncriteriaportal.org/files/epfiles/NSCIB-CC-0228971-CR.pdf`

Inside a CC certificate



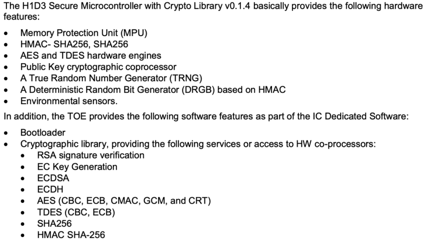
Certification Report

H1D3 Secure Microcontroller with Crypto Library v0.1.4

Sponsor and developer: ***Google LLC***
1600 Amphitheatre Parkway
Mountain View, CA 94043
USA

Evaluation facility: ***SGS Brightsight BV***
Brassersplein 2
2612 CT Delft
The Netherlands

Security features of the target



The H1D3 Secure Microcontroller with Crypto Library v0.1.4 basically provides the following hardware features:

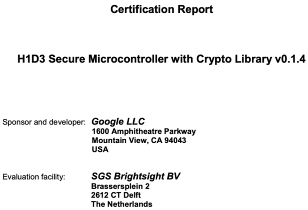
- Memory Protection Unit (MPU)
- HMAC- SHA256, SHA256
- AES and TDES hardware engines
- Public Key cryptographic coprocessor
- A True Random Number Generator (TRNG)
- A Deterministic Random Bit Generator (DRGB) based on HMAC
- Environmental sensors.

In addition, the TOE provides the following software features as part of the IC Dedicated Software:

- Bootloader
- Cryptographic library, providing the following services or access to HW co-processors:
  - RSA signature verification
  - EC Key Generation
  - ECDSA
  - ECDH
  - AES (CBC, ECB, CMAC, GCM, and CRT)
  - TDES (CBC, ECB)
  - SHA256
  - HMAC SHA-256

Certification Report H1D3 Secure Microcontroller with Crypto Library v0.1.4, 12 November 2021, `https://www.commoncriteriaportal.org/files/epfiles/NSCIB-CC-0228971-CR.pdf`

Inside a CC certificate



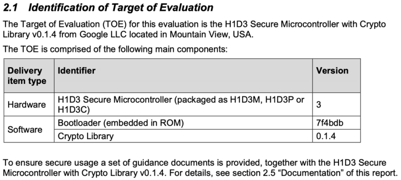
Certification Report

H1D3 Secure Microcontroller with Crypto Library v0.1.4

Sponsor and developer: ***Google LLC***
1600 Amphitheatre Parkway
Mountain View, CA 94043
USA

Evaluation facility: ***SGS Brightsight BV***
Brassersplein 2
2612 CT Delft
The Netherlands

TOE



2.1 Identification of Target of Evaluation

The Target of Evaluation (TOE) for this evaluation is the H1D3 Secure Microcontroller with Crypto Library v0.1.4 from Google LLC located in Mountain View, USA.

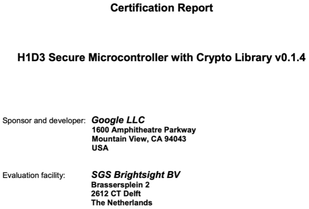
The TOE is comprised of the following main components:

Delivery item type	Identifier	Version
Hardware	H1D3 Secure Microcontroller (packaged as H1D3M, H1D3P or H1D3C)	3
Software	Bootloader (embedded in ROM)	7f4bdb
	Crypto Library	0.1.4

To ensure secure usage a set of guidance documents is provided, together with the H1D3 Secure Microcontroller with Crypto Library v0.1.4. For details, see section 2.5 “Documentation” of this report.

Certification Report H1D3 Secure Microcontroller with Crypto Library v0.1.4, 12 November 2021, `https://www.commoncriteriaportal.org/files/epfiles/NSCIB-CC-0228971-CR.pdf`

Inside a CC certificate

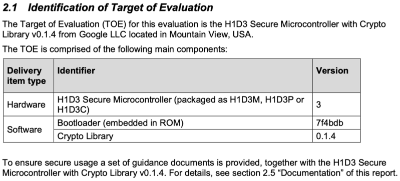


Certification Report

H1D3 Secure Microcontroller with Crypto Library v0.1.4

Sponsor and developer: ***Google LLC***
1600 Amphitheatre Parkway
Mountain View, CA 94043
USA

Evaluation facility: ***SGS Brightsight BV***
Brassersplein 2
2612 CT Delft
The Netherlands



2.1 Identification of Target of Evaluation

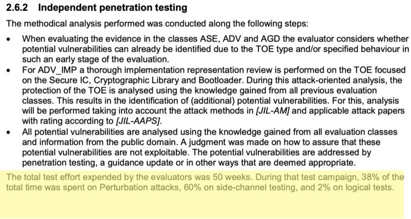
The Target of Evaluation (TOE) for this evaluation is the H1D3 Secure Microcontroller with Crypto Library v0.1.4 from Google LLC located in Mountain View, USA.

