presented by

binarly

Repeatable Supply Chain Security Failures in Firmware Key Management

UEFI 2025 Developers Conference & Plugfest
October 10, 2025

Presented by:
Alex Matrosov, Fabio Pagani

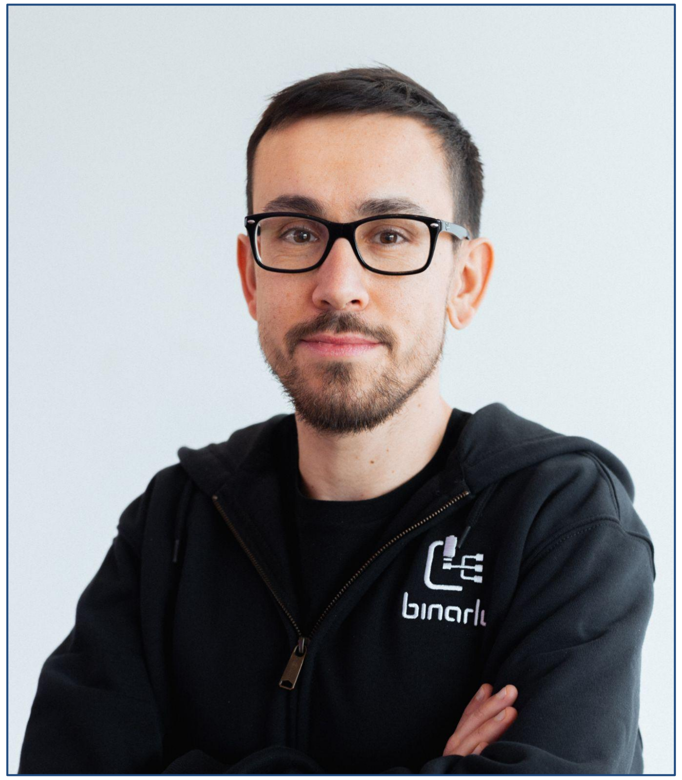
Meet the Presenters

Alex Matrosov

CEO & Head of Research

Alex Matrosov is CEO and Founder of Binarly Inc. where he builds an AI-powered platform to protect devices against emerging firmware threats. Alex has more than two decades of cybersecurity experience. He served as Chief Offensive Security Researcher at Nvidia and Intel Security Center of Excellence (SeCoE). Alex is the Author of numerous research papers and the bestselling award-winning book “Rootkits and Bootkits: Reversing Modern Malware and Next Generation Threats”. He is a frequently invited speaker at security conferences, such as REcon, Black Hat, Offensivecon, WOOT, DEF CON, and many others. Additionally, he was awarded multiple times by Hex-Rays for his open source contributions to the research community.

Meet the Presenters



Fabio Pagani

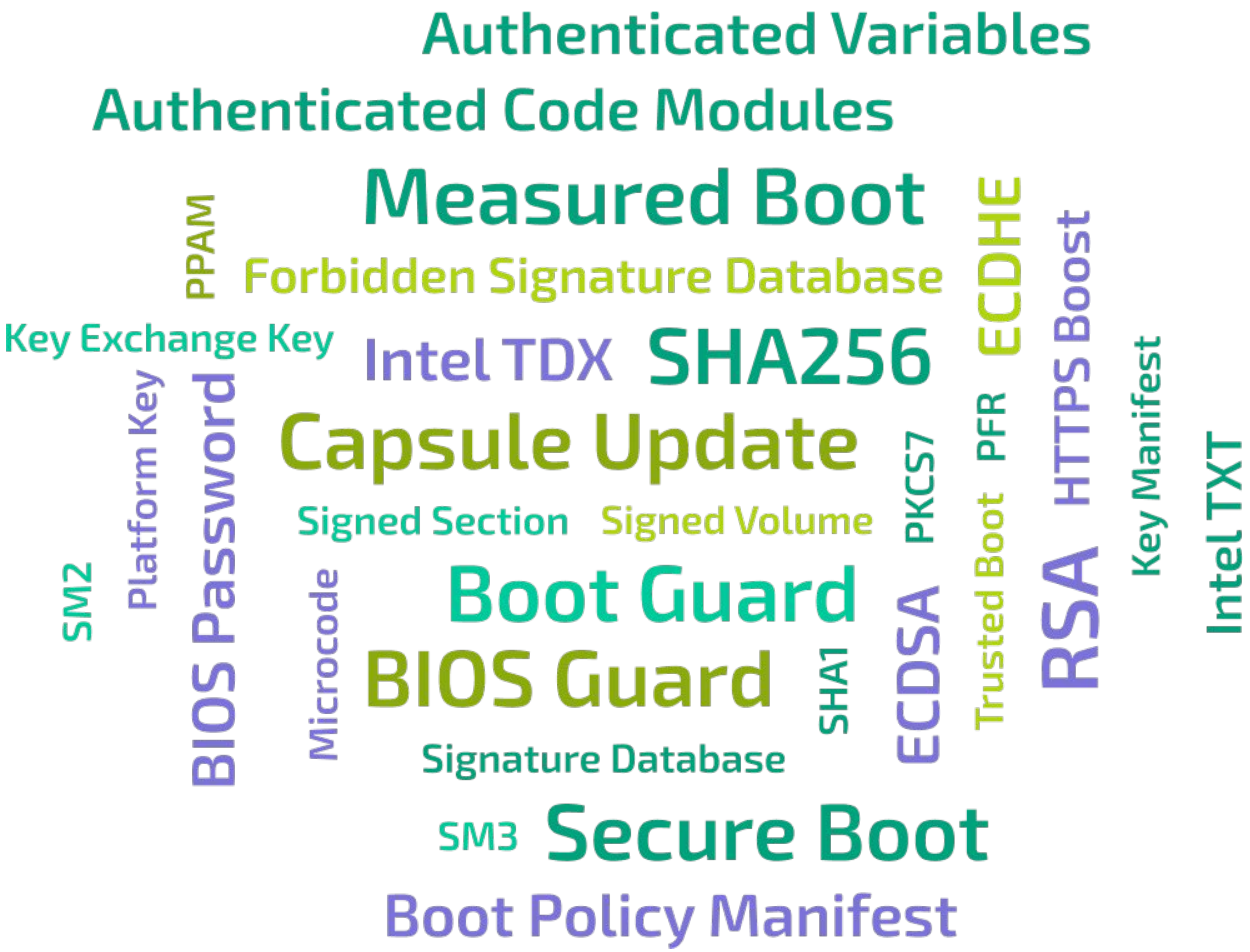
Vulnerability Research Lead

Fabio Pagani is a Vulnerability Research Lead at Binarly, where he works at the intersection of static and dynamic analysis techniques to help secure the UEFI ecosystem. As part of the Binarly REsearch team, he discovered LogoFAIL and helped affected vendors to identify and mitigate this vulnerability. Fabio is always on the lookout for new and impactful firmware vulnerabilities. He also maintains strong connections with the academic community, serving on the program committees of security conferences such as USENIX Security and WOOT.

All this has happened before.

All this will happen again.

Introduction



Authenticated Variables

Authenticated Code Modules

PPAM

Measured Boot

Forbidden Signature Database

ECDHE

HTTPS Boost

Key Exchange Key

Intel TDX

SHA256

Platform Key

BIOS Password

Capsule Update

PKCS7

PFR

Key Manifest

Intel TXT

Signed Section

Signed Volume

SM2

Microcode

Boot Guard

BIOS Guard

SHA1

ECDSA

Trusted Boot

RSA

Signature Database

SM3

Secure Boot

Boot Policy Manifest

Introduction

What can go wrong?

Agenda

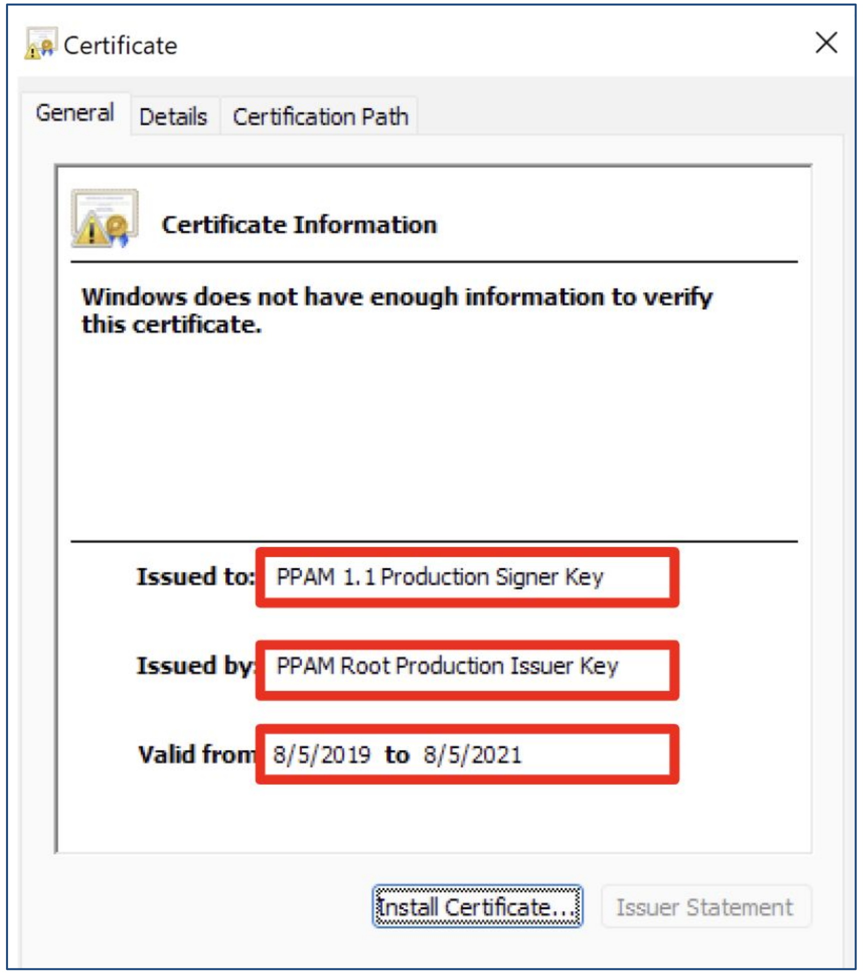
- Intel PPAM expired certificate
- Data breaches and leaked keys
  - Impact on Boot Guard
- Leaked Platform Key (PKfail)
  - Impact on Secure Boot
- Inconsistency in Secure Boot dbx
- Microcode vulnerabilities
- Post-Quantum readiness

[2022] Intel Platform Properties Assessment Module (PPAM)

Expired Certificate Story

[2022] Intel PPAM Expired Certificate

- Platform Properties Assessment Module (PPAM) measures the integrity of SMM code
- Binary signed by Intel that runs before System Management Mode (SMM) entry point
- PKCS7 certificate provides a digital signature for PPAM
- Multiple devices with expired PPAM certificate



Certificate Information

Windows does not have enough information to verify this certificate.

