

LABS CON

# PKfail: Supply-Chain Failures in Secure Boot Key Management

Alex Matrosov | Fabio Pagani



# Binarly REsearch Team — PKfail Edition



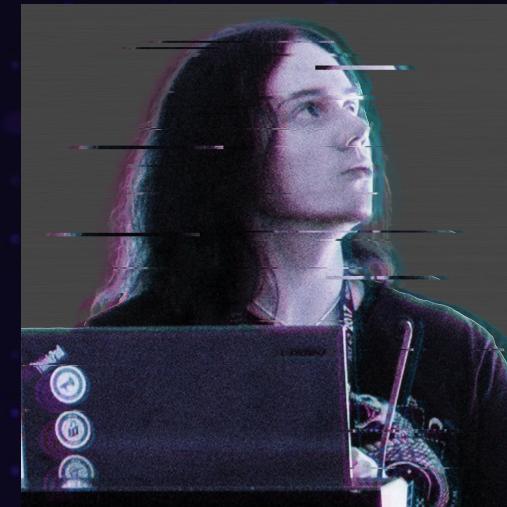
**Fabio Pagani**  
@pagabuc



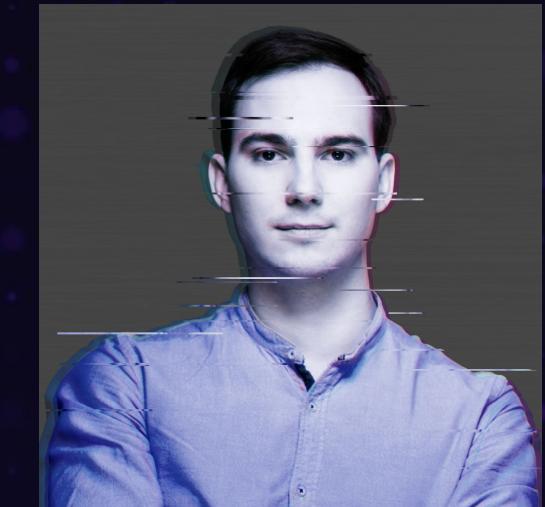
**Alex Matrosov**  
@matrosov



**Yegor Vasilenko**  
@yeggorv



**Sam Thomas**  
@xorpsc



**Anton Ivanov**  
@ant\_av7

# Agenda

- The Beginning
- Discovery and Disclosure
- PKfail: Exploitation and PoC Demo
- New Discovery: MicroFAIL
- Conclusion