The TOE is comprised of the following main components:

Delivery item type	**Identifier**	**Version**
Hardware	H1D3 Secure Microcontroller (packaged as H1D3M, H1D3P or H1D3C)	3
Software	Bootloader (embedded in ROM)	7f4bdb
	Crypto Library	0.1.4

To ensure secure usage a set of guidance documents is provided, together with the H1D3 Secure Microcontroller with Crypto Library v0.1.4. For details, see section 2.5 "Documentation" of this report.

Details about the evalution



2.6.2 Independent penetration testing

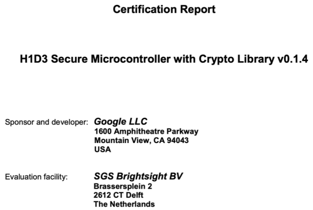
The methodical analysis performed was conducted along the following steps:

- When evaluating the evidence in the classes ASE, ADV and AGD the evaluator considers whether potential vulnerabilities can already be identified due to the TOE type and/or specified behaviour in such an early stage of the evaluation.
- For ADV_IMP a thorough implementation representation review is performed on the TOE focused on the Secure IC, Cryptographic Library and Bootloader. During this attack-oriented analysis, the protection of the TOE is analysed using the knowledge gained from all previous evaluation classes. This results in the identification of (additional) potential vulnerabilities. For this, analysis will be performed taking into account the attack methods in *[JIL-AM]* and applicable attack papers with rating according to *[JIL-AAPS]*.
- All potential vulnerabilities are analysed using the knowledge gained from all evaluation classes and information from the public domain. A judgment was made on how to assure that these potential vulnerabilities are not exploitable. The potential vulnerabilities are addressed by penetration testing, a guidance update or in other ways that are deemed appropriate.

The total test effort expended by the evaluators was 50 weeks. During that test campaign, 38% of the total time was spent on Perturbation attacks, 60% on side-channel testing, and 2% on logical tests.

Certification Report H1D3 Secure Microcontroller with Crypto Library v0.1.4, 12 November 2021, `https://www.commoncriteriaportal.org/files/epfiles/NSCIB-CC-0228971-CR.pdf`

Inside a CC certificate

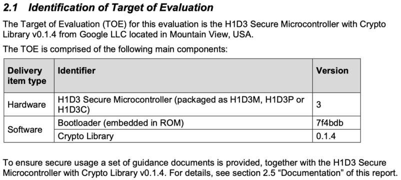


Certification Report

H1D3 Secure Microcontroller with Crypto Library v0.1.4

Sponsor and developer: ***Google LLC***
1600 Amphitheatre Parkway
Mountain View, CA 94043
USA

Evaluation facility: ***SGS Brightsight BV***
Brassersplein 2
2612 CT Delft
The Netherlands



2.1 Identification of Target of Evaluation

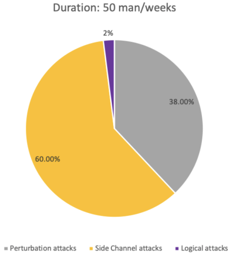
The Target of Evaluation (TOE) for this evaluation is the H1D3 Secure Microcontroller with Crypto Library v0.1.4 from Google LLC located in Mountain View, USA.

The TOE is comprised of the following main components:

Delivery item type	Identifier	Version
Hardware	H1D3 Secure Microcontroller (packaged as H1D3M, H1D3P or H1D3C)	3
Software	Bootloader (embedded in ROM)	7f4bdb
	Crypto Library	0.1.4

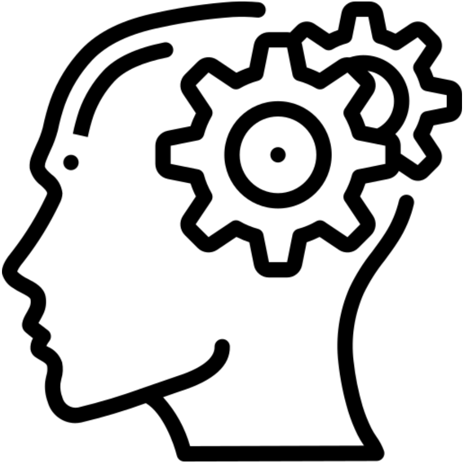
To ensure secure usage a set of guidance documents is provided, together with the H1D3 Secure Microcontroller with Crypto Library v0.1.4. For details, see section 2.5 "Documentation" of this report.

