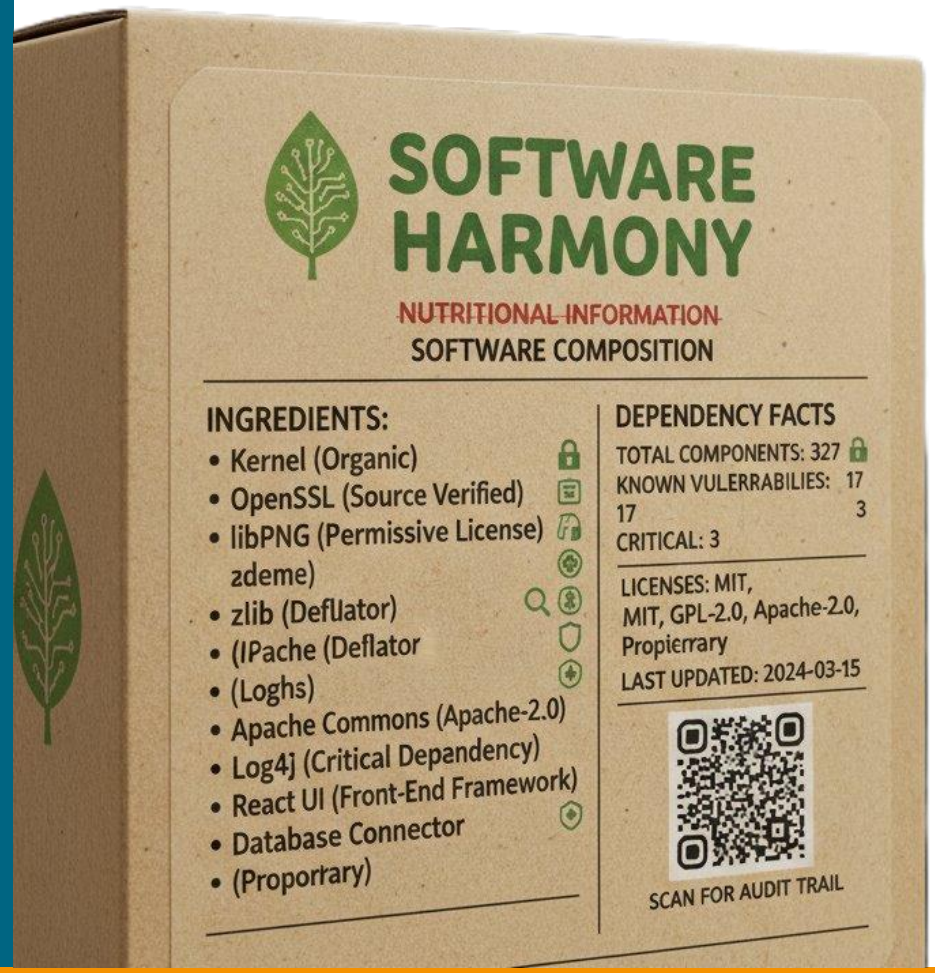
Einsatz von SBOM´s in der Lieferkette

AK CRA Kunden-/Lieferanten-beziehung: Security Supply Chain

Maximilian Korff, Siemens AG

Heiko Mahr, Balluff GmbH

Was ist eine Software Bill of Material (SBOM) und warum ist sie Security relevant?



SBOM ist eine Liste aller Komponenten und Abhängigkeiten, die in einer Software enthalten sind – und sind mit Stücklisten (BOMS), wie sie in Fertigung oft verwendet werden, verwandt

SBOMs liegen i.d.R. als XML oder JSON Datei vor

und sind lesbar von Mensch und Maschine.

es gibt 2 wesentliche Formate: SPDX und Cyclone DX.

SBOMs werden eingesetzt für

- Prüfung von Lizenzkonformität
- Risikomanagement für z.B. CRA
- Identifizierung von Schwachstellen

Was ist Inhalt einer SBOM?

SBOM enthält

- alle wesentlichen Informationen zu den verwendeten Softwarekomponenten einer Firmware/Software.
- die Lizenzbedingungen der Komponenten z.B. von FOSS Komponenten, die das Produkt nutzt
- die Darstellung der Beziehungen zwischen den Komponenten
- Metadatenebene mit zusätzlichen Informationen wie Lizenzen und Herkunft.

Die SBOM enthält KEINEN Code!