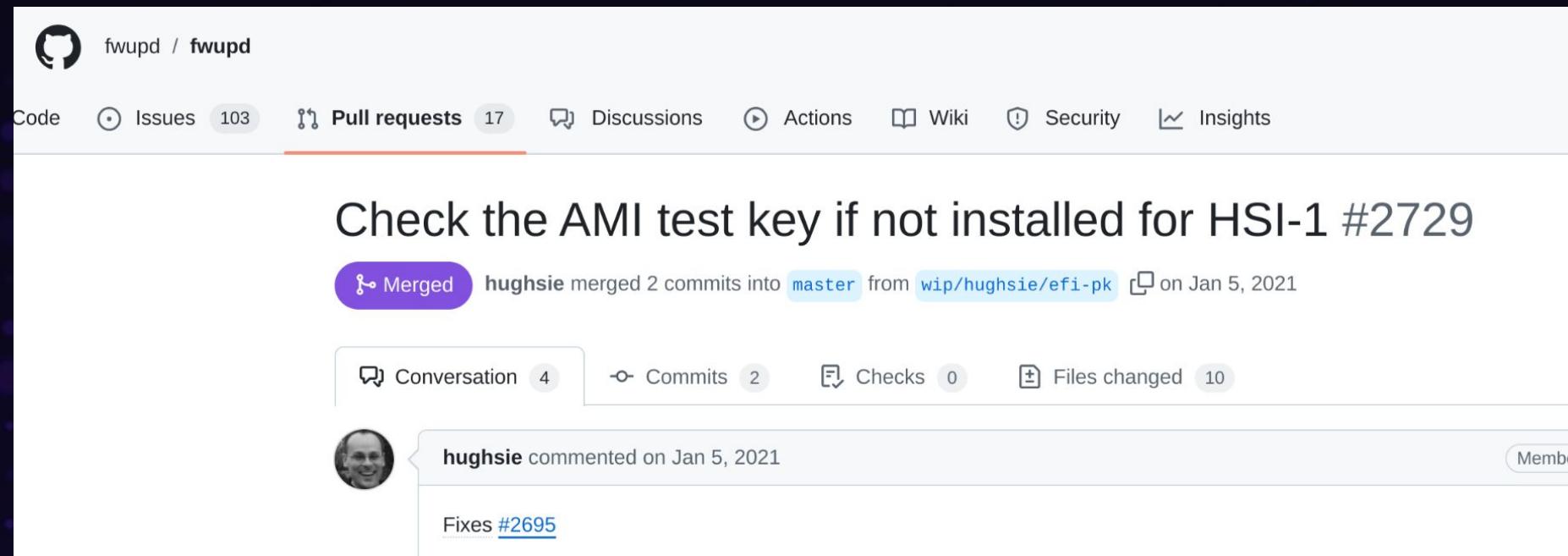
# PKfail: The Beginning

# Retrospective view on PKfail



# Turns out that ...

## This is an already known problem! o\_O



fwupd / fwupd

Code Issues Pull requests 17 Discussions Actions Wiki Security Insights

### Check the AMI test key if not installed for HSI-1 #2729

Merged hughsie merged 2 commits into master from wip/hughsie/efi-pk on Jan 5, 2021

Conversation 4 Commits 2 Checks 0 Files changed 10

hughsie commented on Jan 5, 2021

Fixes #2695

Lenovo SHOP SUPPORT COMMUNITY My Account English Cart

## CVE-2016-5247

Certain BIOS versions may include an AMI Test Key that could compromise Secure Boot protections

