

Firmware SBoM, Annotations and Audits

Daniel Maslowski



Agenda



Fiedka Recap



Firmware Supply Chain Security



Annotation Workflows

Fiedka Recap

Fiedka the Firmware Editor











<https://fiedka.app/>

Fiedka the Firmware Editor



<https://fiedka.app/>

Features












-  analyze firmware images
-  visualize flash usage
-  explore file systems
 -  UEFI
 -  PSP (AMD)
 -  CBFS (coreboot)
-  remove UEFI files
-  embed LinuxBoot

Fiedka the Firmware Editor



<https://fiedka.app/>

Features

-  analyze firmware images
-  visualize flash usage
-  explore file systems
 -  UEFI
 -  PSP (AMD)
 -  CBFS (coreboot)
-  remove UEFI files
-  embed LinuxBoot
-  SBoM, SWID?
-  create annotations?
-  export meta data?

Firmware Supply Chain Security



Timeline

Timeline

2011

NIST: SP800-155 BIOS Integrity Measurement Guidelines (Draft)

[...] intended to facilitate the development of products that can detect problems with the BIOS [...]

Timeline

2011

NIST: SP800-155 BIOS Integrity Measurement Guidelines (Draft)

[...] intended to facilitate the development of products that can detect problems with the BIOS [...]

2021

TCG: PC Client Platform



FIM - Firmware Integrity Measurement



RIM - Reference Integrity Manifest

Timeline

2011

NIST: SP800-155 BIOS Integrity Measurement Guidelines (Draft)

[...] intended to facilitate the development of products that can detect problems with the BIOS [...]

2021

TCG: PC Client Platform



FIM - Firmware Integrity Measurement



RIM - Reference Integrity Manifest

Executive Order 14028 on Improving the Nation's Cybersecurity



includes a lengthy definition of SBOM

Buyers can use an SBOM to perform vulnerability or license analysis, both of which can be used to evaluate risk in a product.

Posted on May 12, 2021

A Host of CVEs



<https://binarily.io/advisories>

First post: July 15, 2021



Software Bill of Materials (SBOM)



Software Bill of Materials (SBOM)



SPEAKER —

DANIEL MASLOWSKI

SBOM ANNOTATIONS AND AUDITS

OPEN SOURCE FIRMWARE CONFERENCE

1 3 18

Richard Hughes
@hughsient

Replying to @osfc_io @_zaolin_ and @OrangeCMS

Hey @OrangeCMS -- I wondered if this would be coSWID or something different? If the former, I've got some test images you can use if that would be helpful.

6:08 PM · Jul 11, 2022 · Twitter Web App



Software Identification (SWID)

Software Identification (SWID)

<https://github.com/veraison/swid>

The swid package provides a golang API for manipulating Software Identification (SWID) Tags as described by ISO/IEC 19770-2:2015, NISTIR-8060, as well as by their “concise” counterpart CoSWID.

NISTIR 8060 Guidelines for the Creation of Interoperable Software Identification (SWID) Tags

<http://dx.doi.org/10.6028/NIST.IR.8060>

ISO/IEC 19770-2:2015 (not open/public, because ISO)

<https://www.iso.org/standard/65666.html>



Auditability?

Auditability?

```
firmware.bin has unknown extension, using uSWID
Found USWID header at offset: 2520032
Loaded:
uSwidContainer([uSwidIdentity(a9032c9d-2aaa-5a25-a0e6-6d865b24e6d2,0,coreboot,
bd34cca50aba130364f362618881693c0478a4a6):
uSwidLink(https://spdx.org/licenses/Apache-2.0.html,None)
uSwidLink(https://spdx.org/licenses/BSD-3-Clause.html,None)
uSwidLink(https://spdx.org/licenses/CC-BY-4.0.html,None)
uSwidLink(https://spdx.org/licenses/CC-BY-SA-3.0.html,None)
uSwidLink(https://spdx.org/licenses/GPL-2.0-only.html,None)
uSwidLink(https://spdx.org/licenses/GPL-2.0-or-later.html,None)
uSwidLink(https://spdx.org/licenses/GPL-3.0-only.html,None)
uSwidLink(https://spdx.org/licenses/GPL-3.0-or-later.html,None)
uSwidLink(https://spdx.org/licenses/ISC.html,None)
uSwidLink(https://spdx.org/licenses/MIT.html,None)
uSwidLink(https://spdx.org/licenses/X11.html,None)
uSwidLink(swid:9579af2b-39d8-59f1-ac5a-5b1fd4c03bd0,None)
uSwidLink(swid:e5a249ad-04bb-5b63-a587-ceb7b0e331c9,None)
uSwidLink(swid:23edb84c-5d68-544e-b389-8a67f6c80247,None)
uSwidLink(swid:8e0d0fd3-1116-50ad-ba5f-599c8117c42b,None)
uSwidEntity(9elements,9elements.com->TAG_CREATOR), uSwidIdentity(9579af2b-39d8-
59f1-ac5a-5b1fd4c03bd0,0,Intel Management Engine,None):
uSwidEntity(9elements,9elements.com->TAG_CREATOR), uSwidIdentity(e5a249ad-04bb-
5b63-a587-ceb7b0e331c9,0,Seabios,d239552ce7220e448ae81f41515138f7b9e3c4db):
uSwidEntity(9elements,9elements.com->TAG_CREATOR), uSwidIdentity(23edb84c-5d68-
544e-b389-8a67f6c80247,0,Intel-Microcode,2019-04-23):
uSwidEntity(9elements,9elements.com->TAG_CREATOR), uSwidIdentity(23edb84c-5d68-
544e-b389-8a67f6c80247,0,Intel-Microcode,2019-04-23):
uSwidEntity(9elements,9elements.com->TAG_CREATOR), uSwidIdentity(8e0d0fd3-1116-
50ad-ba5f-599c8117c42b,0,GCC,None):
uSwidEntity(9elements,9elements.com->TAG_CREATOR))
```



