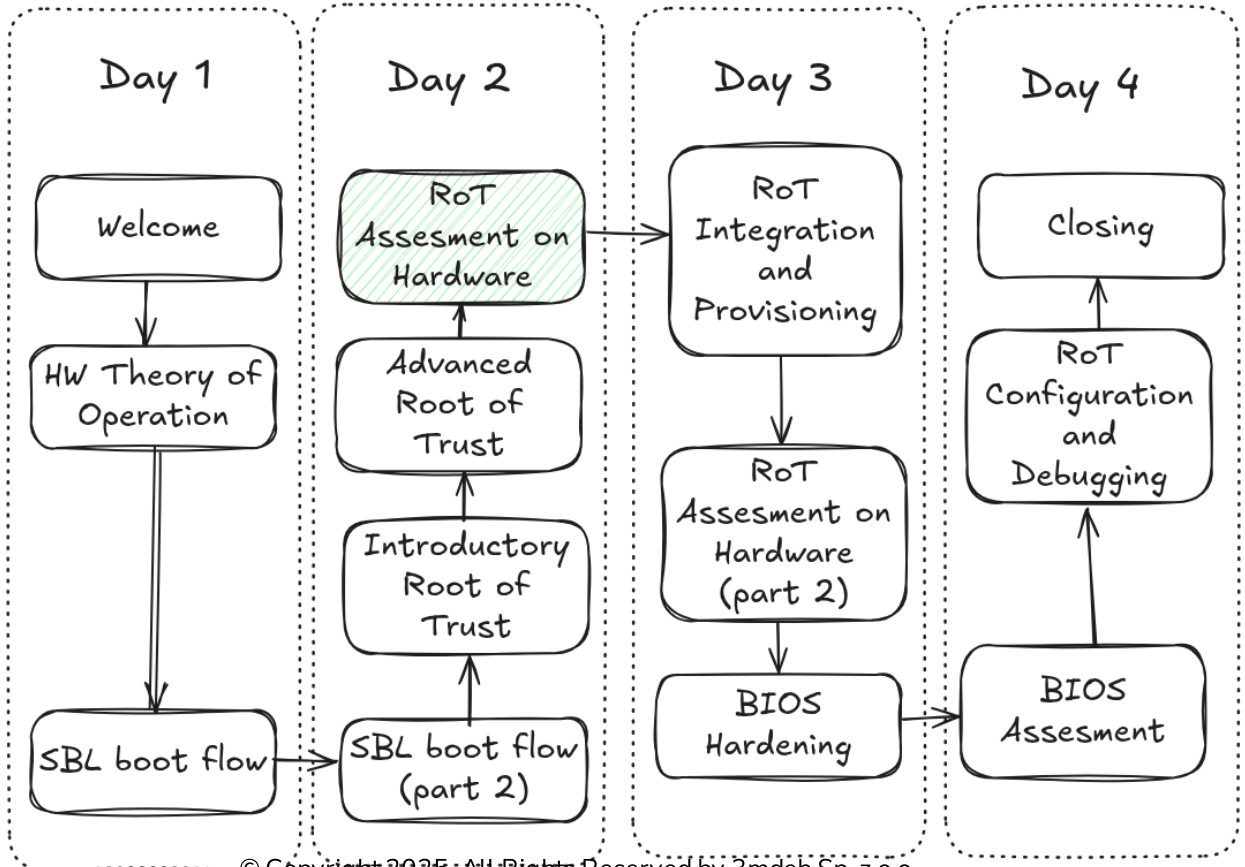
DS09SBL: Dasharo TrustRoot Training

Assessing Root of Trust Implementations

Dasharo Trust Root Training

Where we are in the course



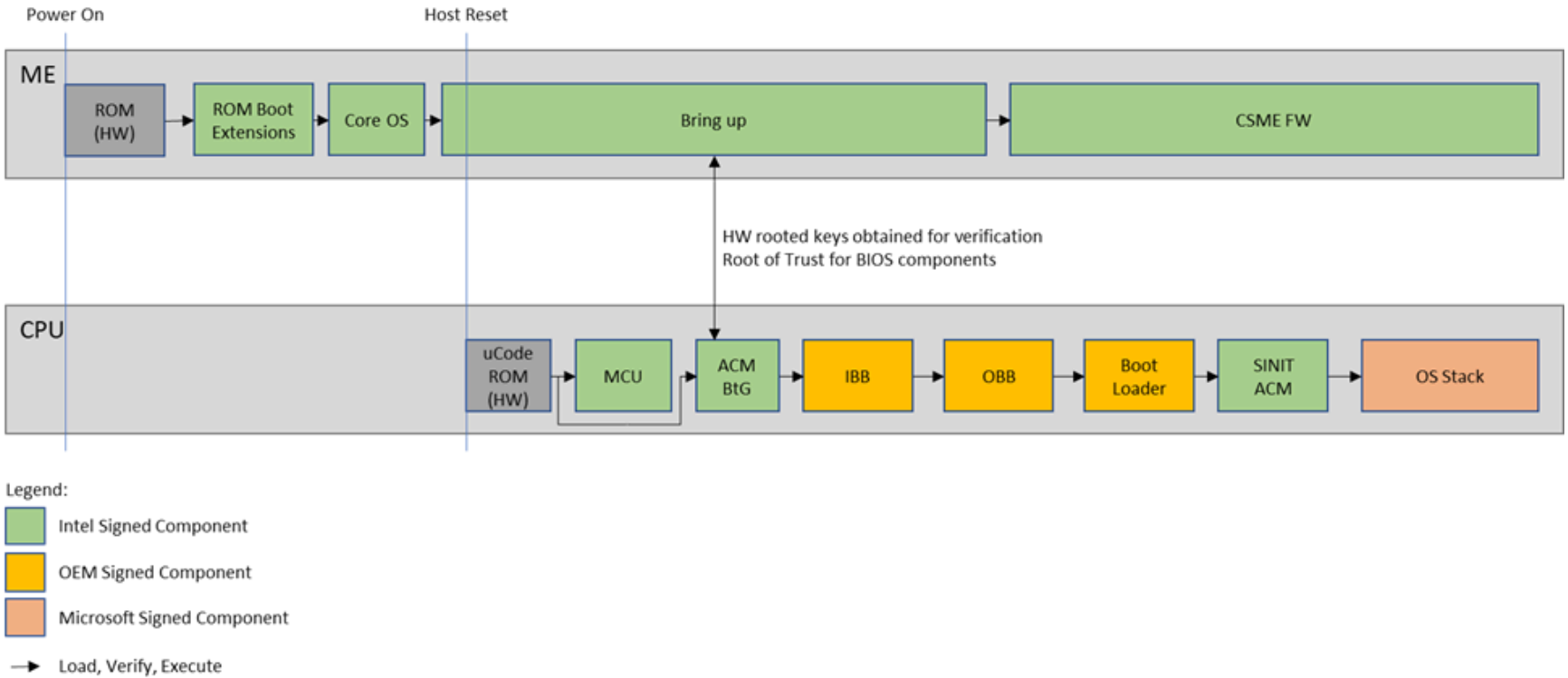
Goals of the Presentation

- Techniques for assessing Root of Trust mechanisms
- Tools and methodologies for evaluating Root of Trust effectiveness
