Cyber Security for Europe

CHESS

# Lessons learned from blackbox analyses of software and hardware cryptographic implementations


## And what can be expected in PQC future?

**Petr Švenda** *svenda@fi.muni.cz* *@rngsec*

Centre for Research on Cryptography and Security, Masaryk University

*Joint work with many amazing people from CRoCS - thank you all!*

CROCS Centre for Research on Cryptography and Security

*Funded by the European Union under Grant Agreement No. 101087529 (CHESS project)

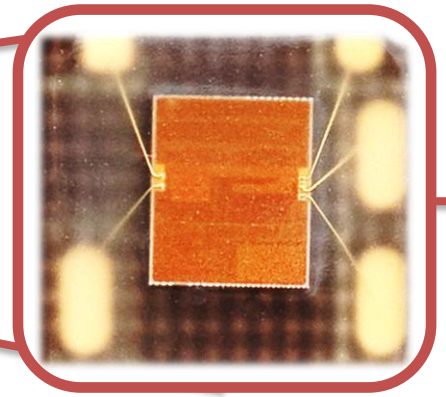
www.fi.muni.cz/crocs

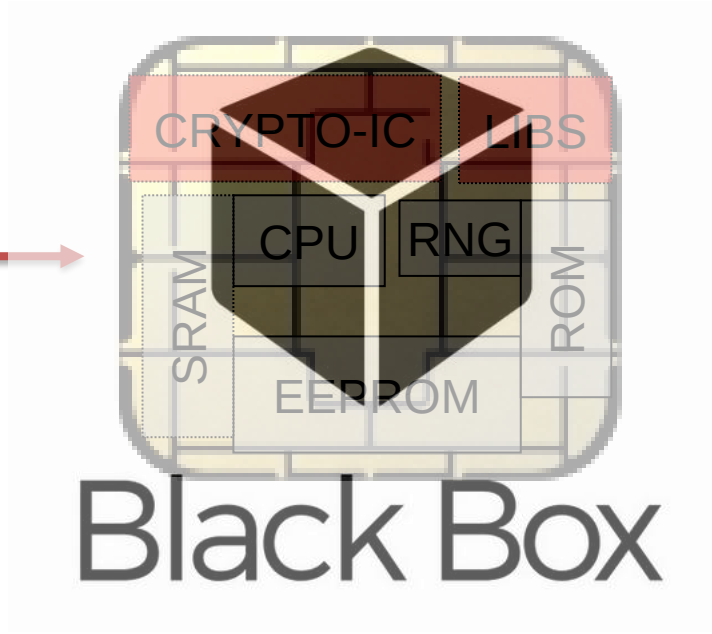
# We analyze blackbox implementations as academics

- As much as possible is kept proprietary (IP, JIL rating, certification req.)
  - Design details, hardware layout, firmware code, JCRE, packages, vendor's API…

**Open JavaCard API** ✓

```
KeyPair.genKeyPair()
Signature.sign()
```



**Vendor**

- Great understanding of target implementation (whitebox)
- Limited knowledge in security testing, advanced attacks, equipment
- Conflict between time-for-testing and time-to-market

**Eval lab**

- Great knowledge in security testing, specialized equip., process knowledge
- Moderate understanding of target implementation (whitebox, but !enough time)
- Conflict between tough analysis and keeping vendor as a customer

**Scheme (BSI, ANSSI, NIAP…)**

- Some experts, no direct testing (“impartial”)
- Aim to keep security bar reasonably high
- Stronger ties to bigger players

**User**

- Lack of knowledge, test outsourcing (certification)
- Do not know what was tested!

# Ideal setup for finding bugs

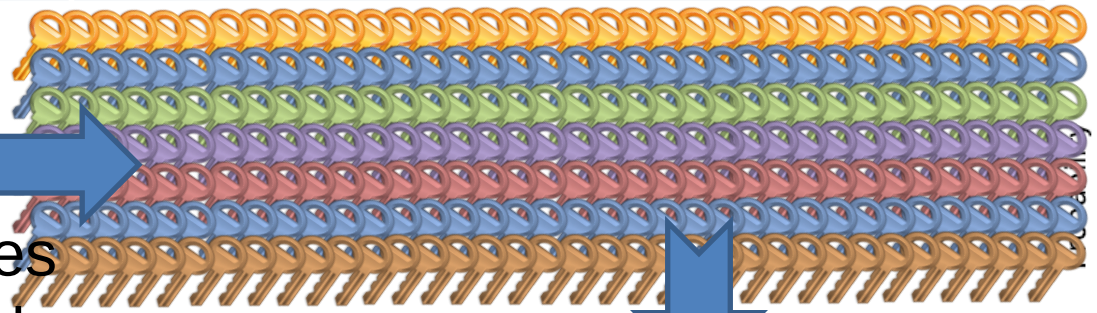
1. Complete knowledge of design and implementation (whitebox)
2. Great experience in security testing, specialized equipment, automated testing, advanced attacks, vulns in related devices
3. A lot of time for testing and creative thinking

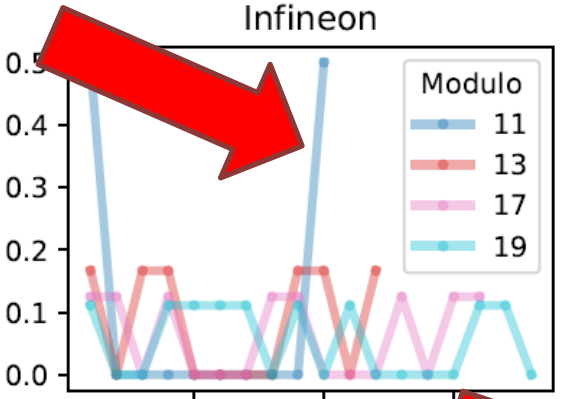
**Academia, security researchers**

- Small understanding of implementation (frequently blackbox)
- Great knowledge of *some* advanced attacks, *some* equipment
- A lot of time, focus on publishable, more complex results
- Not focused on specific device, wide-scale testing suitable

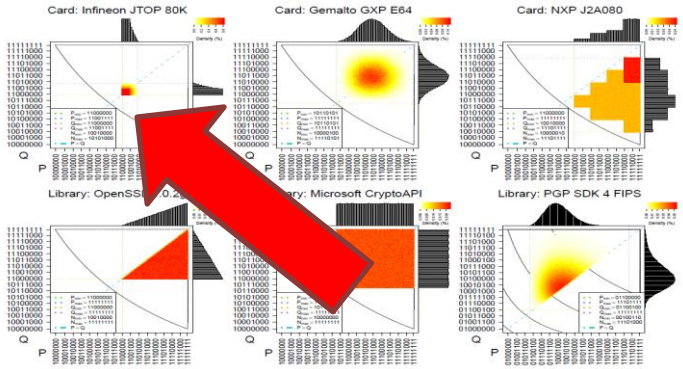
# 60+ million fresh RSA keypairs (P, Q, N)

22 sw. libraries
16 smart cards

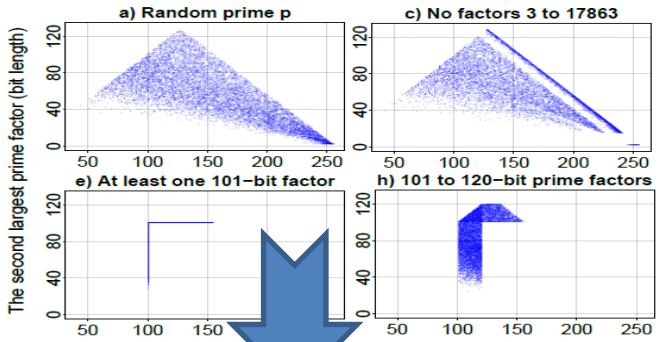




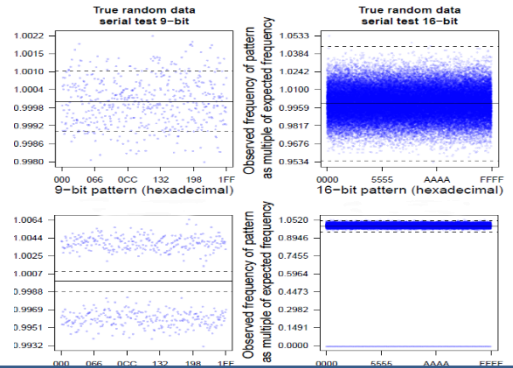

Distribution of primes (MSB)



