Our Journey, from SBOM to ASSBOMB

Davor Frkat, Martin Schmiedecker

Disclaimer

- Our view is limited to our organisation, Bosch Engineering
- Bosch is huge, and you might get different answers for asking similar questions
- Bosch Engineering is <1% of our global workforce

This presentation:

- Images are links to the source
- Crypto means cryptography

BOSCH

Who?

Davor Frkat:

- Security Engineer
- Background in electrical engineering & telecommunications
- Ping me on LinkedIn

Martin Schmiedecker:

- Security Engineer and court certified expert witness
- Background in digital forensics & online privacy
- Ping me at @Fr333k@infosec.exchange

Located in Vienna, Austria

Bosch Engineering
Facts and figures

100%
Bosch

1999
founding

3,200
associates

Active at
>20
locations
in >10 countries

0 – 24
cylinders

125
years of
motorsports

1, 2, 3...
Small-batch development

800 +
customer projects
per year

BOSCH

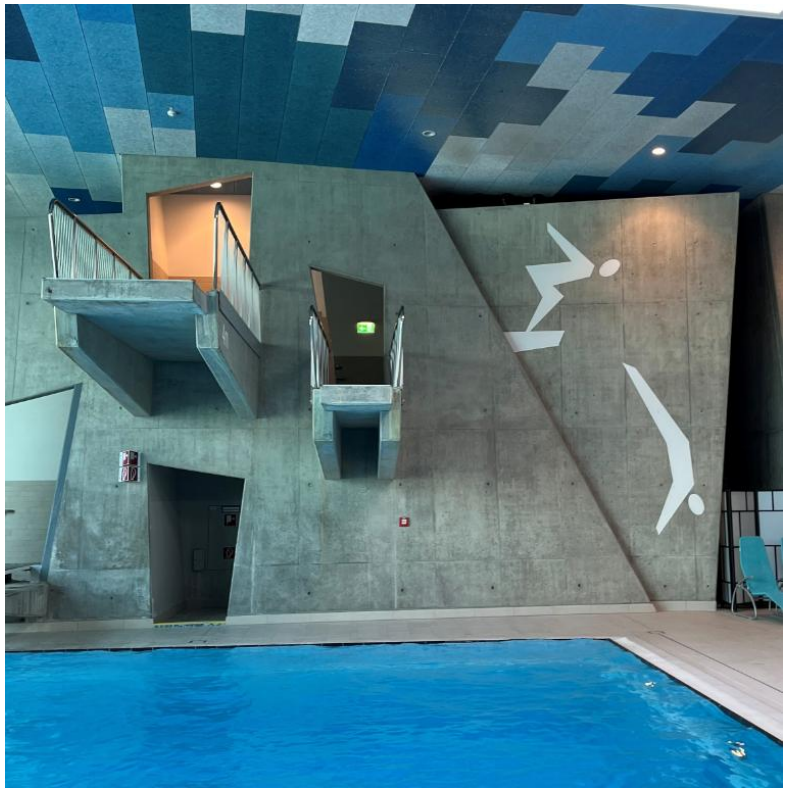
What is *ASSBOMB*?

{*}bill of materials:

- SBOM, HBOM, CBOM, TBOM, AIBOM, SaaSBOM, …
- Soooo many confusing abbreviations!
- People seem to just constantly invent new ones

Here comes the ASSBOMB:

- We made our own: ***„Automotive Security and Software Bill of Material & Beyond“***
- (This is not a real thing!)



