Beyond the Build: Why SBOMs are essential for modern application security

June 2025

Anthony Harrison - anthony@aph10.com

Who am I?

40 years delivering mission critical solutions across multiple sectors

Founder and Director APH10

Co-founder SBOMEurope

Open Source Software

Mentor

CoderDojo

Log4Shell
tm

Intentionally corrupt versions uploaded to public GitHub repository

Trigger infinite loops causing a denial of service

27 million weekly downloads

So who likes Software vulnerabilities?

Particularly on a Friday afternoon….

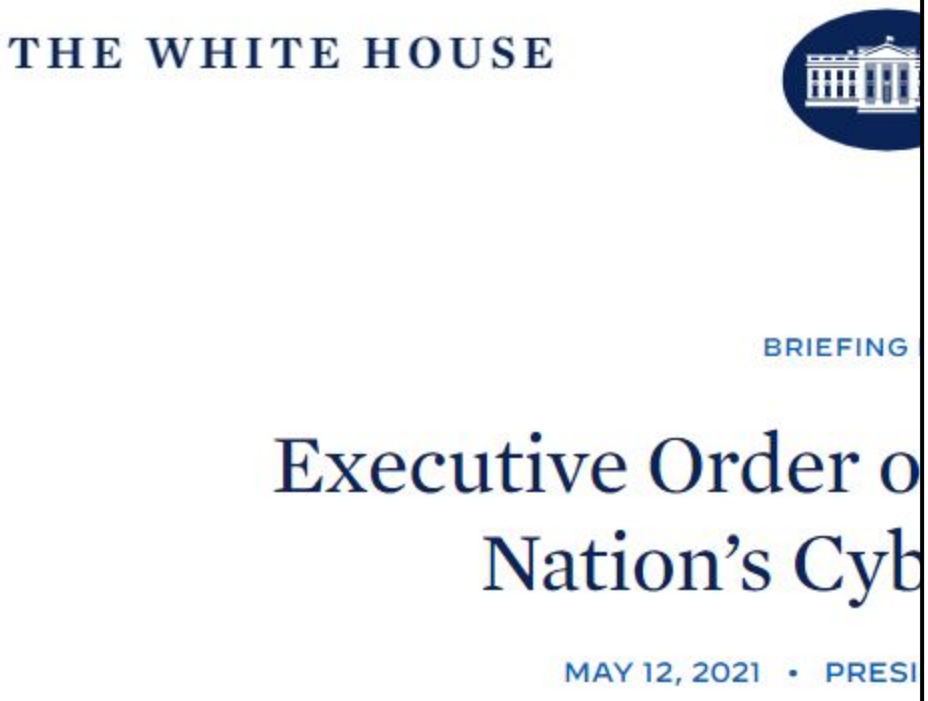
NIST
National Institute of Standards and Technology
U.S. Department of Commerce

TLP:WHITE

Defending Against Software Supply Chain Attacks

April 2021

Regulation



THE WHITE HOUSE

BRIEFING

Executive Order o
Nation's Cyb

MAY 12, 2021 • PRESI

- Improve Software Supply Chain Security
  - The EO will improve the security of software by establishing baseline security standards for development of software sold to the government, including requiring developers to maintain greater visibility into their software and making security data publicly available.
  - It also creates a pilot program to create an "energy star" type of label so the government — and the public at large — can quickly determine whether software was developed securely.

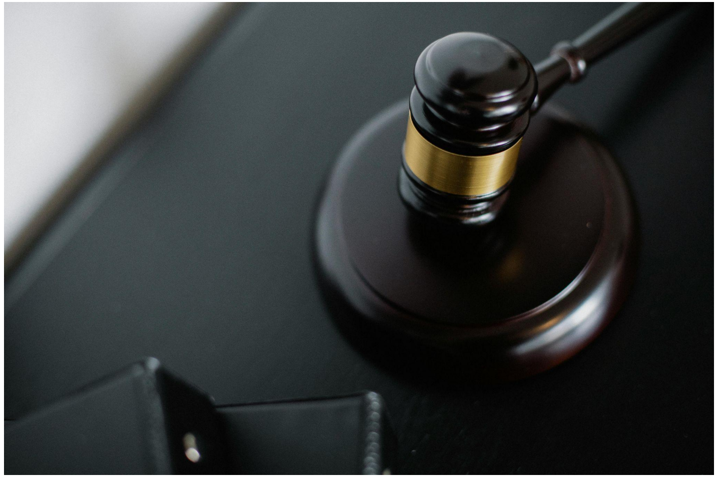
What is the CRA?

EU Cyber Resilience Act (CRA)

Defines the legislative framework of **essential cybersecurity requirements** that must be met when placing **any product with digital elements into the EU market**.

Addresses **cyber risk** in digital products to **protects users of product.**

Applies **throughout the life-cycle** of the product.



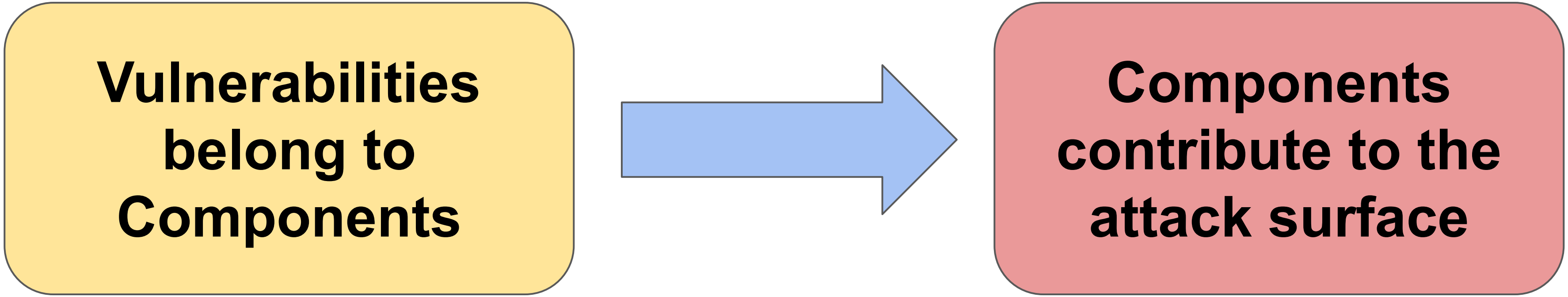
CRA Requirements

“*be made available on the market without known* ***exploitable*** *vulnerabilities*”

(CRA Annex I Part 1 2(a))

Doesn’t all software have defects?

But not all defects become vulnerabilities



Vulnerabilities belong to Components

Components contribute to the attack surface

So how do you know what Components you are using?

Software Bill of Materials (SBOMs)

Understanding the supply chain of software, obtaining an SBOM, and using it to analyze known vulnerabilities are crucial in managing risk.

White House Executive Order 14028, Section 10(j)

Understanding the Software Supply Chain

THE APACHE SOFTWARE FOUNDATION

ECLIPSE FOUNDATION

THE LINUX FOUNDATION

Microsoft

aws

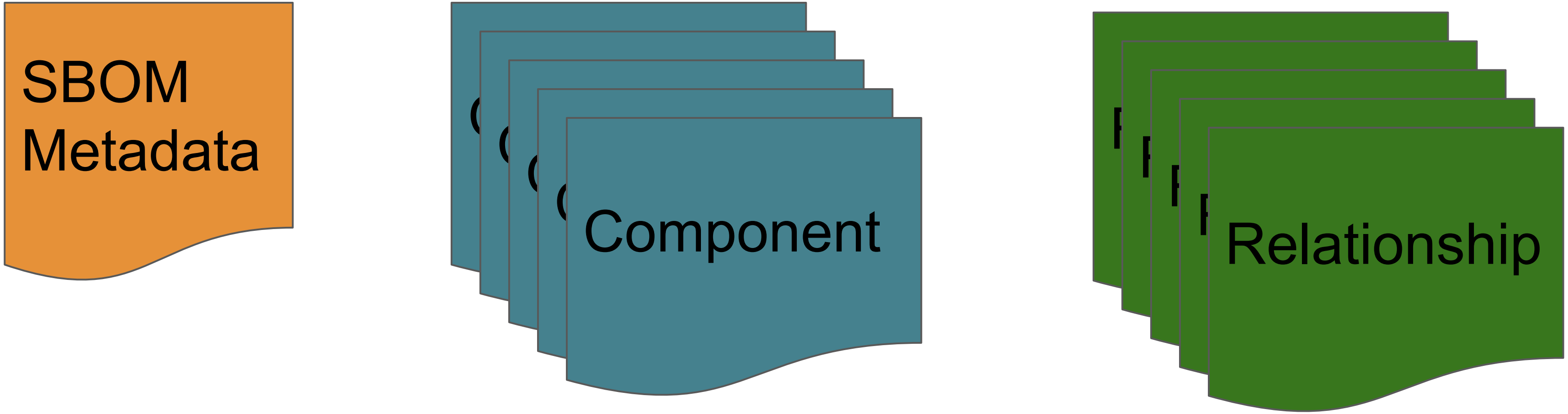
What is an SBOM?