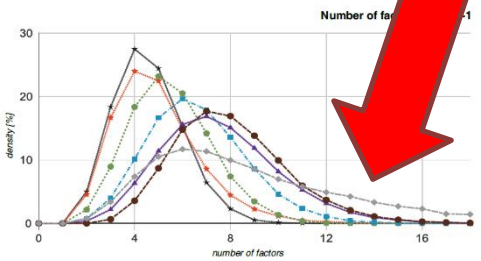

Large factors of p-1 / p+1




Bit stream statistics




Number of factors

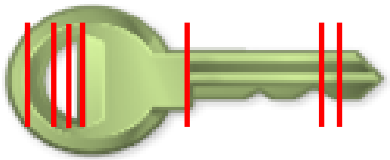



and more…

- Various implementation choice
- Small bias, but enough to attri

$Prime_{expected} = random$ ✓

$Prime_{Infineon} = k * M + 65537^a \bmod M$



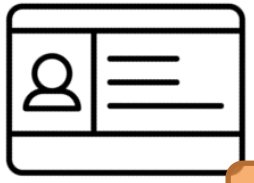

Random 2048b key: 6442450944000000 vCPU years
Infineon 2048b key: <$1000 140 vCPU years

*M. Nemec, M. Sys, P. Svenda, D. Klinec, V. Matyas: The Return of Coppersmith's Attack..., ACM CCS 2017*

# The usage domains affected by the vulnerable library

Austria, Estonia, Slovakia, Spain…

Identity documents (eID, eHealth cards)

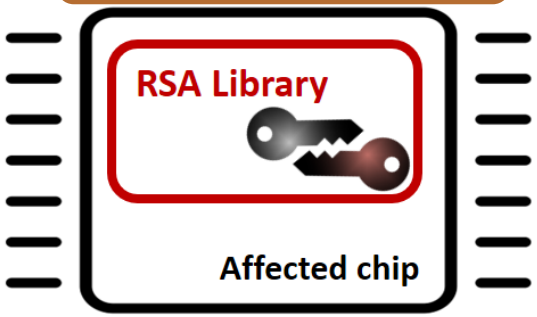


Trusted Platform Modules (Data encryption, Platform integrity)


25-30% TPMs worldwide, BitLocker, ChromeOS… Firmware update available

1-2 billion chips

Software signing




Secure browsing (TLS/HTTPS*)

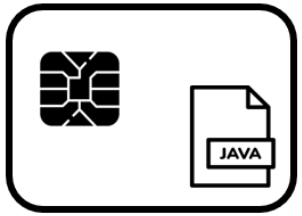
Commit signing, Application signing GitHub, Maven…

Very few keys, but all tied to SCADA management

Authentication tokens

Message protection (S-MIME/PGP)

Programmable smartcards



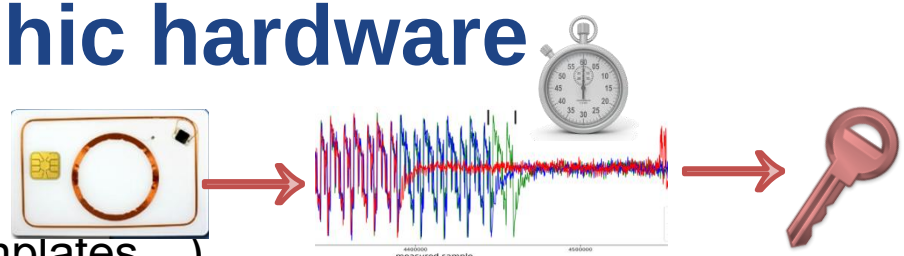

Gemalto .NET Yubikey 4…

Yubikey 4…

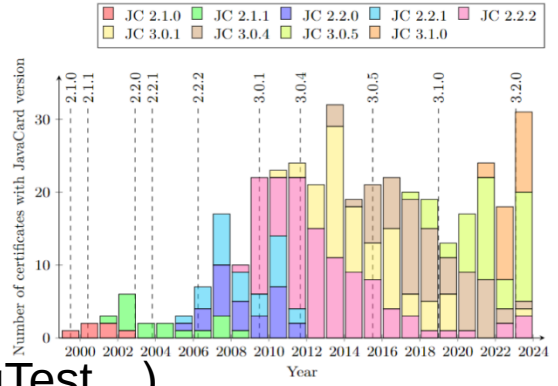
** only a small number of vulnerable keys found*

# DIFFICULTY OF ANALYSIS OF CRYPTOGRAPHIC DEVICES IN TIME

# Independent analysis of cryptographic hardware



- ~2000
  - Fresh powerful side-channel attack ideas (SPA/DPA/CPA/Templates…)
  - No card samples for testing (minus points in JIL rating), heavy reliance on black-box as protection
  - CC/FIPS reports very different variable (no template), little to no public info about testing performed
  - No mature open-source testing tools, scattered knowledge
- ~2010-2017
  - Advanced testing setups available at vendor and evaluation labs
  - Availability of (some) smartcards in small quantities for independent testing
  - Practical application of more attacks (lattice attacks on ECC nonce leaks…)
  - Growing open-source community (Chipwhisperer, Sakura, ASCAD dataset, JCAlgTest…)
- ~2018-now
  - Deep learning SCA attacks (boom from 2016/17), improved lattice-based attacks, PQC SCA…
  - Academic researchers typically focus on white-box targets (FPGA…), not smartcards!
  - Only (somewhat) older cards available for testing (1-2 generations back, unofficial samples…)



# CROCS 's way

- Focusing on new attacks on blackbox targets

1. Design technique to probe cryptographic target
2. Implement open-source tool for testing it
3. Perform the test on wide range of targets (cards, cryptolibs…)
4. Spot biases and develop academically publishable exploitation method

- Ideal outcome: method can be published, real-world impact can be demonstrated, open analysis tool available for others and future (CI)

⇨ Reverse engineering (steps 1.-3.) typically revealed some weakness (step 4.)

  – Wide-range testing is amplifying (otherwise low) changes to find something!