Auditability?

```
firmware.bin has unknown extension, using uSWID
Found USWID header at offset: 2520032
Loaded:
uSwidContainer([uSwidIdentity(a9032c9d-2aaa-5a25-a0e6-6d865b24e6d2,0,coreboot,
bd34cca50aba130364f362618881693c0478a4a6):
uSwidLink(https://spdx.org/licenses/Apache-2.0.html,None)
uSwidLink(https://spdx.org/licenses/BSD-3-Clause.html,None)
uSwidLink(https://spdx.org/licenses/CC-BY-4.0.html,None)
uSwidLink(https://spdx.org/licenses/CC-BY-SA-3.0.html,None)
uSwidLink(https://spdx.org/licenses/GPL-2.0-only.html,None)
uSwidLink(https://spdx.org/licenses/GPL-2.0-or-later.html,None)
uSwidLink(https://spdx.org/licenses/GPL-3.0-only.html,None)
uSwidLink(https://spdx.org/licenses/GPL-3.0-or-later.html,None)
uSwidLink(https://spdx.org/licenses/ISC.html,None)
uSwidLink(https://spdx.org/licenses/MIT.html,None)
uSwidLink(https://spdx.org/licenses/X11.html,None)
uSwidLink(swid:9579af2b-39d8-59f1-ac5a-5b1fd4c03bd0,None)
uSwidLink(swid:e5a249ad-04bb-5b63-a587-ceb7b0e331c9,None)
uSwidLink(swid:23edb84c-5d68-544e-b389-8a67f6c80247,None)
uSwidLink(swid:8e0d0fd3-1116-50ad-ba5f-599c8117c42b,None)
uSwidEntity(9elements,9elements.com->TAG_CREATOR), uSwidIdentity(9579af2b-39d8-
59f1-ac5a-5b1fd4c03bd0,0,Intel Management Engine,None):
uSwidEntity(9elements,9elements.com->TAG_CREATOR), uSwidIdentity(e5a249ad-04bb-
5b63-a587-ceb7b0e331c9,0,Seabios,d239552ce7220e448ae81f41515138f7b9e3c4db):
uSwidEntity(9elements,9elements.com->TAG_CREATOR), uSwidIdentity(23edb84c-5d68-
544e-b389-8a67f6c80247,0,Intel-Microcode,2019-04-23):
uSwidEntity(9elements,9elements.com->TAG_CREATOR), uSwidIdentity(23edb84c-5d68-
544e-b389-8a67f6c80247,0,Intel-Microcode,2019-04-23):
uSwidEntity(9elements,9elements.com->TAG_CREATOR), uSwidIdentity(8e0d0fd3-1116-
50ad-ba5f-599c8117c42b,0,GCC,None):
uSwidEntity(9elements,9elements.com->TAG_CREATOR))
```

We only have a list of names - claimed ingredients.
Imagine nutrition facts, but no lab verifying them.



Attestation

Attestation

The screenshot displays a web-based interface for managing firmware. On the left, a sidebar titled "Devices" lists several firmware entries, with "Lenovo System Firmware" selected. The main panel, titled "Firmware", shows the details for the selected device. It includes a "Device Properties" section with fields for Current Version (0.1.25), Minimum Version (0.0.1), Vendor (Lenovo), and Vendor ID (DMI:LENOVO). An "Attestation" section shows "Not OK" and a "Store" button. Below this is a "GUIDs" section with two entries: "main-system-firmware" and "Plugin Defined". The "Available Releases" section lists four versions of "ThinkPad L14 AMD / L15 AMD" firmware, each with a corresponding button: "Reinstall" for 0.1.25, "Downgrade" for 0.1.20, 0.1.19, and 0.1.15.