Issued to: PPAM 1.1 Production Signer Key

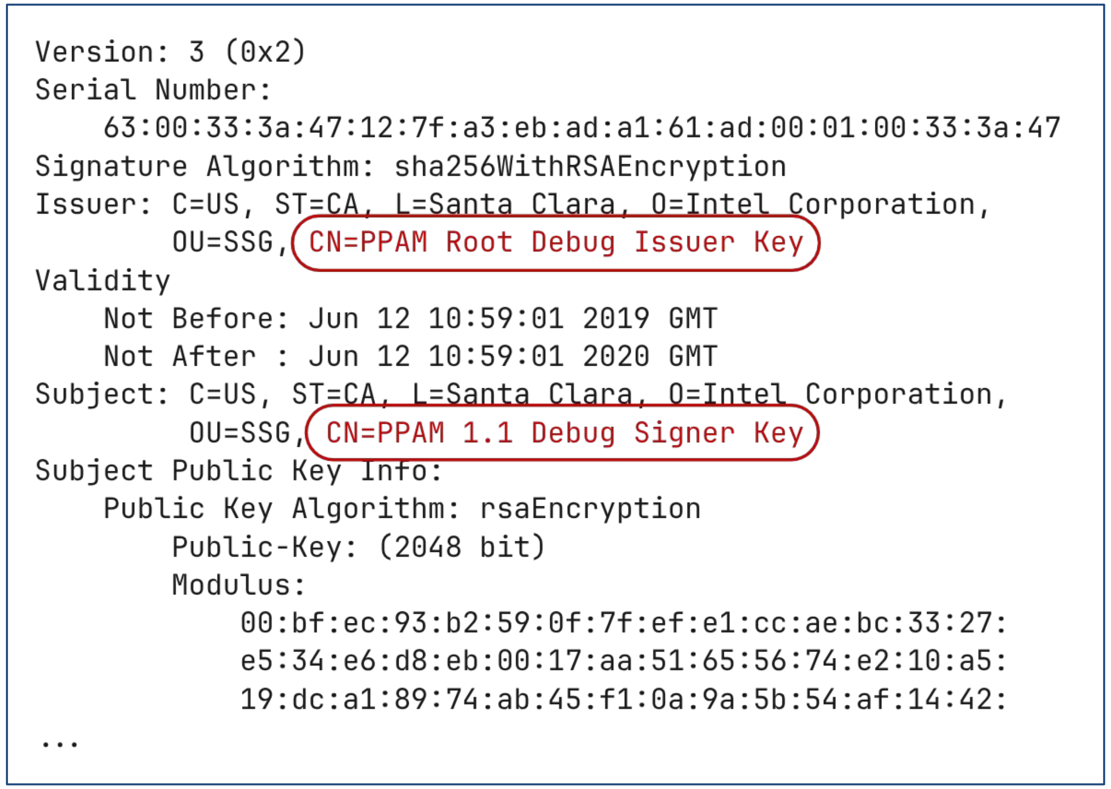
Issued by: PPAM Root Production Issuer Key

Valid from 8/5/2019 to 8/5/2021

https://www.binarly.io/blog/black-hat-2022-the-intel-ppam-attack-story

Revisiting Intel PPAM Expired Certificate

- Retrospective scan on our dataset revealed that 68% of certificates in-the-wild are expired
- We also found some recents devices deployed with PPAM debug certificates
- Not a security vulnerability, but highlights potential for improved security practice



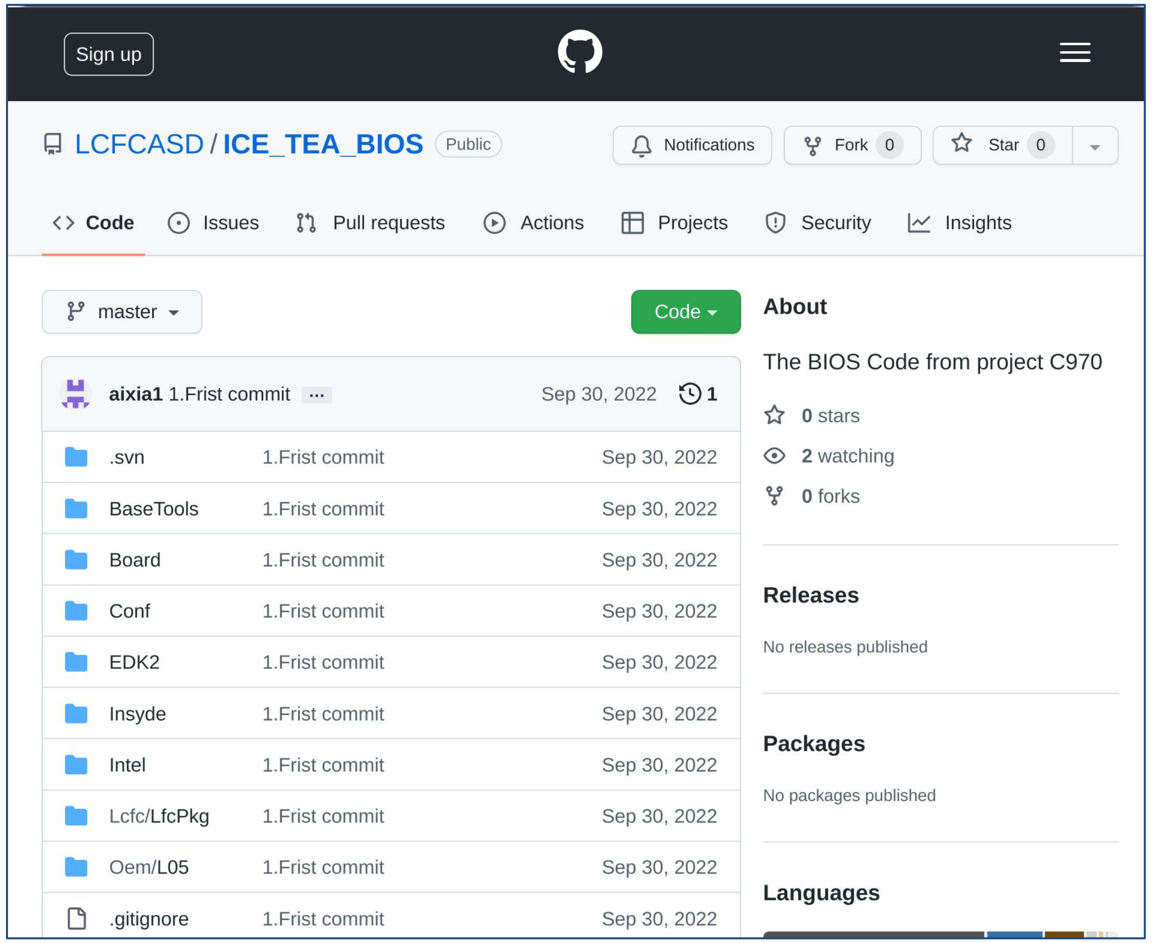
```
Version: 3 (0x2)
Serial Number:
    63:00:33:3a:47:12:7f:a3:eb:ad:a1:61:ad:00:01:00:33:3a:47
Signature Algorithm: sha256WithRSAEncryption
Issuer: C=US, ST=CA, L=Santa Clara, O=Intel Corporation,
        OU=SSG, CN=PPAM Root Debug Issuer Key
Validity
    Not Before: Jun 12 10:59:01 2019 GMT
    Not After : Jun 12 10:59:01 2020 GMT
Subject: C=US, ST=CA, L=Santa Clara, O=Intel Corporation,
         OU=SSG, CN=PPAM 1.1 Debug Signer Key
Subject Public Key Info:
    Public Key Algorithm: rsaEncryption
        Public-Key: (2048 bit)
        Modulus:
            00:bf:ec:93:b2:59:0f:7f:ef:e1:cc:ae:bc:33:27:
            e5:34:e6:d8:eb:00:17:aa:51:65:56:74:e2:10:a5:
            19:dc:a1:89:74:ab:45:f1:0a:9a:5b:54:af:14:42:
...
```

[2022] LC/FC OEM

Data Breach and Leaked Keys

[2022] LC/FC Data Breach

- Alder Lake's UEFI firmware was leaked on GitHub
- Reference implementation (Intel), IBV solution (Insyde) and OEM implementation (Lenovo)
- 6GB of source code, binary blobs, debugging tools and multiple private keys



LCFCASD / ICE_TEA_BIOS Public

aixia1 1.Frist commit ... Sep 30, 2022 1

.svn	1.Frist commit	Sep 30, 2022
BaseTools	1.Frist commit	Sep 30, 2022
Board	1.Frist commit	Sep 30, 2022
Conf	1.Frist commit	Sep 30, 2022
EDK2	1.Frist commit	Sep 30, 2022
Insyde	1.Frist commit	Sep 30, 2022
Intel	1.Frist commit	Sep 30, 2022
Lcfc/LfcPkg	1.Frist commit	Sep 30, 2022
Oem/L05	1.Frist commit	Sep 30, 2022
.gitignore	1.Frist commit	Sep 30, 2022

About

The BIOS Code from project C970

0 stars

2 watching

0 forks

Releases

No releases published

Packages

No packages published

Languages

https://www.binarly.io/blog/leaked-intel-boot-guard-keys-what-happened-how-does-it-affect-the-software-supply-chain