# RSA pubkey origin attribution (no CVE)

[Svenda et.al., The Million-Key Question – Investigating the Origins of RSA Public Keys, USENIX'16]

[Janovsky et.al., Biased RSA private keys: Origin attribution of GCD-factorable keys, ESORICS'20]

# ROCA (CVE-2017-15361)

[Nemec et.al., The Return of Coppersmith's Attack: Practical Factorization of Widely Used RSA Moduli, ACM CCS'17]

# Minerva (CVE-2019-15809)

[J.Jancar, V.Sedlacek, P.Svenda, M.Sys. Minerva: The curse of ECDSA nonces, CHES'20]

[J.Jancar, V.Suchanek, P.Svenda, V. Sedlacek, L. Chmielewski. pyecsca: Reverse engineering black-box elliptic curve cryptography via side-channel analysis, CHES'24]

# TPMScan (CVE-2020-25082, "NineSig" - no CVE)

[Svenda et.al., TPMScan: A wide-scale study of security-relevant properties of TPM 2.0 chips, CHES'24]

# (stay tuned)

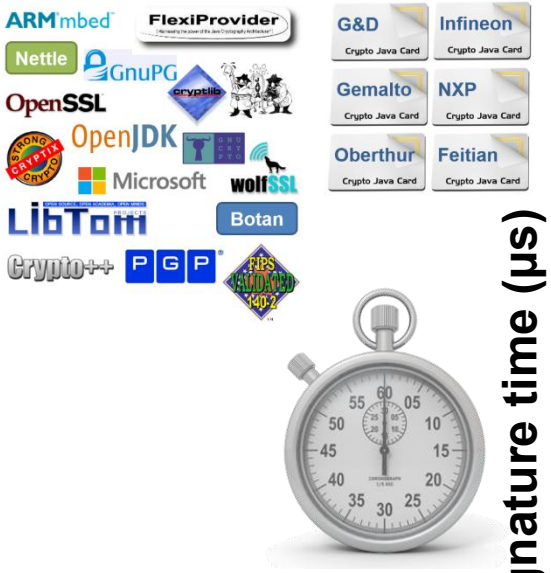
[responsible disclosure period ☺]

# EXAMPLE: MINERVA ECC VULNERABILITIES

Minerva

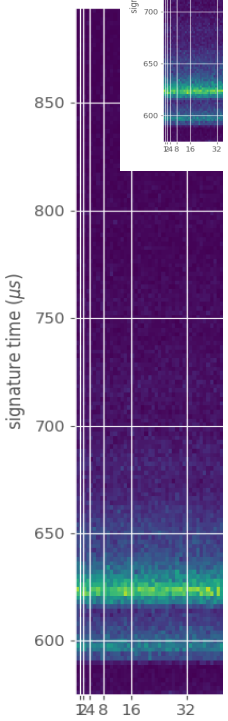
# Gather data → Analyse → Bias found → Impact