- Hands-On Lab: Assessing and validating Root of Trust in provided hardware

Tooling

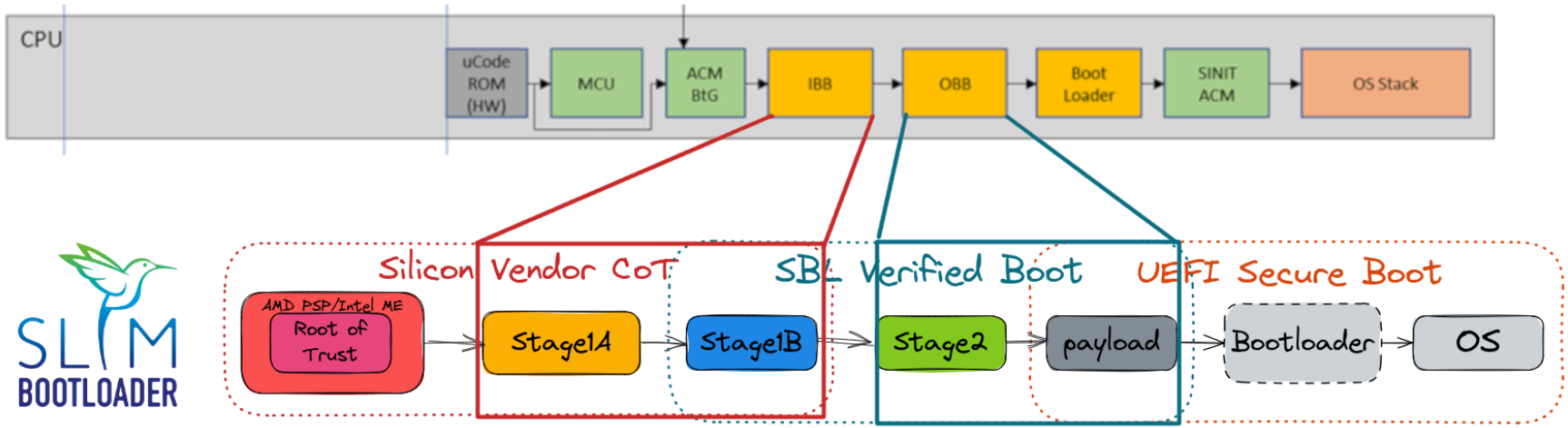
- `MEInfo`
- `FPT`
- Linux sysfs
- firmware debug log (if applicable)
- UEFITool

Managenent Engines role in platform security



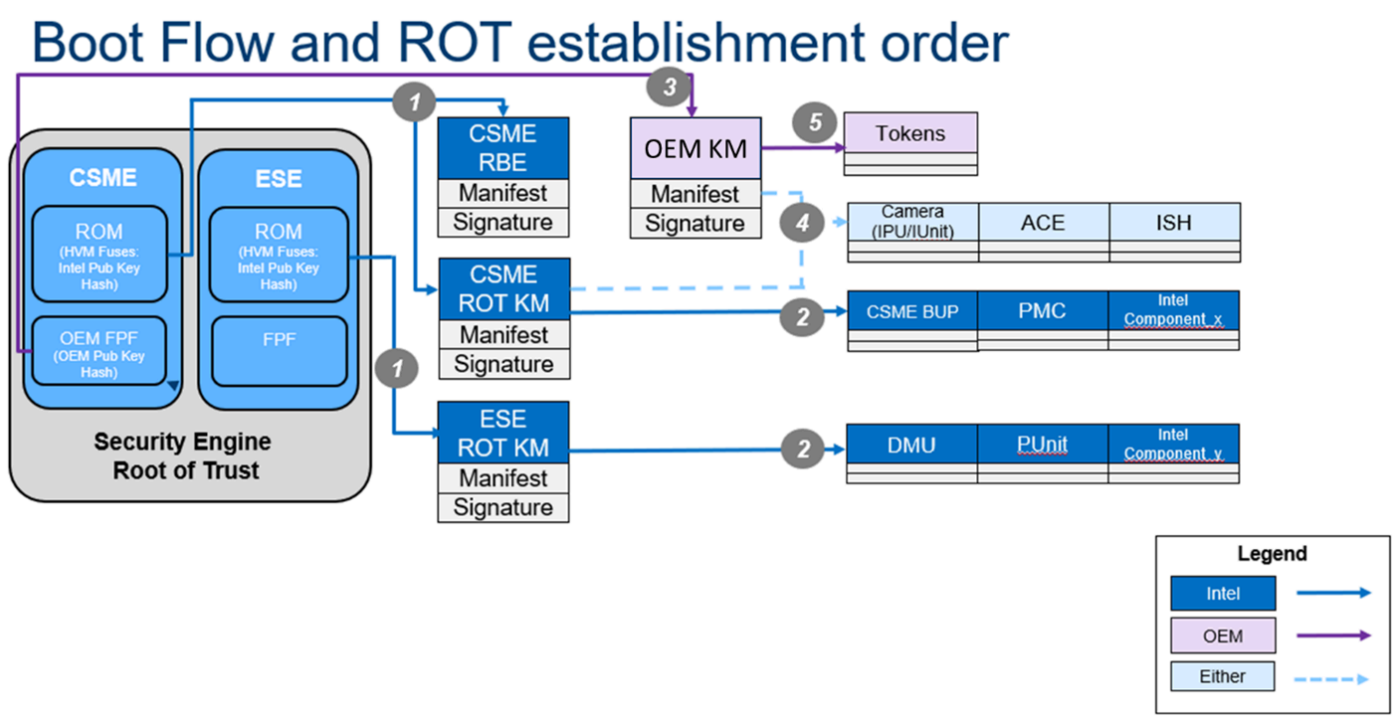
Source: Intel: Key usage in integrated firmware images