➔ A SBOM is a **formal set of machine-readable metadata** describing your software application

➔ SBOMs are designed to be **shared within and across organisations**

➔ SBOMs form an important part of a software **risk strategy**

SBOM Content

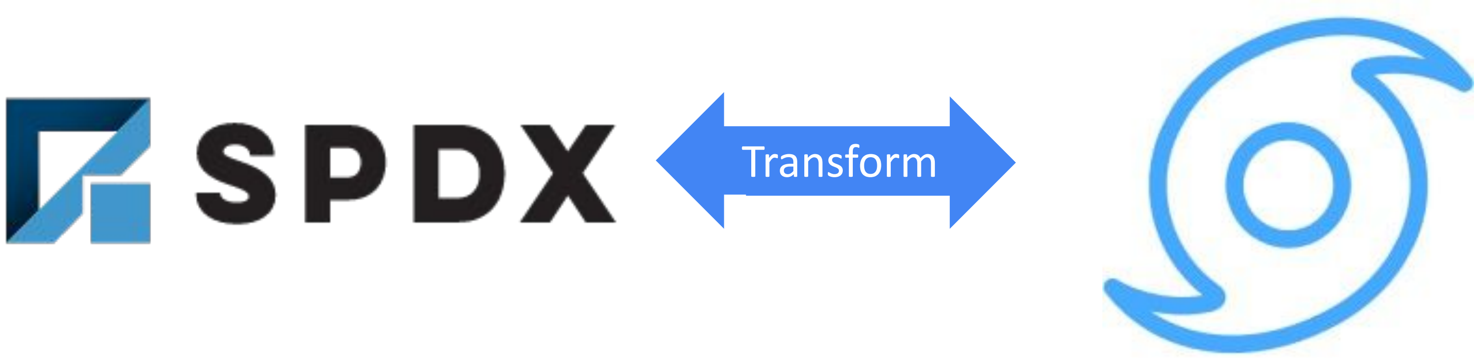


SBOM Metadata

Component

Relationship

Two primary standards and formats….



https://spdx.org/

https://cyclonedx.org/

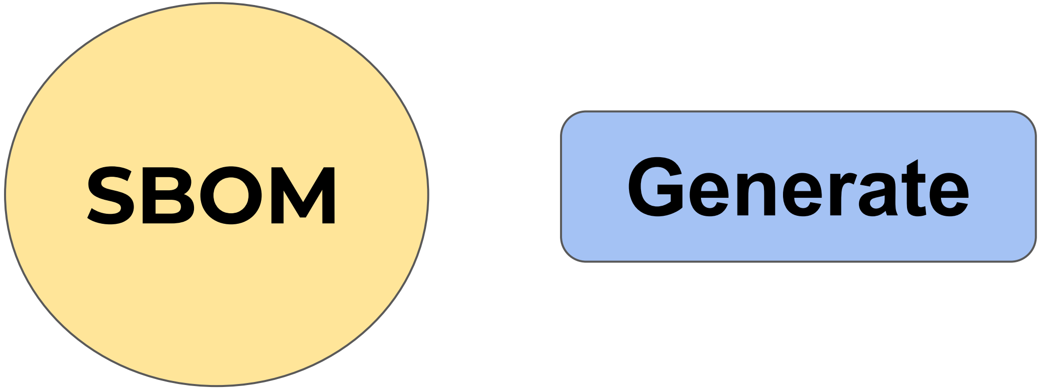
ISO 5962:2021
(version 2.2)

ECMA-424
(version 1.6)

Six Types of SBOMs

- Design
  - Represents the software to be produced
- Source
  - Created from source repository. Often produced as part of SCA tool chain
- Build
  - Represents a releasable product resulting from a build process (e.g. an executable)
- Analysed
  - Represents a set of products e.g., executables, packages, containers, and virtual machine images after a build
- Deployed
  - Represents an inventory of all artefacts installed onto a system
- Runtime
  - Identifies the components executing within a system

SBOM Lifecycle



SBOM

Generate

Generating SBOMs

	Source	Build	Deployed
lib**4sbom**	Y	Y	Y
sbom**4files**	Y		
sbom**4python**	Y	Y	
sbom**4js**		Y	
sbom**4php**		Y	
sbom**4rust**		Y	
sbom**4windows**			Y
distro**2sbom**			Y

Apache-2.0 Licence. Work with both CycloneDX and SPDX

SOURCE SBOM

```
FileName: ./cli.py
SPDXID: SPDXRef-File-2-cli
FileChecksum: SHA1: bc045c292bac3175f80e9b2a0101c32cf5ac4792
FileChecksum: SHA256: dac51da8c922e22d8007152e29dd0fafa7fbef90e59e9f06ec1c2c98ff58f138
FileChecksum: SHA512:
225a4c66726a6c6c5185bbb1326e23343dac15530debbae028e45913b4cc66c13cf5fa6adf1213e96422f42db
FileType: SOURCE
FileType: TEXT
LicenseConcluded: Apache-2.0
LicenseInfoInFile: Apache-2.0
LicenseComments: <text>This information was automatically extracted from the file.</text>
FileCopyrightText: <text>(C) 2023 Anthony Harrison</text>
```

https://pypi.org/project/sbom4files/

BUILD SBOM

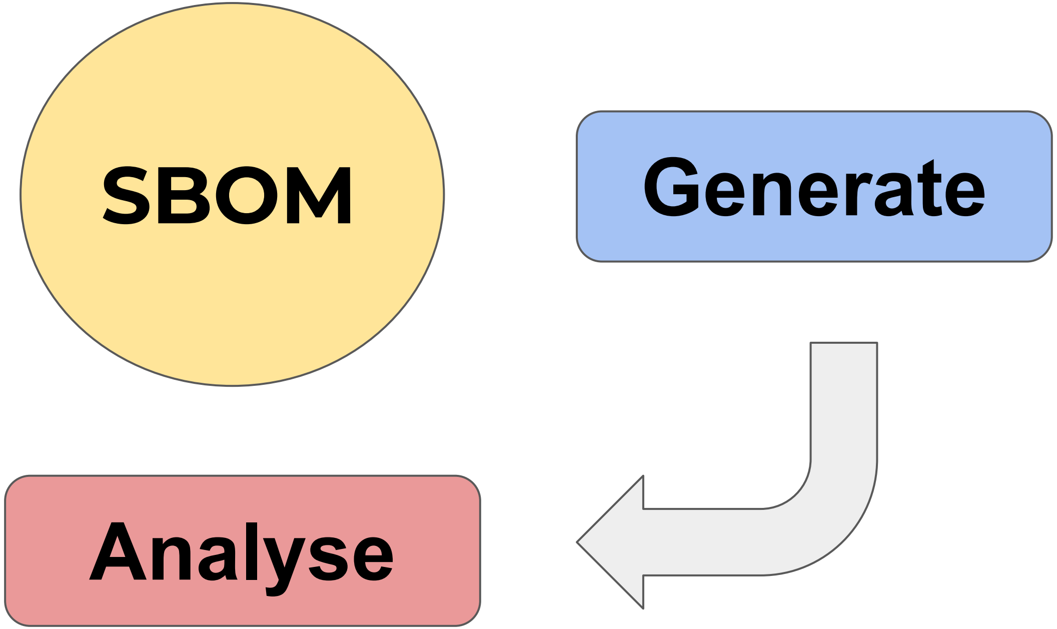
```
PackageName: sbom4files
SPDXID: SPDXRef-Package-6-sbom4files
PackageVersion: 0.2.2
PrimaryPackagePurpose: LIBRARY
PackageSupplier: Person: Anthony Harrison (anthony.p.harrison@gmail.com)
PackageDownloadLocation: https://pypi.org/project/sbom4files/0.2.2
FilesAnalyzed: false
PackageHomePage: https://github.com/anthonyharrison/sbom4files
PackageLicenseDeclared: Apache-2.0
PackageLicenseConcluded: Apache-2.0
PackageCopyrightText: NOASSERTION
PackageSummary: <text>SBOM generator for files in a directory</text>
ExternalRef: PACKAGE-MANAGER purl pkg:pypi/sbom4files@0.2.2
ExternalRef: SECURITY cpe23Type
cpe:2.3:a:anthony_harrison:sbom4files:0.2.2:*:*:*:*:*:*:*
```