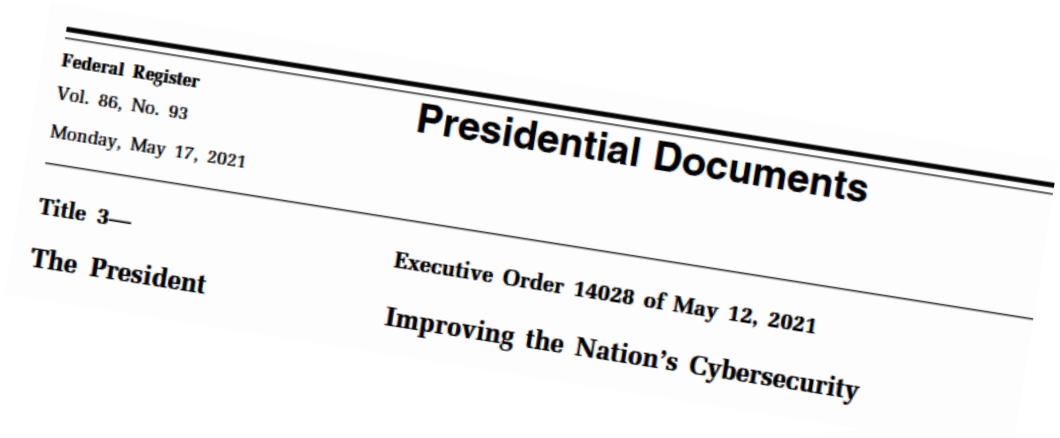
BOSCH

Why SBOMs

- CRA, that‘s why!
- White House Executive Order 14028
- Connected vehicles without Chinese software
- Automotive standard ISO/SAE 21434
- OT security IEC 62443
- Machinery regulation EU 2023/1230

Most importantly:

- Our customers ask for it
- We ask our suppliers for it


Federal Register
Vol. 86, No. 93
Monday, May 17, 2021

Presidential Documents

Title 3—

The President

Executive Order 14028 of May 12, 2021

Improving the Nation's Cybersecurity


EXECUTIVE OFFICE OF THE PRESIDENT
OFFICE OF MANAGEMENT AND BUDGET
WASHINGTON, D.C. 20503

THE DIRECTOR

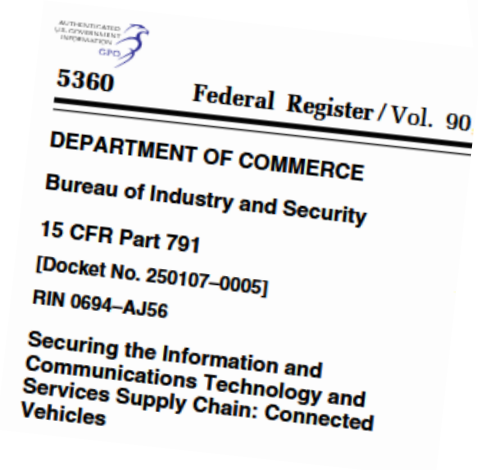
January 23, 2026

M-26-05

MEMORANDUM TO THE HEADS OF EXECUTIVE DEPARTMENTS AND AGENCIES

FROM: Russell T. Vought
Director

SUBJECT: Adopting a Risk-based Approach to Software and Hardware Security


5360 Federal Register / Vol. 90

DEPARTMENT OF COMMERCE

Bureau of Industry and Security

15 CFR Part 791

[Docket No. 250107–0005]

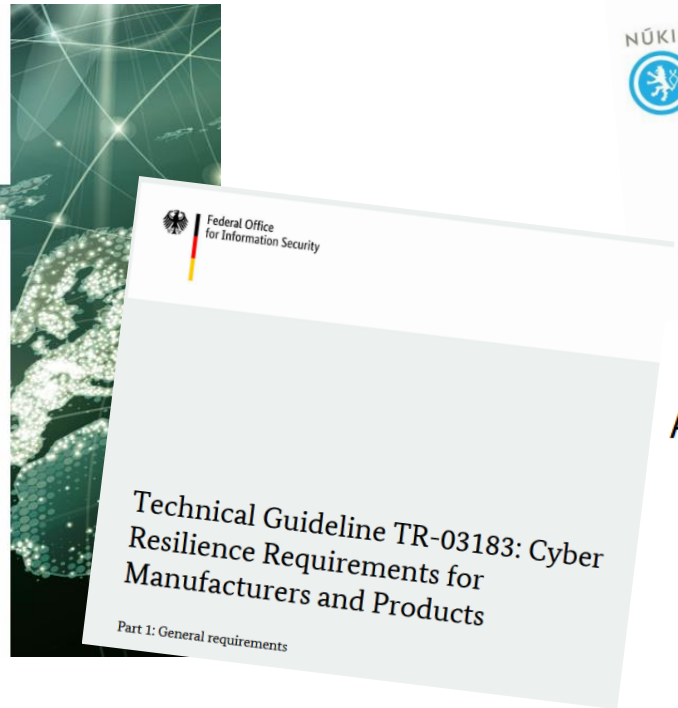
RIN 0694–AJ56

Securing the Information and Communications Technology and Services Supply Chain: Connected Vehicles

