presented by

Microsoft

Microsoft Security Features and Firmware Configurations

Spring 2018 UEFI Seminar and Plugfest
March 26-30, 2018
Presented by Scott Anderson and Jeremiah Cox
Facilitated by Michael Anderson

- Contains pre-release information shared under UEFI Forum NDA
- Please do not record or share this presentation outside of UEFI Forum

- VBS & HVCI everywhere
- UEFI CA and Baby Duck
- Device Firmware Configuration Interface (DFCI)

Virtualization Based Security & Hypervisor Protected Code Integrity (VBS & HVCI)

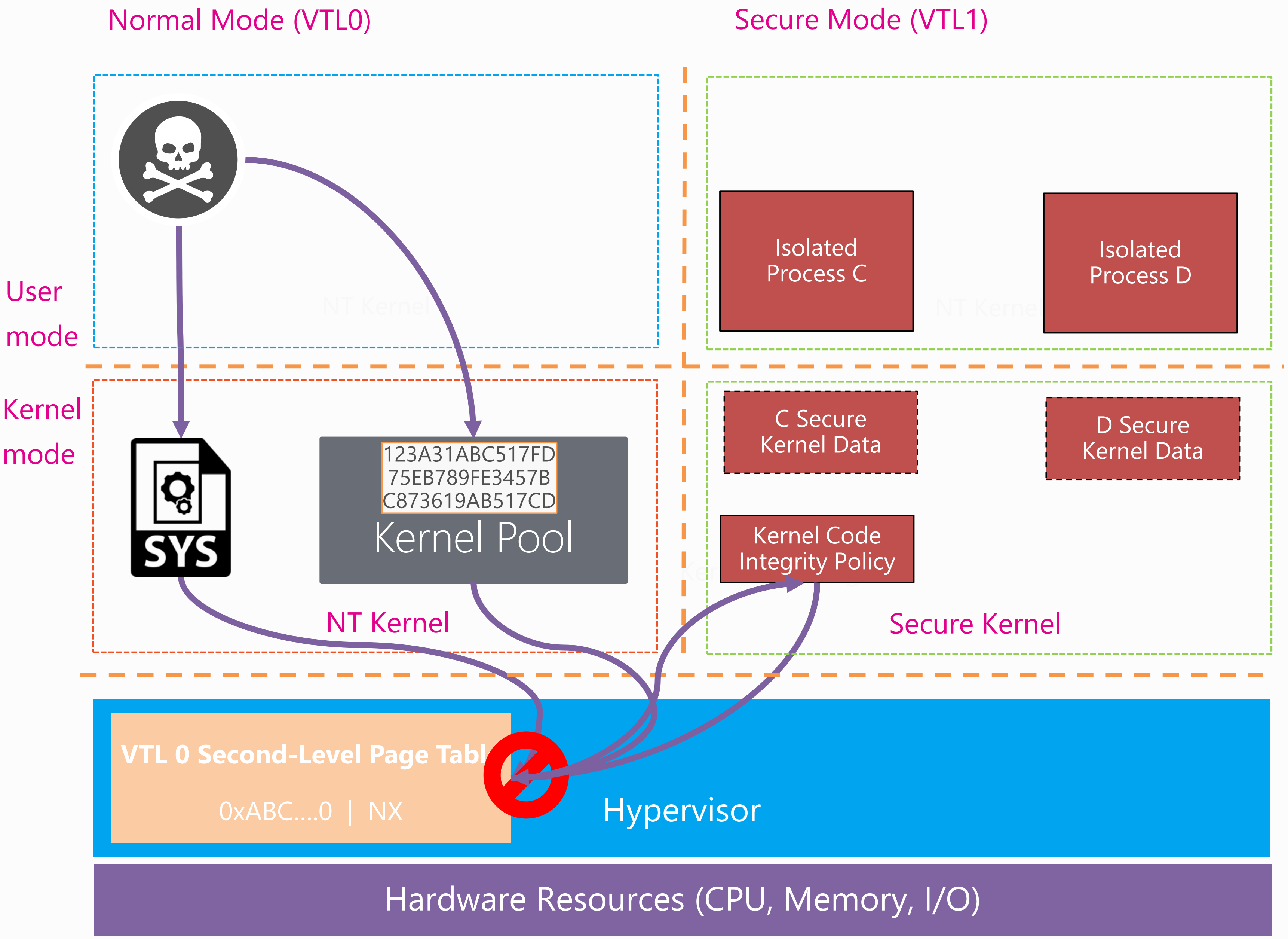
What are these technologies?

Virtualization-based security (VBS) is the 'foundation' for many features we have shipped since the start of Windows 10. VBS uses the hypervisor to create this virtual secure mode, and to enforce restrictions which protect vital system and operating system resources, or to protect security assets such as authenticated user credentials

Hypervisor-protected code integrity (HVCI) has also been shipping since the start of Windows 10, but previously focused on Enterprise customers only and we are bringing it now across all SKU's. It introduces requirements for *kernel* mode drivers only and HVCI compatibility has been a requirement for all new driver submissions since RS1

Memory Integrity Protection

leverages virtualization page tables managed by VTL1 to eliminate W^X memory in VTL0 kernel-mode



SLAT is used to gate enforce RX only

HVCI running in SK validates code pages
If valid set GPA bits to
R=1 W=0 KMX=UMX=1

Mode-Based Execute (MBE) Control

Extended-Extended Page Tables (EPT)

- XU for user pages
- XS for supervisor pages
- KMX and UMX hardware bits.

Kernel Control Flow Integrity

Kernel CFG is used to enforce runtime code flow integrity for kernel drivers

Compile time

```
void Foo(...) {
   // SomeFunc is address-taken
   // and may be called indirectly
   Object->FuncPtr = SomeFunc;
}
```

Metadata is automatically added to the image which identifies functions that may be called indirectly

```
void Bar(...) {
   // Compiler-inserted check to
   // verify call target is valid
   _guard_check_icall(Object->FuncPtr);
   Object->FuncPtr(xyz);
}
```

A lightweight check is inserted prior to indirect calls which will verify that the call target is valid at runtime