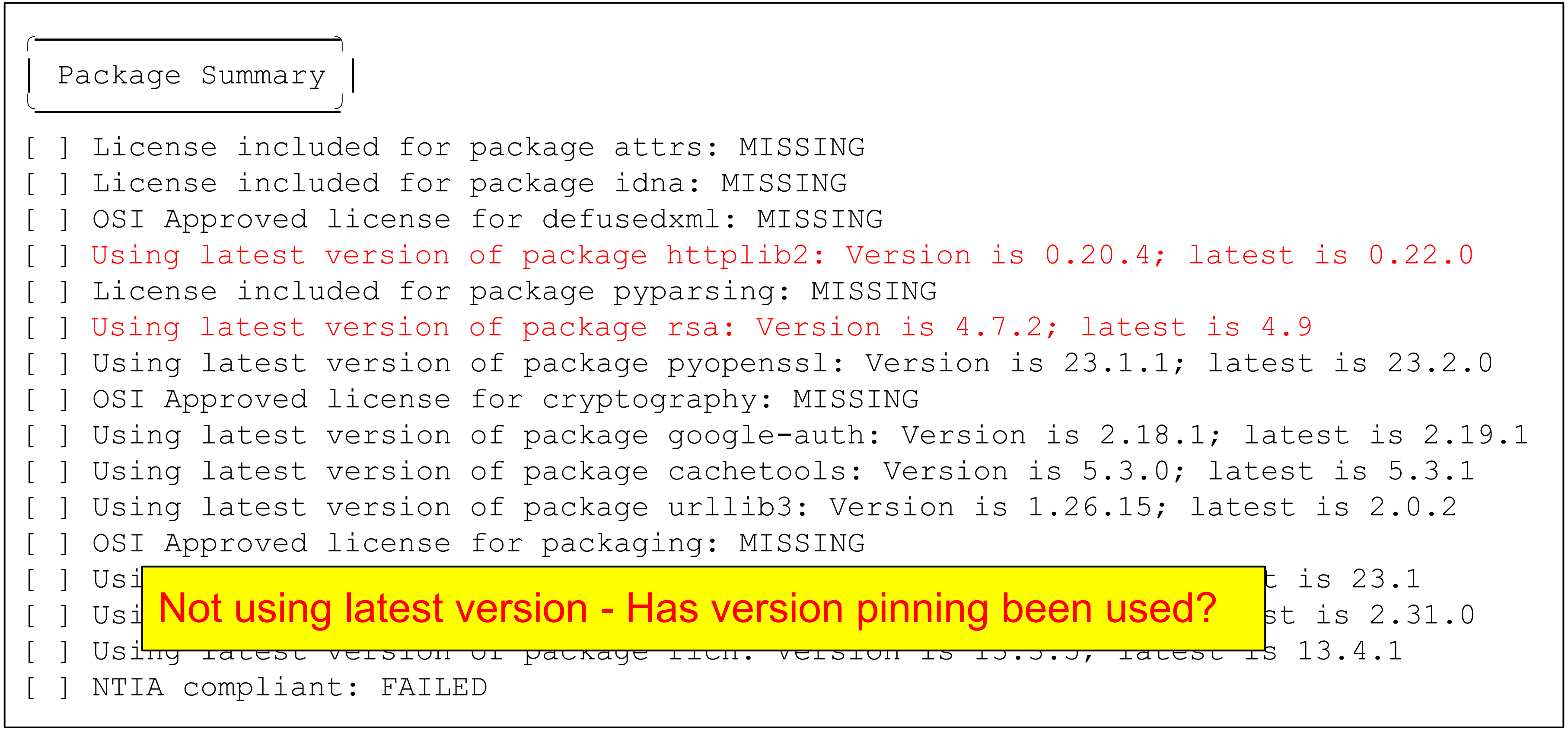
https://pypi.org/project/sbom4python/

SBOM Generation Differences

	Ground Truth	Tool A	Tool B	Tool C	Tool D
Image	447	471	617	0 (Failed)	1
Container	204	149	152	148	148
Application	36	38	44	40	45

SBOM Lifecycle





```
╭───────────────────╮
│  Package Summary  │
╰───────────────────╯
[ ] License included for package attrs: MISSING
[ ] License included for package idna: MISSING
[ ] OSI Approved license for defusedxml: MISSING
[ ] Using latest version of package httplib2: Version is 0.20.4; latest is 0.22.0
[ ] License included for package pyparsing: MISSING
[ ] Using latest version of package rsa: Version is 4.7.2; latest is 4.9
[ ] Using latest version of package pyopenssl: Version is 23.1.1; latest is 23.2.0
[ ] OSI Approved license for cryptography: MISSING
[ ] Using latest version of package google-auth: Version is 2.18.1; latest is 2.19.1
[ ] Using latest version of package cachetools: Version is 5.3.0; latest is 5.3.1
[ ] Using latest version of package urllib3: Version is 1.26.15; latest is 2.0.2
[ ] OSI Approved license for packaging: MISSING
[ ] Usi                                                          t is 23.1
[ ] Usi                                                          st is 2.31.0
[ ] Using latest version of package rich: Version is 13.3.5; latest is 13.4.1
[ ] NTIA compliant: FAILED
```

Not using latest version - Has version pinning been used?

https://pypi.org/project/sbomaudit/

```
[VERSION] google-auth: Version changed from 2.18.1 to 2.19.0
[VERSION] cachetools: Version changed from 5.3.0 to 5.3.1
[VERSION] urllib3: Version changed from 1.26.15 to 1.26.16
[VERSION] requests: Version changed from 2.30.0 to 2.31.0

Summary
-------
Version changes:   4
License changes:   0
Removed packages:  0
New packages:      0
```