Slim Bootloader boot flow with Boot Guard



CoT - Chain of Trust
IBV - Independent BIOS Vendor
IFV - Independent Firmware Vendor
SEC/PEI/DXE/BSD - UEFI boot phases

Management Engine boot flow and RoT establishment



Source: Intel: Key usage in integrated firmware images

Management Engine toolkit

- Set of tools to:
  - Build Intel firmware images
  - Configure Management Engine and flash descriptor straps
  - Generate manifests and sign Management Engine components
- Set of binary components for multiple IP blocks
  - During the image build process, these are glued together into a single binary and put into the Management Engine region in flash
- Can be legally obtained under NDA only through Intel's Resource and Documentation Center (R&DC)

Management Engine toolkit license

- Licenses are available in the `day2/licenses` directory in the shared link to the training materials:
  - `PV Intel OBL Software License Agreement_11.2.2017_1.pdf` - license accompanying the toolkit when downloading from R&DC
  - `SLA_TOOLS.pdf` - license accompanying the tools inside the ME toolkit package

Practice #302 Exercise #1

Finding the right Management Engine toolkit

First we have to find out what ME version we are using/running.

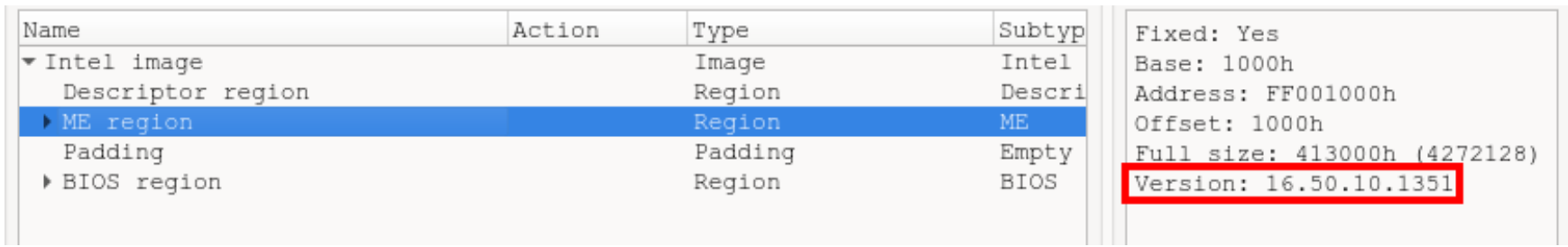
- Checking version on Linux: `cat /sys/class/mei/mei0/fw_ver`

```
0:16.50.5.1303
0:16.50.5.1303
0:16.50.5.1308
```

Note

This method requires Management Engine to be enabled and running normally. May not be the most reliable method. There are three versions listed here: main, recovery, fitc.

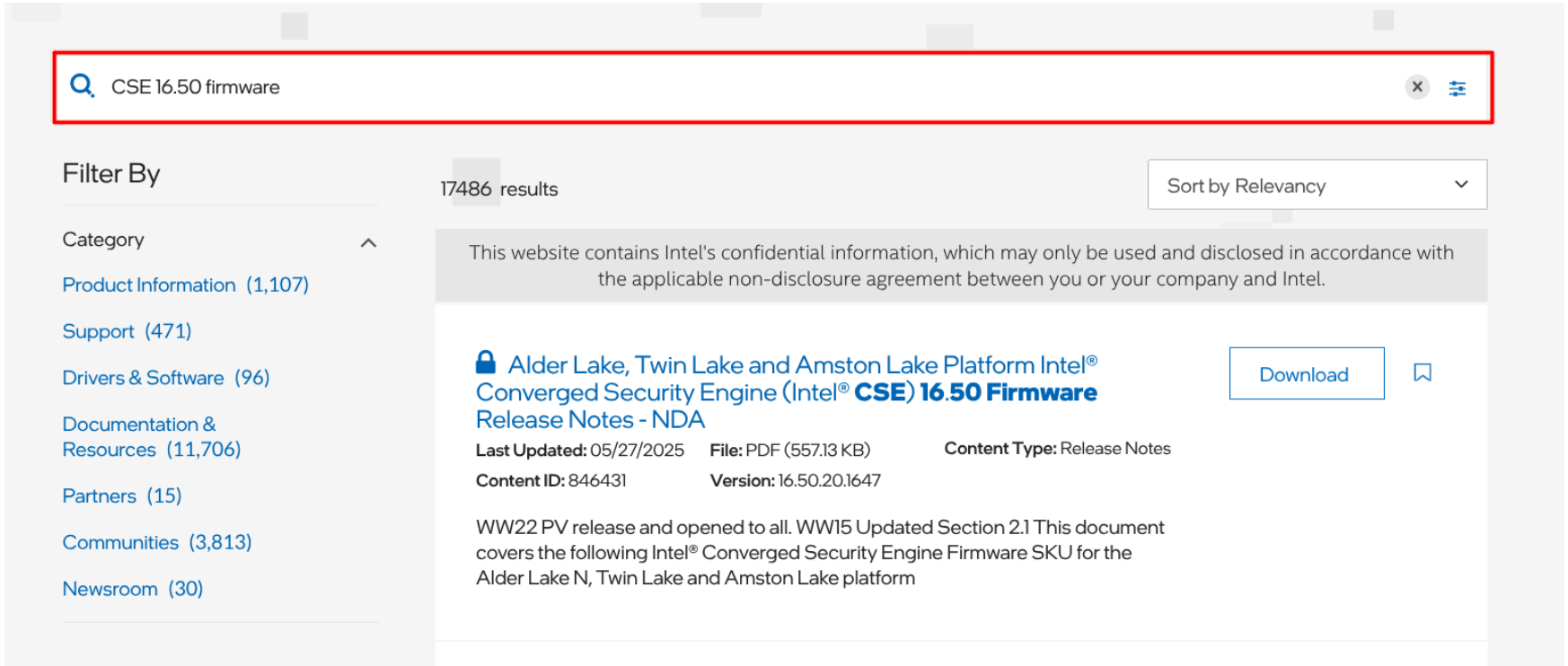
- Checking version with UEFITool
  - Open a full image, e.g. Slim Bootloader in the UEFITool: `UEFITool Outputs/odroid_h4/ifwi-release.bin` or `UEFITool Outputs/odroid_h4/ifwi-debug.bin`
  - Expand the `Intel Image`
  - Click on `ME Region`