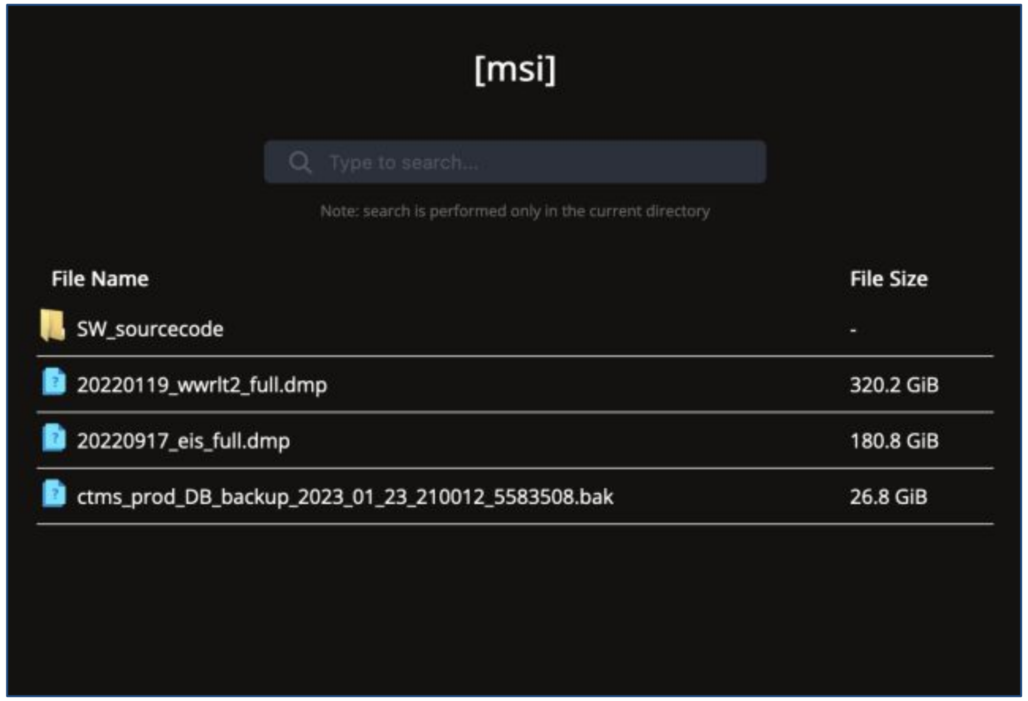
[2023] MSI OEM

Data Breach and Leaked Keys

[2023] MSI OEM Data Breach

- Breach from the Money Message ransomware group
- 1.5TB of source code, production databases and multiple private keys

https://www.binarly.io/blog/leaked-msi-source-code-with-intel-oem-keys-how-does-this-affect-industry-wide-software-supply-chain



[msi]

Type to search...

Note: search is performed only in the current directory

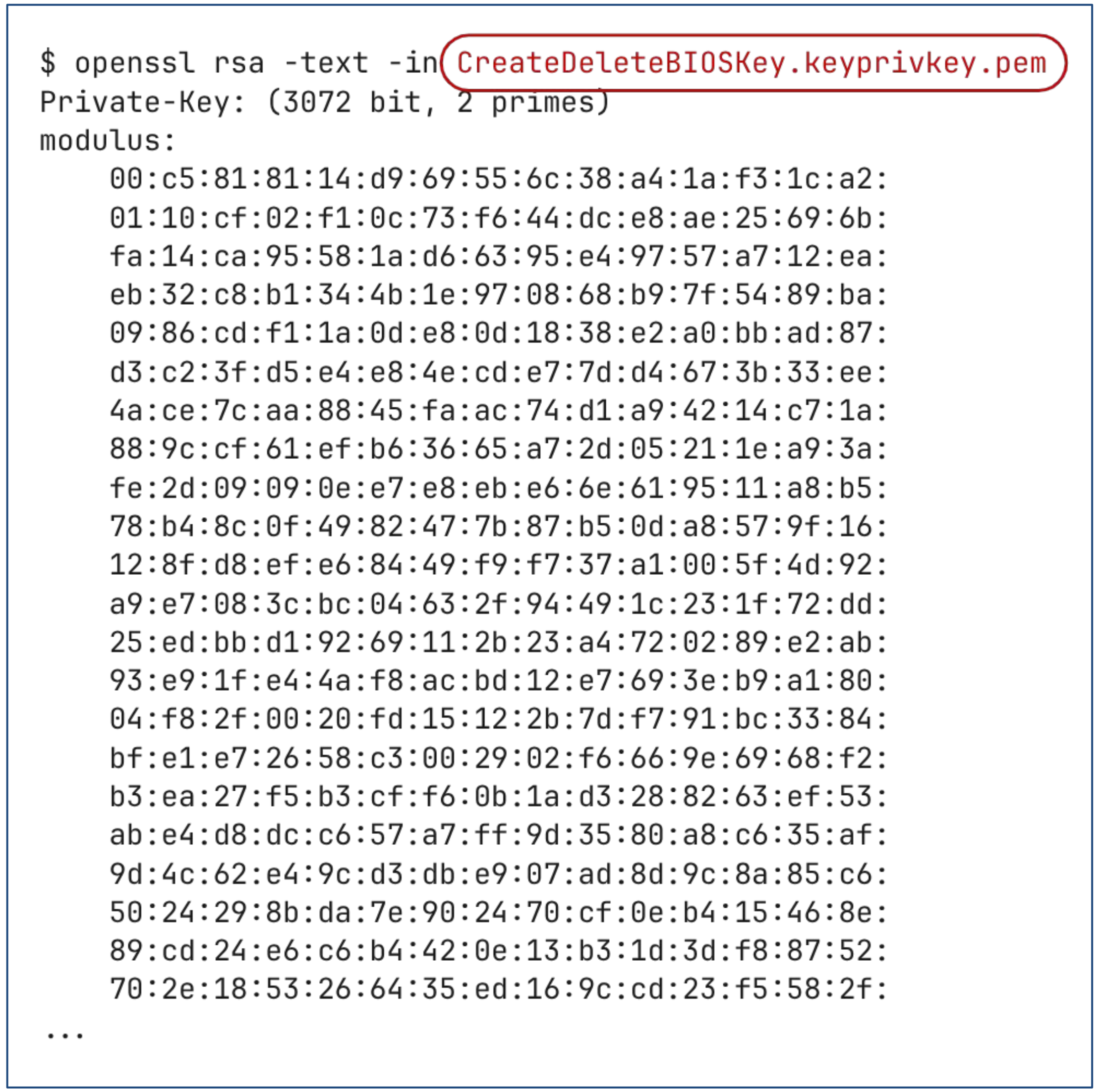
File Name	File Size
SW_sourcecode	-
20220119_wwrlt2_full.dmp	320.2 GiB
20220917_eis_full.dmp	180.8 GiB
ctms_prod_DB_backup_2023_01_23_210012_5583508.bak	26.8 GiB

[2025] Clevo OEM

Leaked Keys

[2025] Clevo Boot Guard Keys Leak

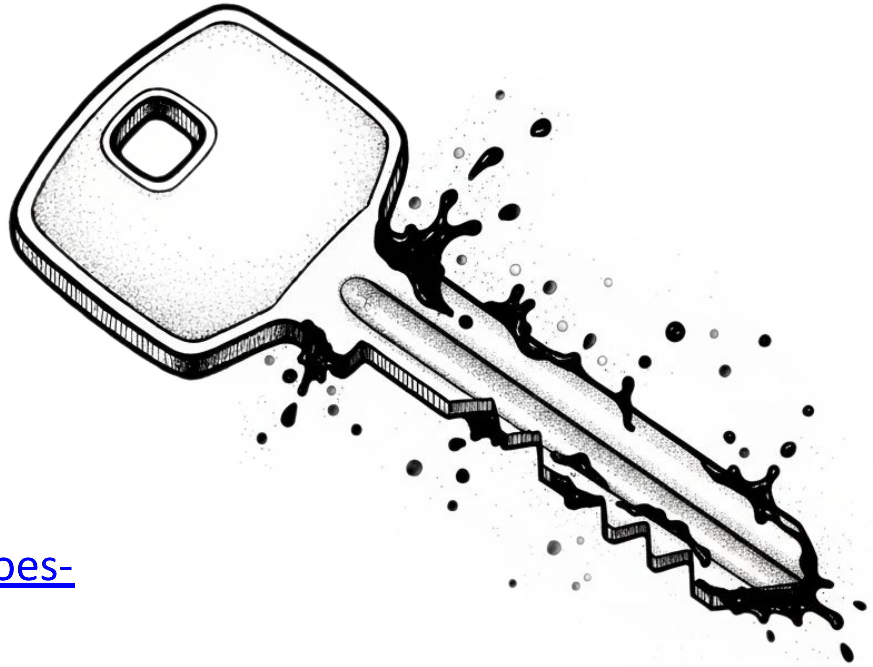
- Binarly was notified by Thierry Laurion about a possible leak of Boot Guard keys from Clevo in March 2025
- Firmware update package (400MB uncompressed size)
- Contains user manuals, internal tooling, firmware images and Boot Guard private keys



```
$ openssl rsa -text -in CreateDeleteBIOSKey.keyprivkey.pem
Private-Key: (3072 bit, 2 primes)
modulus:
    00:c5:81:81:14:d9:69:55:6c:38:a4:1a:f3:1c:a2:
    01:10:cf:02:f1:0c:73:f6:44:dc:e8:ae:25:69:6b:
    fa:14:ca:95:58:1a:d6:63:95:e4:97:57:a7:12:ea:
    eb:32:c8:b1:34:4b:1e:97:08:68:b9:7f:54:89:ba:
    09:86:cd:f1:1a:0d:e8:0d:18:38:e2:a0:bb:ad:87:
    d3:c2:3f:d5:e4:e8:4e:cd:e7:7d:d4:67:3b:33:ee:
    4a:ce:7c:aa:88:45:fa:ac:74:d1:a9:42:14:c7:1a:
    88:9c:cf:61:ef:b6:36:65:a7:2d:05:21:1e:a9:3a:
    fe:2d:09:09:0e:e7:e8:eb:e6:6e:61:95:11:a8:b5:
    78:b4:8c:0f:49:82:47:7b:87:b5:0d:a8:57:9f:16:
    12:8f:d8:ef:e6:84:49:f9:f7:37:a1:00:5f:4d:92:
    a9:e7:08:3c:bc:04:63:2f:94:49:1c:23:1f:72:dd:
    25:ed:bb:d1:92:69:11:2b:23:a4:72:02:89:e2:ab:
    93:e9:1f:e4:4a:f8:ac:bd:12:e7:69:3e:b9:a1:80:
    04:f8:2f:00:20:fd:15:12:2b:7d:f7:91:bc:33:84:
    bf:e1:e7:26:58:c3:00:29:02:f6:66:9e:69:68:f2:
    b3:ea:27:f5:b3:cf:f6:0b:1a:d3:28:82:63:ef:53:
    ab:e4:d8:dc:c6:57:a7:ff:9d:35:80:a8:c6:35:af:
    9d:4c:62:e4:9c:d3:db:e9:07:ad:8d:9c:8a:85:c6:
    50:24:29:8b:da:7e:90:24:70:cf:0e:b4:15:46:8e:
    89:cd:24:e6:c6:b4:42:0e:13:b3:1d:3d:f8:87:52:
    70:2e:18:53:26:64:35:ed:16:9c:cd:23:f5:58:2f:
...
```