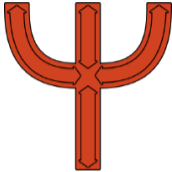
Current Version	0.1.25
Minimum Version	0.0.1
Vendor	Lenovo
Vendor ID	DMI:LENOVO

GUID	Value
main-system-firmware	230c8b18-8d9b-53ec-838b-6cfc6383493a
Plugin Defined	2afc1995-fa50-4d0d-a569-2d9133dc4950

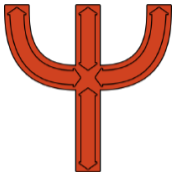
Version	Action
ThinkPad L14 AMD / L15 AMD 0.1.25	Reinstall
ThinkPad L14 AMD / L15 AMD 0.1.20	Downgrade
ThinkPad L14 AMD / L15 AMD 0.1.19	Downgrade
ThinkPad L14 AMD / L15 AMD 0.1.15	Downgrade



Platform System Interface

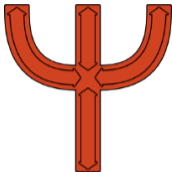


Platform System Interface



Goal: Derive a specification, summarizing firmware projects, their boot flows, how they interact as a platform with the actual operating system.

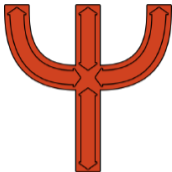
Platform System Interface



Goal: Derive a specification, summarizing firmware projects, their boot flows, how they interact as a platform with the actual operating system.

How: Extract features, compare approaches, reevaluate, improve.

Platform System Interface



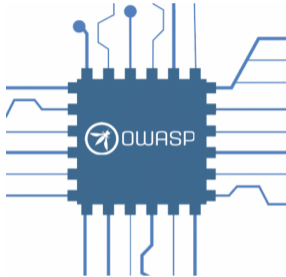
Goal: Derive a specification, summarizing firmware projects, their boot flows, how they interact as a platform with the actual operating system.

How: Extract features, compare approaches, reevaluate, improve.

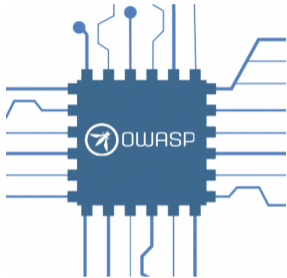
Example: **Auditable Firmware Implementation**

<https://github.com/platform-system-interface/psi-spec/issues/4>

OWASP Firmware Security Testing Methodology



OWASP Firmware Security Testing Methodology



1. Information gathering and reconnaissance
2. Obtaining firmware
3. Analyzing firmware
4. Extracting the filesystem
5. Analyzing filesystem contents
6. Emulating firmware
7. Dynamic analysis
8. Runtime analysis
9. Binary Exploitation

<https://scriptingxss.gitbook.io/firmware-security-testing-methodology/>

Annotation Workflows

Ghidra

```
*****  
*                               FUNCTION                               *  
*****  
void K2_Right_Click(pointer sender, pointer e)  
  <VOID>      <RETURN>  
  Stack[0x4]:4 sender  
  Stack[0x8]:4 e  
  .NET CLR Managed Code  
  K2_Right_Click  
004042d8 7e 4a 01      db[66]  
          00 04 1a  
          33 01 2a ...  
0040431a 36      MethodDe... L.S. Bits 0:1 Flags, Bits 2:7 Si...  
  
*****  
*                               FUNCTION                               *  
*****  
void FormMain_Load(pointer sender, pointer e)  
  <VOID>      <RETURN>  
  Stack[0x4]:4 sender  
  Stack[0x8]:4 e  
  .NET CLR Managed Code  
  FormMain_Load  
0040431b 02 7b 55      db[13] This loads the main form of the |...  
          00 00 04  
          02 6f 27 ...  
00404328 36      MethodDe...  
00404328 36      db      36h      Size+Flags L.S. Bits 0:1 Flag...
```

Right click -> Comment -> EOL Comment -> Type -> Apply ...



Fiedka

undefined bytes, 100 files
7CB8BDC9-F8EB-4F34-AAEA-3EE4AF6516A1 (9E21FD93-9C72-4C15-8C4B-E77F1DB2D792)

FC510EE7-FFDC-11D4-BD41-0080C73C8881

type: EFI_FV_FILETYPE_FREEFORM
size:
checksum:
blocks used: NaN

DxeCore

ReportStatusCodeRouterRuntimeDxe

I don't know what this is, probably removable. Let's see!