Details about the evalution



Certification Report H1D3 Secure Microcontroller with Crypto Library v0.1.4, 12 November 2021, `https://www.commoncriteriaportal.org/files/epfiles/NSCIB-CC-0228971-CR.pdf`

What are the merits/criticism of a CC evaluation?



CC evaluations in a nutshell

A CC certificate cannot guarantee security, but ensures that claims about security are independently verified.

Merits:

- White-box evaluations;
- Attack based evaluation;
- Recognized in multiple markets;
- Vetted evaluation labs;

Criticism:

- Static, certificate valid for the version of HW and SW that was evaluated;
- Formal, a lot of documentation to be reviewed;
- Expensive;

A CC certificate can only be withdrawn when it was issued under misconception, e.g., when it turns out that wrong evidence was submitted, **not if** a vulnerability is found.

What is EMVco?

https://www.youtube.com/watch?v=g1aSWgq0l8s

EMV chip

- Europay Mastercard Visa - first published in 1996
- EMVchip has three key elements:
  1. it can perform processing
  2. can store confidential information very securely
  3. can perform cryptographic processing
- EMV certification
  - similar to CC certification
  - keyword: secure composability: chip, OS, application;
  - accredited labs;
  - manufacturers are sponsors of an evaluation
  - certification body are private companies

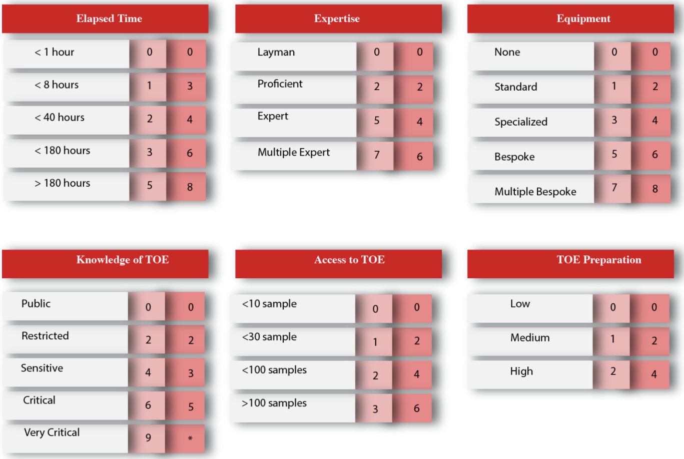
Beyond EMV chip

Notes:

- Online payment world, the Chip&Pin are becoming obsolete
- Support for different integrated payment methods
- Mobile Payment, Payment Tokenization, Wearables, EMV3D
- Technology evolves: HCE, TEE, etc.;
- Secure storage of credentials is hardware related;

Evaluating Hardware Attacks

Joint Interpretation Library describes how to objectively express the effort required to mount a successful attack.[8]



Elapsed Time		
< 1 hour	0	0
< 8 hours	1	3
< 40 hours	2	4
< 180 hours	3	6
> 180 hours	5	8

Expertise		
Layman	0	0
Proficient	2	2
Expert	5	4
Multiple Expert	7	6

Equipment		
None	0	0
Standard	1	2
Specialized	3	4
Bespoke	5	6
Multiple Bespoke	7	8

Knowledge of TOE		
Public	0	0
Restricted	2	2
Sensitive	4	3
Critical	6	5
Very Critical	9	*

Access to TOE		
<10 sample	0	0
<30 sample	1	2
<100 samples	2	4
>100 samples	3	6

TOE Preparation		
Low	0	0
Medium	1	2
High	2	4

[8] SOG-IS: Attack methods for smartcards and similar devices (2020), https://www.sogis.eu/documents/cc/domains/sc/JIL-Application-of-Attack-Potential-to-Smartcards-v3-1.pdf

The cost of a (potential) DPA attack

	Identification		Exploitation	
Elapsed time	$<$ one week	2	$<$one day	3
Expertise	Expert	5	Proficient	2
Equipment	Specialized	3	Specialized	4
Knowledge of TOE	Public	0	Public	0
Access to TOE	$<$ 10 samples	0	$<$ 10 samples	0
ToE Preparation	Medium	1	Low	0
Subtotal		11		9