Keys Leaked From Past Incidents

- Intel Integrated Sensors Hub (ISH) signing key
- FW Image signing Keys
- Intel OEM Platform Key
- Intel Boot Guard KM/BPM keys
- …



https://www.binarly.io/blog/leaked-msi-source-code-with-intel-oem-keys-how-does-this-affect-industry-wide-software-supply-chain

https://www.binarly.io/blog/leaked-intel-boot-guard-keys-what-happened-how-does-it-affect-the-software-supply-chain

https://www.binarly.io/blog/clevo-boot-guard-keys-leaked-in-update-package

Intel Boot Guard

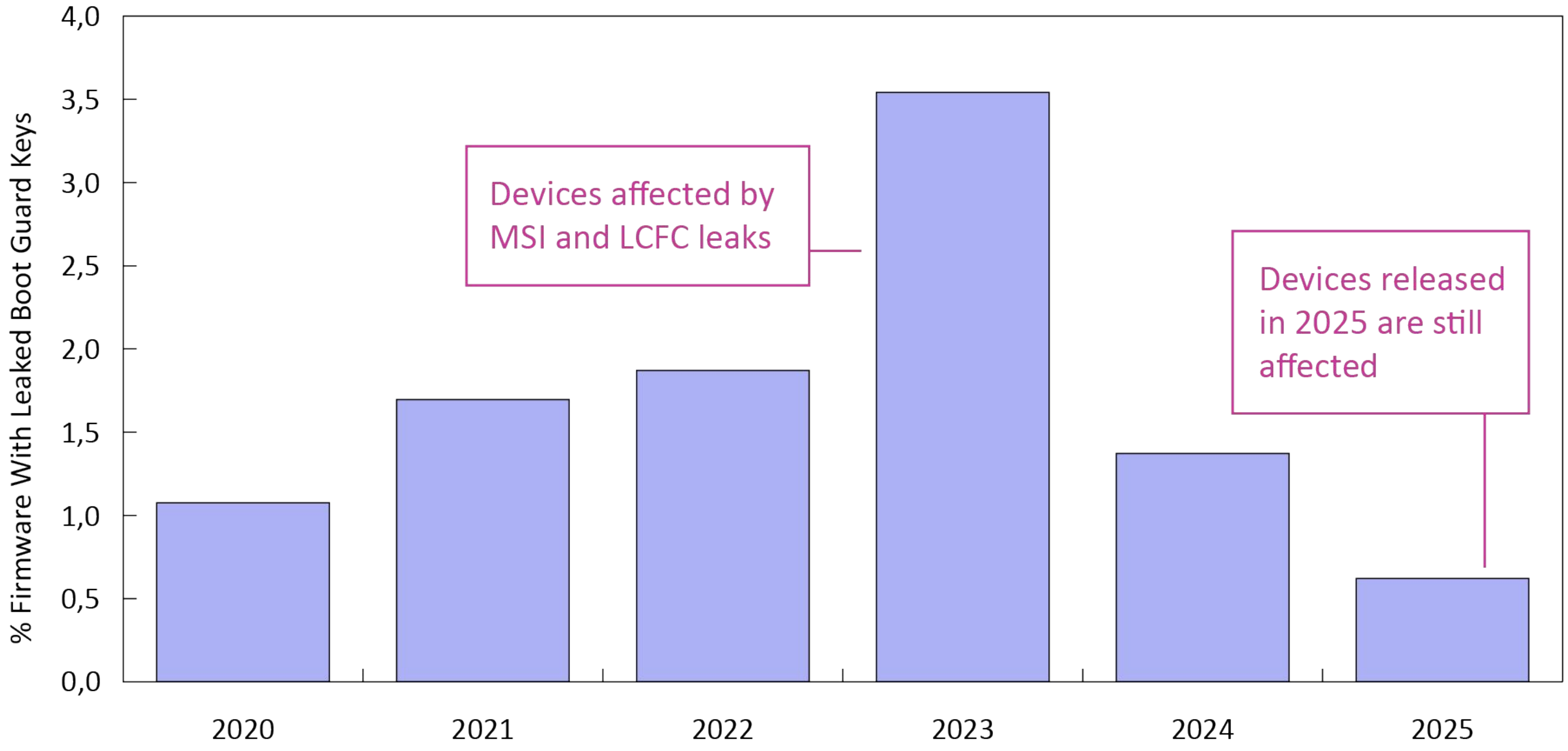
Impact of Leaked Keys

Boot Guard — Introduction

- Hardware-based technology intended to protect against execution of non-genuine UEFI firmware
- Multiple components and cryptographic keys involved:

1. **Authenticated code module (ACM)**: Intel-signed code that runs before the firmware and cryptographically verifies the firmware
2. **Key Manifest (KM)**: verifies Boot Policy Manifest
3. **Boot Policy Manifest (BPM)**: verifies Initial Boot Block

Impact of Boot Guard Keys Leakage



Devices affected by MSI and LCFC leaks

Devices released in 2025 are still affected

Impact of Boot Guard Keys Leakage

Why current devices are still vulnerable to a leak from years ago?

The Boot Guard Key Manifest hash is fused in the platform hardware and it cannot be changed!

Impact of Boot Guard Keys Leakage

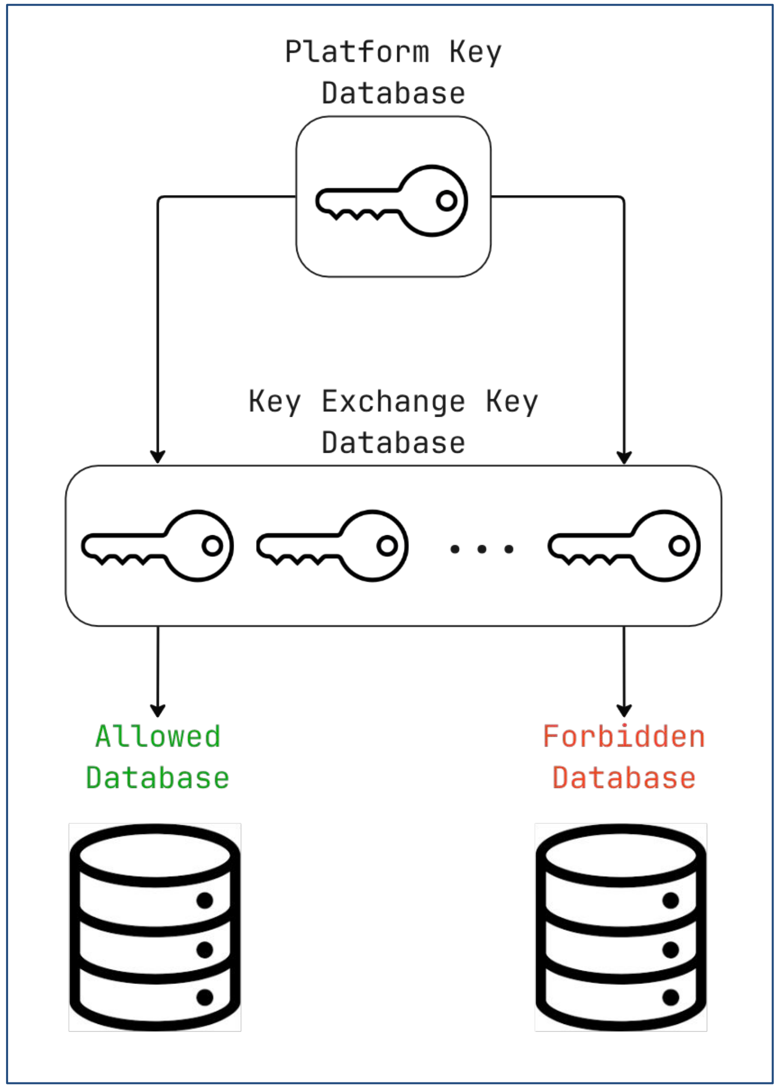
https://hardenedlinux.org/blog/2023-09-07-boot-unguarded-x86-trust-anchor-downfalls-to-the-leaked-oem-internal-tools-and-signing-keys/