BOSCH

So Much Paper Produced Already

- SBOMs, they are everywhere!
- ENISA, CISA, BSI, and many more



Auto-ISAC Software Bill of Materials (SBOM) Informational Report TLP:CLEAR

Auto-ISAC

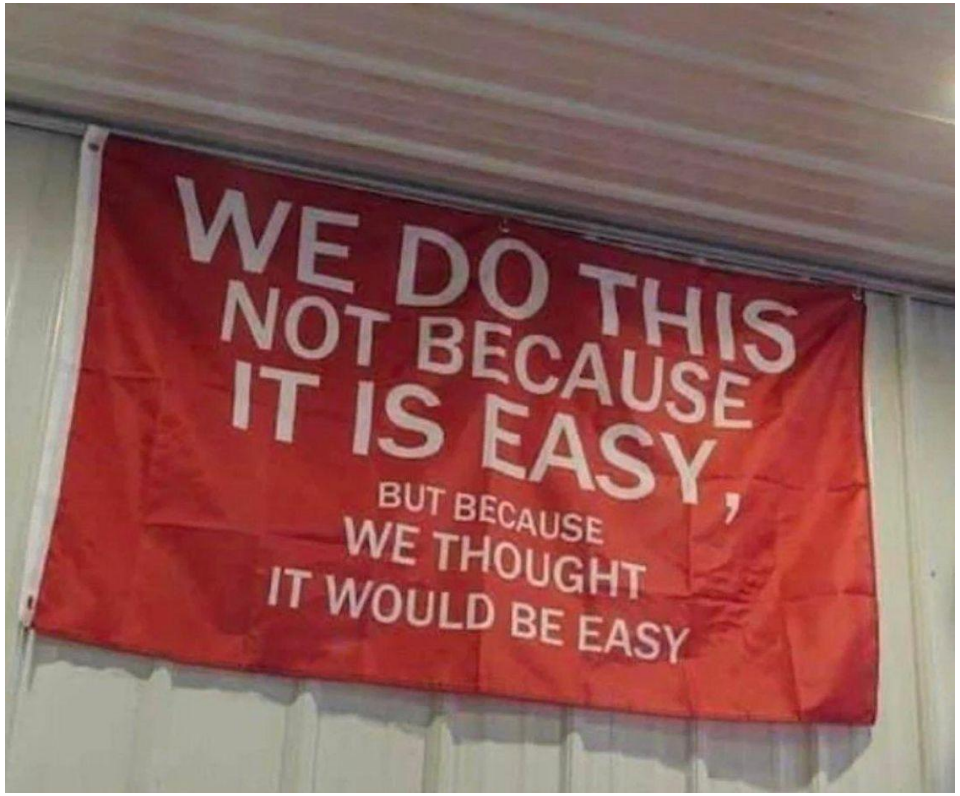
January 17, 2025

BOSCH

Our Timeline

We volunteered for this:

- ~2018: starting to work on this topic
- ~2020: beginning to collecting SBOMs for projects
- 2022: our first ISO/SAE 21434 certification
- 2024: Cyber Resilience Act
- 2026: Netz- und Informationssicherheitsgesetz (NIS2)
- 2027: CRA in full force



Cyber Resilience Act - CRA

- Currently main driver in EU for SBOM implementation

  Annex I, Part II, (1)
  identify and document vulnerabilities and components contained in products with digital elements, including by drawing up a software bill of materials in a commonly used and machine-readable format covering at the very least the top-level dependencies of the products;

- Reporting obligations for manufacturers

  Chapter 2, Article 14
  A manufacturer shall notify any actively exploited vulnerability contained in the product with digital elements that it becomes aware of simultaneously to the CSIRT designated as coordinator, in accordance with paragraph 7 of this Article, and to ENISA. The manufacturer shall notify that actively exploited vulnerability via the single reporting platform established pursuant to Article 16