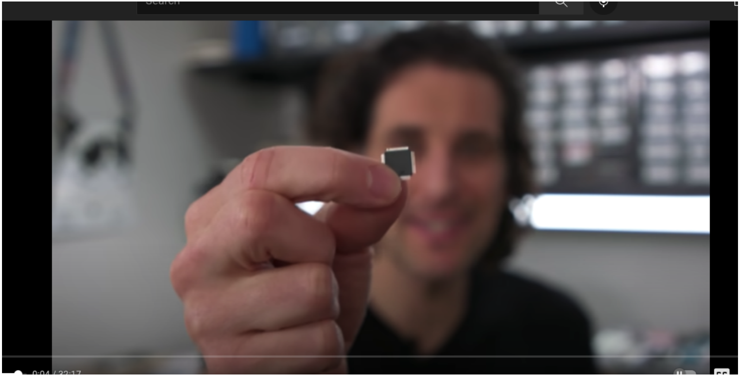
Total attack potential $= 20$;

Range of values	TOE resistant to attackers with potential of:
0-15	No rating
16-20	Basic
21-24	Enhanced -Basic
15-30	Moderate
31 and above	High

CROCS Centre for Research on Cryptography and Security

Exercise for home

Joe Grand, hacks a Trezor Hardware Wallet. We will rate the attack during the seminars so please watch the video:



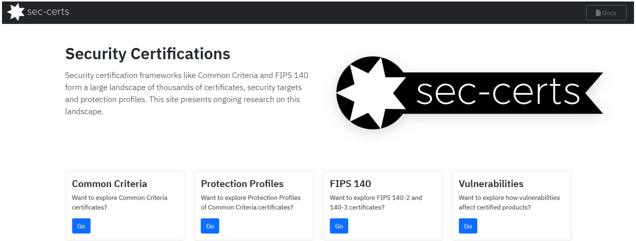
`https://www.youtube.com/watch?v=dT9y-KQbqi4`

Plan

1. Advanced Passive Attacks
   - Higher-Order Attacks
   - Profiled Attacks
2. Introductions/Recall on Fault Attacks
3. Recall on Differential Fault Analysis (RSA)
4. Real-world example: fault attacks on WolfSSL
5. Symmetric DFA
   - DFA on (Triple) DES
   - DFA on AES
6. Advanced Fault Injection Attacks (Selection)
   - Fault Injection And Side-Channel Analysis
   - Statistical Ineffective Fault Attacks
7. Practical Aspects of Fault Injection Attacks
8. The industry perspective
   - Evaluating Hardware attacks
9. **sec-certs**
10. Conclusions and Further Reading
    - Conclusions: SCA and FI
    - Conclusions: Business Perspective

CROCS Centre for Research on Cryptography and Security

Analyzing landscape of certifications.



Security Certifications

Security certification frameworks like Common Criteria and FIPS 140 form a large landscape of thousands of certificates, security targets and protection profiles. This site presents ongoing research on this landscape.

Common Criteria
Want to explore Common Criteria certificates?
Go

Protection Profiles
Want to explore Protection Profiles of Common Criteria certificates?
Go

FIPS 140
Want to explore FIPS 140-2 and 140-3 certificates?
Go

Vulnerabilities
Want to explore how vulnerabilities affect certified products?
Go

- `https://seccerts.org/`
- "Security certification frameworks like Common Criteria and FIPS 140 form a large landscape of thousands of certificates, security targets and protection profiles. This site presents ongoing research on this landscape."
- Paper #1: A. Janovsky, J. Jancar, P. Svenda, Ł. Chmielewski, J. Michalik, V. Matyas, *sec-certs: Examining the security certification practice for better vulnerability mitigation*, in submission, available: `https://arxiv.org/abs/2311.17603`
- Paper #2: A. Janovsky, Ł. Chmielewski, J. Jancar, P. Svenda, V. Matyas, *Chain of trust: Unraveling references among Common Criteria certified products*, IFIPSEC2024.
- For more information ask me :-)

CROCS
Centre for Research on Cryptography and Security

Plan

1. Advanced Passive Attacks
   - Higher-Order Attacks
   - Profiled Attacks
2. Introductions/Recall on Fault Attacks
3. Recall on Differential Fault Analysis (RSA)
4. Real-world example: fault attacks on WolfSSL
5. Symmetric DFA
   - DFA on (Triple) DES
   - DFA on AES
6. Advanced Fault Injection Attacks (Selection)
   - Fault Injection And Side-Channel Analysis
   - Statistical Ineffective Fault Attacks
7. Practical Aspects of Fault Injection Attacks
8. The industry perspective
   - Evaluating Hardware attacks
9. `sec-certs`
10. Conclusions and Further Reading
    - Conclusions: SCA and FI
    - Conclusions: Business Perspective