**Run ECC operations →**



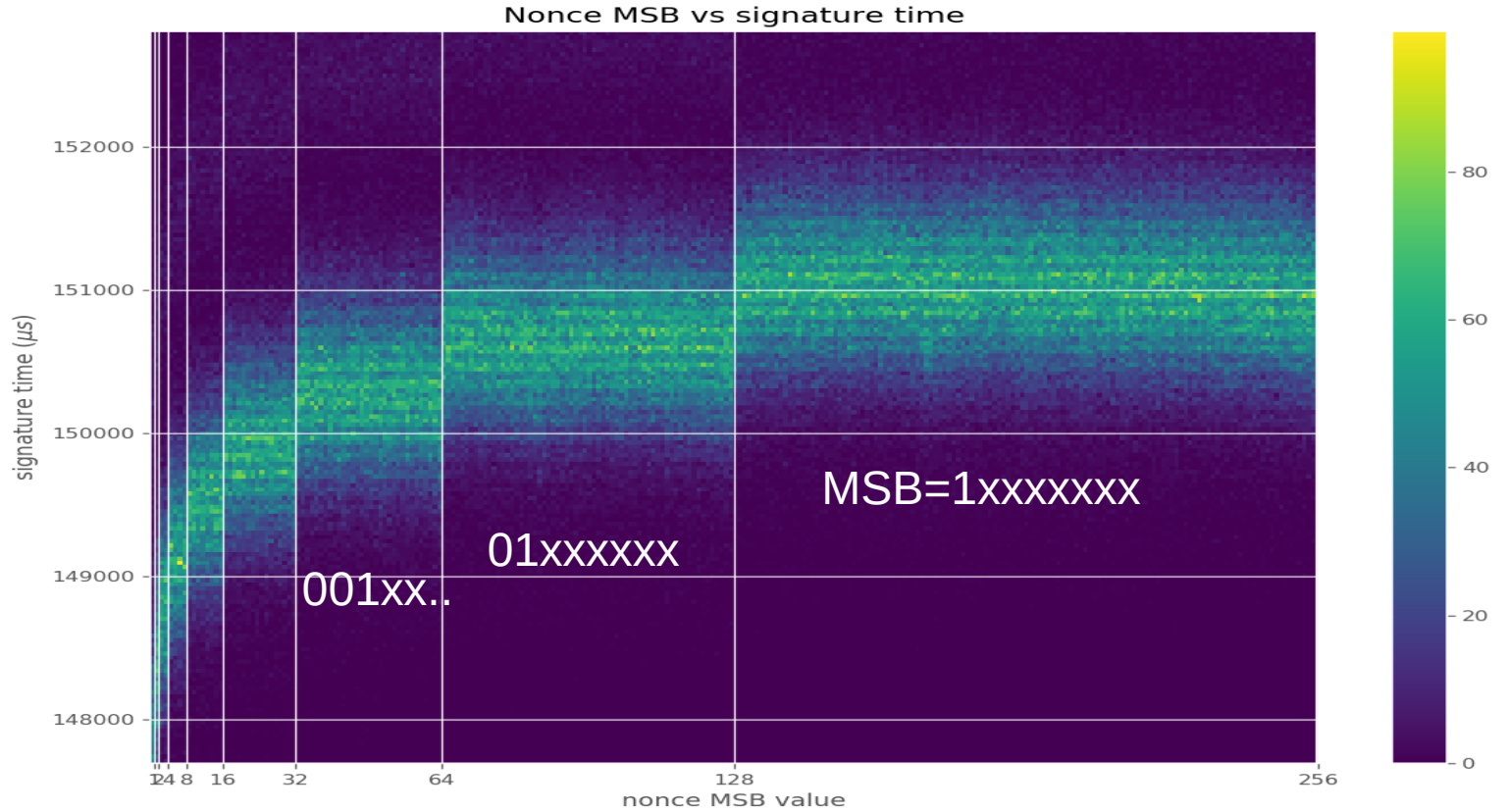



**Signature time (µs)**

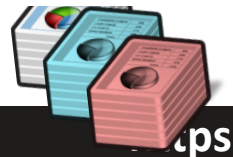




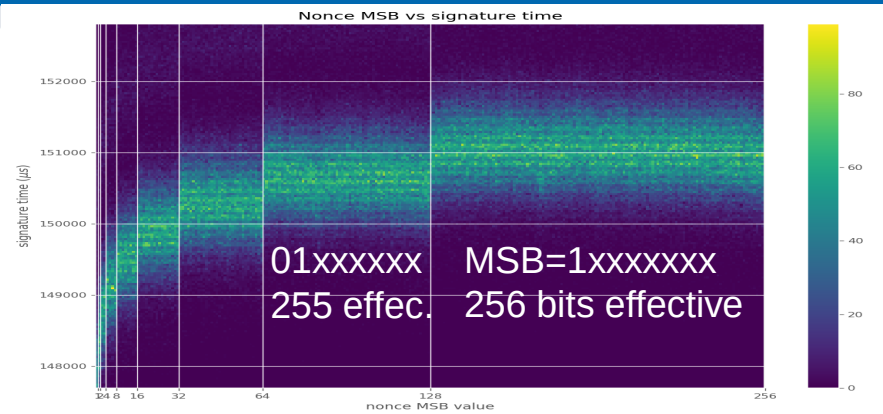


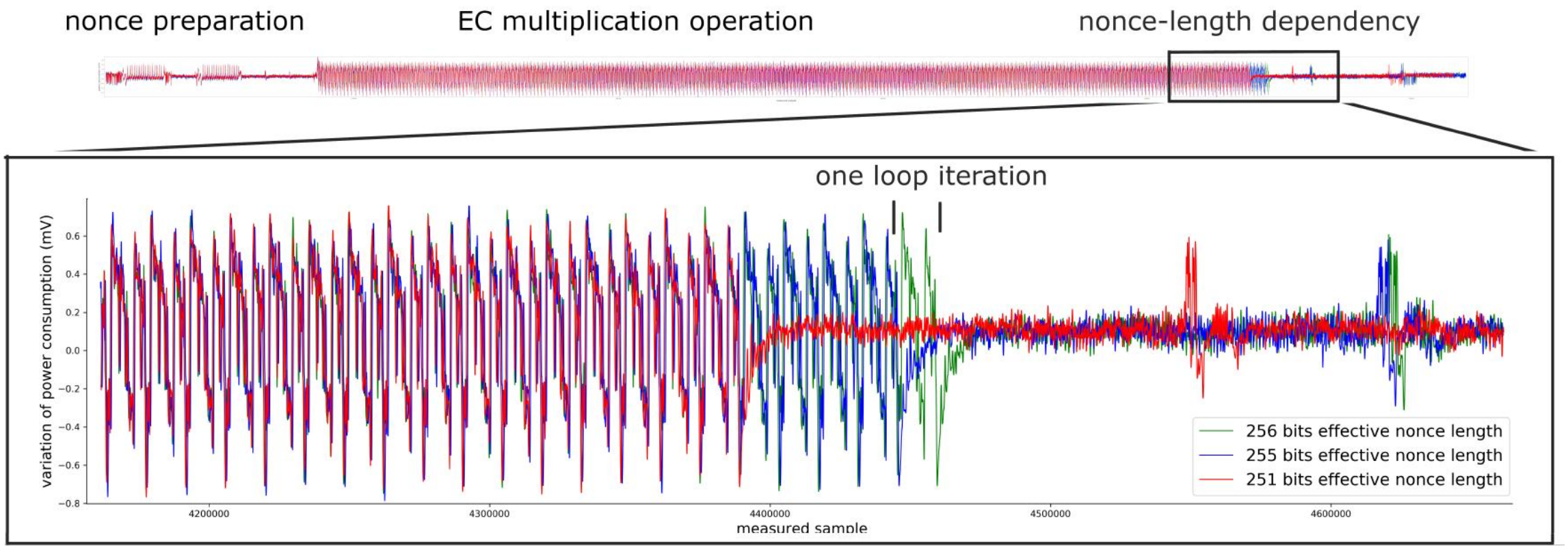



**Nonce MSB value**



# Minerva: Leaking nonce length



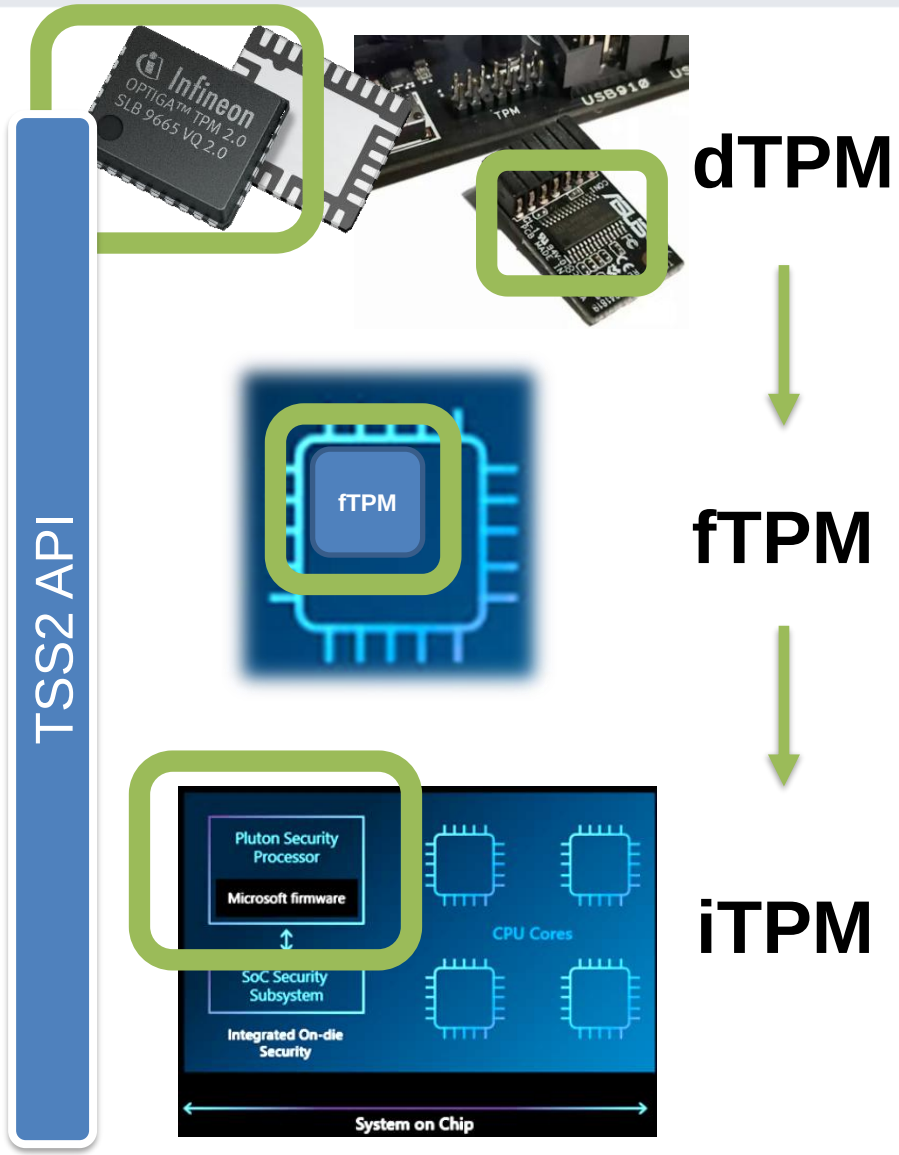
# Minerva vulnerability CVE-2019-15809 (10/2019)

- Athena IDProtect smartcard (CC EAL 4+)
  - FIPS140-2 #1711, ANSSI-CC-2012/23
  - Inside Secure AT90SC28872 Microcontroller
  - Root cause: Upper layer used faster but insecure (unprotected) modMult()
- Similar issue in Libgcrypt, wolfSSL, MatrixSSL, Crypto++, SunEC/OpenJDK/Oracle JDK
- Enough to extract whole ECC private key in 20-30 min
  - ~thousands of signatures + lattice-based attack

| Type | Name | Version/Model | Scalar multiplier | Leakage |
|---|---|---|---|---|
| Library | OpenSSL | 1.1.1d | Montgomery ladder[1] | no |
| | BouncyCasle | 1.58 | Comb method[2] | no |
| | SunEC | JDK 7 - JDK 12 | Window-NAF | no |
| | | | Lopez-Dahab ladder | yes |
| | WolfSSL | 4.0.0 | Sliding window | yes[3] |
| | BoringSSL | 974f4dddf | Window method | no |
| | libtomcrypt | v1.18.2 | Sliding window | no |
| | libgcrypt | 1.8.4 | Double-and-add | yes |
| | Botan | 2.11.0 | Window method[4] | no |
| | Microsoft CNG | 10.0.17134.0 | Window method | no |
| | mbedTLS | 2.16.0 | Comb method | no |
| | MatrixSSL | 4.2.1 | Sliding window | yes |
| | Intel PP Crypto | 2020 | Window-NAF | no |
| | Crypto++ | 8.2 | unknown | yes |
| Card | Athena IDProtect | 010b.0352.0005 | unknown | yes |
| | NXP JCOP3 | J2A081, J2D081, J3H145 | unknown | no |
| | Infineon JTOP | 52GLA080AL, SLE78 | unknown | no |
| | G+D SmartCafe | v6, v7 | unknown | no |