Cyber Resilience Act - CRA

- Currently main driver in EU for SBOM implementation

Annex I, Part II, (1)

identify and document vulnerabilities and components contained in products with digital elements, including by drawing up a software bill of materials in a commonly used and machine-readable format covering at the very least the top-level dependencies of the products;

NEED SBOM

- Reporting obligations for manufacturers

Chapter 2, Article 14

A manufacturer shall notify any actively exploited vulnerability contained in the product with digital elements that it becomes aware of simultaneously to the CSIRT designated as coordinator in accordance with paragraph 7 of this Article, and to ENISA. The manufacturer shall notify that actively exploited vulnerability via the single reporting platform established pursuant to Article 16.

FIND & REPORT VULNERABILITIES

Cyber Resilience Act - CRA

- **Product classes:** default, important and critical products
  - Default: self declaration of conformity (CE)
  - Important class I (self declaration*) & class II (notified body): e.g., OS, Firewalls, µC
  - Critical (notified body): e.g., smart cards, smart meters
  - Explicit exclusion for product types already regulated

- Essential requirements for all products, even though risk-based approach is mandatory
- Critical and/or important products → a **notified body** might be interested in your SBOMs
- Vulnerability monitoring for the lifetime of the product (or at least 5 years)

- What if not followed? **High fin€s! EU market entry?**

* For harmonized standards

What Did We Observe

Our six years of SBOMs:

- Proprietary software requires tailored solutions
- Organically grown software repositories are very tricky
- Consent on format, naming and SBOM details is hard to achieve
- Quality management is your friend (no really!)
- Proprietary software will seldomly match against public sources. Doesn’t mean you can lean back!

Example, from Last Week

Synopsis DWC2 USB controller:

- iOS SecureROM exploit
- Maybe biggest news since *unc0ver* and *checkm8*

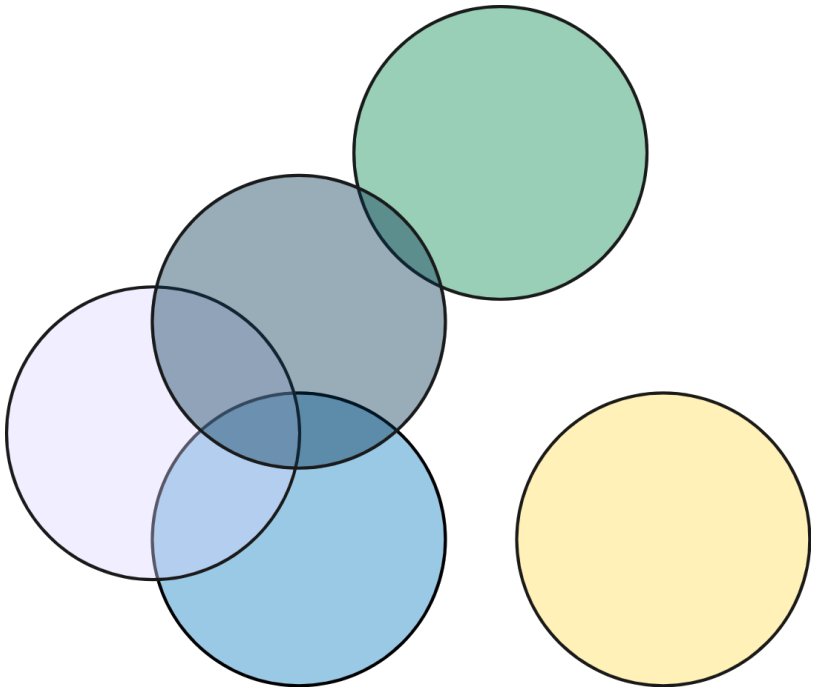
Tricky part:

- Do we have products with this chip?
- Do we have other chips, with similar vulnerability?
- Really popular: ESP32, Raspberry Pi, and more!
- I mean, the PoC is out there …