[2024] PKfail

Leaked Platform Key Story

UEFI Secure Boot

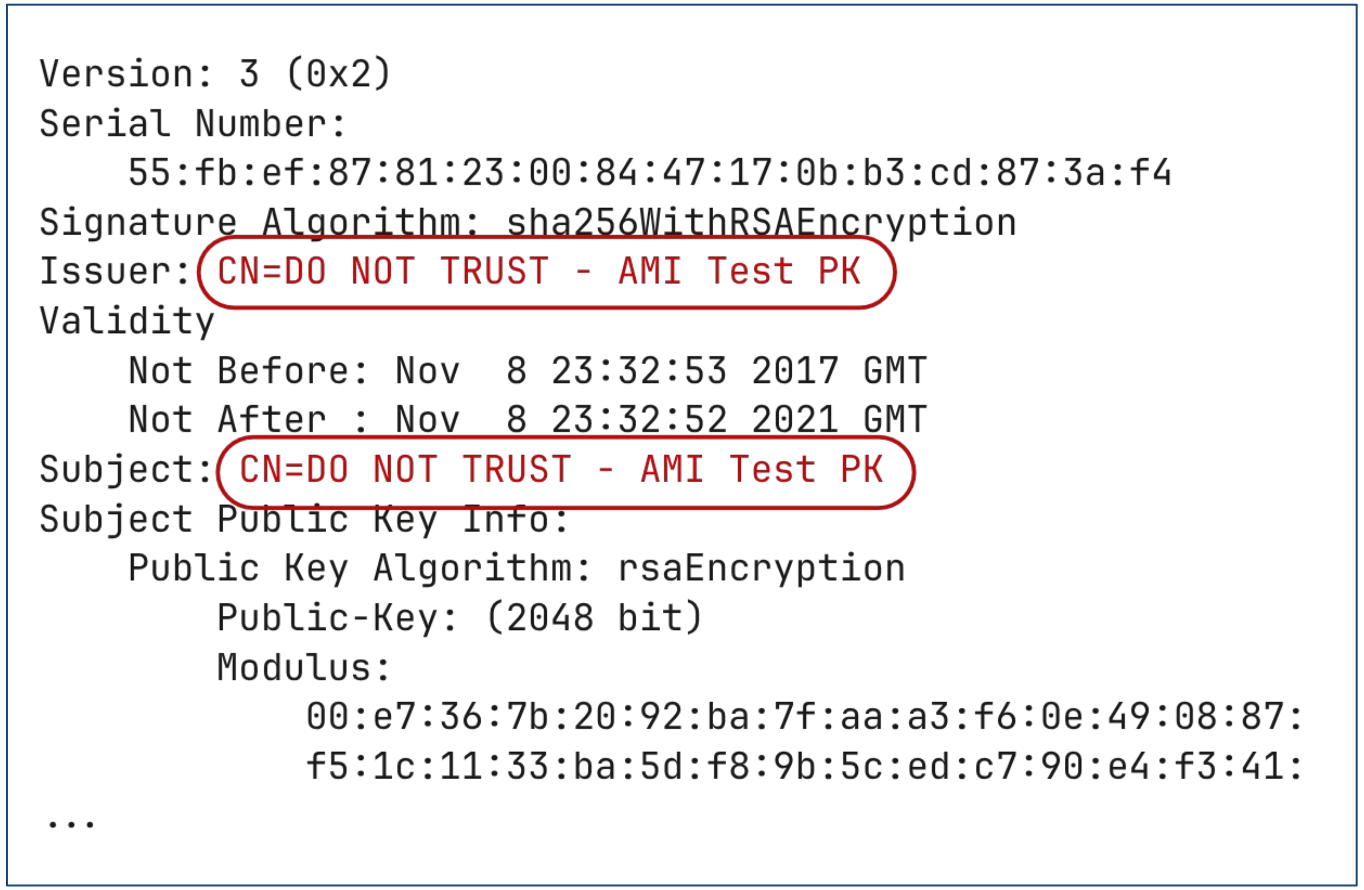
- Allows only trusted, digitally signed software to run during system startup, preventing malware and unauthorized code execution.
- Bypassing Secure Boot allows for bootkit and rootkit execution
- Four databases:
  - PK, KEK, db, dbx



[2024] PKFail

While adding support for Secure Boot to our Binarly Transparency Platform, we found an “*interesting*” Platform Key:

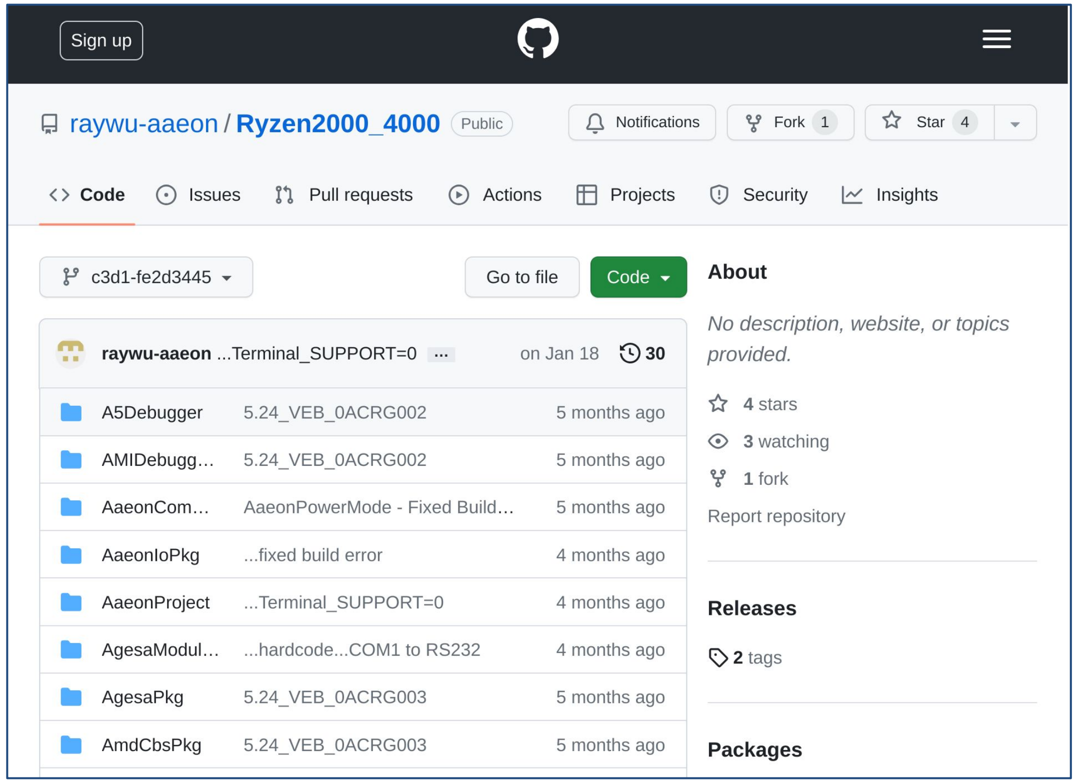
https://www.binarly.io/blog/pkfail-untrusted-platform-keys-undermine-secure-boot-on-uefi-ecosystem



```
Version: 3 (0x2)
Serial Number:
    55:fb:ef:87:81:23:00:84:47:17:0b:b3:cd:87:3a:f4
Signature Algorithm: sha256WithRSAEncryption
Issuer: CN=DO NOT TRUST - AMI Test PK
Validity
    Not Before: Nov  8 23:32:53 2017 GMT
    Not After : Nov  8 23:32:52 2021 GMT
Subject: CN=DO NOT TRUST - AMI Test PK
Subject Public Key Info:
    Public Key Algorithm: rsaEncryption
        Public-Key: (2048 bit)
        Modulus:
            00:e7:36:7b:20:92:ba:7f:aa:a3:f6:0e:49:08:87:
            f5:1c:11:33:ba:5d:f8:9b:5c:ed:c7:90:e4:f3:41:
...
```

[2023] AAeon Leak

- In January 2023, the repository **Ryzen2000_4000** is published on GitHub
- Contains IBV (AMI) reference implementation, ODM (AAeon) implementation and private keys
- Remained public until DMCA sent to GitHub in June 2023



raywu-aaeon / **Ryzen2000_4000** Public

raywu-aaeon ...Terminal_SUPPORT=0 ... on Jan 18 30

A5Debugger	5.24_VEB_0ACRG002	5 months ago
AMIDebugg…	5.24_VEB_0ACRG002	5 months ago
AaeonCom…	AaeonPowerMode - Fixed Build…	5 months ago
AaeonIoPkg	...fixed build error	4 months ago
AaeonProject	...Terminal_SUPPORT=0	4 months ago
AgesaModul…	...hardcode...COM1 to RS232	4 months ago
AgesaPkg	5.24_VEB_0ACRG003	5 months ago
AmdCbsPkg	5.24_VEB_0ACRG003	5 months ago

About

No description, website, or topics provided.

4 stars

3 watching

1 fork

Report repository

Releases

2 tags

Packages

[2023] AAeon Leak

```
$ openssl x509 -noout -text -in FW_pubKey.cer  | rg "Issuer:|Subject:"
        Issuer: CN=DO NOT TRUST - AMI Test PK
        Subject: CN=DO NOT TRUST - AMI Test PK
```

```
$ openssl pkcs12 -in FW_priKey.pfx -nodes
Enter Import Password:
```

Oh, hi! I am a private key that’s been available on GitHub for 6 months! 🤷‍♂️

```
$ cat AmiTestKey.sdl | grep password -C3
TOKEN
        Name  = "FW_PFX_Password"
        Value  = "abcd"
        Help  = "Specifies the password to use when opening a PFX - 
Private Key container file."
        TokenType = Expression
        TargetMAK = Yes
End
```

Retrospective View on PKFail

Dataset with 80,000 UEFI firmware images:

- Spanning over 10 years
- Includes every major vendor (Lenovo, Dell, HP, Intel..)

Results:

- 10% of images use non-production keys
- 8% of images when selecting images released in the past 4 years
- 22 unique non-production keys identified