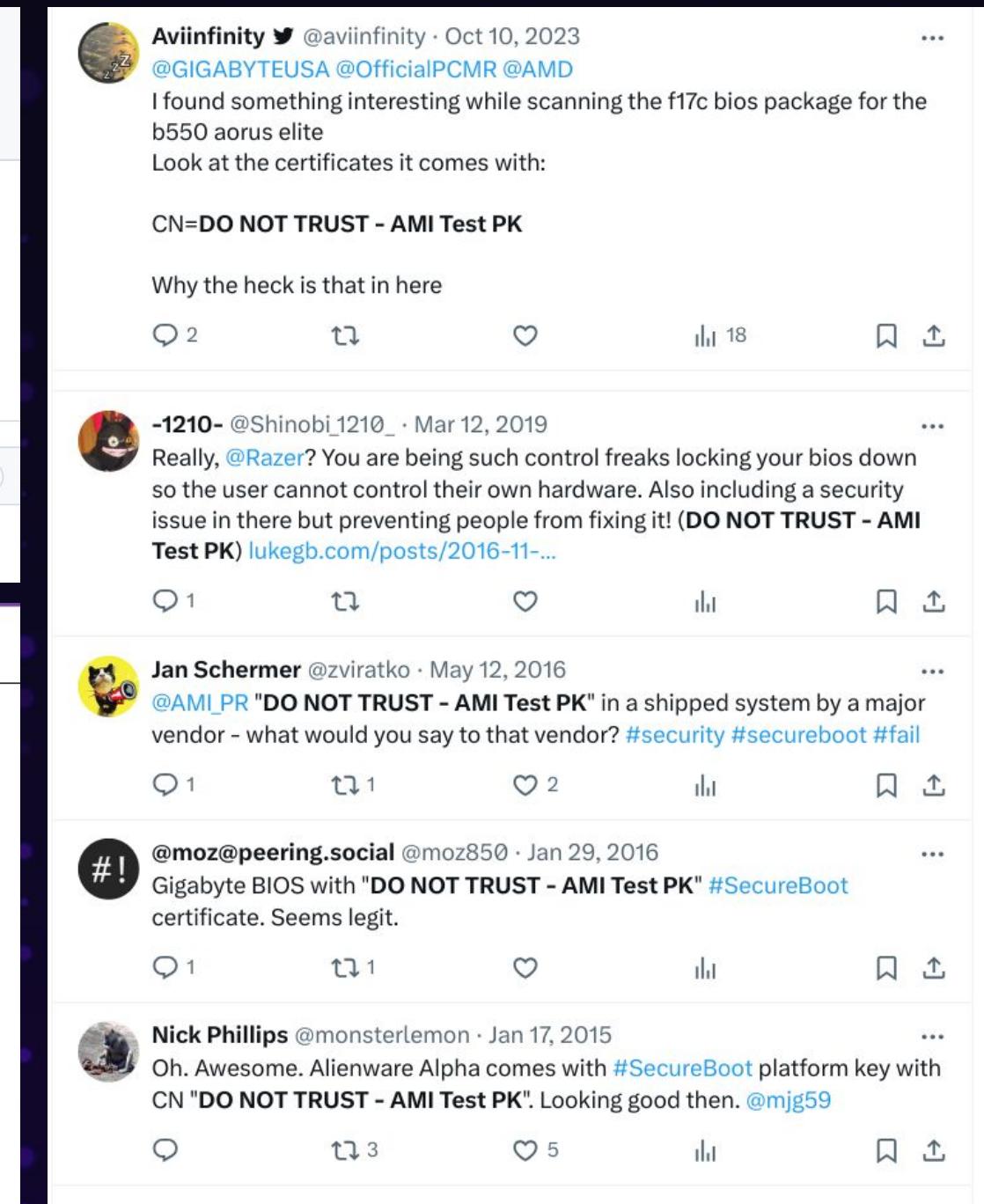
RSS

Lenovo Security Advisory: LEN-7806

Potential Impact: Secure boot may be compromised by an attacker with local access

Severity: High

Scope of Impact: Lenovo-specific



Aviinfinity @aviinfinity · Oct 10, 2023  
@GIGABYTEUSA @OfficialPCMR @AMD  
I found something interesting while scanning the f17c bios package for the b550 aorus elite  
Look at the certificates it comes with:

CN=DO NOT TRUST - AMI Test PK

Why the heck is that in here

-1210- @Shinobi\_1210\_ · Mar 12, 2019  
Really, @Razer? You are being such control freaks locking your bios down so the user cannot control their own hardware. Also including a security issue in there but preventing people from fixing it! (DO NOT TRUST - AMI Test PK) [lukegb.com/posts/2016-11-...](http://lukegb.com/posts/2016-11-...)

Jan Schermer @zviratko · May 12, 2016  
@AMI\_PR "DO NOT TRUST - AMI Test PK" in a shipped system by a major vendor - what would you say to that vendor? #security #secureboot #fail

#! @moz@peering.social @moz850 · Jan 29, 2016  
Gigabyte BIOS with "DO NOT TRUST - AMI Test PK" #SecureBoot certificate. Seems legit.

Nick Phillips @monsterlemon · Jan 17, 2015  
Oh. Awesome. Alienware Alpha comes with #SecureBoot platform key with CN "DO NOT TRUST - AMI Test PK". Looking good then. @mjt59

# Turns out that ...

## This is an already known problem! o\_O

fwupd / fwupd

Code Issues 103 Pull requests 17 Discussions Actions Wiki Security Insights

Check the AMI test key if not installed for HSI-1 #2729

Merged hughsie merged 2 commits into master from v1p/hughsie/efi-pk on Jan 5, 2021

Conversation 4 Commits 2 Checks 0 Files changed 10

hughsie commented on Jan 5, 2021 Fixes #2695

Lenovo SHOP SUPPORT COMMUNITY My Account Cart Support RSS

Certain BIOS versions may include an AMI Test Key that could compromise Secure Boot protections

Lenovo Security Advisory: LEN-7806

Potential Impact: Secure boot may be compromised by an attacker with local access

Severity: High

Scope of Impact: Lenovo-specific

Aviinfinity @aviinfinity · Oct 10, 2023  
@GIGABYTEUSA @OfficialPCMR @AMD  
I found something interesting while scanning the f17c bios package for the b550 aorus elite  
Look at the certificates it comes with:  
CN=DO NOT TRUST - AMI Test PK

Why the heck is that in here

-1210- @Shinobi\_1210\_ · Mar 12, 2019  
Really, @Razer? You are being such control freaks locking your bios down so the user can't control their own hardware. A major security issue is here. I present to people not fixing it. DO NOT TRUST AMI Test PK

Jan Schermer @zviratko · May 12, 2016  
@AMI\_PR "DO NOT TRUST - AMI Test PK" in a shipped system by a major vendor - what would you say to that vendor? #security #secureboot #fail

#! @moz@peering.social @moz850 · Jan 29, 2016  
Gigabyte BIOS with "DO NOT TRUST - AMI Test PK" #SecureBoot certificate. Seems legit.