Kernel Runtime

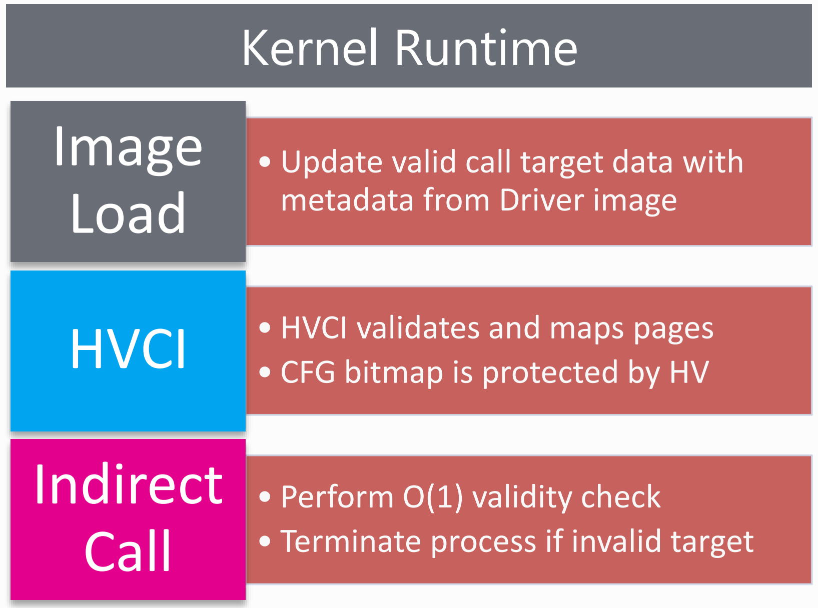
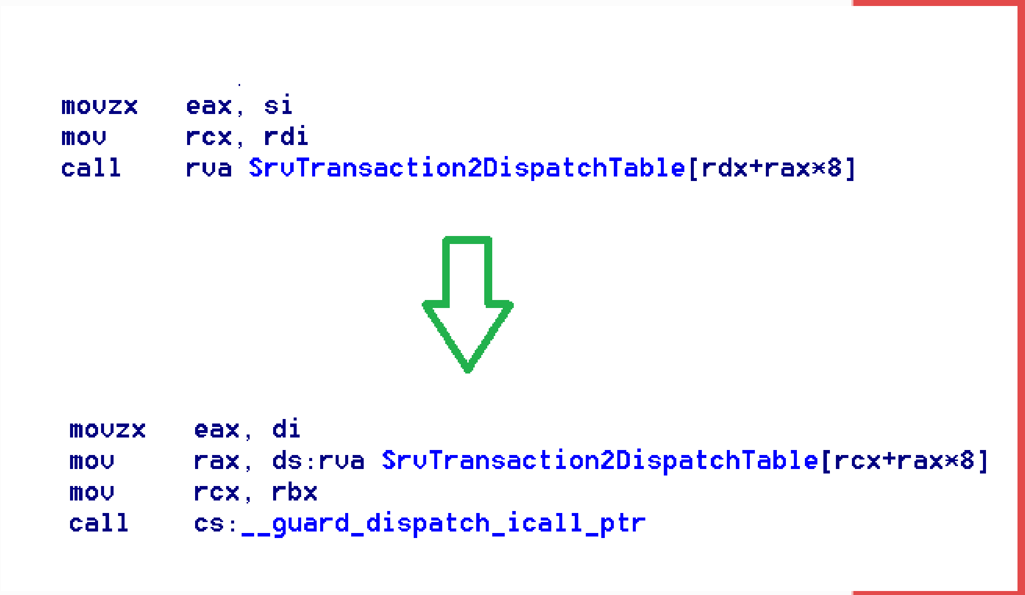


Image Load
- Update valid call target data with metadata from Driver image

HVCI
- HVCI validates and maps pages
- CFG bitmap is protected by HV

Indirect Call
- Perform O(1) validity check
- Terminate process if invalid target



```
movzx   eax, si
mov     rcx, rdi
call    rva SrvTransaction2DispatchTable[rdx+rax*8]
```

```
movzx   eax, di
mov     rax, ds:rva SrvTransaction2DispatchTable[rcx+rax*8]
mov     rcx, rbx
call    cs:__guard_dispatch_icall_ptr
```

- Kernel Control Flow Guard improves protection against control flow hijacking for kernel code
- Paired with HVCI to ensure both code integrity and control flow integrity

How can you as a partner validate?

Hardware:

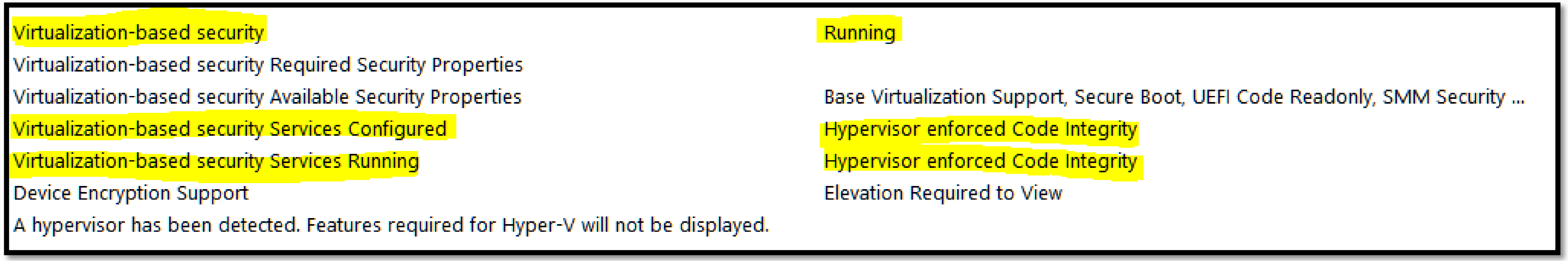
a) Make sure the device you're testing with meets the hardware and firmware configuration requirements for VBS and HVCI: https://docs.microsoft.com/en-us/windows-hardware/design/device-experiences/oem-vbs

b) Specifically on silicon type, this is from Intel's 7th generation, Kabylake or later

c) Example PCs would be:

- » Surface Laptop, Book 2, Surface Pro 2017 and newer
- » Commercial Laptops: Dell XPS Ultrabooks, Dell Latitudes, HP Elitebooks 1000 series, Lenovo Carbon X1's

1. **Use the Device Guard and Credential Guard hardware readiness tool to enable**
2. Once complete, to validate that VBS and HVCI are enabled, check by going to Start → Run → MSINFO32 and you will see the following come up:



```
Virtualization-based security                                       Running
Virtualization-based security Required Security Properties
Virtualization-based security Available Security Properties         Base Virtualization Support, Secure Boot, UEFI Code Readonly, SMM Security ...
Virtualization-based security Services Configured                   Hypervisor enforced Code Integrity
Virtualization-based security Services Running                      Hypervisor enforced Code Integrity
Device Encryption Support                                           Elevation Required to View
A hypervisor has been detected. Features required for Hyper-V will not be displayed.
```

Links for more details

HVCI Driver Compatibility:

https://blogs.msdn.microsoft.com/windows_hardware_certification/2015/05/22/driver-compatibility-with-device-guard-in-windows-10/

HVCI Compatibility Test:

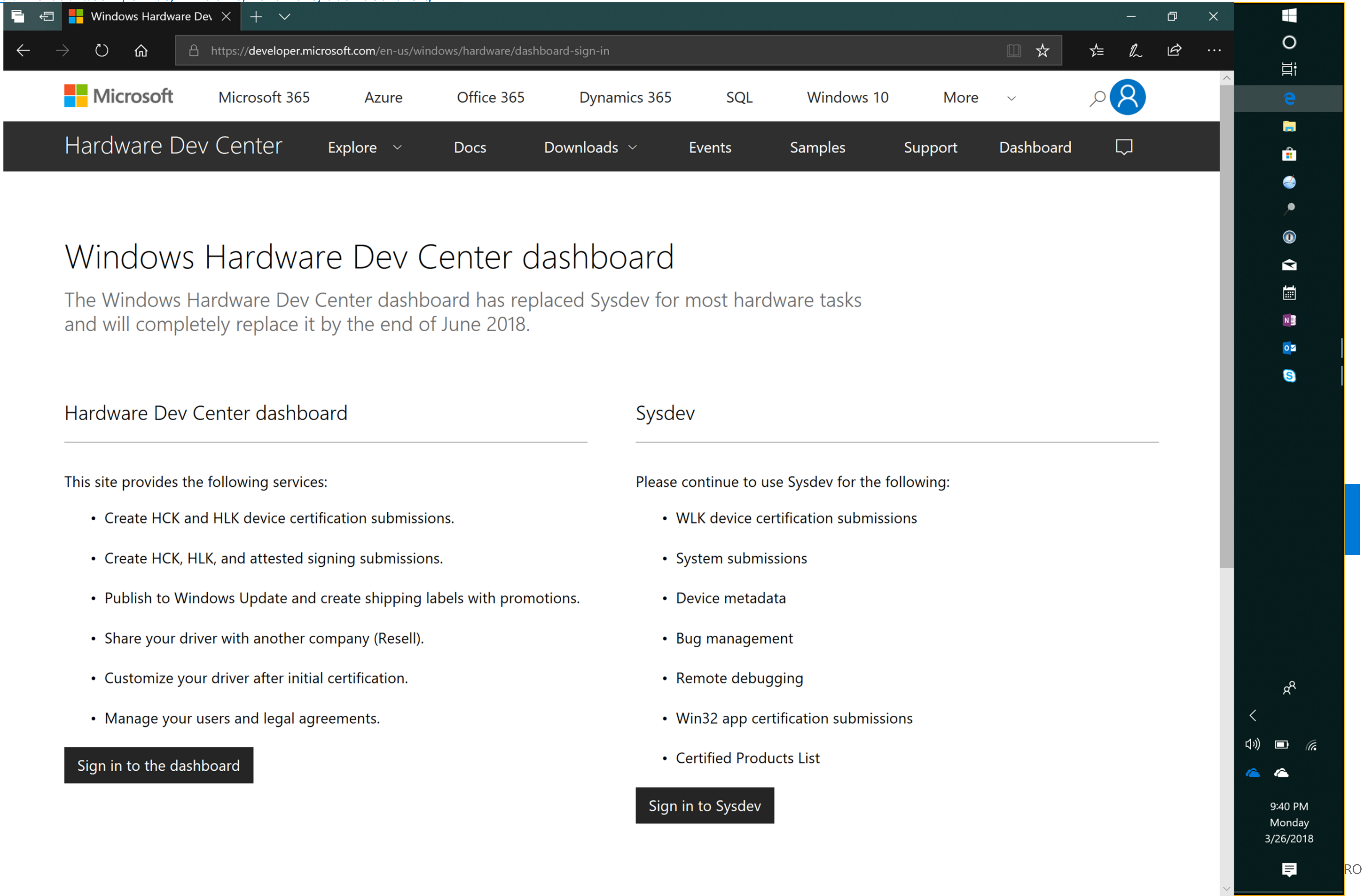
https://docs.microsoft.com/en-us/windows-hardware/test/hlk/testref/b972fc52-2468-4462-9799-6a1898808c86

Device Guard and Credential Guard hardware readiness tool:

https://www.microsoft.com/en-us/download/details.aspx?id=53337

UEFI submission change to Dev Center

https://developer.microsoft.com/en-us/windows/hardware/dashboard-sign-in



Windows Hardware Dev Center dashboard

The Windows Hardware Dev Center dashboard has replaced Sysdev for most hardware tasks and will completely replace it by the end of June 2018.

Hardware Dev Center dashboard

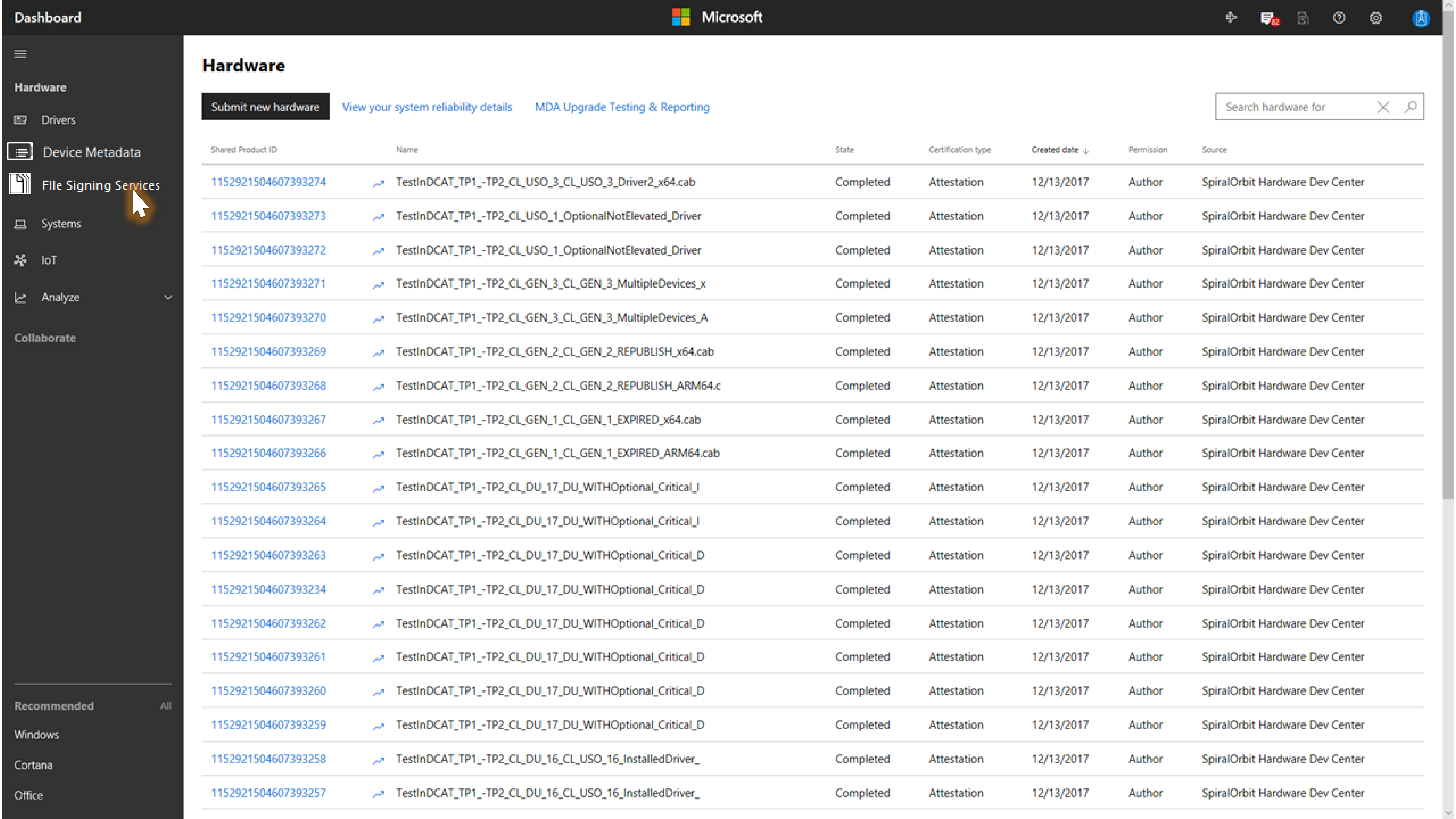
This site provides the following services:

- Create HCK and HLK device certification submissions.
- Create HCK, HLK, and attested signing submissions.
- Publish to Windows Update and create shipping labels with promotions.
- Share your driver with another company (Resell).
- Customize your driver after initial certification.
- Manage your users and legal agreements.