What Did We Observe

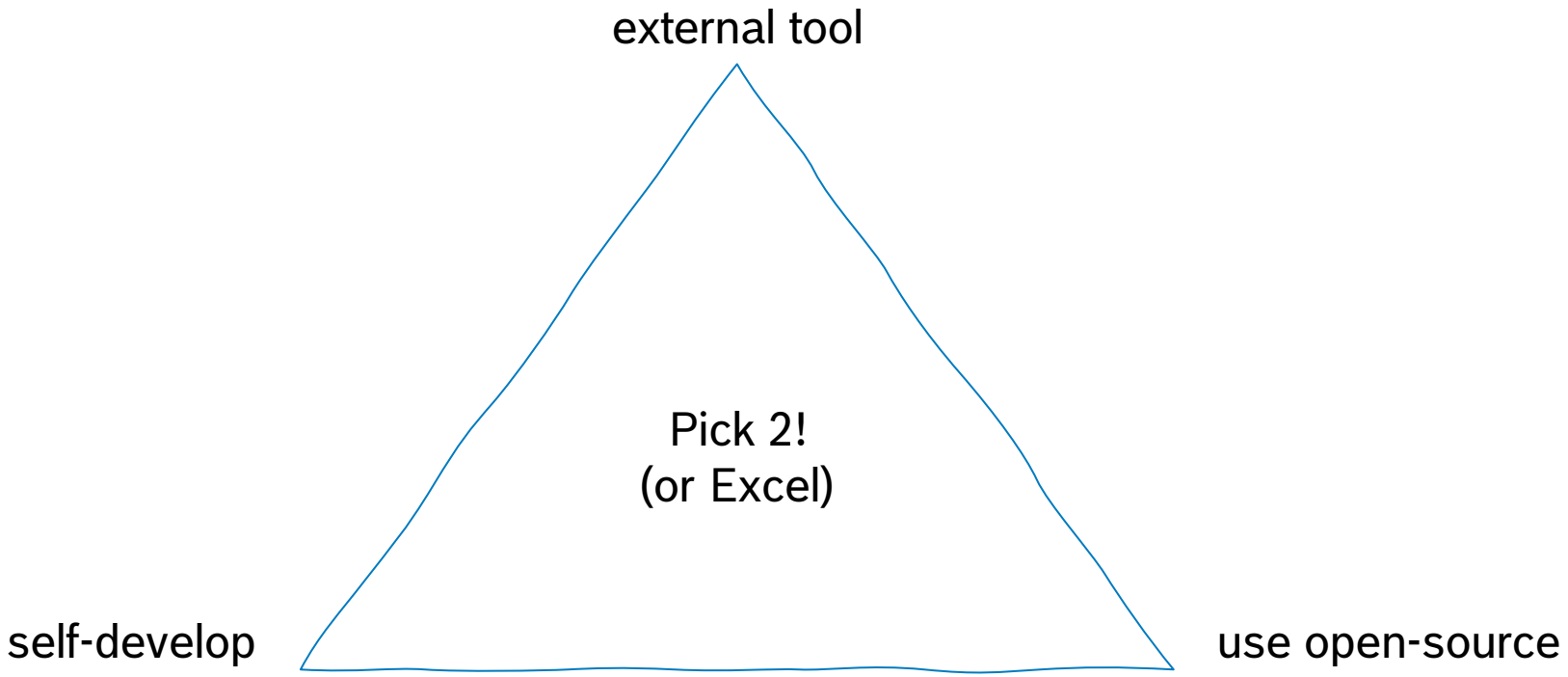
Some of the hardest challenges:

- Hidden „dependencies“: component A is reused under different name and context in software B
- Repackaged software without traces
- Consistent CI/CD integration
- Supplier Management
- But not for open-source software
- Governance



Triad of Tooling

Tooling can be hard!



Consistently Naming Things

- Almost everything we do has e.g. some kind of bootloader
- Everything with AUTOSAR has AUTOSAR components in it
- Versioning & releases are important
- Murphy's law of SBOM: every manual field will be filled inconsistently

Most easy solution: Common Platform Enumeration (CPE)

- cpe:2.3:a:bosch:some_bootloader:127_string:*:*:*:*:*:*:*:*

More elegant: Package-URL (PURL)

- pkg:bosch.swelement/com/bosch/engineering/SOME_NAMESPACE/SOME_COMPONENT/...