Nick Phillips @monsterlemon · Jan 17, 2015  
Oh. Awesome. Alienware Alpha comes with #SecureBoot platform key with CN "DO NOT TRUST - AMI Test PK". Looking good then. @mig59

# So... we are good, RIGHT?

# Retrospective view on PKfail

Dataset with 80,000 UEFI firmware images:

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

Results:

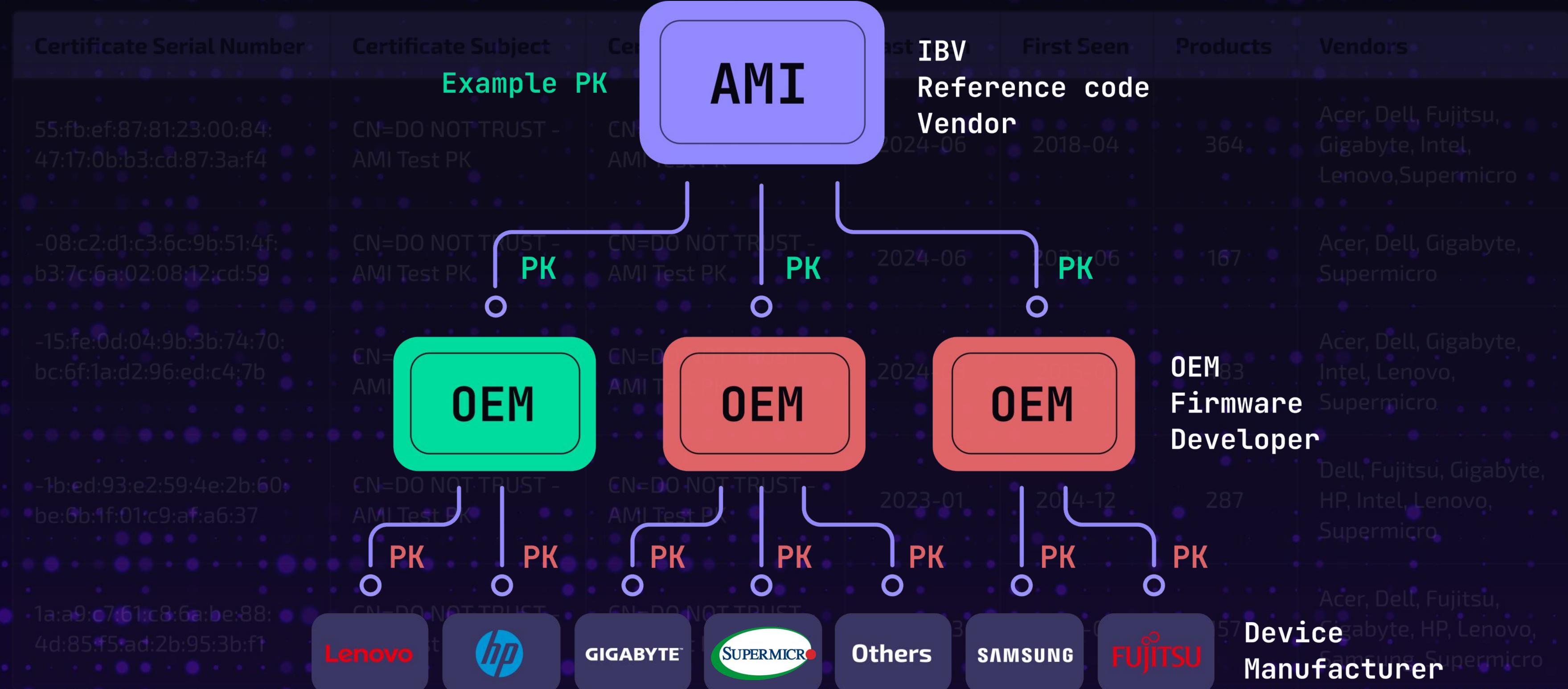
- 10% of images use non-production keys
- 8% of images when selecting images released in the past 4 years
- 22 unique non-production keys identified



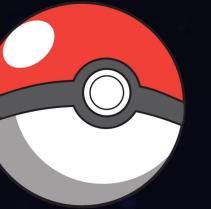
# Retrospective view on PKfail

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

# Retrospective view on PKfail



# A leaked PK appears



Multiple leaks – either by hacking  
or by “accident” – affected UEFI ecosystem

```
pagabuc@trin/tmp/Ryzen2000_4000/Keys/FW/AmiTest$ openssl pkcs12 -in FW_priKey.pfx -nodes -out PK.key  
Enter Import Password:
```

```
pagabuc@trin/tmp/Ryzen2000_4000/Keys/FW/AmiTest$ cat AmiTestKey.sdl
```

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

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



# A leaked PK appears



Multiple leaks – either by hacking  
or by “accident” – affected UEFI ecosystem

**This key has been included in firmware  
released between 2018 and now.**

- 01-18-2023: The [Ryzen2000\\_4000](#) repo is created on GitHub
- 04-14-2023: Repository is uploaded on the Internet Archive
- 05-??-2023: The [Ryzen2000\\_4000](#) repo is deleted by the owner
- 06-06-2023: All remaining forks on GitHub are DMCA'd by AMI