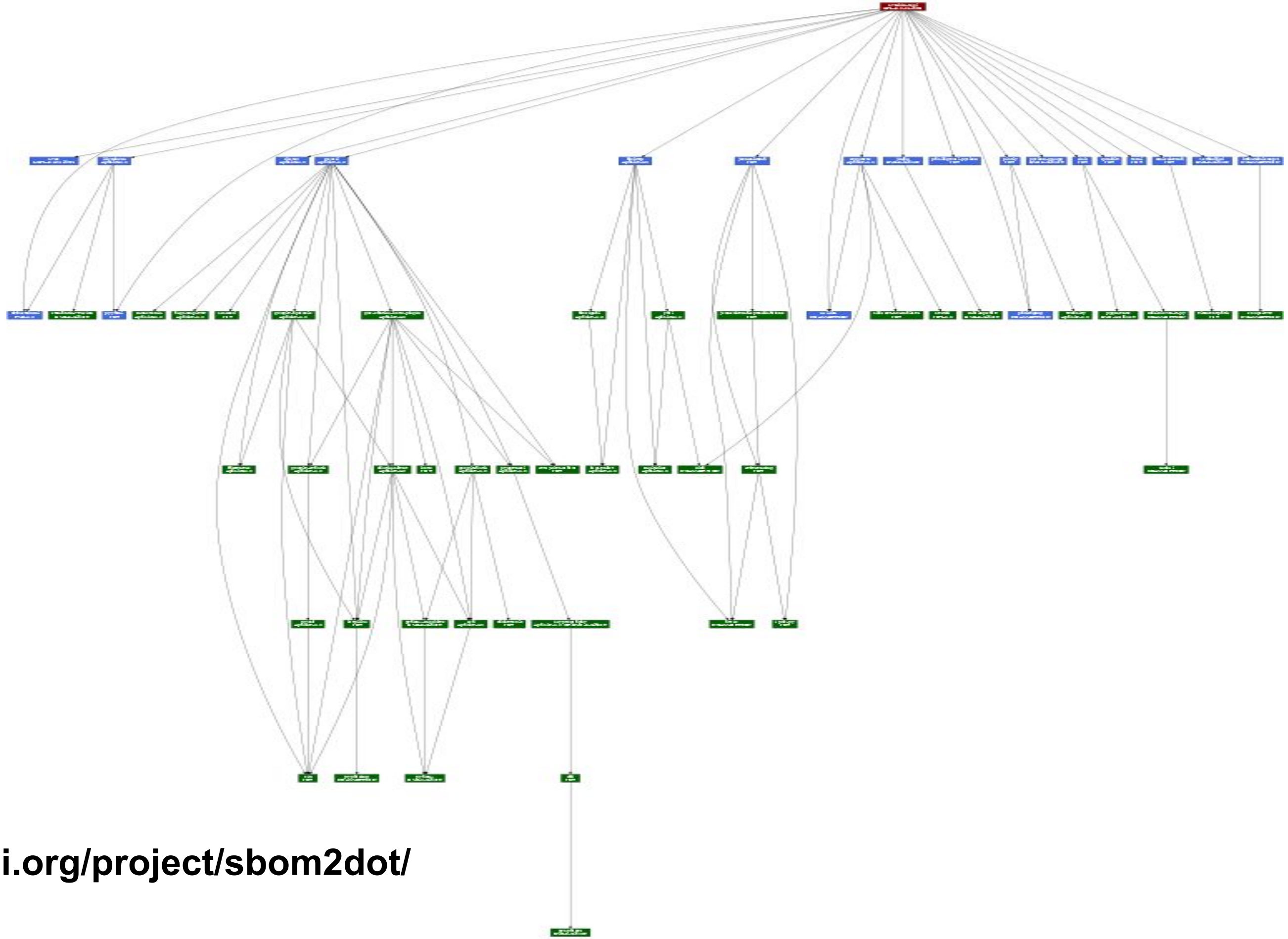
https://pypi.org/project/sbomdiff/

Dependencies

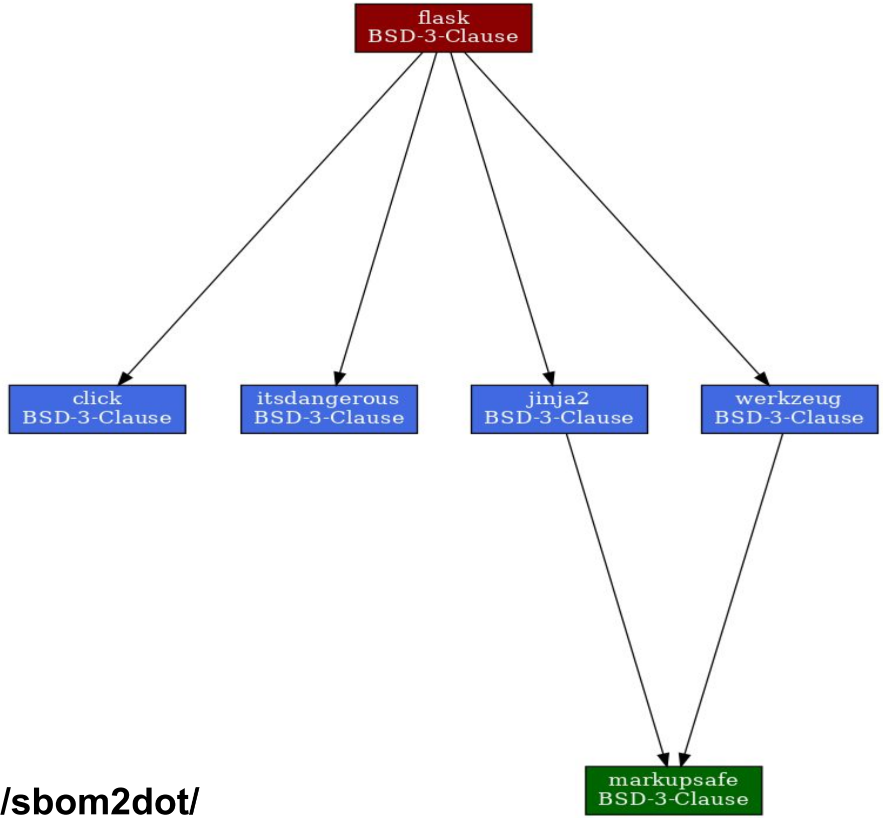
- Dependencies are software constraints - what the software needs to execute
  - Direct components **explicitly** used by the software
  - Transitive components **implicitly** required by other dependencies
- Typically 2 to 10 transitive dependencies for each direct dependency
  - Programming language dependent
- Multiple ways of specifying dependency
  - Can optionally include version of dependency
  - Explicit version numbers v. open (not specified) version numbers
- Conflicting dependencies

Direct dependencies

```
aiohttp[speedups]>=3.7.4
beautifulsoup4==4.1.2
cvss
defusedxml
distro
gsutil
importlib_metadata>=3.6; python_version < "3.10"
importlib_resources; python_version < "3.9"
```