Additional Metadata

Trade-off for additional metadata:

- CryptoBOM, ToolBOM, ...
- They can help!
- But: edge use-cases, with plenty of effort required
- Stay pragmatic!!

Example usbliter8:

- Is pointer authentication enabled?
- Where to draw the line on configurations

False Positives & False Negatives

Alert fatigue:

- So much data, so many notifications
- Dead code or unused libraries – context matters!
- Dependency on certain configurations, hardware variants & used features
- Linux kernel CVEs
- Negotiating risks is often a natural reflex

Imagine no new vulnerabilities come into your SBOM tool

- No stress
- No problems
- But: never trust a green dashboard. The background script might have crashed :)

Data to Work On

What we are currently working with:

- NIST NVD
- CISA KEV
- Bosch Vulnerability Database
- Newsfeeds
- **Automation**, where possible

dependency track

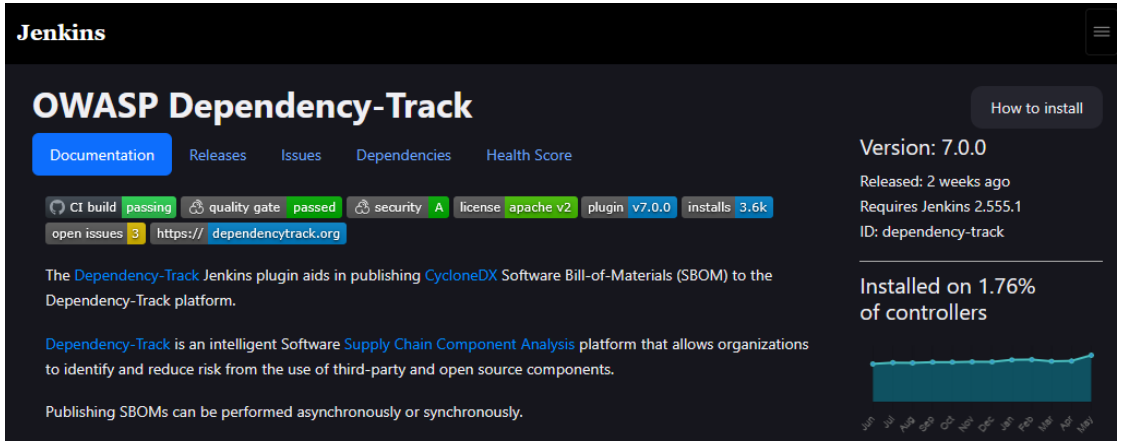
But:

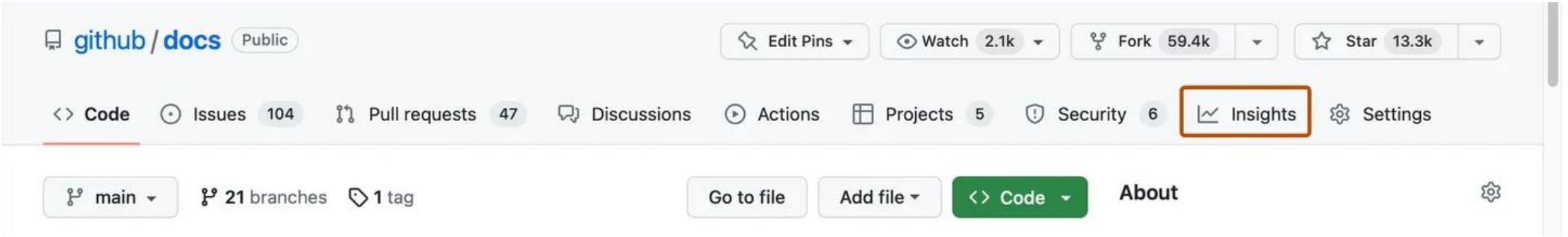
- Highly diverse software development environments
- Greenfield environments not common

Exporting Software Parts

Highly dependent on used toolchains:

- Jekins Dependency Track Plugin
- Yocto sbom export
- *npm audit*
- Github Dependency Graph