Sysdev

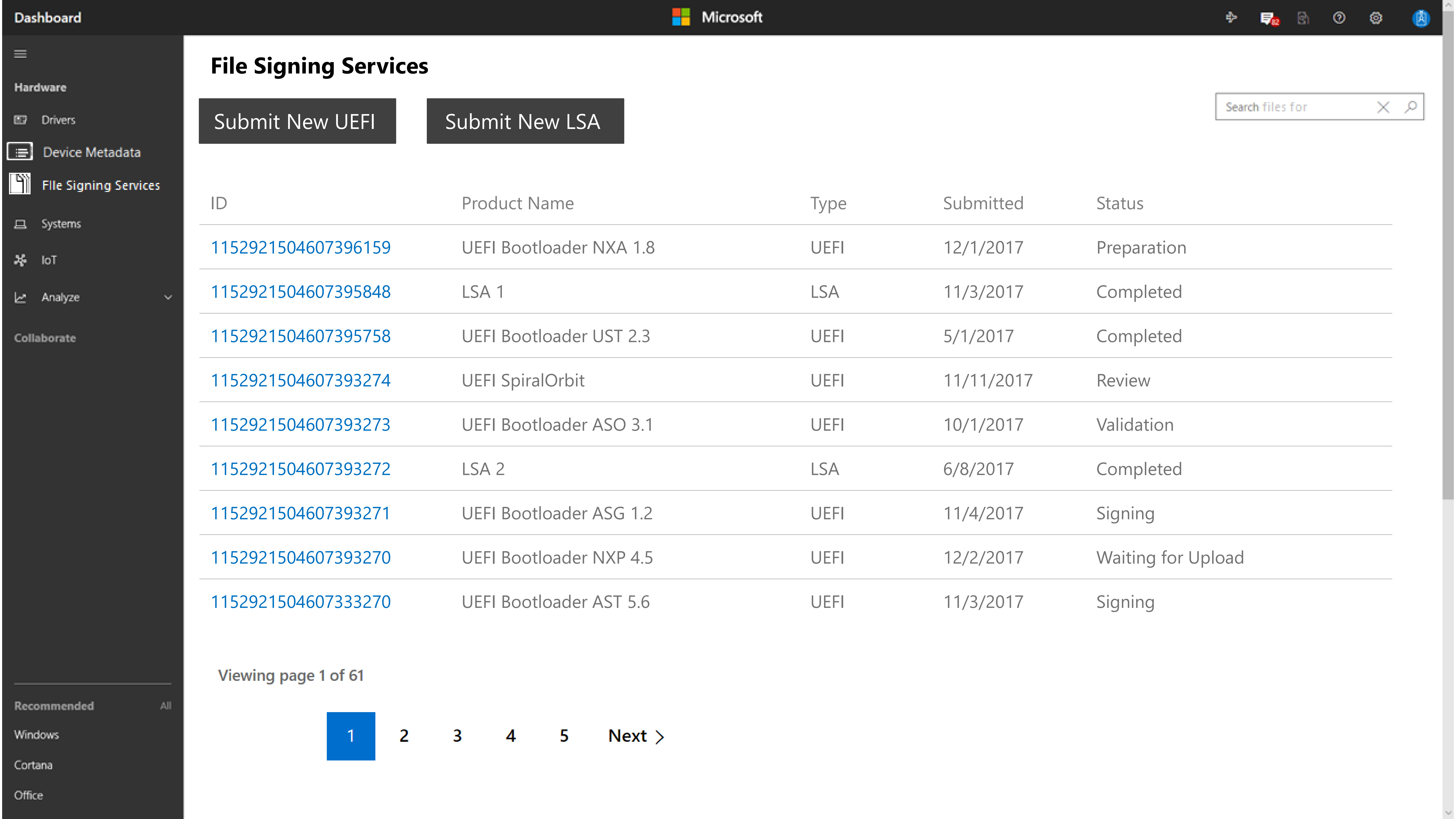
Please continue to use Sysdev for the following:

- WLK device certification submissions
- System submissions
- Device metadata
- Bug management
- Remote debugging
- Win32 app certification submissions
- Certified Products List



Hardware

Shared Product ID		Name	State	Certification type	Created date ↓	Permission	Source
1152921504607393274	↗	TestInDCAT_TP1_-TP2_CL_USO_3_CL_USO_3_Driver2_x64.cab	Completed	Attestation	12/13/2017	Author	SpiralOrbit Hardware Dev Center
1152921504607393273	↗	TestInDCAT_TP1_-TP2_CL_USO_1_OptionalNotElevated_Driver	Completed	Attestation	12/13/2017	Author	SpiralOrbit Hardware Dev Center
1152921504607393272	↗	TestInDCAT_TP1_-TP2_CL_USO_1_OptionalNotElevated_Driver	Completed	Attestation	12/13/2017	Author	SpiralOrbit Hardware Dev Center
1152921504607393271	↗	TestInDCAT_TP1_-TP2_CL_GEN_3_CL_GEN_3_MultipleDevices_x	Completed	Attestation	12/13/2017	Author	SpiralOrbit Hardware Dev Center
1152921504607393270	↗	TestInDCAT_TP1_-TP2_CL_GEN_3_CL_GEN_3_MultipleDevices_A	Completed	Attestation	12/13/2017	Author	SpiralOrbit Hardware Dev Center
1152921504607393269	↗	TestInDCAT_TP1_-TP2_CL_GEN_2_CL_GEN_2_REPUBLISH_x64.cab	Completed	Attestation	12/13/2017	Author	SpiralOrbit Hardware Dev Center
1152921504607393268	↗	TestInDCAT_TP1_-TP2_CL_GEN_2_CL_GEN_2_REPUBLISH_ARM64.c	Completed	Attestation	12/13/2017	Author	SpiralOrbit Hardware Dev Center
1152921504607393267	↗	TestInDCAT_TP1_-TP2_CL_GEN_1_CL_GEN_1_EXPIRED_x64.cab	Completed	Attestation	12/13/2017	Author	SpiralOrbit Hardware Dev Center
1152921504607393266	↗	TestInDCAT_TP1_-TP2_CL_GEN_1_CL_GEN_1_EXPIRED_ARM64.cab	Completed	Attestation	12/13/2017	Author	SpiralOrbit Hardware Dev Center
1152921504607393265	↗	TestInDCAT_TP1_-TP2_CL_DU_17_DU_WITHOptional_Critical_I	Completed	Attestation	12/13/2017	Author	SpiralOrbit Hardware Dev Center
1152921504607393264	↗	TestInDCAT_TP1_-TP2_CL_DU_17_DU_WITHOptional_Critical_I	Completed	Attestation	12/13/2017	Author	SpiralOrbit Hardware Dev Center
1152921504607393263	↗	TestInDCAT_TP1_-TP2_CL_DU_17_DU_WITHOptional_Critical_D	Completed	Attestation	12/13/2017	Author	SpiralOrbit Hardware Dev Center
1152921504607393234	↗	TestInDCAT_TP1_-TP2_CL_DU_17_DU_WITHOptional_Critical_D	Completed	Attestation	12/13/2017	Author	SpiralOrbit Hardware Dev Center
1152921504607393262	↗	TestInDCAT_TP1_-TP2_CL_DU_17_DU_WITHOptional_Critical_D	Completed	Attestation	12/13/2017	Author	SpiralOrbit Hardware Dev Center
1152921504607393261	↗	TestInDCAT_TP1_-TP2_CL_DU_17_DU_WITHOptional_Critical_D	Completed	Attestation	12/13/2017	Author	SpiralOrbit Hardware Dev Center
1152921504607393260	↗	TestInDCAT_TP1_-TP2_CL_DU_17_DU_WITHOptional_Critical_D	Completed	Attestation	12/13/2017	Author	SpiralOrbit Hardware Dev Center
1152921504607393259	↗	TestInDCAT_TP1_-TP2_CL_DU_17_DU_WITHOptional_Critical_D	Completed	Attestation	12/13/2017	Author	SpiralOrbit Hardware Dev Center
1152921504607393258	↗	TestInDCAT_TP1_-TP2_CL_DU_16_CL_USO_16_InstalledDriver_	Completed	Attestation	12/13/2017	Author	SpiralOrbit Hardware Dev Center
1152921504607393257	↗	TestInDCAT_TP1_-TP2_CL_DU_16_CL_USO_16_InstalledDriver_	Completed	Attestation	12/13/2017	Author	SpiralOrbit Hardware Dev Center