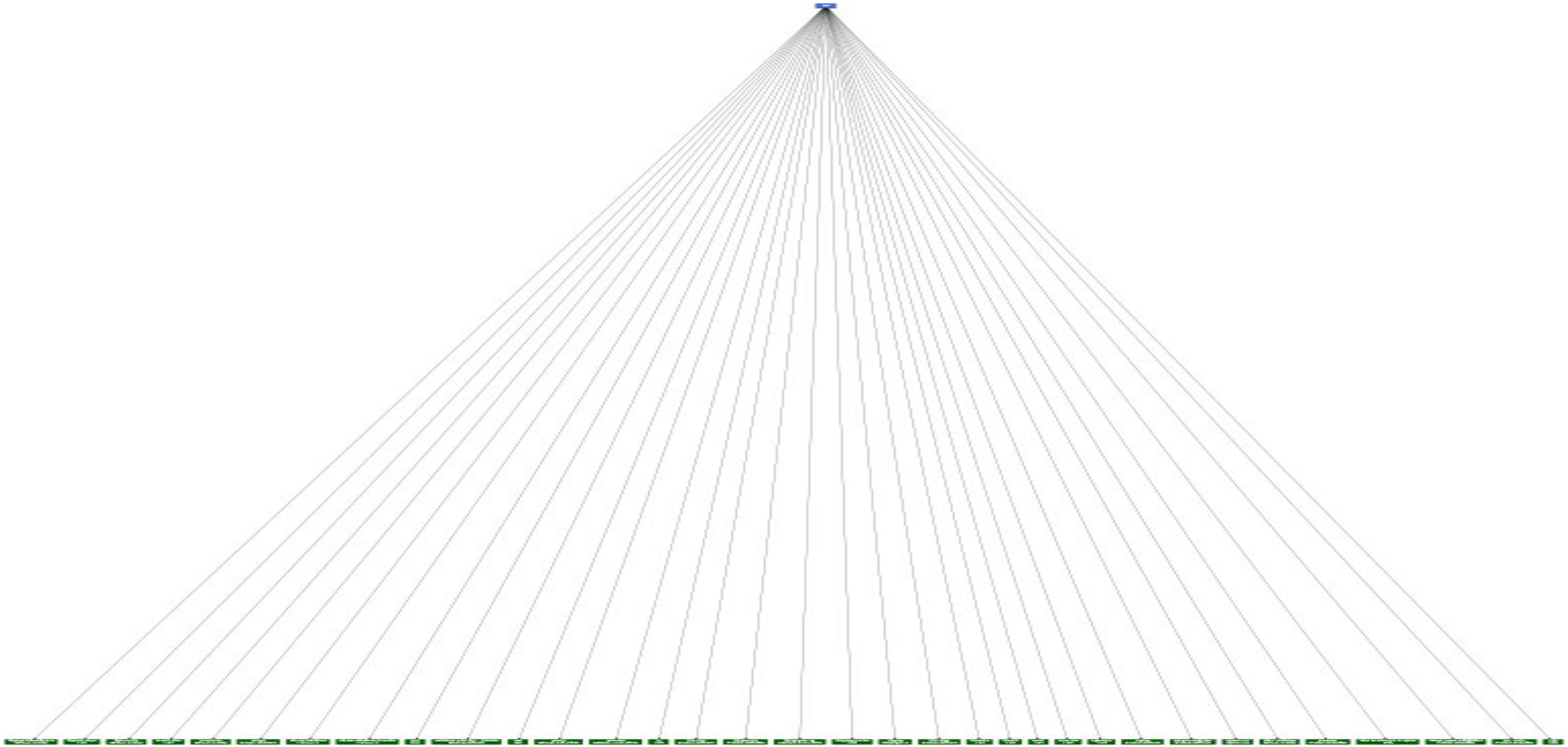


https://pypi.org/project/sbom2dot/

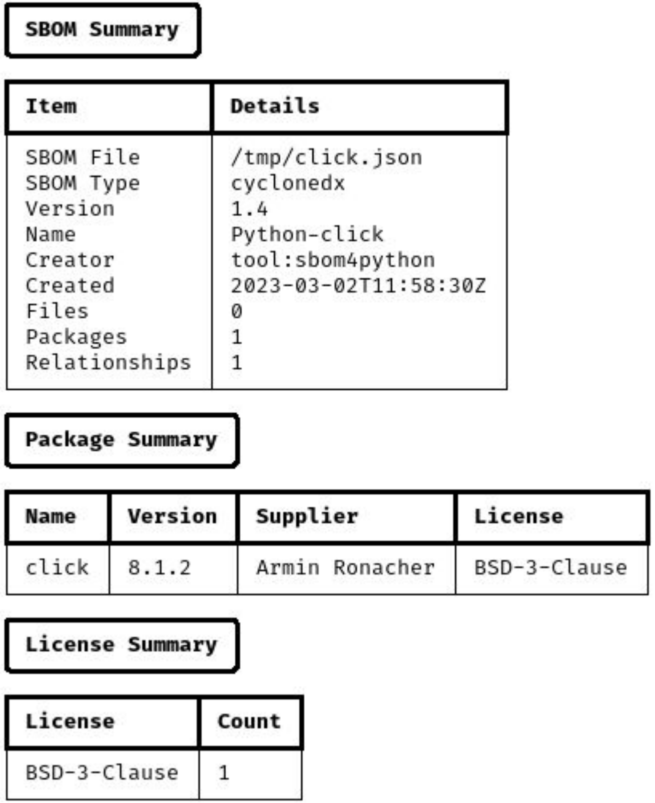


https://pypi.org/project/sbom2dot/

APH10



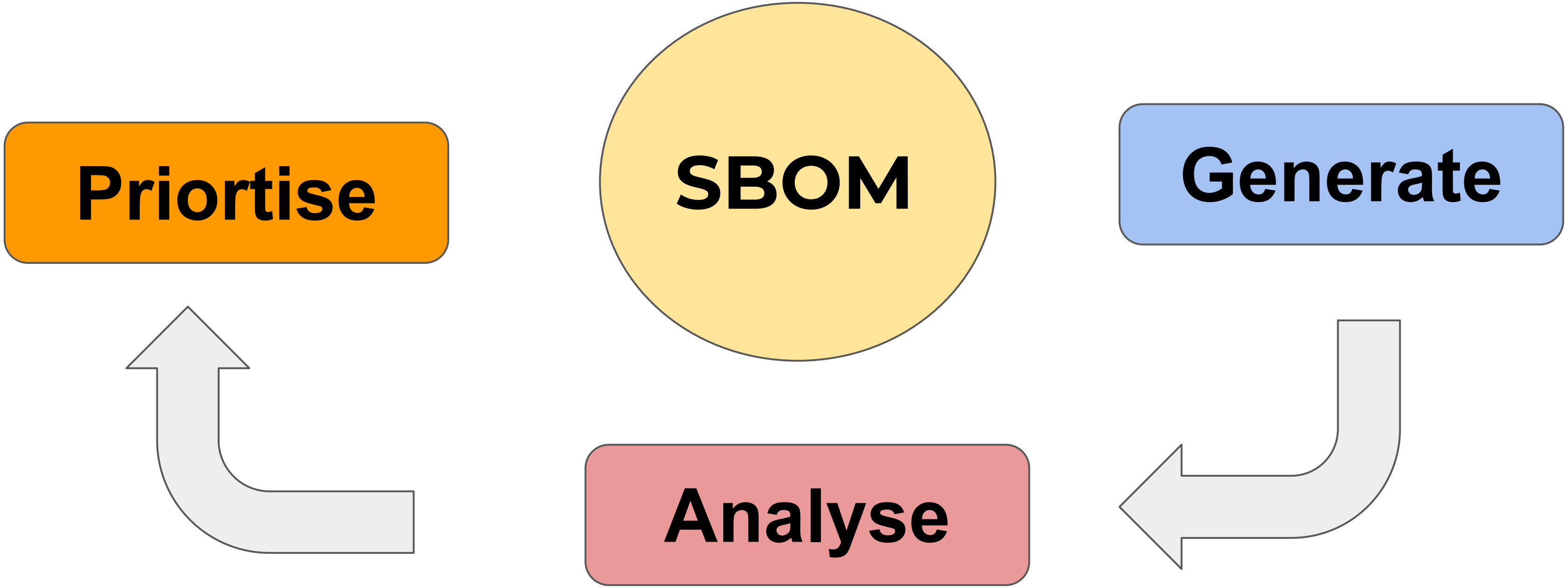
https://pypi.org/project/sbom2dot/



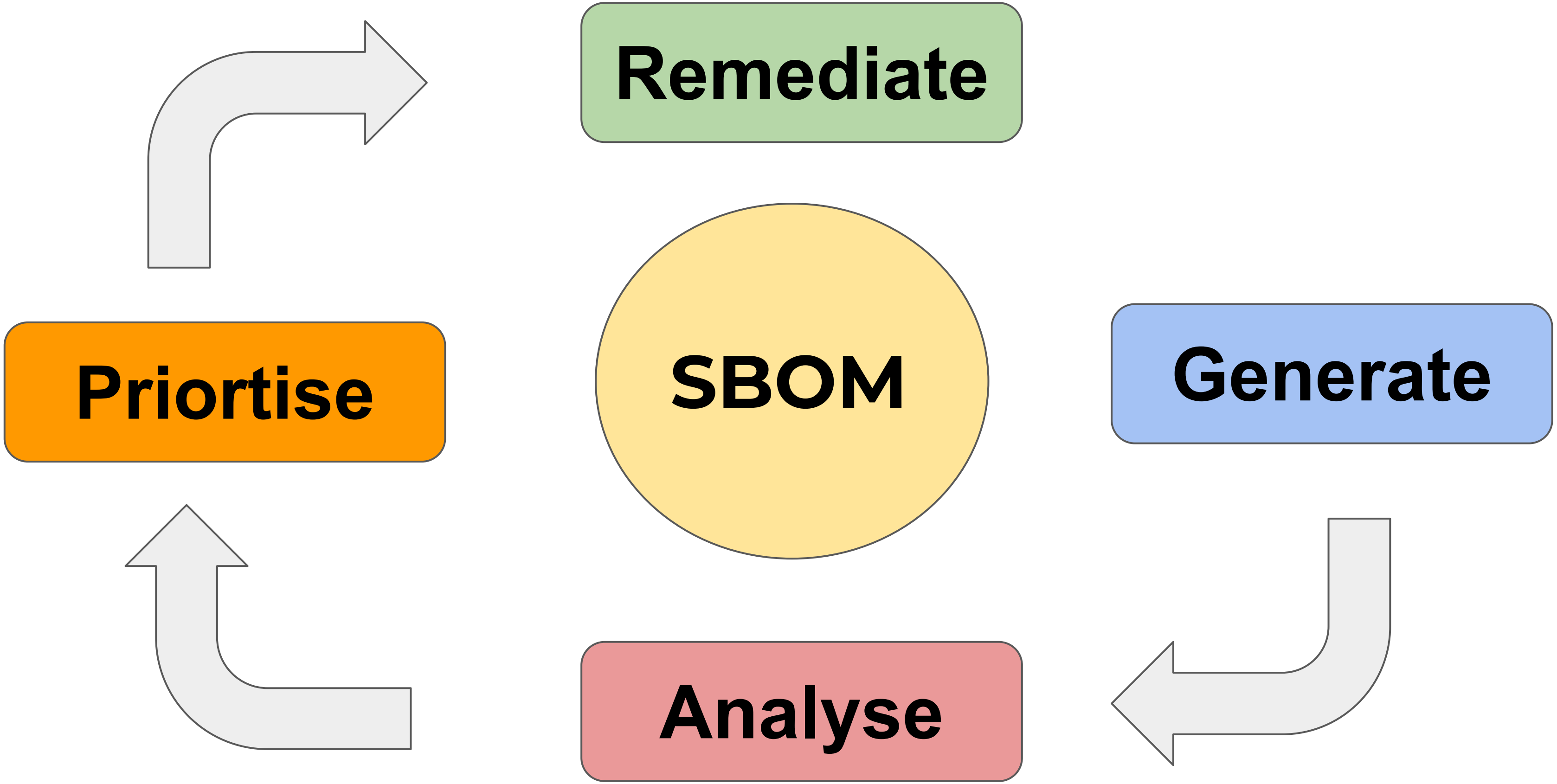
```
┌──────────────┐
│ SBOM Summary │
└──────────────┘
┌───────────────┬──────────────────────┐
│ Item          │ Details              │
├───────────────┼──────────────────────┤
│ SBOM File     │ /tmp/click.json      │
│ SBOM Type     │ cyclonedx            │
│ Version       │ 1.4                  │
│ Name          │ Python-click         │
│ Creator       │ tool:sbom4python     │
│ Created       │ 2023-03-02T11:58:30Z │
│ Files         │ 0                    │
│ Packages      │ 1                    │
│ Relationships │ 1                    │
└───────────────┴──────────────────────┘
┌─────────────────┐
│ Package Summary │
└─────────────────┘
┌───────┬─────────┬────────────────┬──────────────┐
│ Name  │ Version │ Supplier       │ License      │
├───────┼─────────┼────────────────┼──────────────┤
│ click │ 8.1.2   │ Armin Ronacher │ BSD-3-Clause │
└───────┴─────────┴────────────────┴──────────────┘
┌─────────────────┐
│ License Summary │
└─────────────────┘
┌──────────────┬───────┐
│ License      │ Count │
├──────────────┼───────┤
│ BSD-3-Clause │ 1     │
└──────────────┴───────┘
```