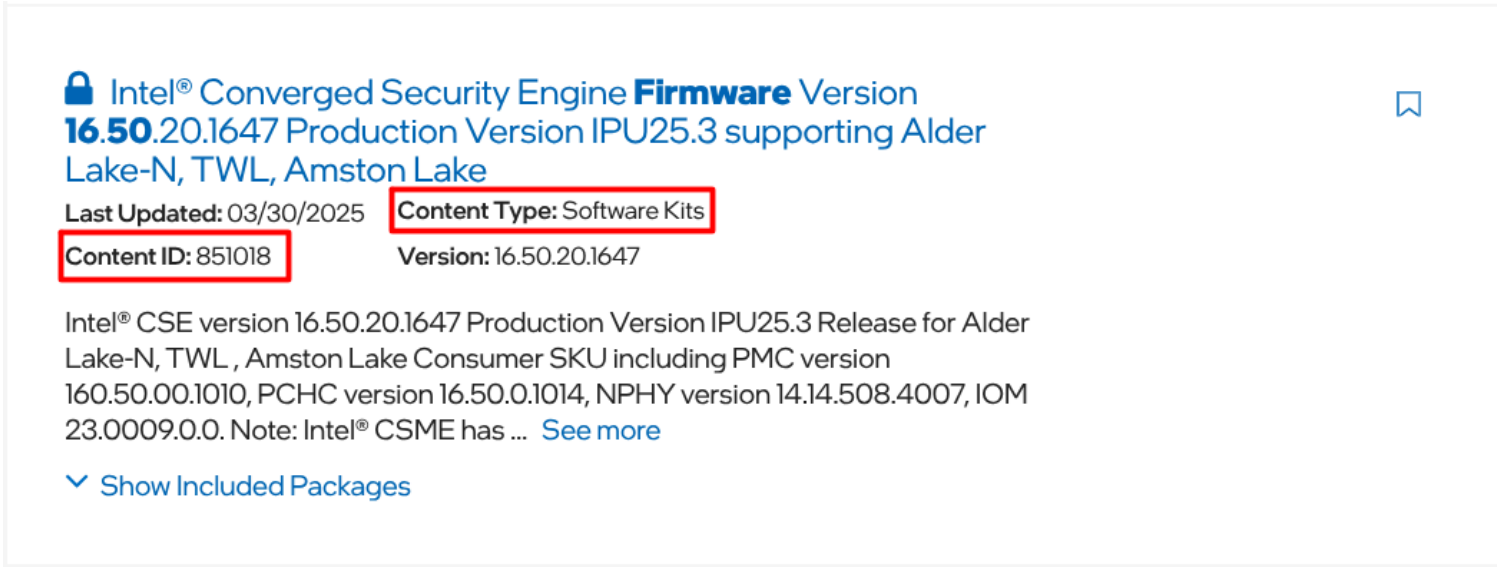


The Management Engine toolkit should match the running ME major and minor version at minimum

> **Note**
>
> Sometimes if the patch version and build number are too far from the toolkit's version, certain tools may have problems in operating on the image

- Login to Intel R&DC portal
- Search for `CSE 16.50 firmware`
- **Important**: Look at links with `Content Type: Software Kits`





tip

If you know the Content ID beforehand, just log in to the R&DC portal and type the URL in the browser to start downloading directly: `https://cdrdv2.intel.com/v1/dl/getContent/<content_id>` replacing the `<content_id>` with the desired ID. But it only works for single-file content. ME toolkits consist of ZIP, license, SBOM and SBOM signature. In such case, searching for the ID gives good results quicker.

Management Engine toolkit structure

```
├── ADP-P Consumer Bring Up Guide.pdf
├── ADP-P MFIT Consumer Bring Up Guide.pdf
├── Alderlake-N Client SPI Programming Guide.pdf
├── Image Components
│   ├── 3rd party Licenses in Security FW
│   ├── CSME
│   ├── NPHY
│   ├── PCHC
│   ├── PMC
│   ├── PRESTITCHED
│   └── TCSS
└── Tools
    ├── 3rd party Licenses in Security Tools
    ├── DnX_Tools
    └── System_Tools
```

Management Engine toolkit structure

```
├── ADP-P Consumer Bring Up Guide.pdf
├── ADP-P MFIT Consumer Bring Up Guide.pdf
├── Alderlake-N Client SPI Programming Guide.pdf
├── Image Components
│   ├── 3rd party Licenses in Security FW
│   ├── CSME
│   ├── NPHY
│   ├── PCHC
│   ├── PMC
│   ├── PRESTITCHED
│   └── TCSS
└── Tools
    ├── 3rd party Licenses in Security Tools
    ├── DnX_Tools
    └── System_Tools
```

Management Engine toolkit structure

```
├── ADP-P Consumer Bring Up Guide.pdf
├── ADP-P MFIT Consumer Bring Up Guide.pdf
├── Alderlake-N Client SPI Programming Guide.pdf
├── Image Components
│   ├── 3rd party Licenses in Security FW
│   ├── CSME
│   ├── NPHY
│   ├── PCHC
│   ├── PMC
│   ├── PRESTITCHED
│   └── TCSS
└── Tools
    ├── 3rd party Licenses in Security Tools
    ├── DnX_Tools
    └── System_Tools
```

Management Engine System Tools

```
└── System_Tools
    ├── DNX
    ├── FPT
    ├── FWUpdate
    ├── FWUpdate_RS
    ├── ICC Tools
    ├── MEInfo
    ├── MEManuf
    ├── MEU
    ├── MFIT
    ├── SLA_TOOLS.PDF
    ├── System Tools User Guide.pdf
    └── Tools Errors User Guide.pdf
```

Management Engine System Tools

```
└── System_Tools
    ├── DNX
    ├── FPT
    ├── FWUpdate
    ├── FWUpdate_RS
    ├── ICC Tools
    ├── MEInfo
    ├── MEManuf
    ├── MEU
    ├── MFIT
    ├── SLA_TOOLS.PDF
    ├── System Tools User Guide.pdf
    └── Tools Errors User Guide.pdf
```