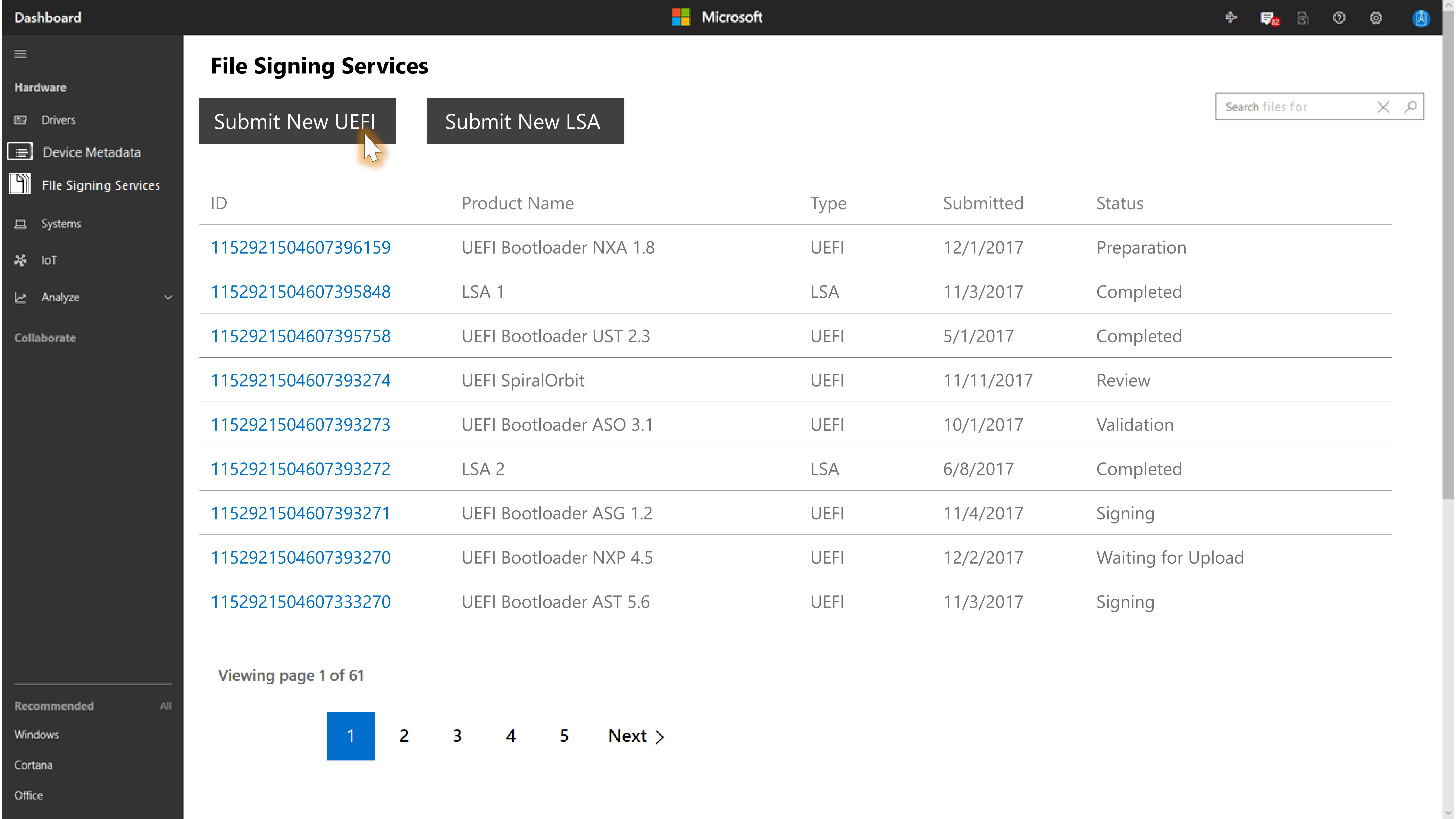


File Signing Services

ID	Product Name	Type	Submitted	Status
1152921504607396159	UEFI Bootloader NXA 1.8	UEFI	12/1/2017	Preparation
1152921504607395848	LSA 1	LSA	11/3/2017	Completed
1152921504607395758	UEFI Bootloader UST 2.3	UEFI	5/1/2017	Completed
1152921504607393274	UEFI SpiralOrbit	UEFI	11/11/2017	Review
1152921504607393273	UEFI Bootloader ASO 3.1	UEFI	10/1/2017	Validation
1152921504607393272	LSA 2	LSA	6/8/2017	Completed
1152921504607393271	UEFI Bootloader ASG 1.2	UEFI	11/4/2017	Signing
1152921504607393270	UEFI Bootloader NXP 4.5	UEFI	12/2/2017	Waiting for Upload
1152921504607333270	UEFI Bootloader AST 5.6	UEFI	11/3/2017	Signing