🔥

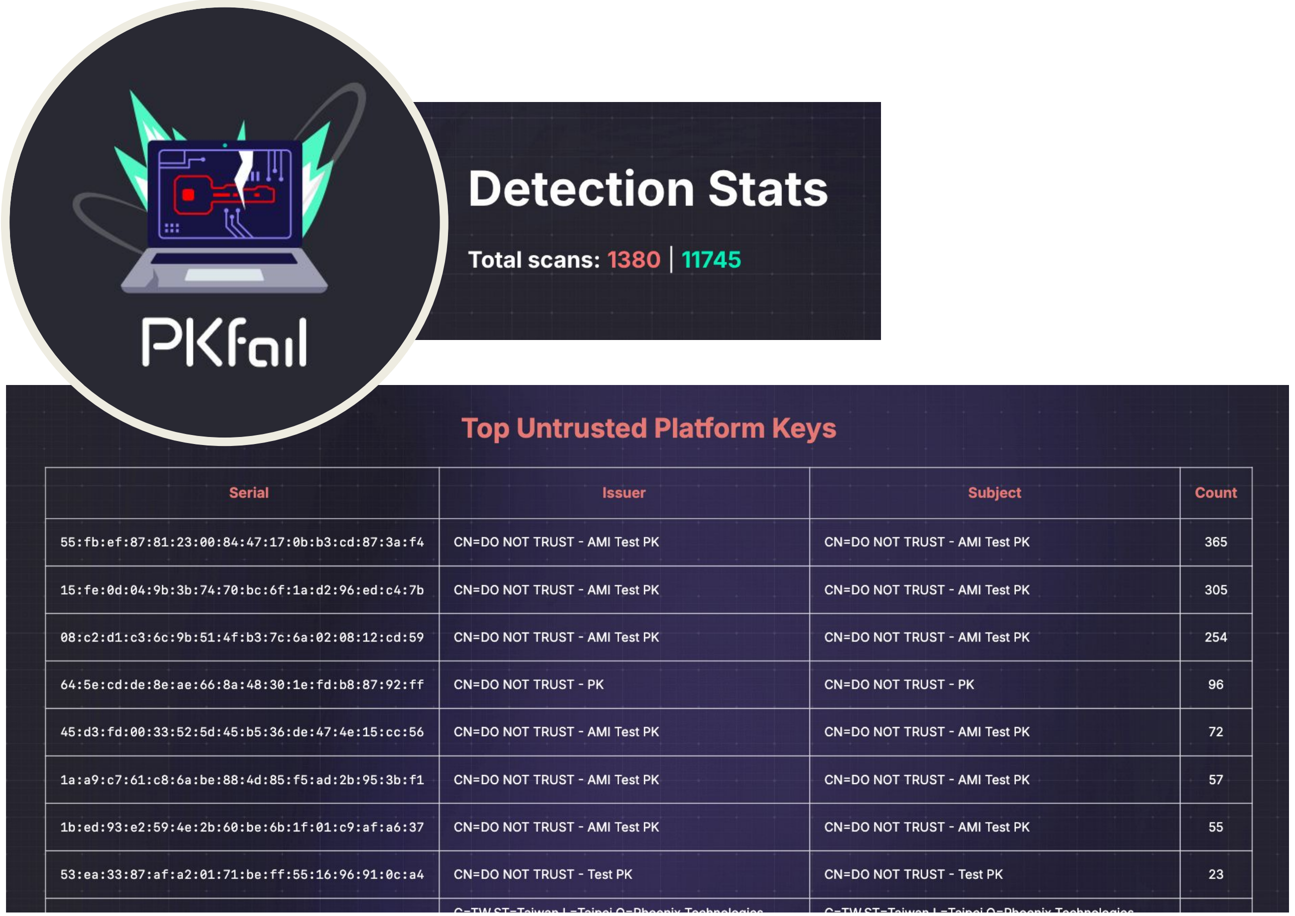
Retrospective View on PKFail

Certificate Serial Number	Certificate Subject	Certificate Issuer	Last Seen	First Seen	Products	Vendors
55:fb:ef:87:81:23:00:84:47:17:0b:b3:cd:87:3a:f4	CN=DO NOT TRUST - AMI Test PK	CN=DO NOT TRUST - AMI Test PK	2024-06	2018-04	364	Acer, Dell, Fujitsu, Gigabyte, Intel, Lenovo, Supermicro
-08:c2:d1:c3:6c:9b:51:4f:b3:7c:6a:02:08:12:cd:59	CN=DO NOT TRUST - AMI Test PK	CN=DO NOT TRUST - AMI Test PK	2024-06	2022-06	167	Acer, Dell, Gigabyte, Supermicro
-15:fe:0d:04:9b:3b:74:70:bc:6f:1a:d2:96:ed:c4:7b	CN=DO NOT TRUST - AMI Test PK	CN=DO NOT TRUST - AMI Test PK	2024-03	2015-01	483	Acer, Dell, Gigabyte, Intel, Lenovo, Supermicro
-1b:ed:93:e2:59:4e:2b:60:be:6b:1f:01:c9:af:a6:37	CN=DO NOT TRUST - AMI Test PK	CN=DO NOT TRUST - AMI Test PK	2023-01	2014-12	287	Dell, Fujitsu, Gigabyte, HP, Intel, Lenovo, Supermicro
1a:a9:c7:61:c8:6a:be:88:4d:85:f5:ad:2b:95:3b:f1	CN=DO NOT TRUST - AMI Test PK	CN=DO NOT TRUST - AMI Test PK	2021-03	2012-05	157	Acer, Dell, Fujitsu, Gigabyte, HP, Lenovo, Samsung, Supermicro

Binarly’s pk.fail Detection Service

Binarly released a free detection service for the community on disclosure date:

- Users uploaded 13,125 firmware images
- Found untrusted keys in 1,380 of them (10.51%)
- The most common key remains the leaked key