Management Engine System Tools

```
└── System_Tools
    ├── DNX
    ├── FPT
    ├── FWUpdate
    ├── FWUpdate_RS
    ├── ICC Tools
    ├── MEInfo
    ├── MEManuf
    ├── MEU
    ├── MFIT
    ├── SLA_TOOLS.PDF
    ├── System Tools User Guide.pdf
    └── Tools Errors User Guide.pdf
```

Management Engine System Tools

```
└── System_Tools
    ├── DNX
    ├── FPT
    ├── FWUpdate
    ├── FWUpdate_RS
    ├── ICC Tools
    ├── MEInfo
    ├── MEManuf
    ├── MEU
    ├── MFIT
    ├── SLA_TOOLS.PDF
    ├── System Tools User Guide.pdf
    └── Tools Errors User Guide.pdf
```

Management Engine System Tools

```
└── System_Tools
    ├── DNX
    ├── FPT
    ├── FWUpdate
    ├── FWUpdate_RS
    ├── ICC Tools
    ├── MEInfo
    ├── MEManuf
    ├── MEU
    ├── MFIT
    ├── SLA_TOOLS.PDF
    ├── System Tools User Guide.pdf
    └── Tools Errors User Guide.pdf
```

Management Engine System Tools

```
└── System_Tools
    ├── DNX
    ├── FPT
    ├── FWUpdate
    ├── FWUpdate_RS
    ├── ICC Tools
    ├── MEInfo
    ├── MEManuf
    ├── MEU
    ├── MFIT
    ├── SLA_TOOLS.PDF
    ├── System Tools User Guide.pdf
    └── Tools Errors User Guide.pdf
```

Management Engine System Tools

```
└── System_Tools
    ├── DNX
    ├── FPT
    ├── FWUpdate
    ├── FWUpdate_RS
    ├── ICC Tools
    ├── MEInfo
    ├── MEManuf
    ├── MEU
    ├── MFIT
    ├── SLA_TOOLS.PDF
    ├── System Tools User Guide.pdf
    └── Tools Errors User Guide.pdf
```

Management Engine System Tools

```
└── System_Tools
    ├── DNX
    ├── FPT
    ├── FWUpdate
    ├── FWUpdate_RS
    ├── ICC Tools
    ├── MEInfo
    ├── MEManuf
    ├── MEU
    ├── MFIT
    ├── SLA_TOOLS.PDF
    ├── System Tools User Guide.pdf
    └── Tools Errors User Guide.pdf
```

Before we begin, let's make sure we have two binaries:

- `Dasharo (Slim Bootloader+UEFI) v0.9.0` debug build (if we don't have one yet).
- Original vendor firmware on second flash (should be there if we haven't written anything to it yet)

Practice #302 Exercise #2

Let's build and flash debug version Dasharo (Slim Bootloader+UEFI) v0.9.0.

Building:

```
user@OST2-VM:~$ cd training_materials/src/slimbootloader
user@OST2-VM:~$ DEBUG=1 ./build.sh odroid_h4
```

Resulting image in `training_materials/src/slimbootloader/Outputs/Odroid_h4/ifwi-debug.bin`.

Copy the image on the DTS USB stick and plug it to ODROID H4.

Boot to DTS and flash it on ODROID H4. Verify that Slim Bootloader prints debug messages on serial port.

We will attempt to run Intel ME tools to read the FPF state to confirm that fuses are not locked and assess feasibility of Intel Boot Guard provisioning.

FPF assessment will let us determine whether the platform is unlocked and capable of provisioning Intel Boot Guard. FPF values are an absolute indicator of the ME configuration state.

For that, we will use the `MEInfo` tool, which you can find in ME firmware toolkit.

Transfer the tool to the USB stick with DTS and plug it in Odroid-H4. Boot the DTS and run the tool. Dump output to file. It would be useful for further comparison.

```
root@ODROID-DTS# chmod +x MEInfo
root@ODROID-DTS# ./MEInfo -VERBOSE > meinfo_sbl.log
```

```
Intel (R) ME Info Version: 16.50.15.1519
Copyright (C) 2005 - 2024, Intel Corporation. All rights reserved.

    FW Status Register1                         0x90000255
    FW Status Register2                         0x69000506
    FW Status Register3                         0x00000020
    FW Status Register4                         0x00004004
    FW Status Register5                         0x00000000
    FW Status Register6                         0x00400002

General FW Information
    Current FW State                            Normal
    Flash Partition Table                       Valid
    FW Memory State                             CM0 with UMA
    FW Initialization                           Complete
    BUP Loading state                           Success
    FW Error Code                               No Error
    FW Mode Of Operation                        Normal
    SPI Flash Log                               Present
    FW Loading Phase                            HOSTCOMM Module
    FW Loading Phase Status                     UNKNOWN
    ME File System Corrupted                    No
    RPMC status                                 OK



    Platform Type                               Mobile
    FW Image Type                               Production
    Last ME Reset Reason                        Other
    BIOS Boot State (EOP)                       Pre Boot
    BIOS Boot State (CBD)                       Pre Boot
    Boot Critical Code Redundancy               Disabled
    Current Boot Partition                      1
    Factory Defaults Restoration Status         Disabled
    CPUID                                       0x806E0
    Factory Defaults Recovery Status            Enabled
    Firmware Update OEM ID                      00000000-0000-0000-0000-000000000000
    Intel(R) ICPS SW SKUing Eligible            Disabled
    Intel(R) ICM Entitlement                    Disabled
    Camera privacy feature control disabled     True
    Crypto HW Support                           Enabled
    Intel(R) ISH Power State                    Disabled
    OEM Tag                                     0x00
    FW Update State                             Enabled
    TLS State                                   Enabled
    CSME Measured Boot to TPM                   Disabled
    BIOS Recovery State                         Disabled

Intel(R) ME Code Versions
    BIOS Version                                Dasharo (Slim Bootloader+UEFI) v0.9.
    FW Version                                  16.50.10.1351 LP Consumer

Extended Platform Services
    License Installed                           False
    License Period                              0 Periods

IUPs Information
    PMC FW Version                              160.50.0.1009
    OEM FW Version                              0.0.0.0000
    IUNT FW Version                             0.10.1.7051
    IOM FW Version                              35.8.0.0000
    NPHY FW Version                             14.14.508.4007
    PCHC FW Version                             16.50.0.1013

PCH Information
    PCH Name                                    ADL
    PCH Device ID                               5481
    PCH Revision ID                             A0
    PCH SKU Type                                Production PRQ Revenue
    PCH Replacement State                       Disabled
```

```
Transactional FW Information
    Original image type                     Consumer
    Current sku type                        Consumer

Flash Information
    Storage Device Type                     SPI
    SPI Flash ID 1                          EF7018
    RPMC                                    Unsupported
    RPMC Bind Counter                       0
    RPMC Bind Status                        Pre-bind
    RPMC Rebind                             Supported
    RPMC Replay Protection Max Rebind       15
    BIOS Read Access                        0×FFFF
    BIOS Write Access                       0×FFFF
    GBE Read Access                         0×FFFF
    GBE Write Access                        0×FFFF
    ME Read Access                          0×FFFF
    ME Write Access                         0×FFFF
    EC Read Access                          0×FFFF
    EC Write Access                         0×FFFF

FW Capabilities                             0×31309200
    Intel(R) Protected Audio Video Path     Present/Enabled
    Intel(R) Dynamic Application Loader     Present/Enabled
    Intel(R) Platform Trust Technology      Present/Enabled
    Persistent RTC and Memory               Present/Enabled

End Of Manufacturing
    NVAR Configuration State                Unlocked
    EOM Settings                            Lock(Flash,Config)
    EOM Flow                                Not set
    HW Binding State                        Enabled
    Flash Protection Mode                   Unprotected
    FPF Committed                           No
```

```
Intel(R) Protected Audio Video Path
    PAVP State                              Yes

Security Version Numbers
    Trusted Computing Base SVN              1

    Firmware Version Control SVNs
        PMC                                 0   [minimum allowed: 0]
        CSE                                 1   [minimum allowed: 0]
        ROT KM                              0   [minimum allowed: 0]
        IDLM                                0   [minimum allowed: 0]
        CSME bootstrap                      0   [minimum allowed: 0]
        OEM KM                              0   [minimum allowed: 0]

HW Glitch Detection                         0×1749
    TRC Polarity                            Rising Trans
    TRC Mode                                Full-cycle polarity trans
    TRC State                               Enabled

Intel(R) Platform Trust Technology
    Intel(R) PTT initial power-up state     Enabled
    Intel(R) PTT State                      Enabled
    SMx State                               Enabled
    RSA1K Support                           Disabled

Debug Information
    Token Present                           No
    Token Handling Status                   Not Available
    Invalid Knob Detected                   Not Available
    Consent                                 Unknown
    DFx Policy                              Unprivileged Public Debug Enabled
```