PcdDxe

DepEx

> EFI_DEVICE_PATH_UTILITIES_PROTOCOL_GUID
guid: 80CF7257-87AB-47F9-A3FE-D50B76D89541
type: EFI_FV_FILETYPE_DRIVER
size:
checksum:
blocks used: NaN

RuntimeDxe

DepEx

> EFI_PCD_PROTOCOL_GUID
> EFI_DEVICE_PATH_UTILITIES_PROTOCOL_GUID
guid: B601F8C4-43B7-4784-95B1-F4226CB40CEE
type: EFI_FV_FILETYPE_DRIVER
size:
checksum:
blocks used: NaN

SecurityStubDxe

DepEx

> EFI_PCD_PROTOCOL_GUID
> EFI_DEVICE_PATH_UTILITIES_PROTOCOL_GUID
guid: F80697E9-7FD6-4665-8646-88E33EF71DFC
type: EFI_FV_FILETYPE_DRIVER
size:
checksum:
blocks used: NaN

Legacy8259

DepEx

> EFI_PCD_PROTOCOL_GUID
guid: 245CB4DA-8E15-4A1B-87E3-9878FFA07520
type: EFI_FV_FILETYPE_DRIVER
size:
checksum:
blocks used: NaN

CpuIo2Dxe

DepEx

> EFI_PCD_PROTOCOL_GUID
guid: A19B1FE7-C1BC-49F8-875F-54A5D542443F
type: EFI_FV_FILETYPE_DRIVER
size:
checksum:
blocks used: NaN

CpuDxe

DepEx





> EFI_PCD_PROTOCOL_GUID
> EFI_DEVICE_PATH_UTILITIES_PROTOCOL_GUID
guid: 1A1E4886-9517-440E-9FDE-3BE44CEE2136
type: EFI_FV_FILETYPE_DRIVER
size:
checksum:
blocks used: NaN

Click on notepad button and type!

Data





Data

Why data?

-  enrich analysis, exchange, gain insight
-  feed back into tooling, e.g., Binaryly integrated in LVFS
-  back claims, e.g., regarding security and financial risks
-  data drives business decisions

Data





Why data?

-  enrich analysis, exchange, gain insight
-  feed back into tooling, e.g., Binary integrated in LVFS
-  back claims, e.g., regarding security and financial risks
-  data drives business decisions

Remember Sigsum, transparency logs? That's data.

Data

Why data?



-  enrich analysis, exchange, gain insight
-  feed back into tooling, e.g., Binarly integrated in LVFS
-  back claims, e.g., regarding security and financial risks
-  data drives business decisions

Remember Sigsum, transparency logs? That's data.

Previous Work


Mimoja's Firmware Toolkit for unpacking and analyzing firmware images

<https://github.com/mimoja/mft>

-  fetchers for obtaining lots of images
-  analyzers for different vendors

Mimoja's Firmware Toolkit (MFT)

Used in Fiedka prototype (utk-web) - help wanted with reintegration!

 **Daniel Maslowski aka CyReVolt** 🦋
@OrangeCMS

Time for a break for today. For a preview, check the mft branch on GitHub, or look here directly:
hostile.education/utk-web/A3MSTX...
The output from MFT has more meta data, yes! :-)

localhost:3000/A3MSTX_3_60.mft

Jump to Dir: 0xa8000 0xa7000 0x188000 0x4f0000 0xb8000 0x3a0000 0x168000 0x4d0000 0x268000 0x63f000

type: PSP	magic: SPSP	address: 0xa8000	18 files
AMD_PUBLIC_KEY		PSP_FW_BOOT_LOADER	
address: 0xff0a9000	version: 0.0.0.0	address: 0xff2c0000	version: 0.5.0.45
hash	signature: B7AESD81	hash	signature: FFFFFFFF
size: 576	signing key: 08000800	size: 32768	signing key: C25D8C55
signed: 1435262402	encryption key: 4813CBDD	signed: 30876	encryption key: 00000000
uncompressed	blocks used: 1	uncompressed	blocks used: 9
packed		packed	
SMU_OFFCHIP_FW		PSP_FW_RECOVERY_BOOT_LOADER	
address: 0xff2c8000	version: 0.0.0.0	address: 0xff0aa000	version: 0.5.0.45
hash	signature: FFFFFFFF	hash	signature: FFFFFFFF
size: 81920	signing key: 00000000	size: 24576	signing key: C25D8C55
signed: 78095	encryption key: 00000000	signed: 22908	encryption key: 00000000
uncompressed	blocks used: 21	uncompressed	blocks used: 7
packed		packed	

10:51 PM · Jan 24, 2021 · Twitter Web App



Call to Action

Let's collect data, build up a knowledge base, and see what we can derive from it!



Call to Action

Let's collect data, build up a knowledge base, and see what we can derive from it!

Tracking Issue

<https://github.com/fiedka/fiedka/issues/69>