Viewing page 1 of 61

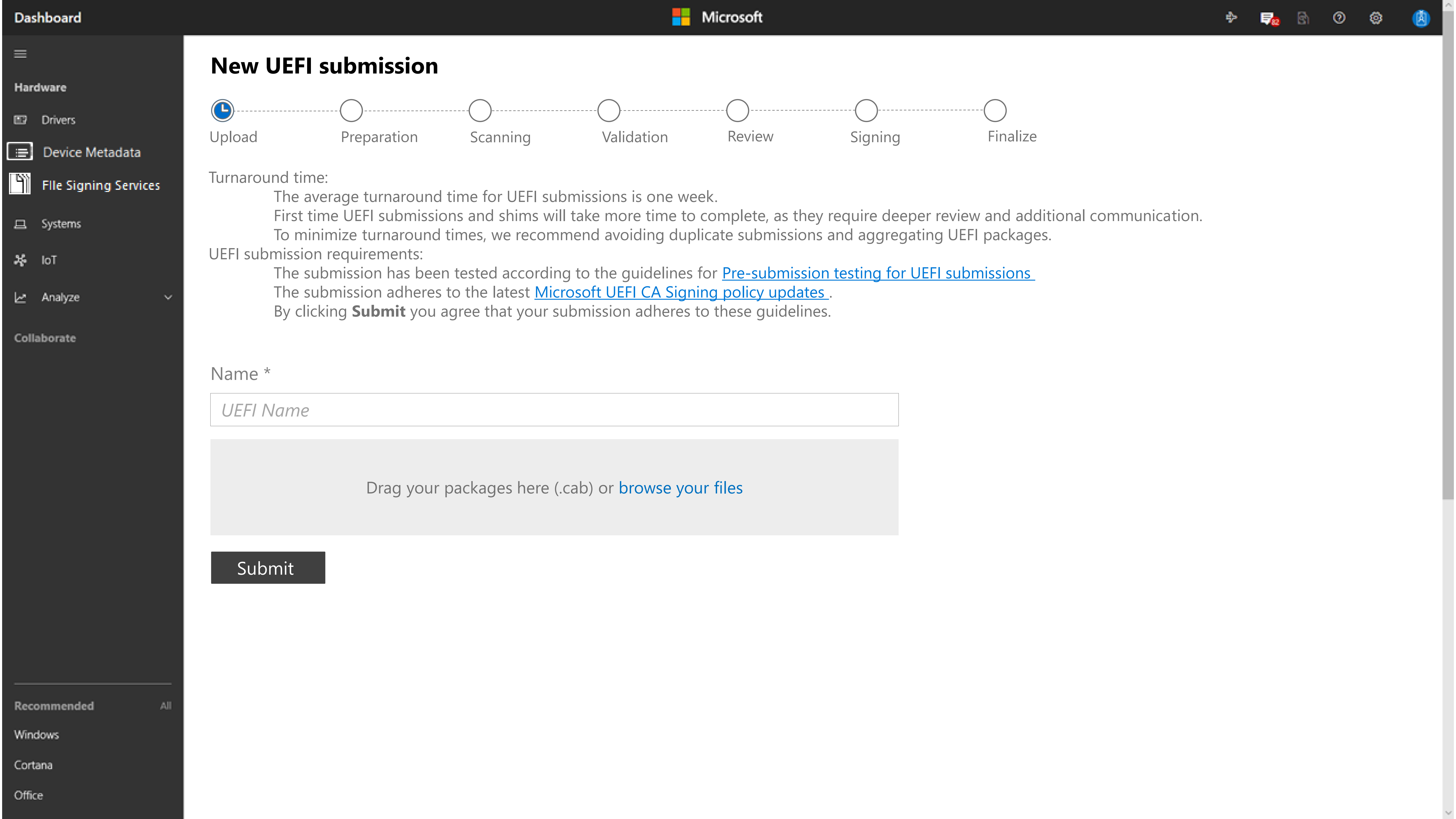
Submit new UEFI Signing Request



File Signing Services

ID	Product Name	Type	Submitted	Status
1152921504607396159	UEFI Bootloader NXA 1.8	UEFI	12/1/2017	Preparation
1152921504607395848	LSA 1	LSA	11/3/2017	Completed
1152921504607395758	UEFI Bootloader UST 2.3	UEFI	5/1/2017	Completed
1152921504607393274	UEFI SpiralOrbit	UEFI	11/11/2017	Review
1152921504607393273	UEFI Bootloader ASO 3.1	UEFI	10/1/2017	Validation
1152921504607393272	LSA 2	LSA	6/8/2017	Completed
1152921504607393271	UEFI Bootloader ASG 1.2	UEFI	11/4/2017	Signing
1152921504607393270	UEFI Bootloader NXP 4.5	UEFI	12/2/2017	Waiting for Upload
1152921504607333270	UEFI Bootloader AST 5.6	UEFI	11/3/2017	Signing

Viewing page 1 of 61



New UEFI submission

Turnaround time:

The average turnaround time for UEFI submissions is one week.
First time UEFI submissions and shims will take more time to complete, as they require deeper review and additional communication.
To minimize turnaround times, we recommend avoiding duplicate submissions and aggregating UEFI packages.

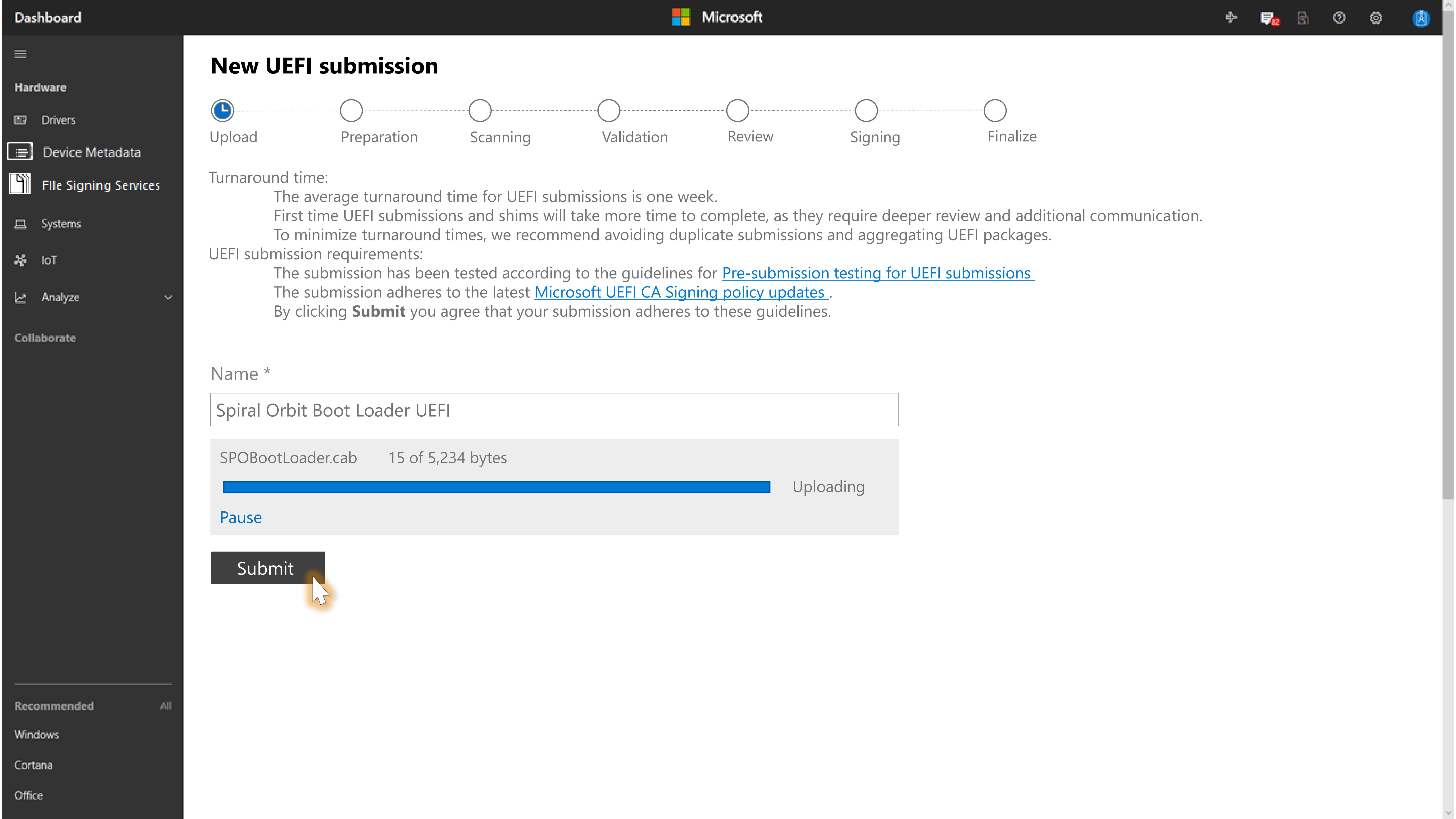
UEFI submission requirements:

The submission has been tested according to the guidelines for Pre-submission testing for UEFI submissions
The submission adheres to the latest Microsoft UEFI CA Signing policy updates .
By clicking **Submit** you agree that your submission adheres to these guidelines.

Name *

UEFI Name

Drag your packages here (.cab) or browse your files



New UEFI submission

Turnaround time:

The average turnaround time for UEFI submissions is one week.

First time UEFI submissions and shims will take more time to complete, as they require deeper review and additional communication.

To minimize turnaround times, we recommend avoiding duplicate submissions and aggregating UEFI packages.