FW Supported FPFs	FPF	UEP
		*In Use
	---	---
1st OEM Key Hash Revoked	Not set	Disabled
1st OEM Key Hash size	Not set	Enabled
1st OEM RSA Key size	Not set	Enabled
2nd OEM Key Hash Revoked	Not set	Disabled
2nd OEM Key Hash size	Not set	Enabled
2nd OEM RSA Key size	Not set	Enabled
BSMM Firmware Version Control	Not set	Enabled
CSE FW Firmware Version Control	Not set	Enabled
CSME Bootstrap Firmware Version Control	Not set	Enabled
DNX Firmware Version Control	Not set	Enabled
Error Enforcement Policy 0	Not set	Disabled
Error Enforcement Policy 1	Not set	Disabled
Flash Descriptor Verification	Not set	Disabled
Glitch Detection Disabled	Not set	Enabled
IDLM Firmware Version Control	Not set	Enabled
Intel PTT Encryption Key	Not set	Not Revoked
Intel(R) Manageability HW Fuse Status	Not set	Enabled
Intel(R) PTT	Not set	Enabled
OEM ID	Not set	0×00
OEM KM Firmware Version Control	Not set	Enabled
OEM Key Manifest	Not set	Enabled
OEM Key Revocation State	Not set	Disabled
OEM Platform ID	Not set	0×00
OEM Secure Boot Policy	Not set	0×42
CPU Debugging	Not set	Disabled
BSP Initialization	Not set	Enabled
Protect BIOS Environment	Not set	Disabled
Measured Boot	Not set	Disabled
Verified Boot	Not set	Disabled
Key Manifest ID	Not set	0×01
Force Boot Guard ACM	Not set	Disabled

OEM key Hash RSA key size	Not set	Enabled
PID Refurbish Counter	Not set	0×00
PMC Firmware Version Control	Not set	Enabled
PTT Lockout Override Counter	Not set	0×00
Persistent PRTC Backup Power	Not set	Enabled
ROT Firmware Version Control	Not set	Enabled
RPMB Monotonic Counters	Not set	0×00
RPMC Rebinding	Not set	Enabled
RPMC Support	Not set	Enabled
SOC Config Lock State	Not set	Disabled
SPI Boot Source	Not set	Enabled
SPIRAL CPU	Not set	Disabled
Secure boot KM Firmware Version Control	Not set	Enabled
TXT Supported	Not set	Disabled
UFS Boot Source	Not set	Disabled
uCode Firmware Version Control	Not set	Enabled
DNX SVN	Not set	0×00
IDLM SVN	Not set	0×00
OEM KM SVN	Not set	0×00
PMC SVN	Not set	0×00
ROT KM SVN	Not set	0×00
Secure boot ACM SVN	Not set	0×00
Secure boot BSMM SVN	Not set	0×00
Secure boot KM SVN	Not set	0×00
Ucode SVN	Not set	0×00
1st OEM Public Key Hash FPF	Not set	
1st OEM Public Key Hash UEP		
0638EFCBB322E655C912D984070450BD22151A6BA96E53E8		
6DDBAD48273EEE2BAB15DBE819696D989ABDC5EB9EB68A11		
2nd OEM Public Key Hash FPF	Not set	
2nd OEM Public Key Hash UEP		
0000000000000000000000000000000000000000000000000000		
0000000000000000000000000000000000000000000000000000		

- `1st/2nd OEM Public Key Hash FPF`

- `1st/2nd OEM Public Key Hash FPF`
- `1st/2nd OEM Public Key Hash ME/UEP`

- `1st/2nd OEM Public Key Hash FPF`
- `1st/2nd OEM Public Key Hash ME/UEP`
- `Secure boot ACM SVN`

- `1st/2nd OEM Public Key Hash FPF`
- `1st/2nd OEM Public Key Hash ME/UEP`
- `Secure boot ACM SVN`
- `Secure boot KM SVN FPF`

- `1st/2nd OEM Public Key Hash FPF`
- `1st/2nd OEM Public Key Hash ME/UEP`
- `Secure boot ACM SVN`
- `Secure boot KM SVN FPF`
- `Secure boot BSMM SVN FPF`

- `1st/2nd OEM Public Key Hash FPF`
- `1st/2nd OEM Public Key Hash ME/UEP`
- `Secure boot ACM SVN`
- `Secure boot KM SVN FPF`
- `Secure boot BSMM SVN FPF`
- `FPF Committed No`

Practice #302 Exercise #4

Run the `MEInfo` tool, but this time on original vendor firmware.