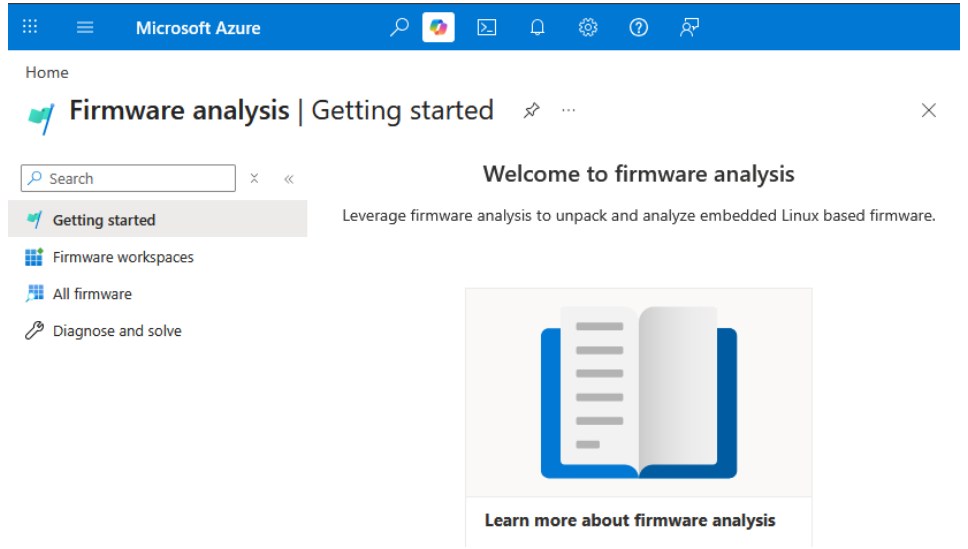
Scanning for Software Parts

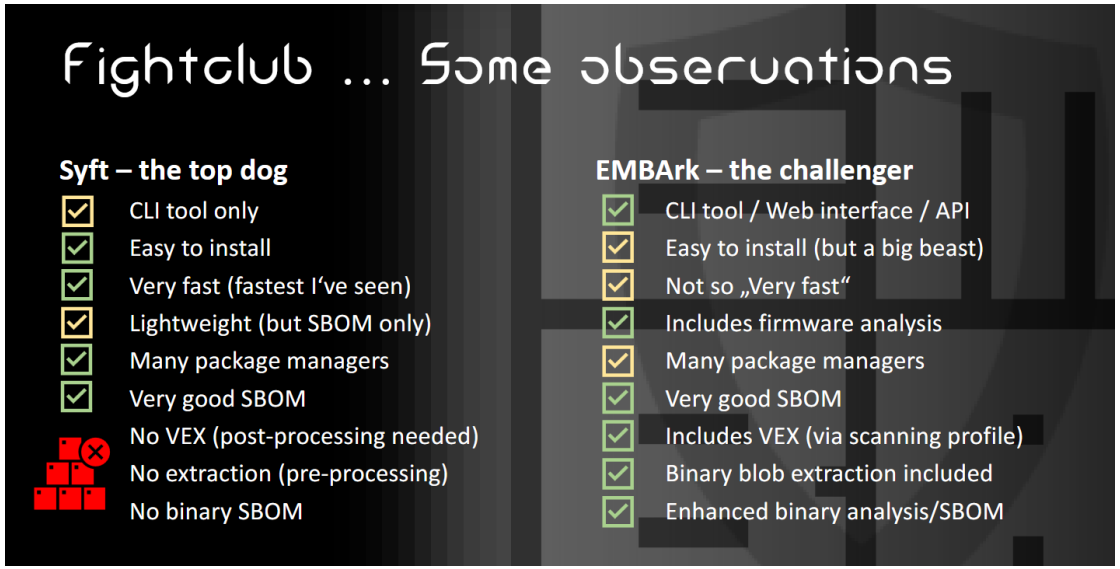
Sometimes you have no source code:

- Yet still responsible!
- Binary scanning to the rescue
- EMBA, Syft, Microsoft Firmware Analysis, …
- Commercial products

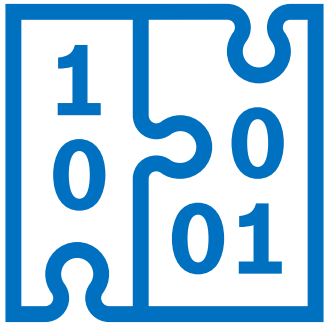
But:

- Not often allowed
- Not often preferred solution
- Not working for proprietary software





Alternative SBOM approaches



Maybe in the future, for embedded software:

- binary pattern matching
- Integration in CI/CD pipeline

Solution for:

- Challenge of hidden dependencies
- Legacy software without existing SBOM information

If you haven’t started yet … like, at all!