https://pypi.org/project/sbom2doc/

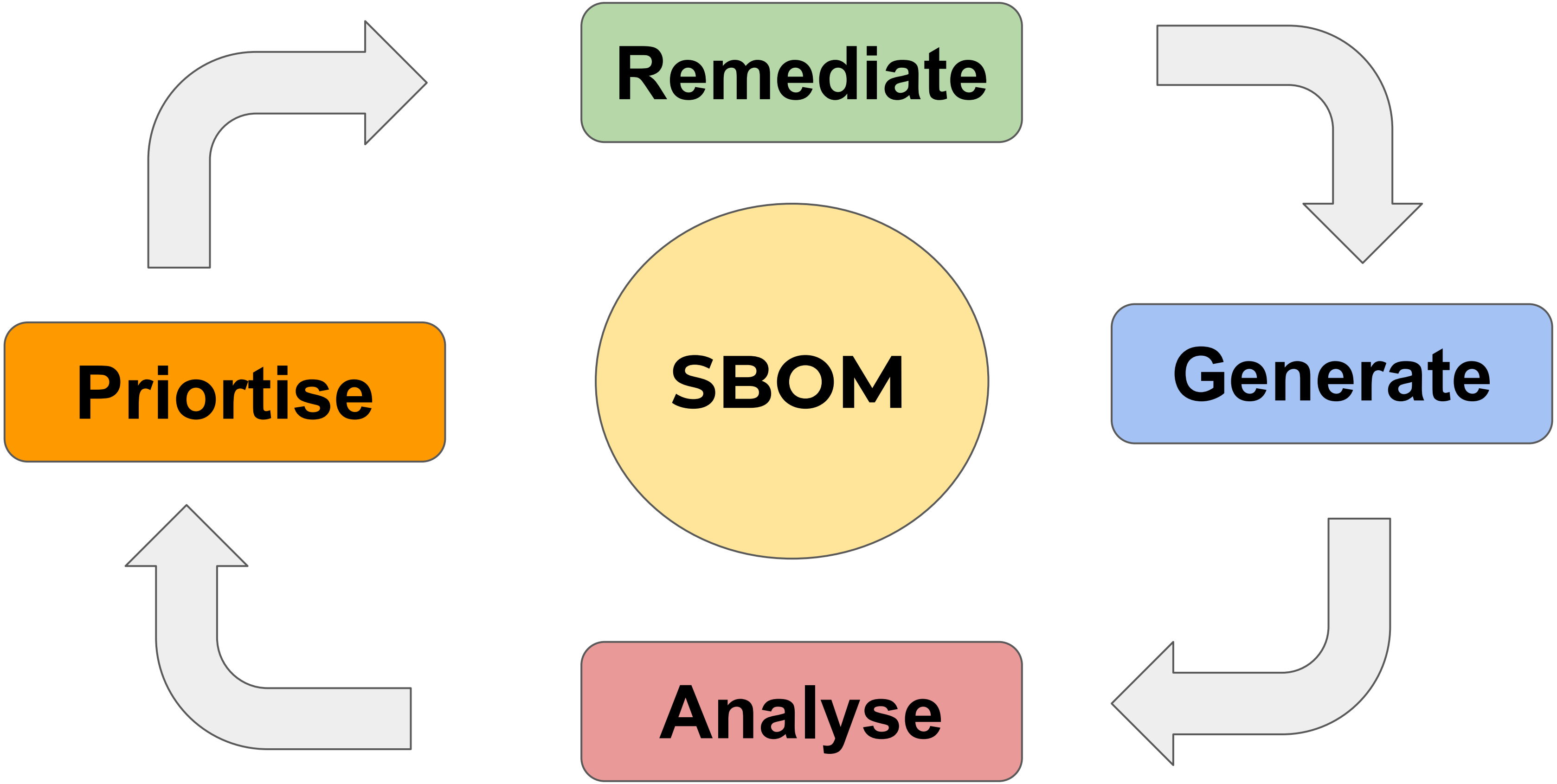
SBOM Lifecycle



SBOM Lifecycle



SBOM Lifecycle



Remediate

Generate

Analyse

Priortise

SBOM

Decade	Languages
1950	*COBOL*, Fortran
1970	**C**, *SQL*, Smalltalk
1980	**C++**, Ada, Perl, Erlang, **Delphi/Object Pascal**
1990	**Python**, **Java**, *R*, *Ruby*, **Javascript**, *PHP*, **Visual Basic**
2000	**C#**, Scala, Powershell, **Go**
2010	Kotlin, *Rust*, *Swift*

https://www.tiobe.com/tiobe-index/ (May 2025)

Metadata - Component Names

- Common Platform Enumeration (CPE)
  - A standardized method of describing and identifying classes of applications, operating systems, and hardware devices
  - May include product names, vendors, architecture
  - *cpe:2.3:a:the_purl_authors:packageurl-python:0.11.1:*:*:*:*:*:*:*
- Package URL (PURL)
  - A string used to identify and locate a software package in a mostly universal and uniform way across programing languages, package managers
  - *pkg:pypi/packageurl-python@0.11.1*
  - *https://github.com/package-url/purl-spec/blob/main/PURL-SPECIFICATION.rst*

- *Open Source makes up 27 percent of UK tech sector economy in 2022*
- *Up to £326.6m planned in UK company investment in open source*
- *44% of companies will increase investment in Open Source in next 12 months*
- *3.4M open source developers in the UK (5% per capita. More than India/US)*

OpenUK - Jul 2023

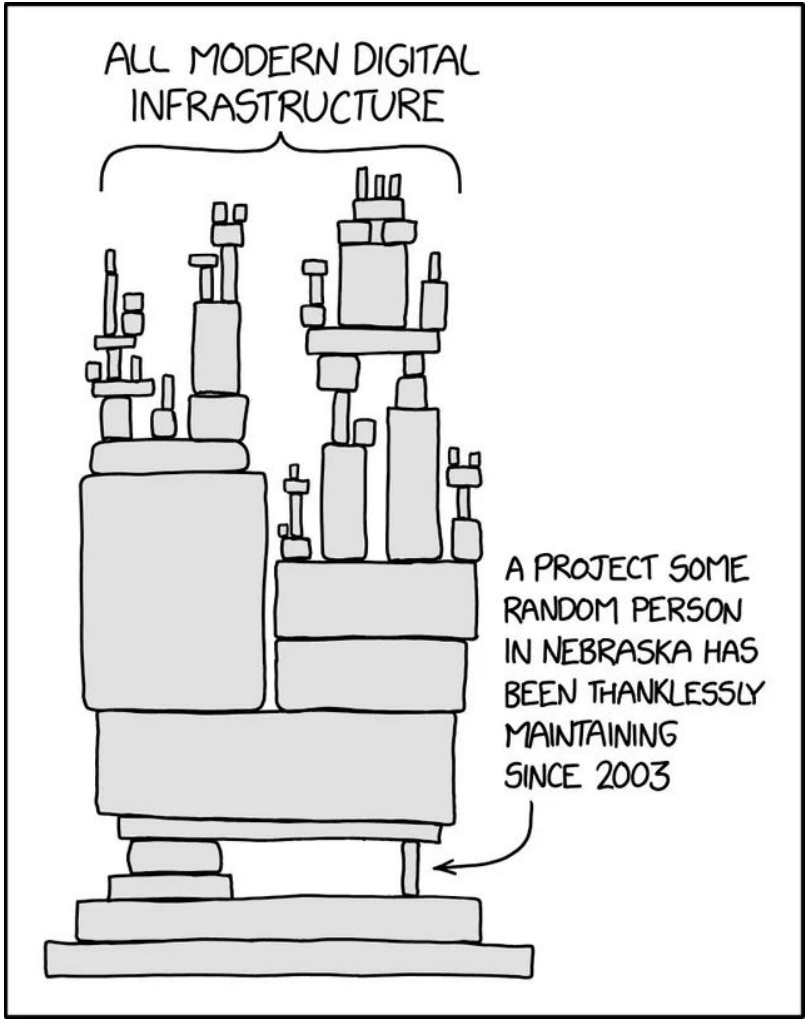
96% of all tech stacks need open source software

OSS is everywhere. OSS runs our lives.