# PKfail: Discovery and Disclosure

# The discovery of PKfail

Earlier this year, we were adding support to our platform for reporting outdated **Forbidden Secure Boot database\*** when...

```
Certificate:  
  Data:  
    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:  
          02:06:41:f9:17:1e:52:aa:99:1a:b4:8a:5a:56:ee:  
          5b:ef:77:59:07:10:6e:91:6f:f7:91:61:4d:fa:30:  
          f5:67:49:f5:80:ad:75:54:0d:a4:dc:68:ad:e1:63:  
          8a:1f:59:23:b0:9e:f9:19:f6:a0:e8:7d:3b:c1:d9:  
          b1:1f:d6:95:96:12:b2:08:fd:20:75:ba:7d:22:81:
```

\* Stay tuned for an upcoming blogpost on this topic!



Certificate:  
Data:  
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 CMT  
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:  
02:06:41:f9:17:1e:52:aa:99:1a:b4:8a:5a:56:ee:  
5b:ef:77:59:07:10:6e:91:6f:f7:91:61:4d:fa:30:  
f5:67:49:f5:80:ad:75:54:0d:a4:dc:68:ad:e1:63:  
8a:1f:59:23:b0:9e:f9:19:f6:a0:e8:7d:3b:c1:d9:  
b1:1f:d6:95:06:12:b2:08:fd:20:75:b0:7d:22:84:

# Disclosure

- **2024-04-17:** Notified CERT/CC with complete advisory
- **2024-07-24:** Public disclosure

Insecure Platform Key (PK) used in UEFI system  
firmware signature

**Vulnerability Note VU#455367**

Original Release Date: 2024-08-30 | Last Revised: 2024-08-30



<https://kb.cert.org/vuls/id/455367>

# PK.FAIL Data Points

10,095 unique firmware images

8.5% vulnerable rate



- Users uploaded 10,095 firmware images
- Found untrusted key in 791 images (8.5%)
- Bulk of the submissions in the week after the disclosure
- Still getting on average 25 submissions per day

# A closer look at the submissions

10,095 unique firmware images

8.5% vulnerable rate



791

Untrusted PK

9,304

Safe

- Detected keys **match** results from our original research
- **Four** unseen keys (3 from AMI, 1 from Supermicro)
- The **most common** key is the one that **leaked** on GitHub 🔥
- Keys found on **desktops, laptops, servers** but also **gaming consoles, ATMs, POS terminals, voting machines**

# A closer look at the submissions

We might have **underestimated** the **impact** for other IBVs:

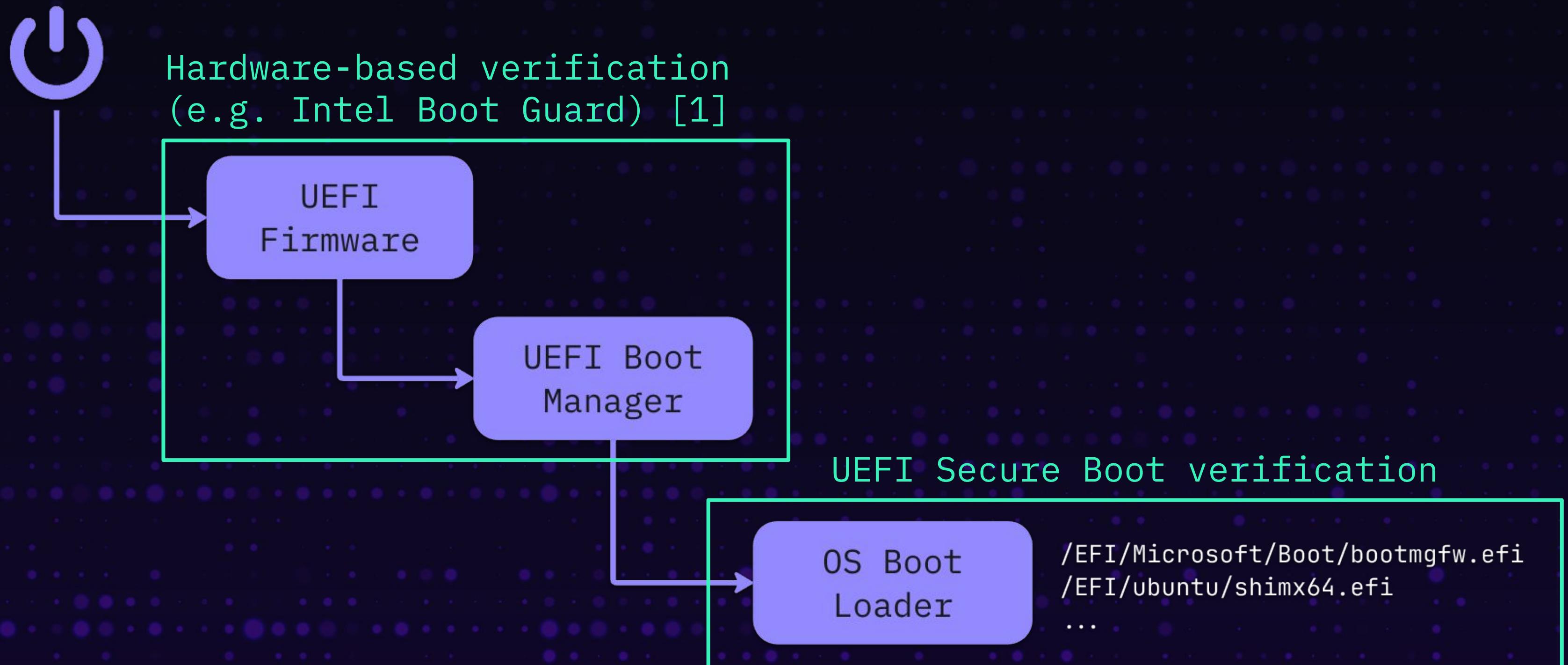
- We received 61 images with non-production key generated by Insyde
- Firmware for devices **currently** on the market

```
Certificate:  
Data:  
Version: 3 (0x2)  
Serial Number:  
32:11:5d:28:e8:84:80:af:43:d5:02:fd:99:eb:bb:4b  
Signature Algorithm: sha256WithRSAEncryption  
Issuer: CN= [REDACTED] UEFI CA 2024, OU= [REDACTED], O= [REDACTED], L=San Jose, ST=California, C=US  
Validity  
Not Before: Jun 11 05:21:42 2024 GMT  
Not After : Jun 11 05:21:41 2054 GMT  
Subject: CN= [REDACTED] UEFI CA 2024, OU= [REDACTED], O= [REDACTED], L=San Jose, ST=California, C=US
```