SBOM ermöglichen Einsicht – z.B. in die Struktur & Abhängigkeiten

Beispielvisualisierung einer SBOM Erstellt mit:

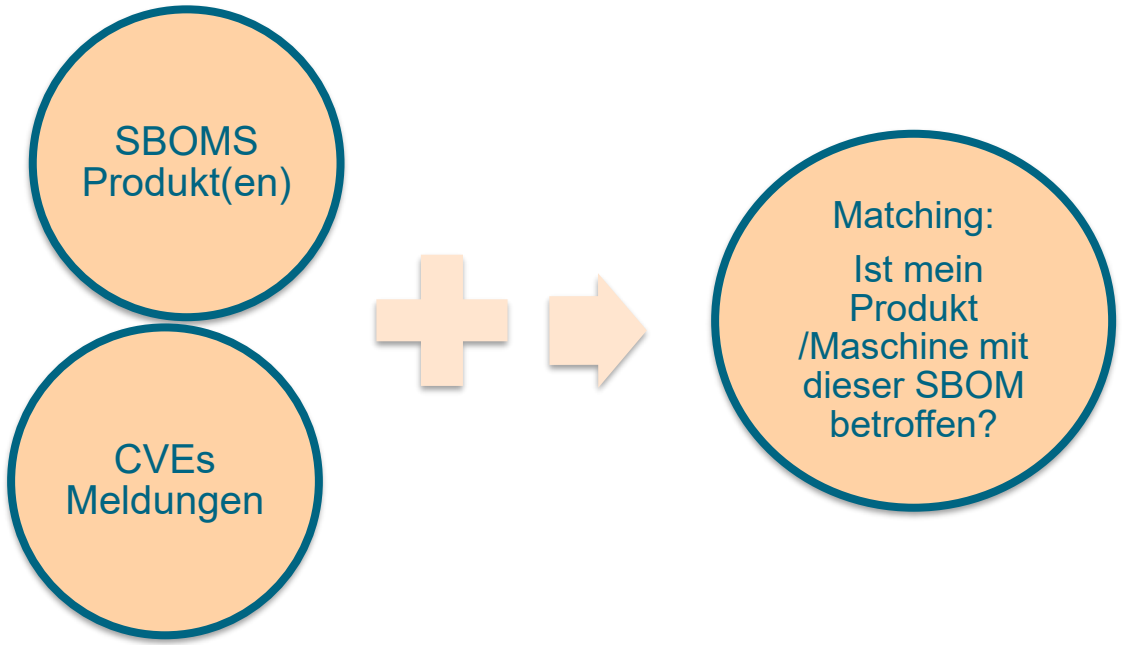
Sunshine - SBOM visualization tool

jedes Segment = 1 Komponente .

Innen nach aussen:

hierarchische Ansicht der Abhängigkeitsstruktur,

Schwachstellenmonitoring mit SBOMs



- **SBOMS der Produkte/FOSS liegen vor**
- **Schwachstellenmeldungen eingesammelt (PSIRT, CERT@VDE..)**
- **Matching in einem Monitoringsystem ?**

➔ Warnung, wenn Schwachstellen in meinem Produkt oder Komponente vorhanden sind

Beginn des Schwachstellenmanagementprozess :

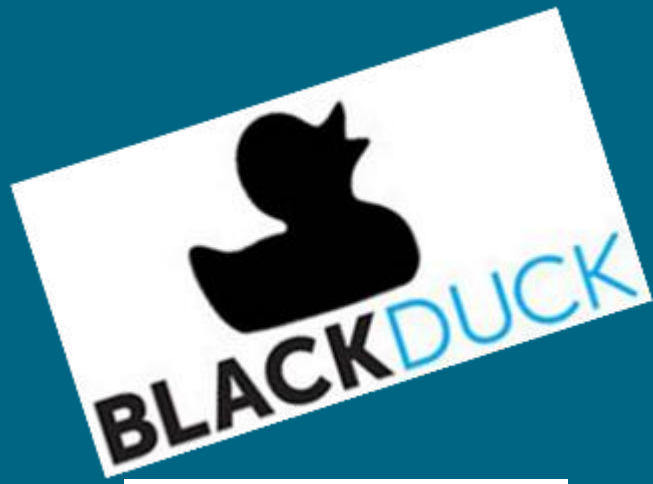
- **Ggf. Information an Behörden**
- **Beurteilen, wie ich damit umgehe**
- **Beheben der Schwachstelle**
- **Offenlegung der Schwachstelle**

Toolvorschläge für Schwachstellenmonitoring

Open Source

anchore

Kommerziell



Beispiel eines Schwachstellen Monitorings (Dependency Track)

Herausforderungen für Komponentenhersteller

Verantwortlich, dass Gerät/Stack/Software frei von ausnutzbaren Schwachstellen ist

➔ Kompatibilität der SBOMS & Vorkomponenten– welches Format, welche Vollständigkeit?