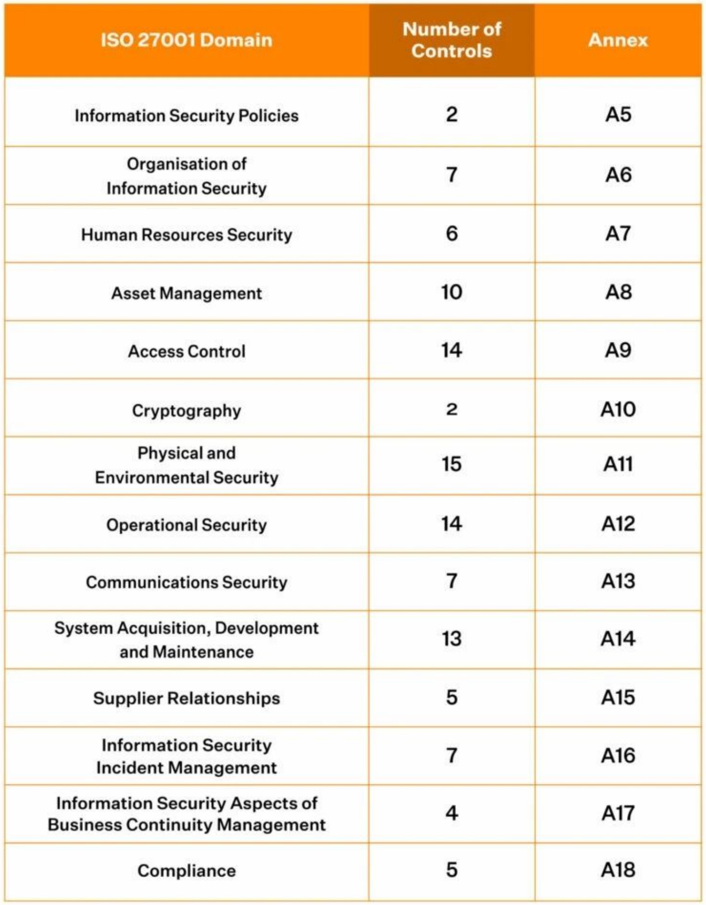
Allan Friedman CISA

Metadata - Licence Naming

Apache 2.0	**Many Python packages**
Apache License, Version 2.0	
Apache 2	
Apache	
Apache-2.0	**Valid SPDX License identifier**


ALL MODERN DIGITAL INFRASTRUCTURE
A PROJECT SOME RANDOM PERSON IN NEBRASKA HAS BEEN THANKLESSLY MAINTAINING SINCE 2003

https://xkcd.com/2347/

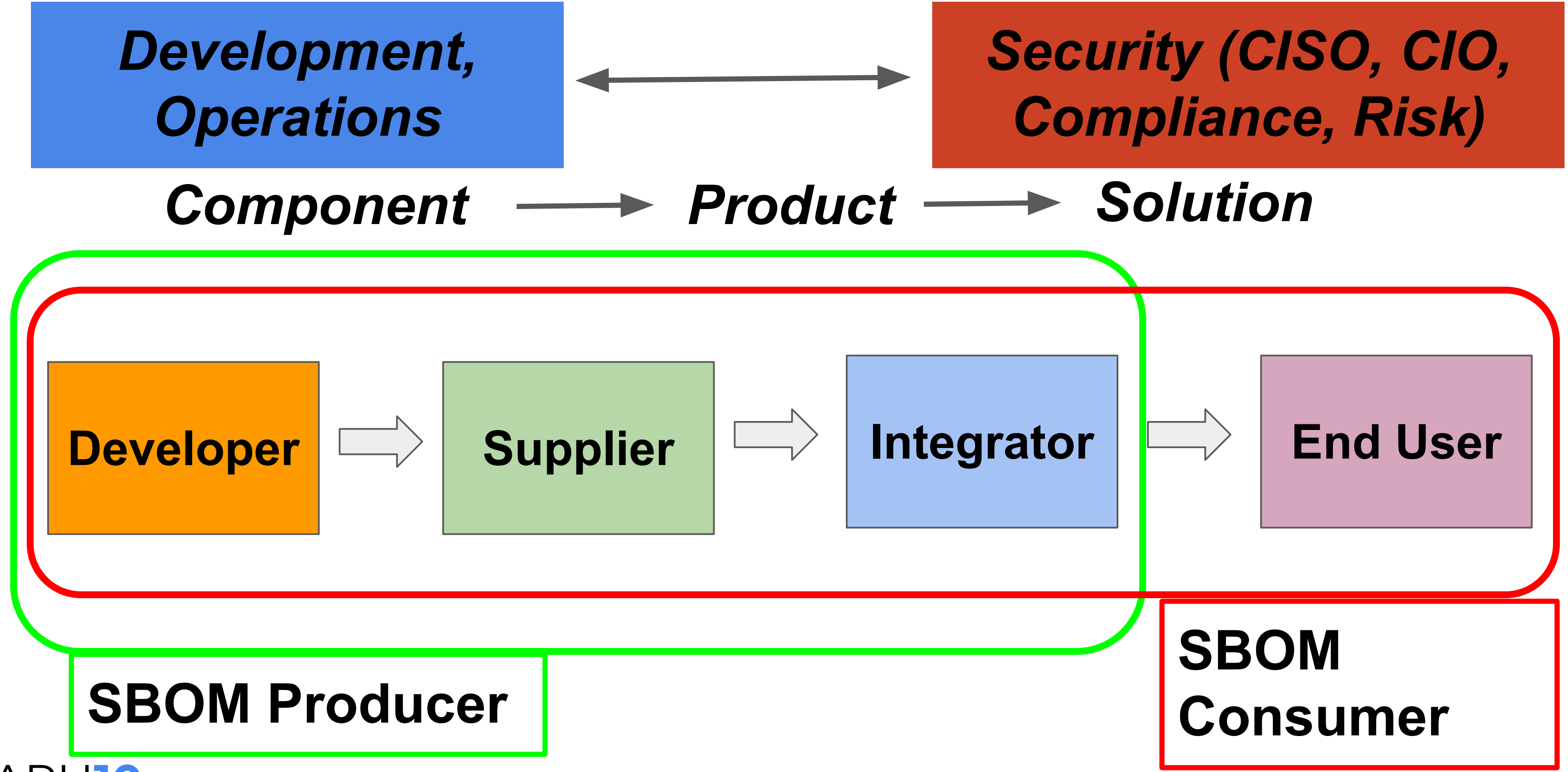


ISO 27001 Domain	Number of Controls	Annex
Information Security Policies	2	A5
Organisation of Information Security	7	A6
Human Resources Security	6	A7
Asset Management	10	A8
Access Control	14	A9
Cryptography	2	A10
Physical and Environmental Security	15	A11
Operational Security	14	A12
Communications Security	7	A13
System Acquisition, Development and Maintenance	13	A14
Supplier Relationships	5	A15
Information Security Incident Management	7	A16
Information Security Aspects of Business Continuity Management	4	A17
Compliance	5	A18

Supply Chain Management

- A8 Asset Management
- A12 Operational Security (defence against malware, backups, logging & monitoring, **change management, patch management, vulnerability management** and penetration tests, etc)
- A14 System Acquisition, Development and Maintenance
- A15 Supplier Relationships

SBOM Interested Parties



Development, Operations

Security (CISO, CIO, Compliance, Risk)

Component → ***Product*** → ***Solution***

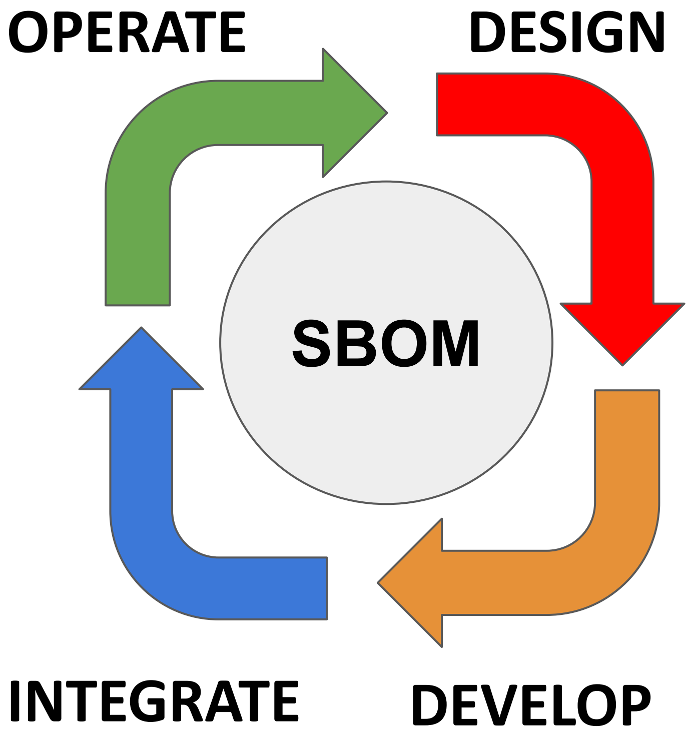
Developer → **Supplier** → **Integrator** → **End User**