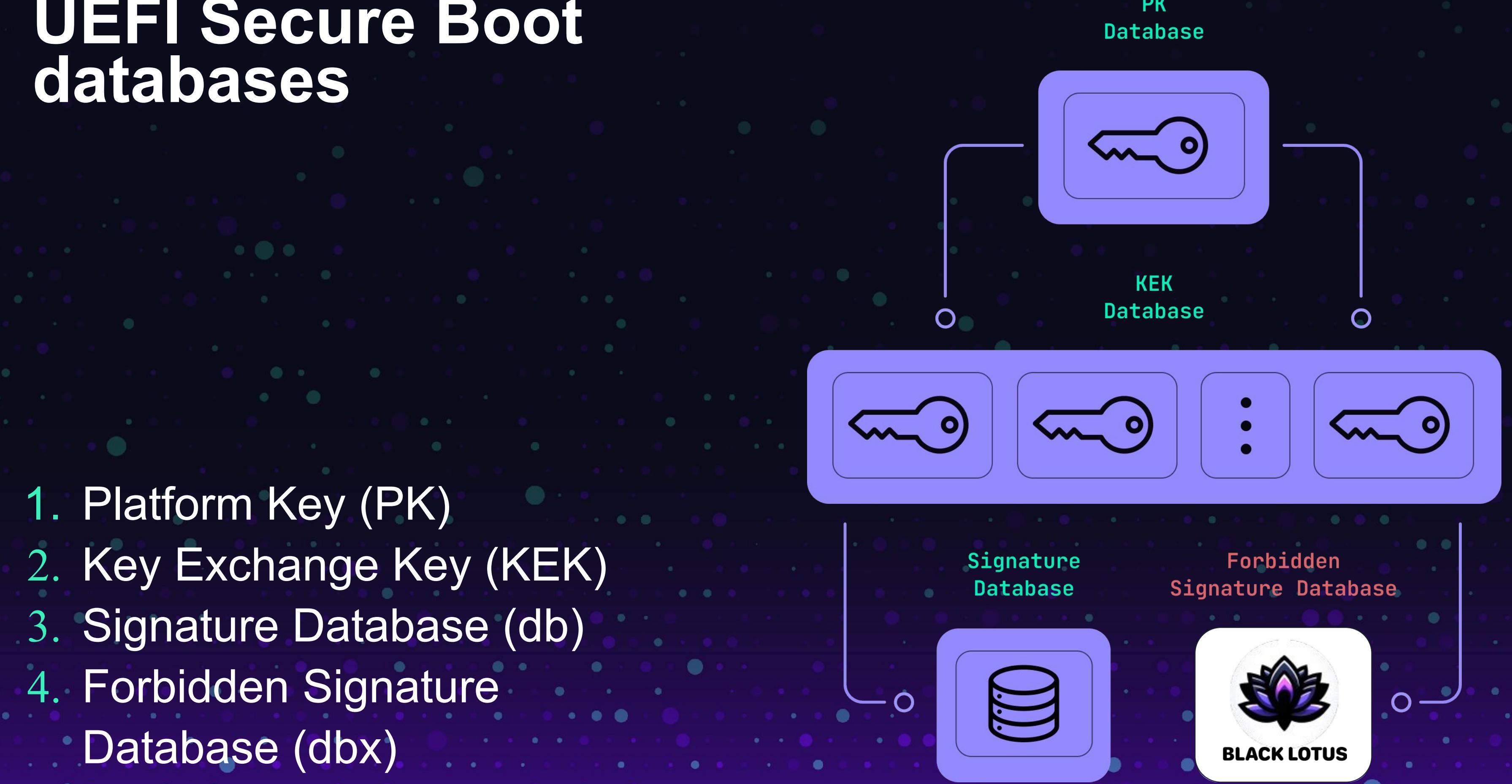
# PKfail: Exploitation and PoC Demo

# UEFI (Secure) Boot process



[1] Leaked Intel Boot Guard keys: What happened? How does it affect the software supply chain? Binarly, 2022

# UEFI Secure Boot databases



# Developing a PoC



# Developing a PoC



**Proof of Concept for PKfail**

<https://www.youtube.com/watch?v=SP17zfc-CmQ>



**Proof of Concept for PKfail (Linux version)**

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

binarly

# Proof of Concept for PKfail

# New Discovery: Supermicro BMC Test Key

# The problem is bigger than we thought...

Certificate:

Data:

Version: 3 (0x2)

Serial Number:

1a:da:e6:cf:23:66:6a:36:d9:dd:69:4c:2f:ba:30:14:90:f7:3d:5e

Signature Algorithm: sha512WithRSAEncryption

Issuer: C = US, ST = CA, L = SanJose, O = Super Micro Computer Inc., CN = RD1 BMC Test Key – DO NOT TRUST

Validity

Not Before: Feb 14 03:14:28 2020 GMT

Not After : Feb 1 03:14:28 2070 GMT

Subject: C = US, ST = CA, L = SanJose, O = Super Micro Computer Inc., CN = RD1 BMC Test Key – DO NOT TRUST

Subject Public Key Info:

Public Key Algorithm: rsaEncryption

Public-Key: (4096 bit)

Modulus:

00:c6:b3:42:c9:36:c3:a1:24:0c:ec:e5:1a:31:96:

5b:1d:a6:c7:85:66:50:bf:59:78:9c:2d:8d:07:5e:

6f:9b:f0:a0:70:7a:42:f0:0a:68:bd:e1:aa:80:ef:

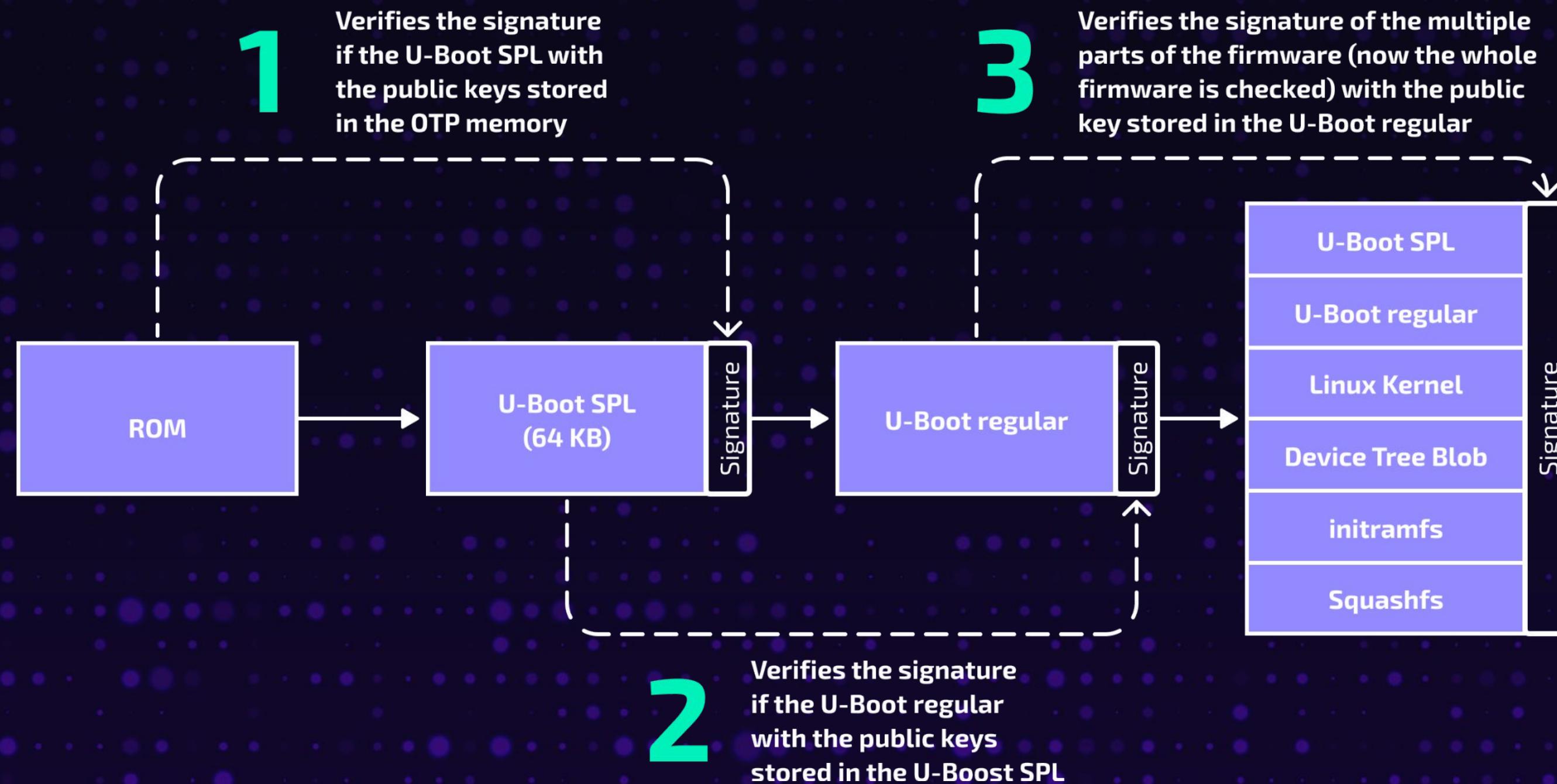
2c:70:bd:7a:36:59:6a:ca:2a:1d:21:f1:1c:a1:31:

f3:d6:3d:2c:ea:32:0f:d6:62:99:01:57:99:e1:13:

fd:82:1a:78:c7:29:2b:4c:2d:70:43:b0:c5:28:94:

# Supermicro BMC Test Key

## Aspeed Root of Trust chain



# Supermicro Response

```
Production key
Issuer: O = "Super Micro Computer Inc. (ENG=Engineering; HSM=HSM; SB=SecureBoot)", OU = ENG, CN = R12FWSigningKey4K
Validity
| Not Before: Dec 14 01:24:22 2022 GMT
| Not After : Dec 14 01:34:20 2037 GMT

Test key
Issuer: C = US, ST = CA, L = SanJose, O = Super Micro Computer Inc., CN = RD1 BMC Test Key - DO NOT TRUST
Validity
| Not Before: Feb 14 03:14:28 2020 GMT
| Not After : Feb 1 03:14:28 2070 GMT
```

Supermicro rejected the issue:

- Test key hasn't been leaked – **agree**
- Test key is NIST compliant – **agree**
- Additional code exists in ROM, that checks the whole image with only production keys – **possible**
- Test key has the same access control as production key – **disagree**
- Test key has the same security level as production key – **disagree**

<https://www.binarly.io/blog/repeatable-failures-test-keys-used-to-sign-production-software-again>

# Conclusion

- Cryptographic keys are widely reused
- Cryptographic materials not correctly stored
- Accidentally, keys are leaked sometimes
- The entire industry is impacted
- Lessons learned and will never be repeated, right?



LABS CON

# Thank you



<https://binarly.io/pkfail>