- Swap the BIOS2 jumper position
- Boot with vendor firmware
- Run the tool and save output `./MEINFO -VERBOSE meinfo_ami.log`
- Compare the output with MEInfo output from Slim Bootloader

```
root@ODROID-DTS# diff meinfo_ami.log meinfo_sbl.log
29,30c29,30
<     BIOS Boot State (EOP)                       Post Boot
<     BIOS Boot State (CBD)                       Post Boot
---
>     BIOS Boot State (EOP)                       Pre Boot
>     BIOS Boot State (CBD)                       Pre Boot
49,50c49,50
<     BIOS Version                                1.0
<     FW Version                                  16.50.5.1303 LP Consumer
---
>     BIOS Version                                Dasharo (Slim Bootloader+UEFI) v0.9.0
>     FW Version                                  16.50.10.1351 LP Consumer
```

Evaluation of FPF emulation capability

FPF emulation capability is crucial feature when developing and testing Root of Trust. FPF emulation allows to use SPI flash-backed configuration values (aka emulated FPFs) to be used in place of real hardware FPFs. In short it avoids accidental burning of fuses with undesired value and wasting the hardware.

For this exercise, we will use Intel `FPT` tool from the ME toolkit to modify the emulated FPFs. Then we will confirm the changes with MEInfo.

Flash Programming Tool - FPT

- FPT can:
  - Program the flash chip on Intel paltform
  - Read and manipulate OEM FPF values
  - Change ME configuration variables
- Useful commands:
  - `./FPT -FPFS` - lists settable OEM FPFs
  - `./FPT -R <VAR_NAMNE>` - read a variable, e.g. OEM FPFs from the above list
  - `./FPT -U -N <VAR_NAMNE> -V <NEW_VALUE>` - updates variable `<VAR_NAME>` value

Let's try to run `FPT` and then run `MEInfo` to realize what was changed in the output.

Now we will test our binary

1. Copy `FPT` to USB stick with DTS.
2. Plug the stick to ODROID and boot into DTS
3. Enter shell
4. Run

```
root@ODROID-DTS# chmod +x FPT
root@ODROID-DTS# ./FPT -R "BSP Initialization"
root@ODROID-DTS# ./FPT -U -N "BSP Initialization" -V Disabled
root@ODROID-DTS# ./MEInfo |grep "BSP Initialization"
```

5. Power off the board, unplug the power cord.
6. Plug the power cord again and boot to DTS shell.
7. Run `./MEInfo |grep "BSP Initialization"` to check if changes persisted.

```
bash-5.2# ./FPT -R "BSP Initialization"
Intel (R) Flash Programming Tool Version: 16.50.15.1519
Copyright (C) 2005 - 2024, Intel Corporation. All rights reserved.

Variable: "BSP Initialization"
Value: Enabled / 00

FPT Operation Successful.

bash-5.2# ./FPT -U -N "BSP Initialization" -V Disabled
Intel (R) Flash Programming Tool Version: 16.50.15.1519
Copyright (C) 2005 - 2024, Intel Corporation. All rights reserved.

FPT Operation Successful.

bash-5.2# ./FPT -R "BSP Initialization"
Intel (R) Flash Programming Tool Version: 16.50.15.1519
Copyright (C) 2005 - 2024, Intel Corporation. All rights reserved.

Variable: "BSP Initialization"
Value: Disabled / 01

FPT Operation Successful.

bash-5.2# ./MEInfo |grep "BSP Initialization"
BSP Initialization                          Not set     Disabled
```

```
bash-5.2# ./FPT -R "BSP Initialization"
Intel (R) Flash Programming Tool Version: 16.50.15.1519
Copyright (C) 2005 - 2024, Intel Corporation. All rights reserved.

Variable: "BSP Initialization"
Value: Enabled / 00

FPT Operation Successful.

bash-5.2# ./FPT -U -N "BSP Initialization" -V Disabled
Intel (R) Flash Programming Tool Version: 16.50.15.1519
Copyright (C) 2005 - 2024, Intel Corporation. All rights reserved.

FPT Operation Successful.

bash-5.2# ./FPT -R "BSP Initialization"
Intel (R) Flash Programming Tool Version: 16.50.15.1519
Copyright (C) 2005 - 2024, Intel Corporation. All rights reserved.

Variable: "BSP Initialization"
Value: Disabled / 01

FPT Operation Successful.

bash-5.2# ./MEInfo |grep "BSP Initialization"
BSP Initialization                          Not set     Disabled
```

```
bash-5.2# ./FPT -R "BSP Initialization"
Intel (R) Flash Programming Tool Version: 16.50.15.1519
Copyright (C) 2005 - 2024, Intel Corporation. All rights reserved.

Variable: "BSP Initialization"
Value: Enabled / 00

FPT Operation Successful.

bash-5.2# ./FPT -U -N "BSP Initialization" -V Disabled
Intel (R) Flash Programming Tool Version: 16.50.15.1519
Copyright (C) 2005 - 2024, Intel Corporation. All rights reserved.

FPT Operation Successful.

bash-5.2# ./FPT -R "BSP Initialization"
Intel (R) Flash Programming Tool Version: 16.50.15.1519
Copyright (C) 2005 - 2024, Intel Corporation. All rights reserved.

Variable: "BSP Initialization"
Value: Disabled / 01

FPT Operation Successful.

bash-5.2# ./MEInfo |grep "BSP Initialization"
BSP Initialization                          Not set     Disabled
```

```
bash-5.2# ./FPT -R "BSP Initialization"
Intel (R) Flash Programming Tool Version: 16.50.15.1519
Copyright (C) 2005 - 2024, Intel Corporation. All rights reserved.

Variable: "BSP Initialization"
Value: Enabled / 00

FPT Operation Successful.

bash-5.2# ./FPT -U -N "BSP Initialization" -V Disabled
Intel (R) Flash Programming Tool Version: 16.50.15.1519
Copyright (C) 2005 - 2024, Intel Corporation. All rights reserved.

FPT Operation Successful.

bash-5.2# ./FPT -R "BSP Initialization"
Intel (R) Flash Programming Tool Version: 16.50.15.1519
Copyright (C) 2005 - 2024, Intel Corporation. All rights reserved.

Variable: "BSP Initialization"
Value: Disabled / 01

FPT Operation Successful.

bash-5.2# ./MEInfo |grep "BSP Initialization"
BSP Initialization                              Not set     Disabled
```