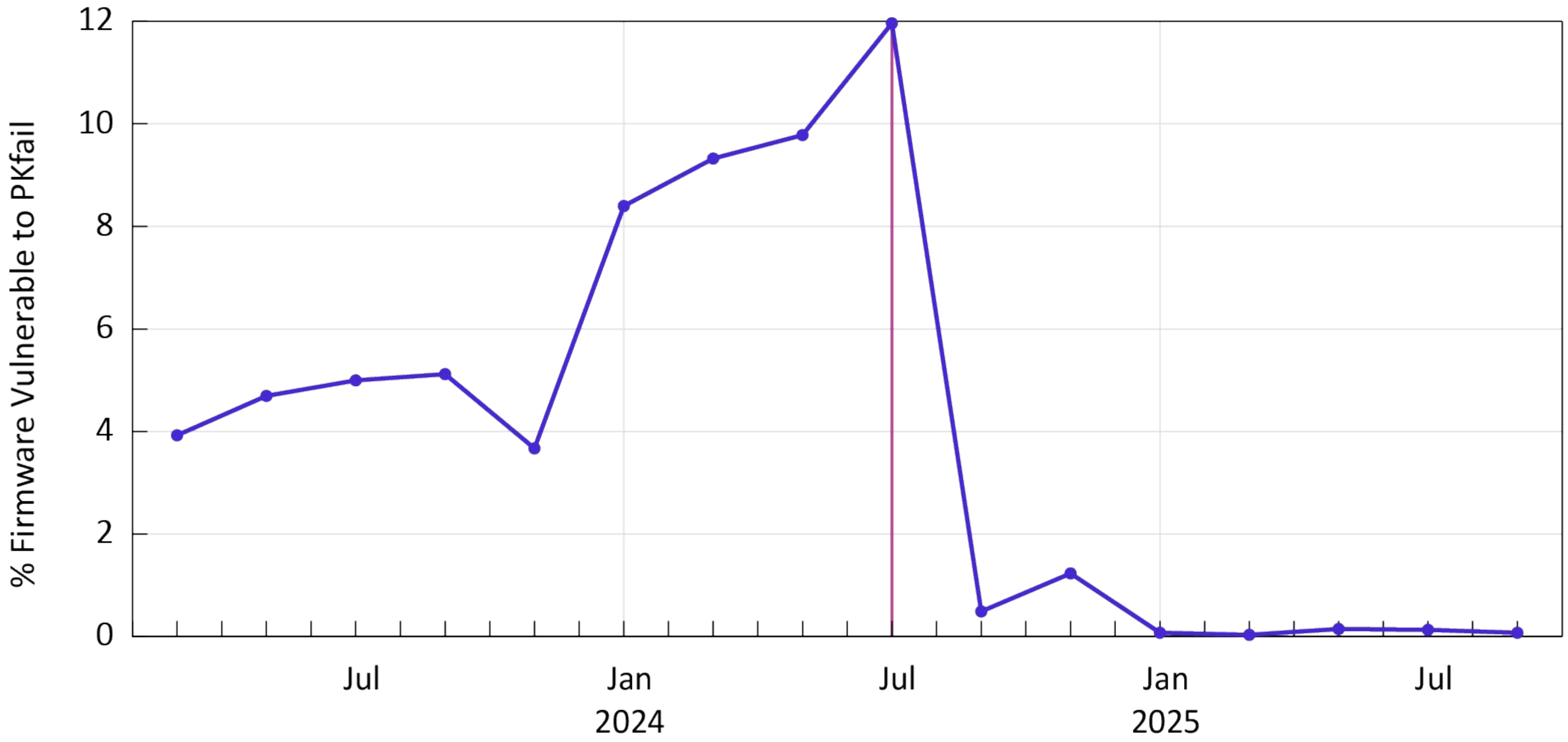
Detection Stats

Total scans: 1380 | 11745

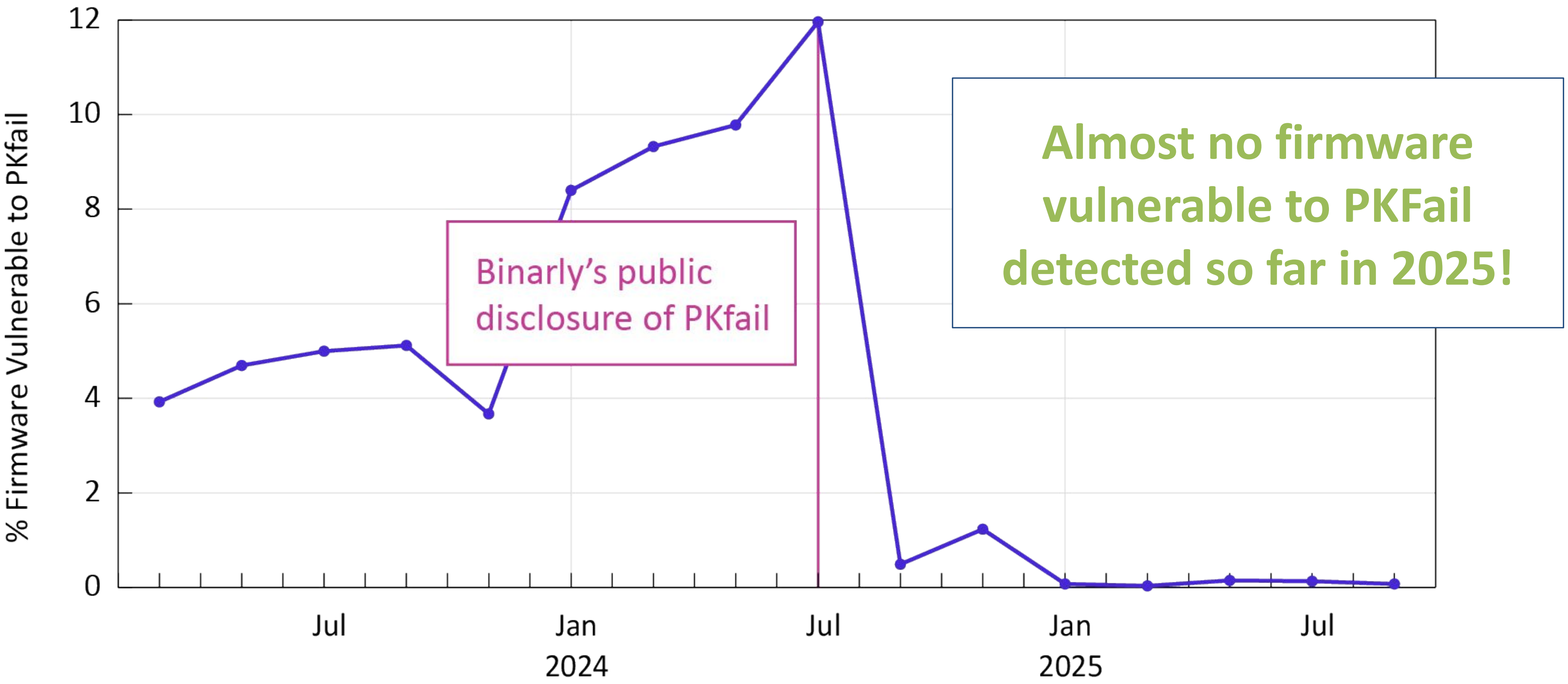
Top Untrusted Platform Keys

Serial	Issuer	Subject	Count
55:fb:ef:87:81:23:00:84:47:17:0b:b3:cd:87:3a:f4	CN=DO NOT TRUST - AMI Test PK	CN=DO NOT TRUST - AMI Test PK	365
15:fe:0d:04:9b:3b:74:70:bc:6f:1a:d2:96:ed:c4:7b	CN=DO NOT TRUST - AMI Test PK	CN=DO NOT TRUST - AMI Test PK	305
08:c2:d1:c3:6c:9b:51:4f:b3:7c:6a:02:08:12:cd:59	CN=DO NOT TRUST - AMI Test PK	CN=DO NOT TRUST - AMI Test PK	254
64:5e:cd:de:8e:ae:66:8a:48:30:1e:fd:b8:87:92:ff	CN=DO NOT TRUST - PK	CN=DO NOT TRUST - PK	96
45:d3:fd:00:33:52:5d:45:b5:36:de:47:4e:15:cc:56	CN=DO NOT TRUST - AMI Test PK	CN=DO NOT TRUST - AMI Test PK	72
1a:a9:c7:61:c8:6a:be:88:4d:85:f5:ad:2b:95:3b:f1	CN=DO NOT TRUST - AMI Test PK	CN=DO NOT TRUST - AMI Test PK	57
1b:ed:93:e2:59:4e:2b:60:be:6b:1f:01:c9:af:a6:37	CN=DO NOT TRUST - AMI Test PK	CN=DO NOT TRUST - AMI Test PK	55
53:ea:33:87:af:a2:01:71:be:ff:55:16:96:91:0c:a4	CN=DO NOT TRUST - Test PK	CN=DO NOT TRUST - Test PK	23
	C=TW,ST=Taiwan,L=Taipei,O=Phoenix Technologies	C=TW,ST=Taiwan,L=Taipei,O=Phoenix Technologies	

Impact of PKFail on the UEFI Ecosystem



Impact of PKFail on the UEFI Ecosystem



Binarly's public disclosure of PKfail

Almost no firmware vulnerable to PKFail detected so far in 2025!

Distribution of PK across vendors

	2021	2022	2023	2024	2025	Total (Unique)
Acer	4	3	1	3	3	7
Dell	18	22	16	17	12	28
Fujitsu	5	7	8	6	8	9
Gigabyte	6	10	12	11	6	15
HP	3	3	3	3	3	3
HPE	2	2	2	2	1	2
Intel	5	10	5	1	1	10
Lenovo	37	106	120	92	94	178
Msi	4	5	5	3	3	5
Supermicro	3	3	3	1	1	4

PKfail PoC

https://www.youtube.com/watch?v=SPl7zfC-CmQ

PKfail PoC (Linux)

https://www.youtube.com/watch?v=CveWt3gFQTE

[2025] DBX Inconsistency

Another Secure Boot bypass

[2025] DBX Inconsistency

- dbx is a crucial component of Secure Boot: it contains what **must be considered untrusted**
- Single source of truth for the entire ecosystem: UEFI Forum
- In July 2024, Microsoft publishes the DBX2024 update, blocking modules related to CVE-2024-28924 (Secure Boot Bypass)
- This update wasn’t included in the UEFI Forum’s dbx, so it didn’t propagate to non-MS devices (e.g. LVFS)
- For around 6 months, a Secure Boot bypass has been publicly known but not included in non-MS dbx
- New source of truth: https://github.com/microsoft/secureboot_objects

https://www.binarly.io/blog/from-trust-to-trouble-the-supply-chain-implications-of-a-broken-dbx

AMD Microcode

Broken Signature Validation

[2025] EntrySign (CVE-2024-56161)

- Google researchers found an AMD microcode vulnerability that allows crafting valid microcode updates
- The microcode controls the low-level operations of the CPU:
  - Allows to override any CPU instruction (rdrand always returns 4)
  - Very difficult to detect, it basically infects the CPU
- Root cause: *“We noticed that the key from an old Zen 1 CPU was the example key of the NIST SP 800-38B publication and was reused until at least Zen 4 CPUs”.*

https://bughunters.google.com/blog/5424842357473280/zen-and-the-art-of-microcode-hacking

https://www.binarly.io/blog/binarly-tracking-updates-for-cve-2024-56161-a-high-risk-microcode-flaw-in-amd-cpus

Post Quantum Readiness

Device Security Implications

Post-Quantum Readiness

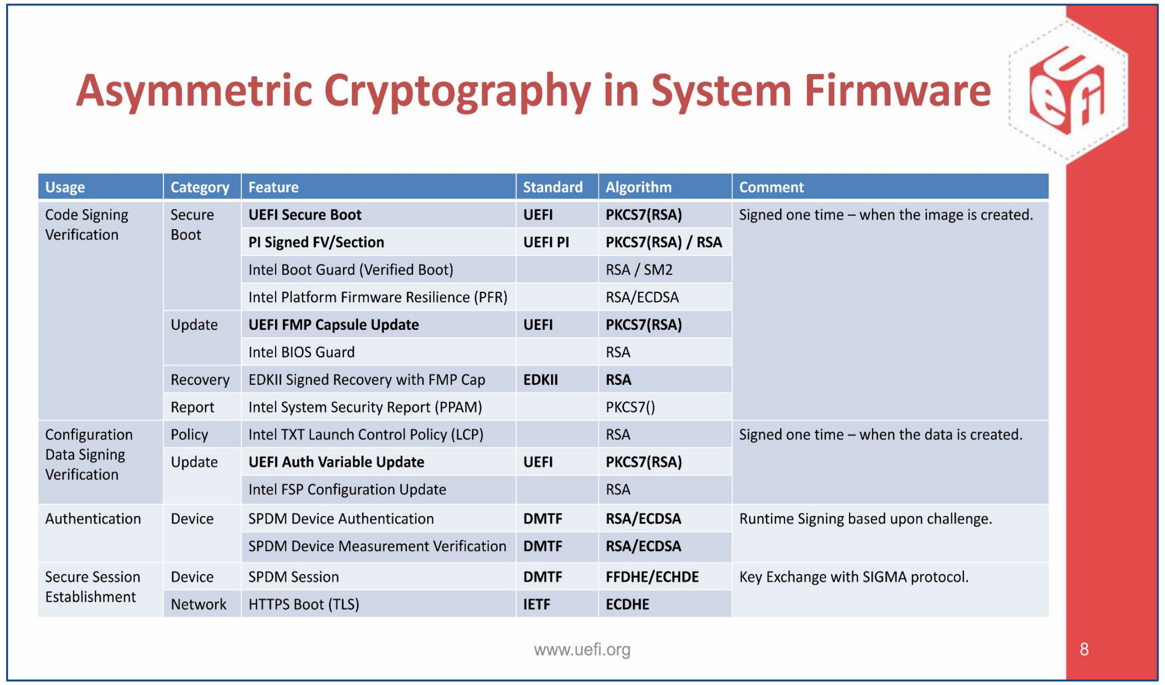
*“The migration will take time and will be more complex than people think. This is actually the driver. Even though **7–10 years sounds a long time** away, in reality the extent of the work needed might mean you are already too late.”*

Phil Venables, former CISO @ Google Cloud

https://www.philvenables.com/post/post-quantum-cryptography-migration-time-to-get-going

Post-Quantum Readiness in UEFI

- Ongoing discussion and few proof-of-concepts
- It will take years to update every component (huge complexity in firmware)