- Understand your products!
- Talk to your suppliers
- Push for consistent naming and versioning
- Find yourself someone to do the CRA reporting

Long-game:

- Automate!
- Get a PSIRT
- Become a CNA

We Know The Complications

CVE data enrichment:

- Tricky, but not unsolvable
- Completeness is really hard

But note:

- High quality SBOMs are not a given thing
- Engineers will do engineering things
- Often without the full picture, or context

But what about AI???

Why is AI not helping?

- You can’t replace talking to people with AI
- Needs to be methodical & complete
- Templates & reproducibility
- Maybe some vibe-coding

Really, the best results we had so far:

- Tags per product category
- Like i.e. which other projects is the same thing, more or less

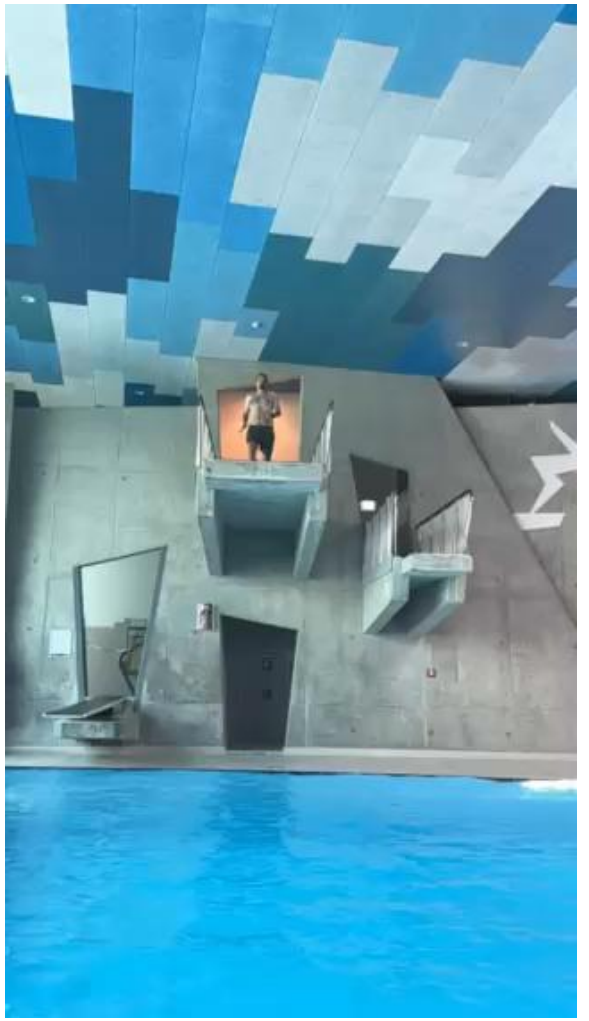
Outlook

The obvious:

- Reporting timeline beginning 2026-09-11 for CRA/EU

We want more!

- ENISA KEV catalogue
- Linux Kernel CVEs
- Global CVE initiative: gcve.eu
- Chinese databases: CNVD, CNNVD, and CAVD (for automotive type approvals)



Summary

There is real momentum:

- Time is ticking
- CRA enacted since 2024
- Automate all the things!
- Understand you products!

Chat us up, what is your perspective?

Thanks for listening!!

Bosch Engineering
Our contacts – let’s talk!

Davor Frkat

Security engineer

+43 676 6580362

davor.frkat@bosch.com

linkedin.com/in/priordice/

Martin Schmiedecker

Security engineer

martin.schmiedecker@bosch.com

https://infosec.exchange/@fr333k

You can find more information on our website at **bosch-engineering.com**.