Conclusions: SCA and FI

- Both SCA and Fault injection are powerful and practical attack methods
    - Arguably it is harder to protected against FI than SCA
- Not too many generic FI attacks are know. Usually they are tailored for one use case and sometimes specific to a target.
- Not only crypto can be attacked with FI (DFA), but also security mechanisms, like secure boot, among others.
- There is a strong relationship between FI and SCA attacks that can be successfully exploited.
- This lecture is just a glimpse into more advanced side-channel and fault injection attacks :-) I hope that you enjoyed it!

Further reading: FI

- Eloi Sanfelix, Cristofaro Mune, and Job de Haas *Unboxing the White-Box Practical attacks against Obfuscated Ciphers*, Black Hat 2015.
- C. Dobraunig, M. Eichlseder, T. Korak, S. Mangard, F. Mendel, and R. Primas *SIFA: Exploiting Ineffective Fault Inductions onSymmetric Cryptography*
- Albert Spruyt, Alyssa Milburn, and Łukasz Chmielewski *Fault Injection as an Oscilloscope: Fault Correlation Analysis*, CHES 2021.
- P. Dusart, G. Letournex and O. Vivolo *Differential fault analysis on AES*, Springer, 2003.
- Christophe Clavier, *Secret external encodings do not prevent transient fault analysis*
- T. Fuhr, É. Jaulmes, V. Lomné, and A. Thillard. *Fault attacks on AES with faulty ciphertexts only*

Interesting Talks:

- *20 Ways Past Secure Boot* by Joob de Haas (2014): `https://www.youtube.com/watch?v=74SzIe9qiM8&ab_channel=TROOPERScon`
- *Bypassing Secure Boot using Fault Injection* by N. Timmers and A. Spruyt (2017): `https://www.youtube.com/watch?v=s_PzQsWfhsU&ab_channel=SHA2017`
- *Divide & Conquer Revisited: FI As A SW EXP Primitive* by R. Boix Carpi and Federico Menarini (2021): `https://www.youtube.com/watch?v=n8y1_pc1Rvw&ab_channel=hardwear.io`

Conclusions: Business Perspective

- Security certification regulate the evaluation and testing in the security industry;
- Security certifications exist in a complex eco-systems (who pays / who tests / what is tested / who bears the risks);
- CC certifications and EMV, shape the testing of high-end SoCs to physical attacks in the industry;
- Challenges ahead:
  - technology shifts;
  - costs;
  - impact of remote attacks is unclear;
  - white/grey-box evaluation;
  - scoring of attacks is subjective to the experts interpretation ;
- Some schemes do not look at the final product but the development process.

Take away: Business Perspective

Key terminology:

- security certification: CC and EMV;
- JIL rating
- TOE and the physical adversary

Reflect:

- Think of a scenario where the physical adversary is not a threat.
- Name some advantages and disadvantages of the JIL rating scheme.
- Name three advantages provided by a security certification scheme.

Compute hypothetical rating:

- `https://tinyurl.com/mv7kjx8w`

Further reading: Business Perspective

1. Application of Attack Potential to Hardware Devices with Security Boxes , Joint Interpretation Library, Version 3.0, July 2020, https://www.sogis.eu/documents/cc/domains/hardware_devices/JIL-%20Application-of-Attack-Potential-to-Hardware-Devices-with-Security-Boxes-v3-0.pdf
2. Arrangement on the Recognition of Common Criteria Certificates In the field of Information Technology Security, Common Criteria, May 2000, https://www.commoncriteriaportal.org/files/operatingprocedures/cc-recarrange.pdf
3. Titan, H1D3 Secure Microcontroller with Crypto Library v0.1.4, Security Target Lite, Version: 2.4, Release: November 5th, 2021, https://www.commoncriteriaportal.org/files/epfiles/[STL][2.4]%20Titan%20H1D3%20Security%20Target%20Lite%20v2.4.pdf
4. H1D3 Secure Microcontroller with Crypto Library v0.1.4, Version: 2.4 Release: November 5th, 2021, https://www.commoncriteriaportal.org/files/epfiles/[STL][2.4]%20Titan%20H1D3%20Security%20Target%20Lite%20v2.4.pdf

Questions