UEFI submission requirements:

The submission has been tested according to the guidelines for Pre-submission testing for UEFI submissions

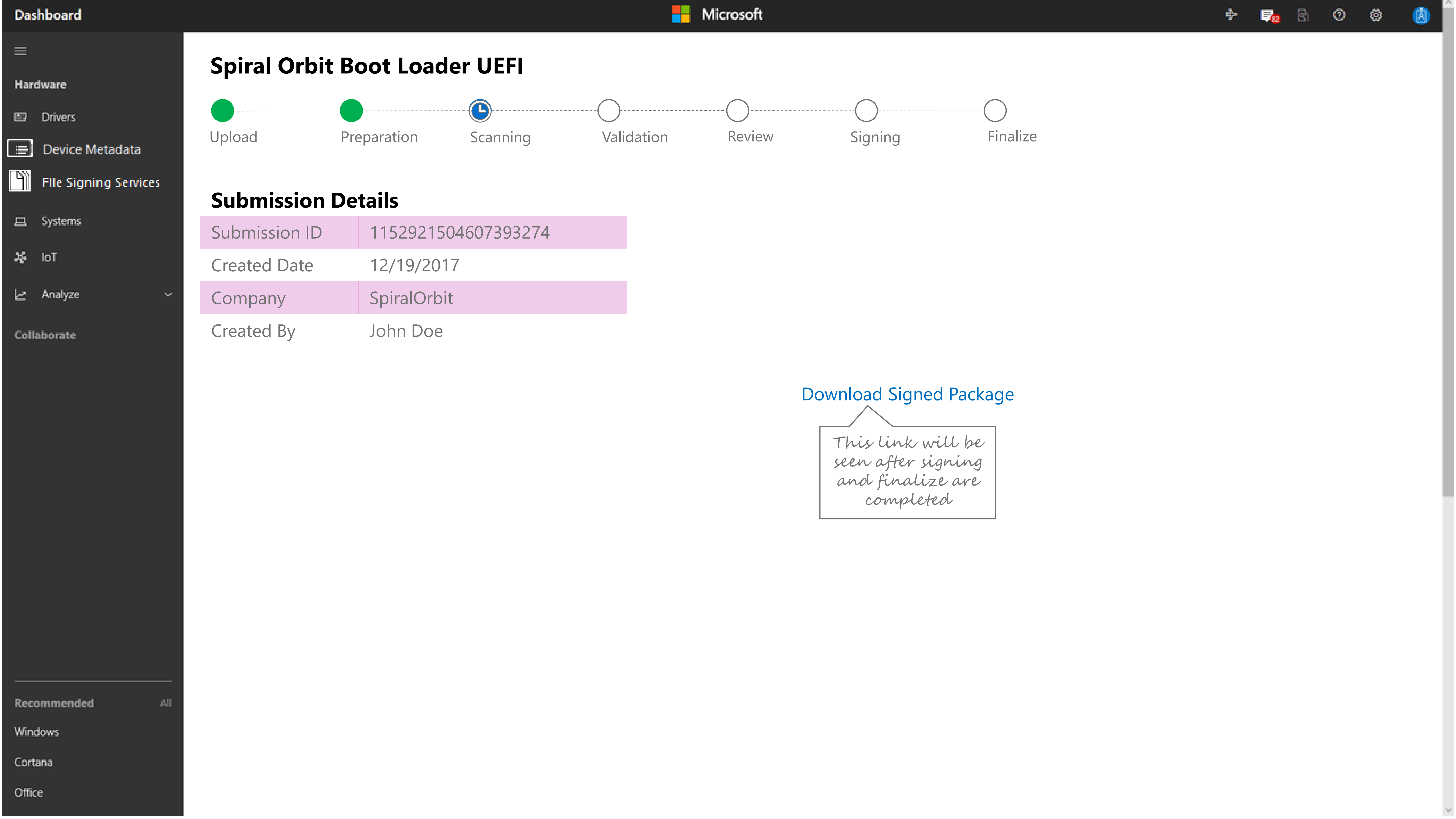
The submission adheres to the latest Microsoft UEFI CA Signing policy updates .

By clicking **Submit** you agree that your submission adheres to these guidelines.

Name *

Spiral Orbit Boot Loader UEFI

SPOBootLoader.cab 15 of 5,234 bytes



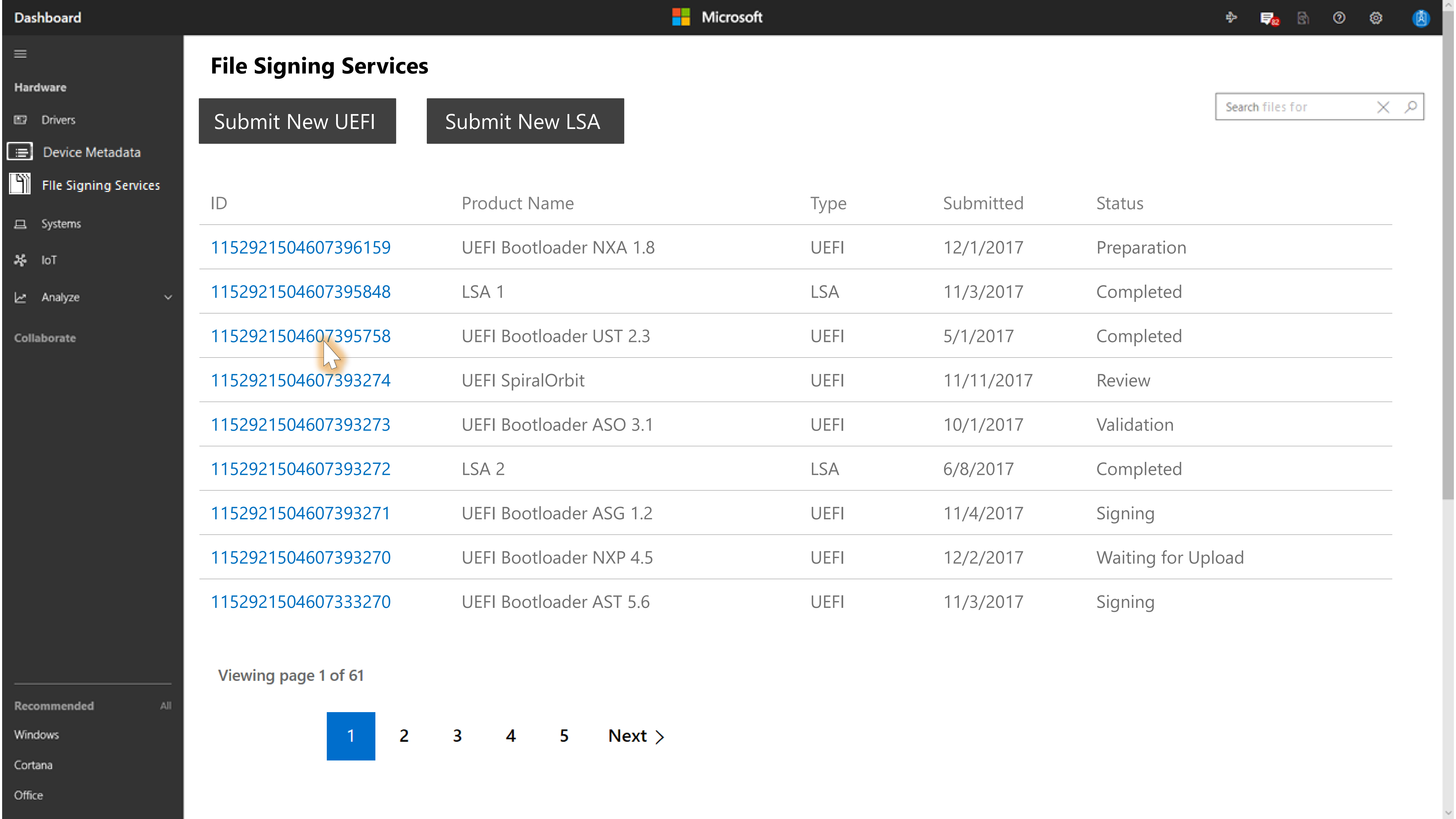
Spiral Orbit Boot Loader UEFI

Submission Details

Submission ID	1152921504607393274
Created Date	12/19/2017
Company	SpiralOrbit
Created By	John Doe