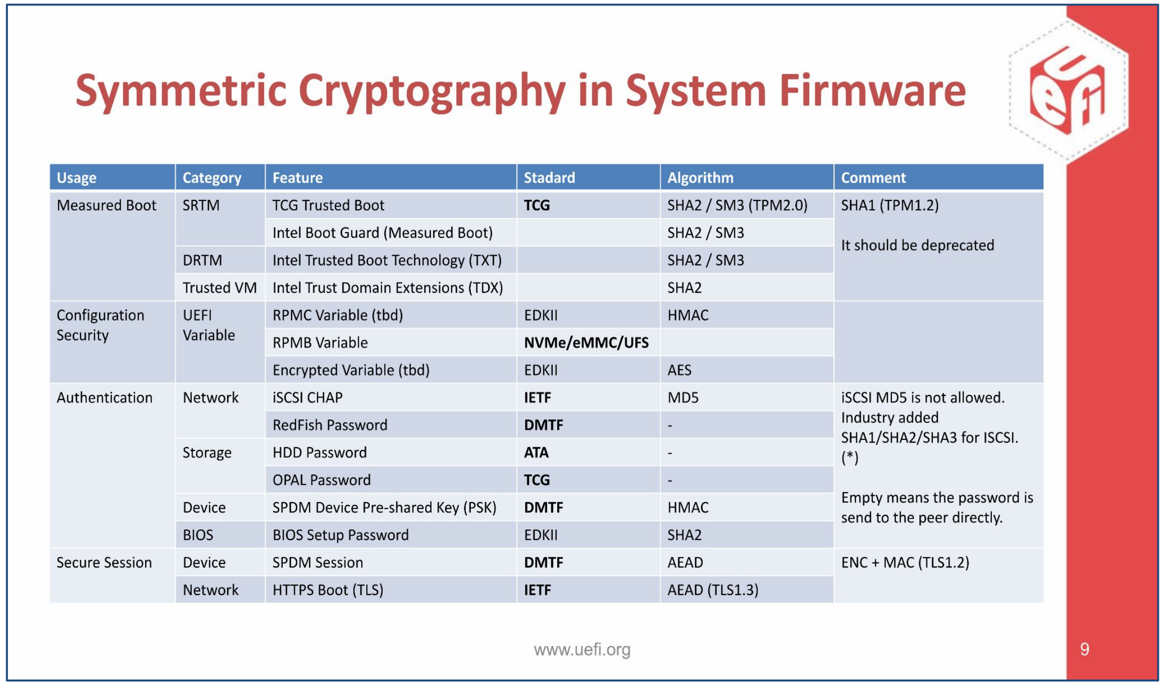


Asymmetric Cryptography in System Firmware

Usage	Category	Feature	Standard	Algorithm	Comment
Code Signing Verification	Secure Boot	**UEFI Secure Boot**	**UEFI**	**PKCS7(RSA)**	Signed one time – when the image is created.
		PI Signed FV/Section	**UEFI PI**	**PKCS7(RSA) / RSA**	
		Intel Boot Guard (Verified Boot)		RSA / SM2	
		Intel Platform Firmware Resilience (PFR)		RSA/ECDSA	
	Update	**UEFI FMP Capsule Update**	**UEFI**	**PKCS7(RSA)**	
		Intel BIOS Guard		RSA	
	Recovery	EDKII Signed Recovery with FMP Cap	**EDKII**	**RSA**	
	Report	Intel System Security Report (PPAM)		PKCS7()	
Configuration Data Signing Verification	Policy	Intel TXT Launch Control Policy (LCP)		RSA	Signed one time – when the data is created.
	Update	**UEFI Auth Variable Update**	**UEFI**	**PKCS7(RSA)**	
		Intel FSP Configuration Update		RSA	
Authentication	Device	SPDM Device Authentication	**DMTF**	**RSA/ECDSA**	Runtime Signing based upon challenge.
		SPDM Device Measurement Verification	**DMTF**	**RSA/ECDSA**	
Secure Session Establishment	Device	SPDM Session	**DMTF**	**FFDHE/ECHDE**	Key Exchange with SIGMA protocol.
	Network	HTTPS Boot (TLS)	**IETF**	**ECDHE**	

www.uefi.org

8



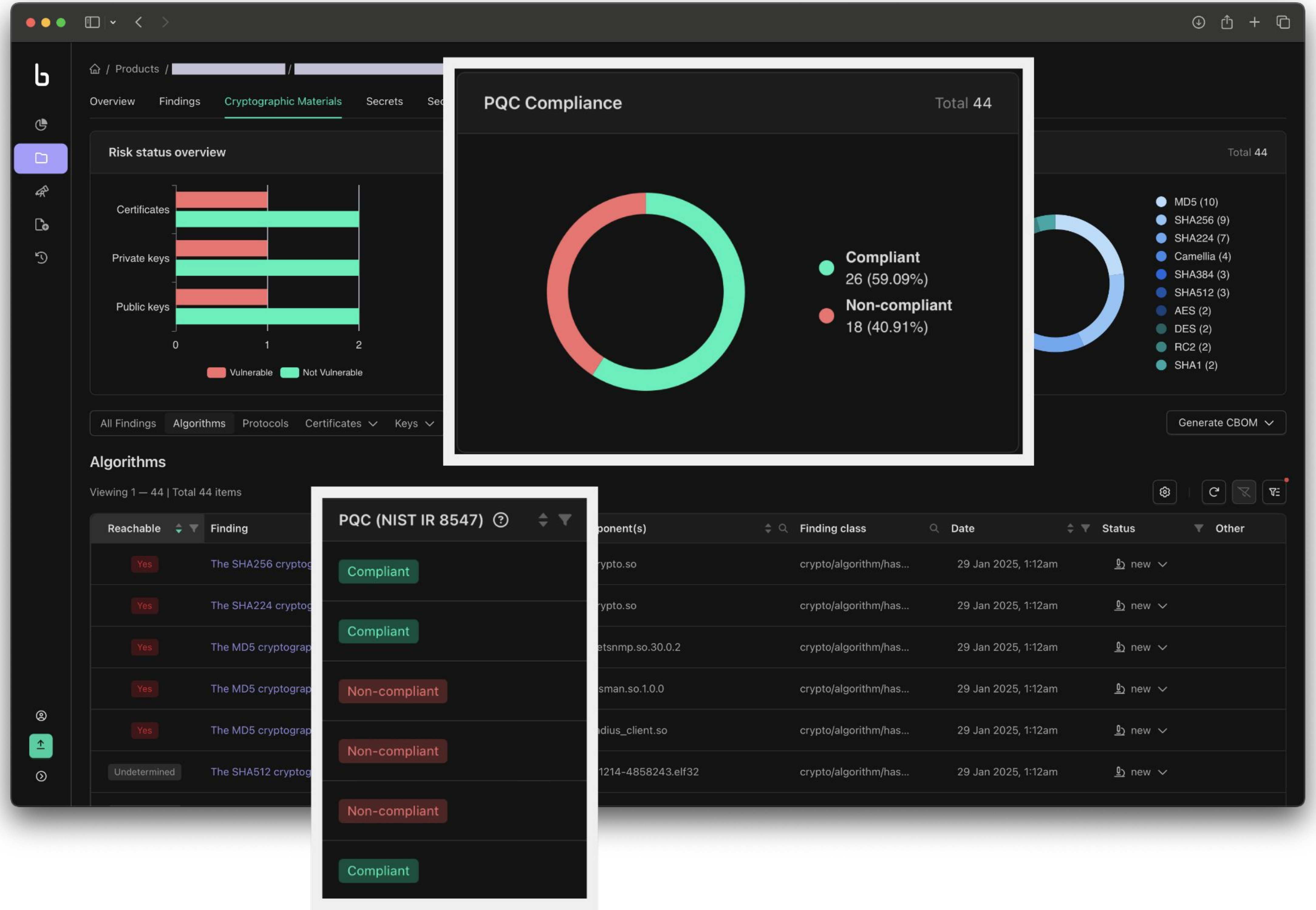
Symmetric Cryptography in System Firmware

Usage	Category	Feature	Stadard	Algorithm	Comment
Measured Boot	SRTM	TCG Trusted Boot	**TCG**	SHA2 / SM3 (TPM2.0)	SHA1 (TPM1.2) It should be deprecated
		Intel Boot Guard (Measured Boot)		SHA2 / SM3	
	DRTM	Intel Trusted Boot Technology (TXT)		SHA2 / SM3	
	Trusted VM	Intel Trust Domain Extensions (TDX)		SHA2	
Configuration Security	UEFI Variable	RPMC Variable (tbd)	EDKII	HMAC	
		RPMB Variable	**NVMe/eMMC/UFS**		
		Encrypted Variable (tbd)	EDKII	AES	
Authentication	Network	iSCSI CHAP	**IETF**	MD5	iSCSI MD5 is not allowed. Industry added SHA1/SHA2/SHA3 for ISCSI. (*) Empty means the password is send to the peer directly.
		RedFish Password	**DMTF**	-	
	Storage	HDD Password	**ATA**	-	
		OPAL Password	**TCG**	-	
	Device	SPDM Device Pre-shared Key (PSK)	**DMTF**	HMAC	
	BIOS	BIOS Setup Password	EDKII	SHA2	
Secure Session	Device	SPDM Session	**DMTF**	AEAD	ENC + MAC (TLS1.2)
	Network	HTTPS Boot (TLS)	**IETF**	AEAD (TLS1.3)	

www.uefi.org

9

The Impact of Post Quantum Cryptography on UEFI BIOS
UEFI 2021 Virtual Plugfest, Presented by: Jiewen Yao & Vincent Zimmer, Intel Corporation
https://uefi.org/events/impact-post-quantum-cryptography-uefi-bios



Algorithms

Viewing 1 — 44 | Total 44 items

Reachable	Finding	PQC (NIST IR 8547)	ponent(s)	Finding class	Date	Status	Other
Yes	The SHA256 cryptog	Compliant	ypto.so	crypto/algorithm/has...	29 Jan 2025, 1:12am	new	
Yes	The SHA224 cryptog	Compliant	ypto.so	crypto/algorithm/has...	29 Jan 2025, 1:12am	new	
Yes	The MD5 cryptograp	Non-compliant	etsnmp.so.30.0.2	crypto/algorithm/has...	29 Jan 2025, 1:12am	new	
Yes	The MD5 cryptograp	Non-compliant	sman.so.1.0.0	crypto/algorithm/has...	29 Jan 2025, 1:12am	new	
Yes	The MD5 cryptograp	Non-compliant	dius_client.so	crypto/algorithm/has...	29 Jan 2025, 1:12am	new	
Undetermined	The SHA512 cryptog	Compliant	1214-4858243.elf32	crypto/algorithm/has...	29 Jan 2025, 1:12am	new	

All this has happened before.

All this will happen again.

Summary / Call to Action

- UEFI firmware ecosystem has been affected by the leak of many private keys
- Intricate UEFI supply-chain exacerbates this problem:
  - Keys leaked from vendor A can be deployed on devices from vendor B
- Cryptographic key management in the ecosystem must improve:
  - Test keys intended for development end up in real devices
  - Private keys stored unencrypted or encrypted with weak and easily guessable passwords
  - Integrate HSM and cryptographic key management best practices

Questions?

References

1. https://www.binarly.io/blog/leaked-msi-source-code-with-intel-oem-keys-how-does-this-affect-industry-wide-software-supply-chain
2. https://www.binarly.io/blog/leaked-intel-boot-guard-keys-what-happened-how-does-it-affect-the-software-supply-chain
3. https://www.binarly.io/blog/clevo-boot-guard-keys-leaked-in-update-package
4. https://uefi.org/events/impact-post-quantum-cryptography-uefi-bios