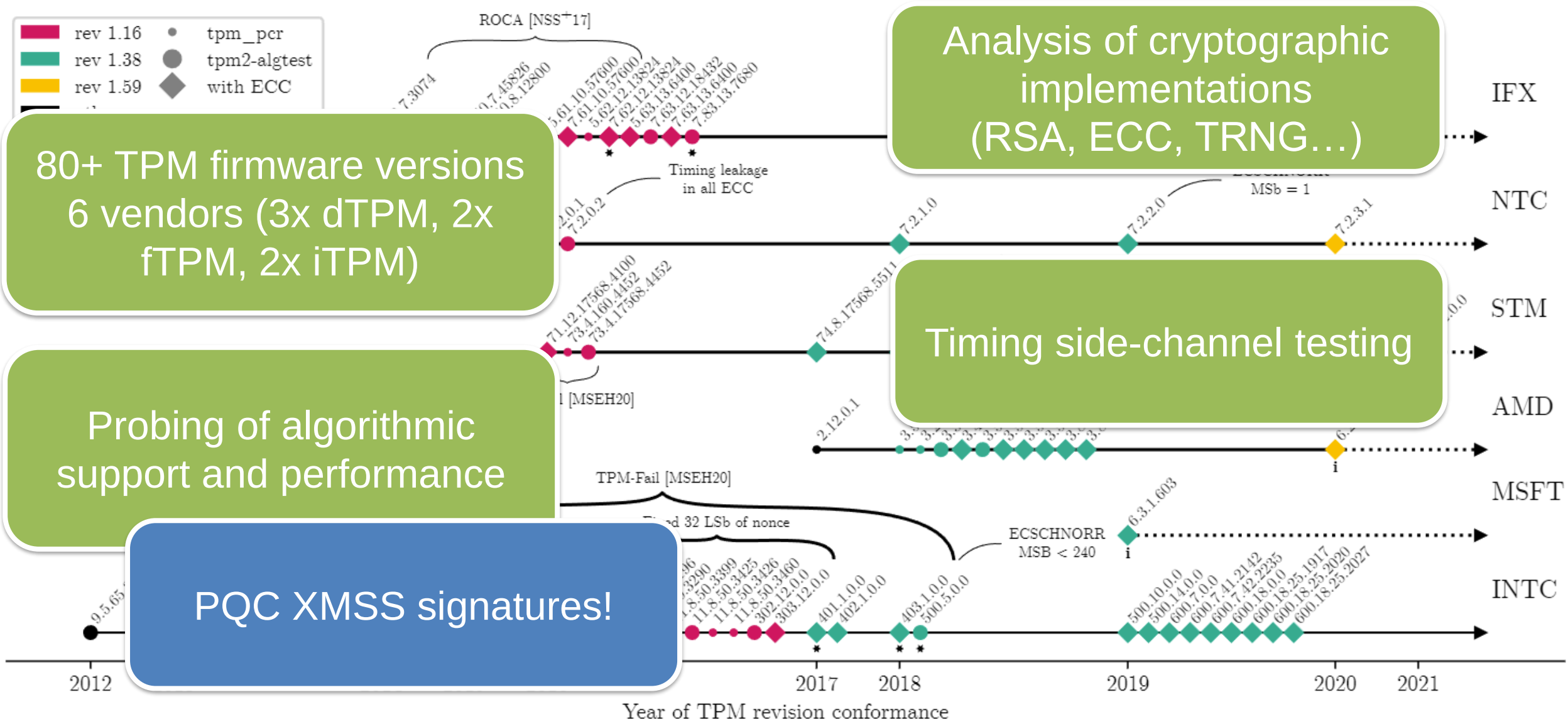
[1] Applies the fixed bit-length mitigation.
[2] Uses many scalar multiplication algorithms.
[3] Likely not exploitable, due to a small amount of leakage.
[4] Uses additive scalar blinding.

# EXAMPLE: TRUSTED PLATFORM MODULES




*Credits: Infineon, Asus, Microsoft*

# Trusted Platform Modules (TPMs), from 2016




80+ TPM firmware versions
6 vendors (3x dTPM, 2x fTPM, 2x iTPM)

Analysis of cryptographic implementations (RSA, ECC, TRNG…)

Timing side-channel testing

Probing of algorithmic support and performance

PQC XMSS signatures!

# TPMScan https://tches.iacr.org/index.php/TCHES/article/view/11444

RSA private key extraction ROCA, passive, one public key

ECC private key extraction, active (EM-SCA), ~40 signatures

**EUCLEAK [4] (EM SCA)**

ECC private key extraction, active, ~1000s of signatures

ECC private key extraction, active, ~1000s of signatures

ECDSA private key extraction "NineSig", passive, 9 signatures

ECC private key extraction, active, ~1000s of signatures

| Year | Vulnerability / nickname | Hardware target | Attack class | Public ID | Discoverer |
|---|---|---|---|---|---|
| 1996 | RSA timing attack | Smart-card | Side-channel – timing | — | Academic (Kocher) |
| 1997 | Boneh–DeMillo–Lipton RSA-CRT fault | Smart-card | Physical – fault | — | Academic (Stanford) |
| 1999 | Differential Power Analysis (DPA) | Smart-card | Side-channel – power | — | Academic (Kocher et al.) |
| 2004 | “Sorcerer’s Apprentice” voltage/laser survey | Smart-card | Physical – fault | — | Evaluation-lab + academic |
| 2008 | MIFARE Classic / CRYPTO-1 break | Contactless card | Logic + side-channel | — | Academic (Radboud U.) |
| 2011 | Java Card operand-stack laser fault | Java Card | Physical – laser | — | Academic (Télécom ParisTech) |
| 2011 | Mifare DESFire MF3ICD40 | Smart-card | Side-channel – power | — | Academic (Oswald&Paar) |
| 2013 | Weak RSA key generation (factorable) | Taiwanese Citizen-ID | Logic – keygen, TRNG | — | Academic (Bernstein et al.) |
| 2013 | Yubikey 2 OTP key extraction | Smart-card | Side-channel – power | — | Academic (Oswald et al.) |
| 2014 | EMV foreign-currency limit bypass | EMV chip card | Logic – protocol | — | Academic (Newcastle U.) |
| 2015 | HMAC_SHA1 key extraction | DS28E01/DS2432 IC | Side-channel – power | — | Academic (Oswald) |
| 2017 | ROCA weak RSA keygen | Infineon cards & TPMs | Logic – math | CVE-2017-15361 | Academic (Masaryk U.) |
| 2019 | Java Card RI multiple bugs | Java Card | Logic – code | Oracle CPU (no CVE) | Commercial (Security Explorations) |
| 2019 | Yubikey reduced initial randomness on FIPS | Yubikey FIPS token | Logic –randomness | YSA-2019-02 | Vendor (Yubico) |
| 2019 | Deep learning on RSA implementations | ARM Core SC 100 EAL4+ IC | Side-channel – EM | — | Evaluation-lab + vendor |
| 2019 | TPM-Fail ECDSA timing leaks | Intel fTPM & ST dTPM | Side-channel – timing | CVE-2019-11090, -16863 | Academic |
| 2020 | Estonian ID-card duplicate keys | National ID smart-card | Logic – key mgmt | — | Academic (Paršovs) |
| 2021 | NXP SmartMX EM leak (Titan/SJTT) | SmartMX & A7x secure MCUs | Side-channel – EM | CVE-2021-3011 | Commercial (NinjaLab) |
| 2021 | STSAFE-J / J-SAFE3 ECDSA leak | Java Card SE | Side-channel – timing | CVE-2021-43392/43393 | Vendor (ST) |
| 2023 | TPM 2.0 ref-lib OOB R/W | Discrete & firmware TPM | Logic – mem-corr | CVE-2023-1017/1018 | Commercial (Quarkslab) |
| 2023 | faulTPM full-state extraction | AMD fTPM | Physical – fault | — | Academic (TU Berlin) |
| 2023 | STSAFE-A1 middleware overflow | Secure element | Logic – buffer-overflow | CVE-2023-50096 | Commercial (Elttam) |
| 2024 | TPMScan nonce-bias & timing | Multi-vendor TPM 2.0 | Side-channel | — | Academic (Masaryk U.) |
| 2024 | NineSig: Intel fixed nonce bits | Intel fTPM | Logic – randomness | — | Academic (Masaryk U.) |
| 2024 | EUCLeak EM leak in Infineon lib | Infineon SE (YubiKey 5/HSM 2) | Side-channel – EM | CVE-2024-45678 | Commercial (NinjaLab) |
| 2025 | YubiKey CTAP v2 partial-sig check | YubiKey 5/Bio/HSM 2 | Logic – protocol | CVE-2025-29991 | Vendor (Yubico) |