This link will be seen after signing and finalize are completed

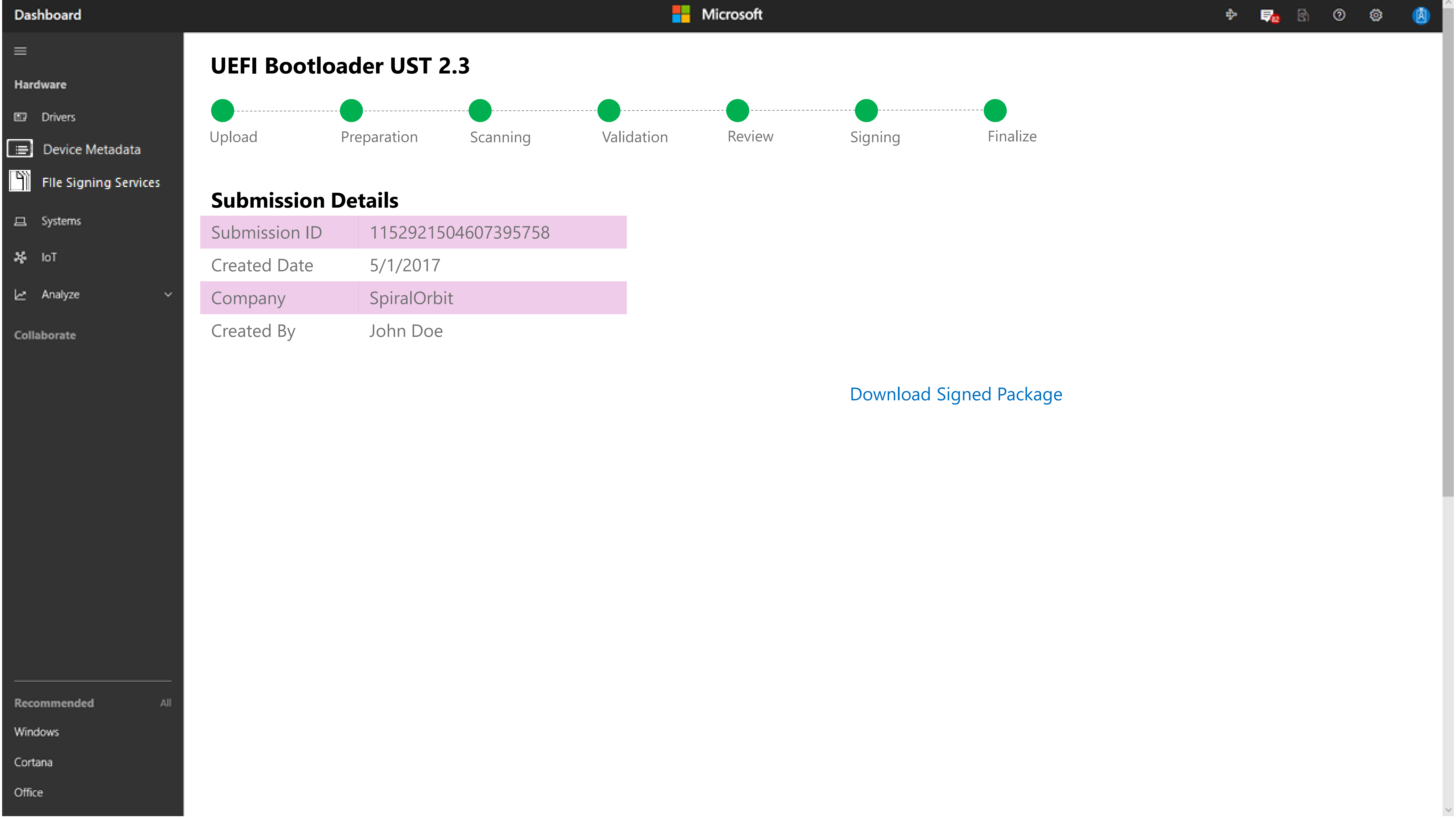
View Existing UEFI Submissions



File Signing Services

ID	Product Name	Type	Submitted	Status
1152921504607396159	UEFI Bootloader NXA 1.8	UEFI	12/1/2017	Preparation
1152921504607395848	LSA 1	LSA	11/3/2017	Completed
1152921504607395758	UEFI Bootloader UST 2.3	UEFI	5/1/2017	Completed
1152921504607393274	UEFI SpiralOrbit	UEFI	11/11/2017	Review
1152921504607393273	UEFI Bootloader ASO 3.1	UEFI	10/1/2017	Validation
1152921504607393272	LSA 2	LSA	6/8/2017	Completed
1152921504607393271	UEFI Bootloader ASG 1.2	UEFI	11/4/2017	Signing
1152921504607393270	UEFI Bootloader NXP 4.5	UEFI	12/2/2017	Waiting for Upload
1152921504607333270	UEFI Bootloader AST 5.6	UEFI	11/3/2017	Signing

Viewing page 1 of 61



UEFI Bootloader UST 2.3

Submission Details

Submission ID	1152921504607395758
Created Date	5/1/2017
Company	SpiralOrbit
Created By	John Doe

UEFI CA – ARM Signing

ARM Signing

- Microsoft is relaxing the policy to permit signing of AARCH64 EFI modules
- Some new, global prerequisites to follow...

Baby Duck – Secure Boot Settings

Signing Challenges

- Multi-signing
  - Overhead to create/manage IHV and ISV and OEM CAs
    - Long lived certs are a challenge
  - OEM/enterprise management of certs
  - Device and System compatibility
    - Increased size

Signing Solution Proposal

- Replace with .RSRC section + Opus Info
  - UEFI CA signer and process remains as-is
  - Opus added by Microsoft UEFI CA to signature
    - specifies Product Vendor from name on file
  - Action to UEFI Submitters: add .RSRC section (trivial)
    - specifies Product Name, File Name & Version
    - Provides improved revocation of insecure versions without flooding dbx with hashes or rolling signing keys
    - This will be a requirement for signing by the UEFI CA going forward
    - No compat issue with downlevel
  - Action for System Firmware:
    - New authentication of EFI images based upon Opus & .RSRC
    - Will be a requirement of future Windows OS

Deployment Challenges

- Unenlightened algorithm for reliable db configuration is impractical
- Per-device targeting
  - Only current option is per-device-unique PK/KEK, does not scale
- Resistance to consistent workflows

Deployment Alternatives

- Infeasible heuristic replaced with explicit profiles & custom mode
- Targeting: SMBIOS Vendor, Model, Serial #, & UUID
- End-user workflow minimized

Building it

Are all blocking issues mitigated?

What is the value prop?

- Replace the UEFI CA with stricter trust ⚓

Cost/benefit justified?

Device Firmware Configuration Interface (DFCI)

Contact SAUEFI@Microsoft.com

Discussion of non-public Windows plans

Questions?

Thanks for attending the Spring 2018 UEFI Plugfest

For more information on the UEFI Forum and UEFI Specifications, visit http://www.uefi.org

Questions regarding this presentation mailto:SAUEFI@Microsoft.com

presented by

Microsoft