➔ Vielzahl der Produkte managen

➔ Pflegeaufwand für die SBOMs

➔ Ggf. Prüfung/Qualifizierung neuer Versionen von Vorkomponenten

➔ Monitoringsystem und Fachkraft für Monitoring notwendig

➔ Free und Open Source Komponenten: Schwachstellen muss ich evtl. selbst beheben, bzw. sogar im Projekt eintragen

➔ Wie gehe ich mit einem Produkt um, von dem wir wissen, dass eine Schwachstelle in den Abhängigkeiten vorhanden ist, aber der Lieferant noch nicht gemeldet hat ?

- Braucht es eine Meldung auch, wenn eine Schwachstelle nicht ausnutzbar ist?

Herausforderung als Betreiber

Diverse Maschinen unterschiedlicher Hersteller

– Jede Maschine ist aus Betreiber Sicht 1 Produkt mit digitaler Komponente.

➔ Holschuld: Information über Schwachstellen und Updates beim Lieferanten abholen, aber Updates müssen mit eigenem Personal installiert werden. (IEC 62443-2-3)

➔ Betreiber muss im Kontext seiner Anlage prüfen, ob Patch verträglich ist

➔ Rückgriff auf Hersteller macht evtl. Service-Level Verträge notwendig

➔ Aber: dann keine eigene Wartung mehr möglich, auch defekte Komponenten müssten durch Hersteller ausgetauscht werden

Herausforderungen für Integratoren -1

Neu durch CRA: Maschine / Software muss frei von ausnutzbaren Schwachstellen sein.

SBOMs für eigenen Code, für Free & Open Source Anteile, für Komponenten notwendig

- ➔ **Kompatibilität der SBOMS – woher bekommen, welches Format, welche Vollständigkeit?**
- ➔ **Pflegeaufwand für die eigene SBOMs**
- ➔ **Monitoringsystem und Fachkraft für Monitoring/PSIRT notwendig**
- ➔ **Kosten für Updates beim Kunden: Updates müssen kostenfrei zur Verfügung gestellt werden.**
  - ➔ Was, wenn die Schwachstelle in der Hardware der Maschine steckt?

Herausforderungen für Integratoren -2

- **Free und Open Source Komponenten: Schwachstellen muss ich evtl.. selbst beheben, bzw. sogar im Software-Projekt eintragen**
  - Werde ich ungewollt Free & Open Source Stewart?
- **Wie gehe ich mit einem Produkt um, von der wir wissen, dass eine Schwachstelle in den Abhängigkeiten vorhanden ist, aber der Lieferant sie noch nicht gemeldet hat ?**
  - Brauchte es eine Meldung auch, wenn eine Schwachstelle nicht ausnutzbar ist?
- **Wie kann ich Abarbeitung von Schwachstellenmeldungen meiner Lieferanten / FOSS automatisieren?**
  - Beispiel Linux Kernel: 91 Schwachstellen vom 1-10. September 2025
  - Workload steigt, da mehr und mehr Schwachstellen gemeldet werden
  - Empfehlungen:
    - CSAF Format für automatisierten Import
    - ISDuBA Tool vom BSI/Fraunhofer – automatisiertes Bewerten von CSAFs
    - Renovate Open Source Tool für automatisiertes Updaten der Dependcies

https://wid.cert-bund.de/portal/wid/kurzinformationen

Security Advisories

91 Datensätze

1 2 50 pro Seite RSS-Feed

Stand	Risiko	CVSS Base	ID	Titel	betroffene Produkte	CVE	Status	Mitigation
01.09.2025 - 10.09.2025				Linux Kernel				
10.09.2025, 13:36		8.6	WID-SEC-2025-1465	Linux Kernel: Mehrere Schwachstellen ermöglichen Denial of Service	Debian Linux, Amazon Linux 2, Red Hat Enterprise Linux ...	CVE-2025-38177, CVE-2025-38178, CVE-2025-38179 ...	UPDATE	JA
10.09.2025, 13:36		5.5	WID-SEC-2025-1270	Linux Kernel: Mehrere Schwachstellen ermöglichen Denial of Service	Google Container-Optimized OS, Debian Linux, Amazon Linux 2 ...	CVE-2025-38000, CVE-2025-38001, CVE-2025-38002 ...	UPDATE	JA
10.09.2025, 13:36		4.4	WID-SEC-2025-1098	Linux Kernel: Mehrere Schwachstellen ermöglichen nicht spezifizierten Angriff	Debian Linux, Google Cloud Platform, Amazon Linux 2 ...	CVE-2025-37890, CVE-2025-37891	UPDATE	JA
10.09.2025, 13:36		8.6	WID-SEC-2025-0698	Linux Kernel: Mehrere Schwachstellen	Google Container-Optimized OS, Debian Linux, Amazon Linux 2 ...	CVE-2025-21987, CVE-2025-21988, CVE-2025-21989 ...	UPDATE	JA
10.09.2025, 07:32		8.6	WID-SEC-2025-0861	Linux Kernel: Mehrere Schwachstellen	Debian Linux, Google Cloud Platform, Amazon Linux 2 ...	CVE-2020-36789, CVE-2021-47668, CVE-2021-47669 ...	UPDATE	JA
10.09.2025, 07:32		8.6	WID-SEC-2025-0844	Linux Kernel: Mehrere Schwachstellen	Debian Linux, Amazon Linux 2, Red Hat Enterprise Linux ...	CVE-2023-53034, CVE-2024-58093, CVE-2024-58094 ...	UPDATE	JA