Commercial + Vendors not public with results

# Bias in past publicly reported vulnerabilities

- Based on public reports, vulnerabilities seems to be primarily found by academic researchers and freelancers/independent labs
  - People operating in black-box attacker model => harder to spot problems (than in white-box)

Public disclosure might be a side-effect of academics desire to publish and independent commercial labs to advertise themselves

- Rarely vendor itself (Yubico seems to be exception)

How likely is that blackbox analysis spotted all the vulnerabilities?

  - Likely many problems completely missed (blackbox) or neglected (if second defensive layer)
  - Likely many problems undisclosed (vendor/eval.labs: before/after deployment/certification)

- Reported vulnerabilities with low visibility (no CVE, no public security bulletin…)
  - E.g. Intel fTPM “NineSig” vulnerability (TPMScan), no CVE because product is “outdated”

# HOW THE SITUATION WILL CHANGE FOR PQC?

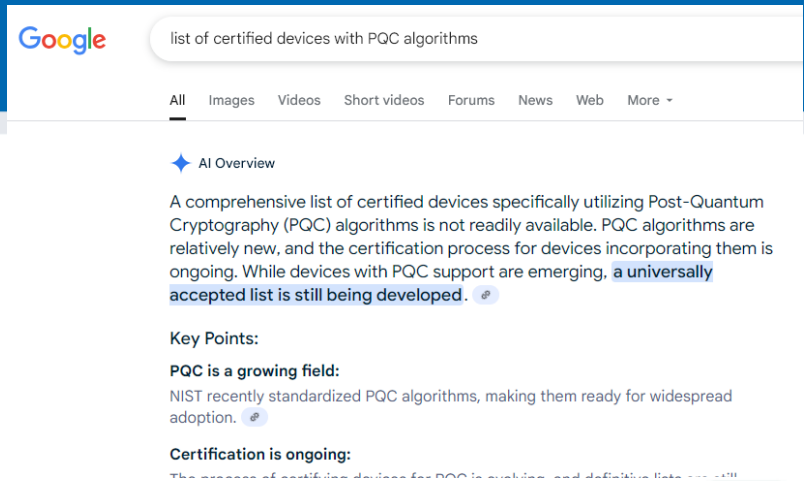
# Attacker capabilities will inevitably improve over time

1. Completely novel attack principle (e.g., DPA by Kocher)
   – Hard to defend against, but (fortunately) infrequent
2. Significant improvement of previously known attack (e.g., ML SCA)
   – Security margin is important, pre-emptively upping device security even if not immediate threat (not favored by certification)
3. Device-specific tweaks making known attacks practical (e.g., ROCA)
   – Following best practices, using the most mature and battle-tested code (which is typically not vendor's own proprietary implementation!)
4. Insufficient testing (e.g., TPMFail, Minerva, NineSig…)
   – Using mature implementation, continuous testing

# Common cause of vulnerabilities presented

- Long time to discovery => high impact due to penetration of market
- As much as possible is kept proprietary
  - Intellectual Property reasons, Certification scheme requirements (JIL rating…)
  - Design details, hardware layout, firmware code, JCRE, packages, vendor's API…
  - Another layer of defense, but also false sense of security
- Certification is expensive, change requires recertification => no change
  - Clear bugs will be fixed (e.g., ROCA) or possibly put out of scope (e.g., EUCLeak)
  - Improved protection measures will not be included unless necessary
- Analysis performed by vendor and eval lab is time-bounded
  - State-level attacker or academia researchers can devote years of work (interleaved)
  - Creative attack ideas typically cannot be squeezed defined 4 weeks evaluation period!

# PQC ADOPTION IN CERTIFIED PRODUCTS




# Certified devices with PQC algorithms?

1. Google search for certified products with PQC
2. ChatGPT o3 PQC search: not bad, but only 3+5 certs (CC+FIPS140)

sec-certs

3. https://sec-certs.org PQC search: Common Criteria (29), FIPS140 (20)
   - "post quantum" OR "post-quantum" OR "PQC" OR "KYBER" OR "SPHINCS" OR "NTRU" OR "XMSS" OR "LWE" OR "CSIDH" OR "BLISS" OR "RLCE" OR "McEliece" OR "CRYSTALS" OR "Dilithium" OR "ML-KEM"
4. https://sec-certs.org PQC search with query generated by LLM
   - All PQC algorithms in Whoosh language syntax" + search
5. https://sec-certs.org with LLM-based chatbot (Llama 4 Scout)

# PQC certified products in CC

- Multi-Party Cryptographic Appliance "Trident v2.1.3“ OCSI/CERT/CCL/02/2020/RC (from 09/2020)
  - SPHINCS+ algorithm (inside larger HSM)
- 16x Infineon OPTIGA TPM SLB9672 (from 05/2021)
  - XMSS: eXtended Merkle Signature Scheme (large signs)
- 2x Samsung chips with PQC (03,07/2024)
  - CRYSTALS engine (Kyber, Dilithium)
  - Smartcard chip EAL6+ ANSSI-CC-2024/26 (22.07.2024)
    - *“No security functionality is claimed”*
  - Mobile SoC EAL5+ NSCIB-CC-2300085-01-CR (26.03.2024)
    - *“Out of ToE”*

https://sec-certs.org/cc/ec832cac23bb8b42/

**1.2.2 TOE Definition**

6 The S3SSE2A single-chip CMOS micro-controller is designed and packaged specifically for "Smart Card" applications.

7 The SC300 CPU architecture of the S3SSE2A microcontroller follows the Harvard style, that is, it has separate program memory and data memory. Both instruction and data can be fetched simultaneously without causing a stall, using separate paths for memory access.

The main security features of the S3SSE2A integrated circuit are:

- Environmental & Life time detector & filters
- Active shield
- Dedicated tamper-resistant design based on synthesizable glue logic and secure topology
- Dedicated hardware mechanisms against side-channel attacks
- Secure DES and AES Symmetric Cryptography support
- Secure TORNADO-E coprocessor supports a Montgomery type multiplication, a modular addition/subtraction, a modular exponentiation (out of TSF), and a computation for the square of a montgomery constant up to 4128-bit operand sizes
- PQC(Post-Quantum Cryptography) engine(CRYSTALS)(It is out of TOE)
- PARITY/ CRC-32 calculators

https://sec-certs.org/cc/54e6eb5c1bd57036/#st-info

NOTE 1: No security functionality is claimed for the following hardware blocks in this TOE or use cases:

- DMA (PL080)
- SSP_DMA
- Key manager KEYMGT in SYSCON

SAMSUNG ELECTRONICS 13/131 SAMSUNG

*Public*

ST Lite 1 ST INTRODUCTION

- Code execution through the Secure AXI bridge (eXecute In Place, XIP)
- PQC

NOTE 2: Secure functionality is claimed for the AES in the Security Controller and the AES in the CRYPTO block.