Evaluation based on firmware logs

- When running open-source firmware like coreboot or Slim Bootloader, the firmware may print information about Boot Guard state
- May be useful when not having utilities at hand or need to debug whether the Boot Guard configuration has been applied properly after provisioning
- Disadvantages:
  - Often requires NDA knowledge and documents how to interpret register values
  - Firmware often prints raw hex values, which are not human-readable
  - May not always give the most important information needed

Practice #302 Exercise #6

Run the debug version of Slim Bootlaoder and capture the output to a file:

```
user@OST2-VM:~$ minicom -D /dev/ttyUSB0 -b 115200 -C /tmp/output.log
```

Grep the output for `Boot Guard`:

```
user@OST2-VM:~$ cat /tmp/output.log |grep "Boot Guard"
```

```
user@OST2-VM:~$ cat /tmp/output.log |grep "Boot Guard"
[Boot Guard] AcmStatus : 0×00000000
[Boot Guard] BootStatus: 0×00000000
[Boot Guard] Boot Guard Failed or is Disabled!
[Boot Guard] Acm Info: 0×7000000000
[Boot Guard] Verified Boot Status: Disabled
[Boot Guard] Measured Boot Status: Disabled
Processor does not support Boot Guard.
Boot Guard Support status: 0
```

Evaluation based on firmware logs

- Slim Bootloader logs lack information about FPF fuse state.
- Fuse state can be checked in ME firmware status registers on Linux:

```
root@ODROID-DTS# cat /sys/class/mei/mei0/fw_status
90000255
02F10506
00000020
00004004
00000000
00400002
```

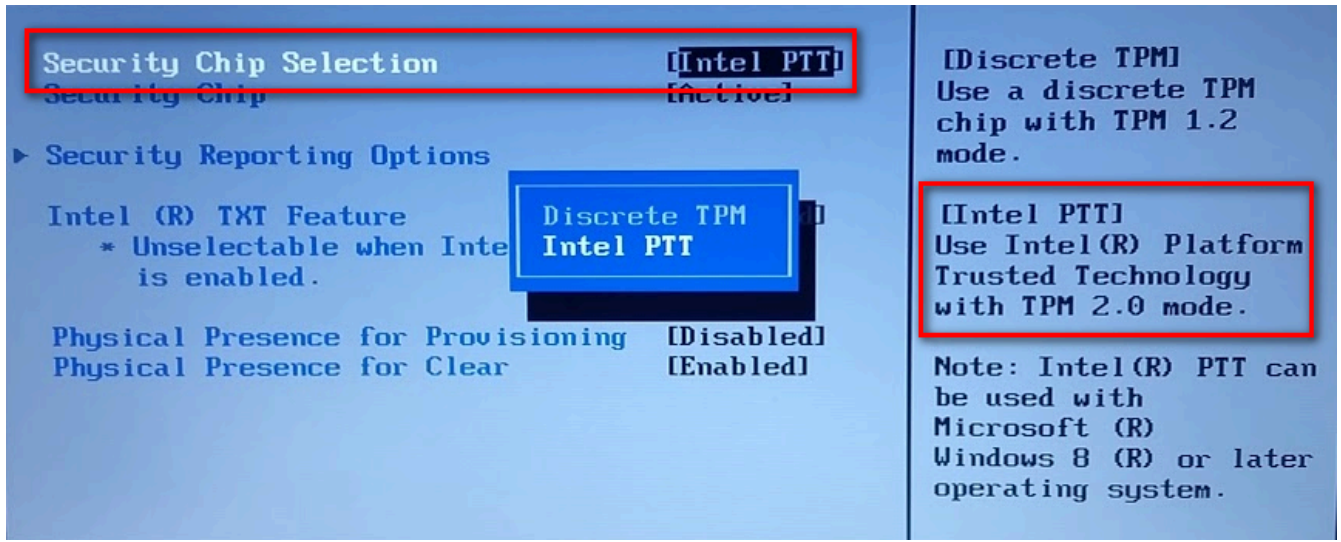
- The bit 30 of the 6th value indicates the fuse state, aka Field Programmable Fuses (FPF) SoC Configuration Lock. If the bit is set, it means the fuses are blown and Boot Guard would not be possible to be disabled or enabled.
- Decoding all the bits is possible with (CS)ME BIOS specification for given ME version
  - For ODROID it will be CSE 16.50 BIOS specification available under document number 730719.
  - For Kontron COMe TGL it will be CSME 15.0 BIOS specification available under document number 612229.

Evaluation using UEFITool

- UEFITool is able to parse the Boot Guard Manifests in the UEFI images.
  - Boot Guard protected regions are highlighted with red color
  - Yellow highlight indicates partially protected parts
- It can only tell us if the BIOS region of the image has been properly prepared to run with Boot Guard configured ME. It doesn't give 100% confidence Boot Guard is enabled, just a hint.

Sample view of Intel's Tiger Lake Reference Validation Platform BIOS image:

Intel Platform Trust Technology assessment



- Intel Platform Trust Technology is Intel's implementation of TPM 2.0 running in the ME firmware
  - Also referred to as fTPM (Firmware TPM)
- Integration with PCH allows for smaller board size and lower power consumption
- Better isolated due to lack of an exposed connection
  - dTPM with session encryption can provide adequate protection

Practice #302 Exercise #7

Assessment of fTPM application in CSME.

Analyze `MEInfo` output to check whether the fTPM enablement is possible on ODROID-H4+.

Practice #302 Exercise #8

Analysis of the binary blobs.

Based on information gained so far, what binary blobs and proprietary tools do you think have to go into firmware to provision Intel Boot Guard.

Conclusion

Let’s discuss the results of the assessment.

- Did we asses the whole boot chain?

Q&A