Wie geht der Weg von der BOM zur SBOM in der Maschine?

BOM Erstellen für digitale Komponenten erstellen

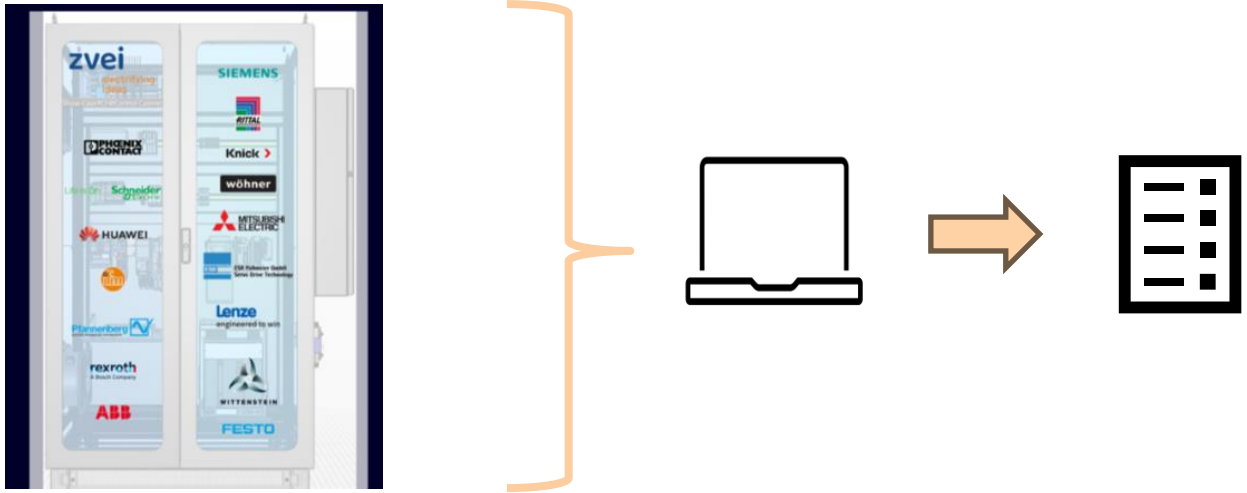
* Quelle 1 aus Engineering Daten (E-Plan, CAD/CAM, TIA Portal,…
* Quelle 2: aktiver Scan der Maschine (PRONETA, Balluff Engineering Tool..)

Ergebnis: BOM aller Komponenten liegt als Liste vor – inklusive Daten wie HW, FW Version, Produktname, Seriennummer, MAC Adressen u.ä. direkt aus der Komponente.

Damit: Konvertierung in Cyclone DX (oder SPDX)

➔ Grundlegende SBOM für die Maschine

➔ Anreicherung um Einträge für selbsterstelle Software/Firmware Anteile



Handlungsempfehlungen

➔ Frühzeitiges Auseinandersetzen mit den Themen rund um SBOM

- Verständnis über Schwachstellenhandling generell aufbauen
- Welche Tools passen zu meiner Entwicklungsumgebung und meinen Prozessen
- Wie tief mache ich selbst SBOMs und in welcher Tiefe will ich SBOMs an meine Kunden weitergeben

➔ Frühzeitig an Standardisierung denken

- Klären des Formats (Cyclone DX von BSI favorisiert, SPDX)
- Vereinbarung mit Lieferanten treffen
- Verbinden mit technischer Dokumentation des Produktes und Archivierung

Weiterführende Informationen

Bundessicherheitsamt für Informationssicherheit (BSI)

Technische Richtlinie des BSI zur SBOM

Technische Richtlinie des BSI zum CSAF Format - TR-03191

CSAF Org

Common Security Advisory Framework (CSAF) Org

VDMA

Übersicht zum Thema Cybersecurity beim VDMA