# PQC certified products

- IFX_CCI_000068h… optional PQ Crypto Suite v5.00.012… (BSI-DSZ-CC-1249-2024, 12/2024) https://sec-certs.org/cc/5e551e06e7b71400
- Finally in certification scope \o/ (but not much details provided)

sec-certs | Common Criteria | Protection Profiles | FIPS 140 | Vulnerabilities | About

Sections

1. Certificate summary
2. Certificate
3. Certification report
4. Security target
5. Heuristics
6. References
7. Updates
8. Raw data

IFX_CCI_000068h, IFX_CCI_000080h design step G12 with firmware version 80.505.04.1, optional optional HSL v04.05.0040, optional UMS guidance documents

Compare 2 | Subscribe 0 | Share

CSV information

| | |
|---|---|
| Status | active |
| Valid from | 17.12.2024 |
| Valid until | 17.12.2029 |
| Scheme | DE DE |
| Manufacturer | Infineon Technologies AG |
| Category | ICs, Smart Cards and Smart Card-Related Devices and Systems |
| Security level | ALC_FLR.1, EAL6+ |
| Protection profiles | • Security IC Platform Protection Profile with Augmentation Packages |

public

**TEGRION™ SLC21 Post-Quantum Edition**

**Security Target Lite**

**Security Requirements (ASE_REQ)**

**Table 42 Cryptographic table for FCS_COP.1/CS/MLKEM/<iter>**

| <iter> | [assignment: list of cryptographic operations] | [assignment: cryptographic algorithms] | [assignment: cryptographic key sizes] | [assignment: list of standards] |
|---|---|---|---|---|
| ENC | encapsulation | ML-KEM-512<br>ML-KEM-768<br>ML-KEM-1024 | ML-KEM-512,<br>ML-KEM-768,<br>ML-KEM-1024 | [FIPS 203] ch. 6.2 |
| DEC | decapsulation | ML-KEM-512<br>ML-KEM-768<br>ML-KEM-1024 | ML-KEM-512,<br>ML-KEM-768,<br>ML-KEM-1024 | [FIPS 203] ch. 6.3 |

# POSITIVE EXAMPLE: HARDWARE WALLETS

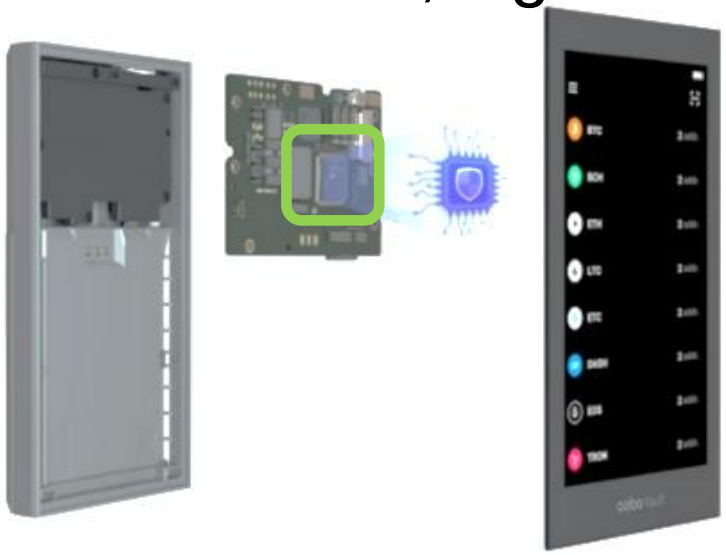
# Testbed with hardware wallets, from 2020

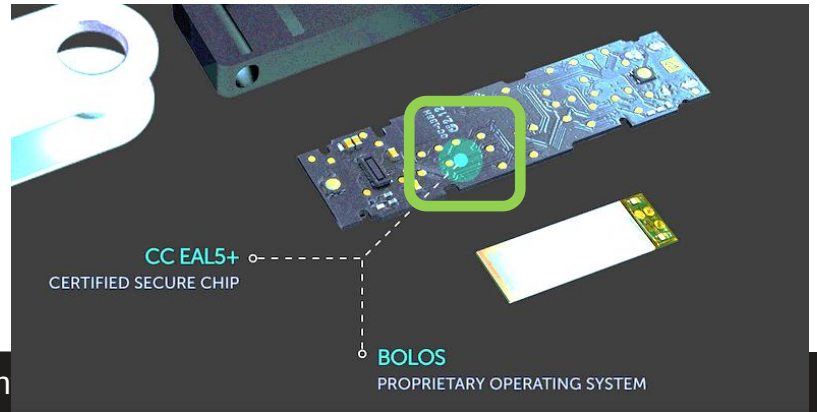
25+ different hardware wallets

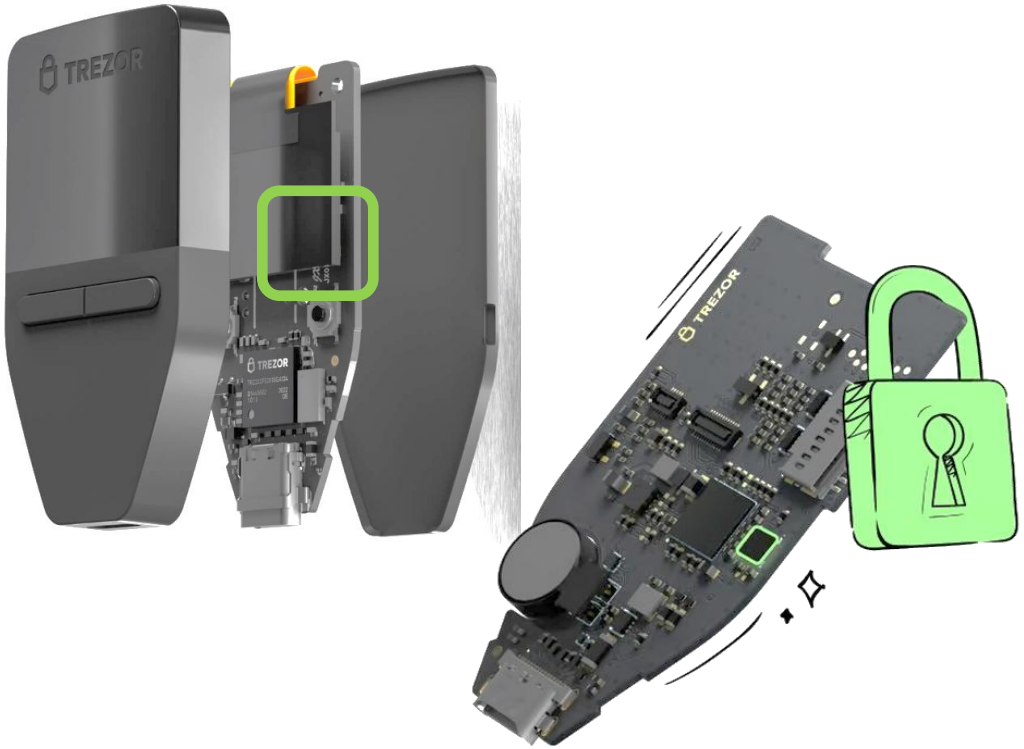
Analysis of cryptographic implementations (ECDSA, BIP32, TRNG…)

# Secure element frequently used (ECDSA, EdDSA, TRNG)

- Generate new wallet seed, sign transaction, protected seed storage

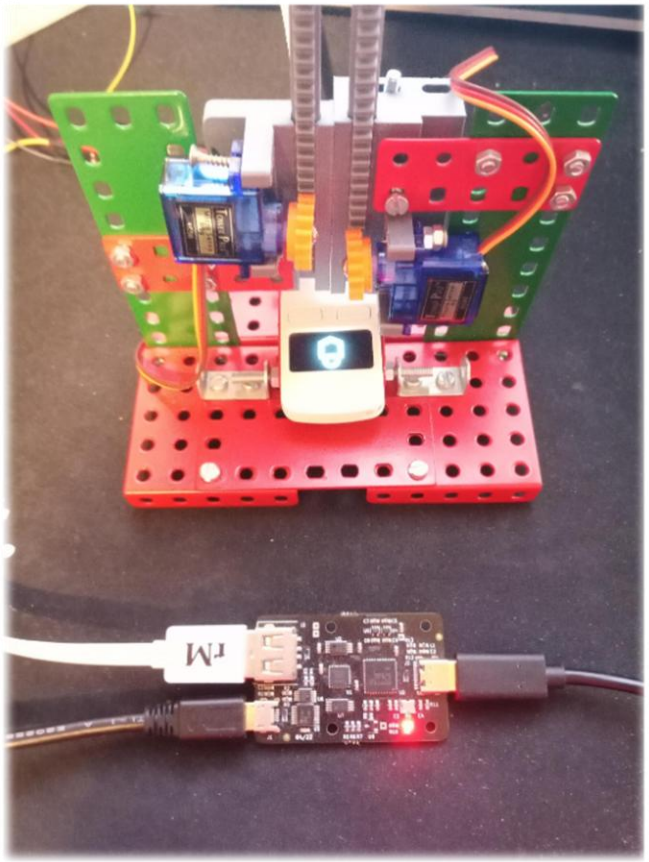


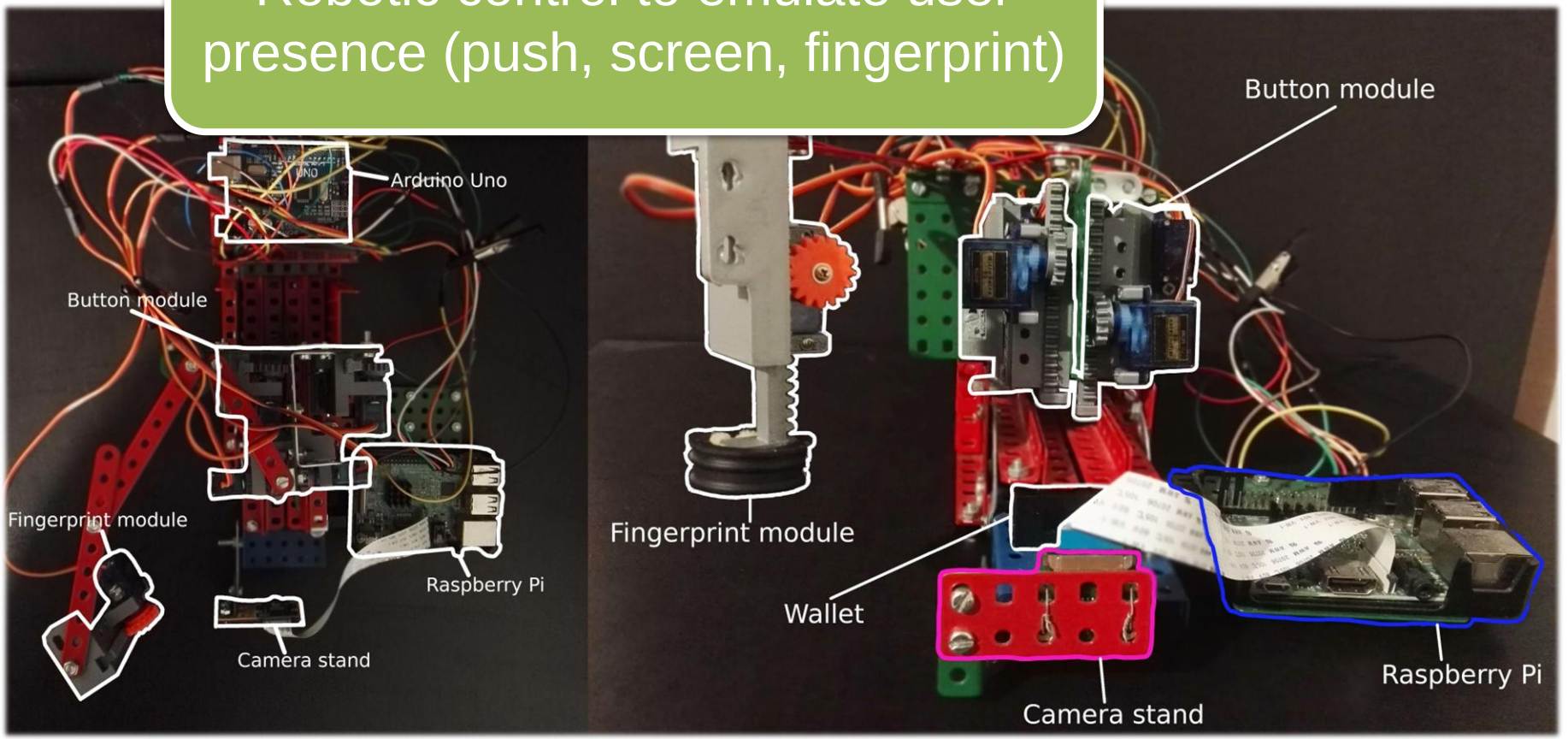




*https://shop.cobo.com/products/cobo-vault-essential https://coldcard.com https://trezor.io https://ledger.com*

# Automatically extracting data from (intentionally) non-automatic device (cryptocurrency hardware wallet)

Robotic control to emulate user presence (push, screen, fingerprint)





# Data collected for hardware wallets

- Random entropy as generated by wallets (100 000x)
- Signatures and nonces used for ECDSA operation (100 000x)
- (Coarse) timing of cryptographic operations
  - Host clock measurements, visual trigger, sound trigger
  - (for more time demanding SCA setups see Ledger Donjon, Kraken Labs…)
- What we found:
  - Not much ☺
  - (no biased seed, no biased nonce, no timing dependency…)
  - Unexpected: some wallets undertook audits, but not CC/FIPS140 certifications

# Why we failed for cryptocurrency hardware wallets?

- Wallets have very limited interface (limits your odds of finding something)
  - Only very narrow set of functions exposed, only high-level API available
  - Fixed EC curve domain parameters, pre-defined data structures…
- Secure-by-default primitives
  - Deterministic nonce generation for ECDSA (RFC6979)
  - Hash entropy before its use (initial entropy for mnemonic backup)
  - Seed generated from mnemonics (PBKDF2-HMAC512-2048)
- High-quality open-source implementations with permissive license
  - SatoshiLabs Trezor wallet, Ledger wallet, BitcoinDevKit…
  - Easy to copy from, easy to adopt security patches (same/similar code)…
  - (Smartcards/TPMs… have closed and different implementations)
    - => limited/no learning from vulnerability of other vendors, even if vulnerability disclosed

# Conclusions

- Blackbox analysis of cryptographic targets is viable even for academics
- Wide-scale, long-running analysis is the crucial factor
  - Increase your odds by focusing on multiple target devices / libraries in parallel
  - Allows to involve more people more quickly (undergrads to research)
  - Creative attack ideas needs time (prolonged analysis timeframe)
- Disadvantage (initial lack of details) turned into advantage
  - Security with addition of obscurity layer results in obscured bugs
  - Open tools to increase transparency (no NDA burden)
- Worked for classic crypto algs, likely to work for PQC ones as well
  - But maybe with less real impact…?

# Conclusions (cont.)

Questions

- Not much changed in certification ecosystem
  - System still not accommodating academic/independent researchers
  - More openness and availability of information needed
- Advent of open cryptographic hardware?
  - Open-source and mature implementations, learning from flows of others?
  - PQC harder on hardware => will come even later