SBOM Producer

SBOM Consumer

APH10

SBOMs are used throughout the lifecycle

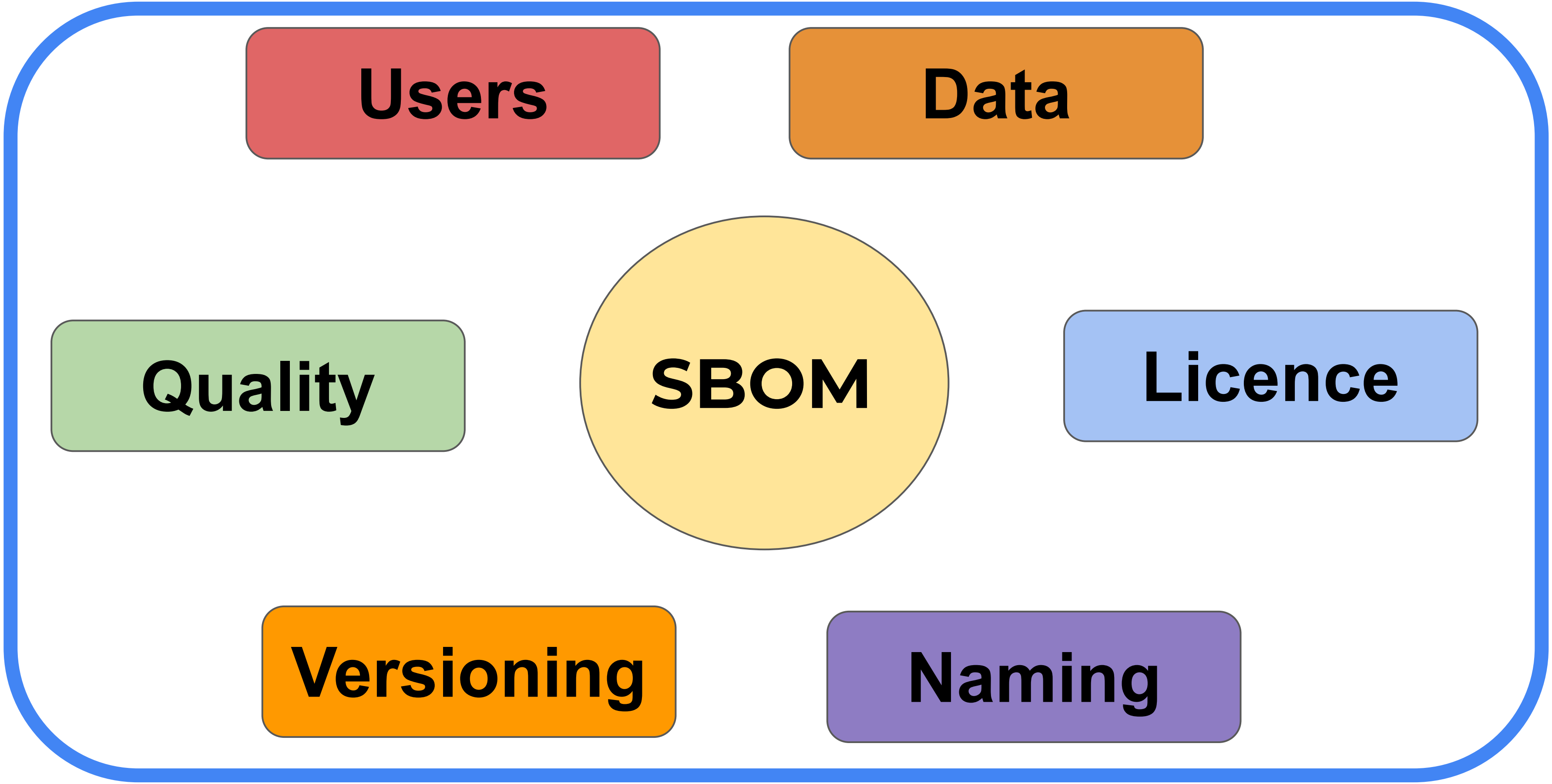


- **3rd party component selection**
- **Source files in build**
- **Applications built**
- **Component dependencies**
- **Build composition**
- **Build integrity**
- **License Compliance**
- **Change Management**
- **Vulnerability Monitoring**
- **Obsolete software detection**

APH10

Insights and Benefits

- Component Use
  - Do you have multiple versions across applications
  - Actively being maintained?
  - Vulnerable?
- Dependent suppliers and communities
  - Are you supporting them?
- Licence Compliance
  - Do all components match development policy?
- Improved Software Security and Resilience



Users

Data

Quality

SBOM

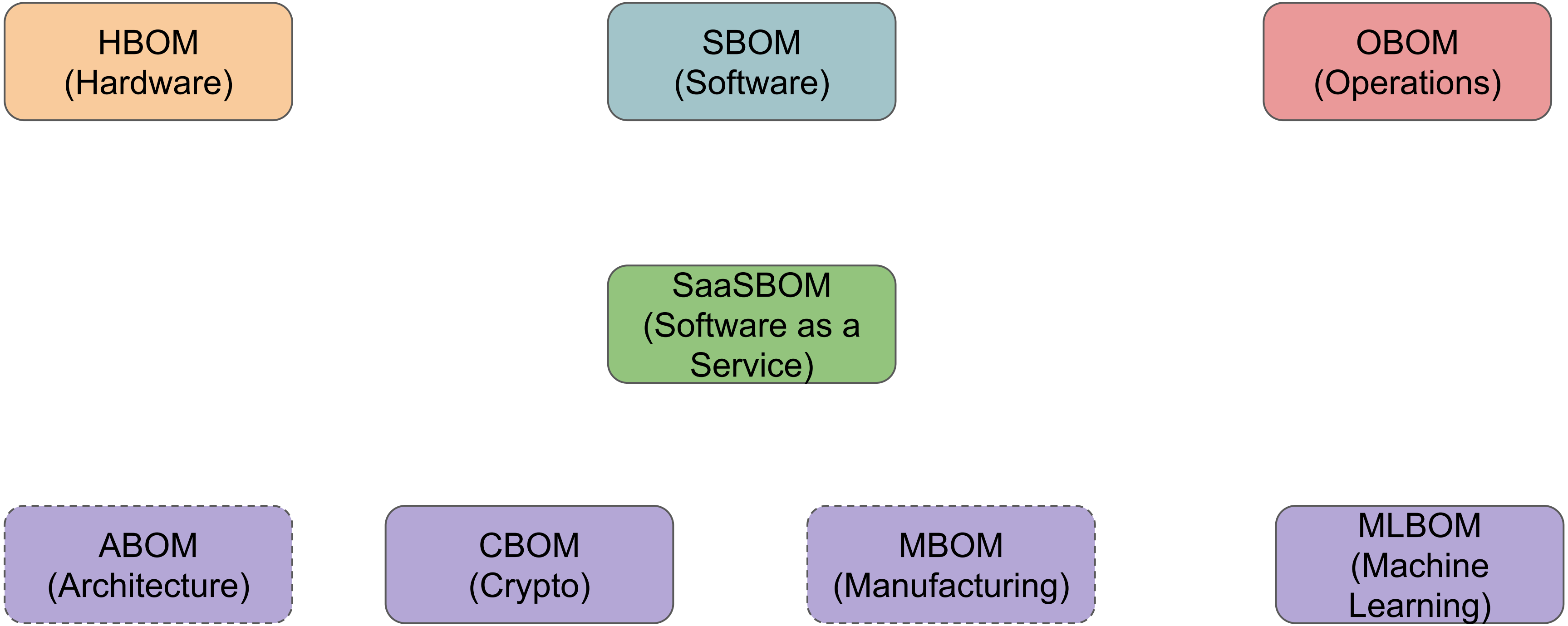
Licence

Versioning

Naming

APH10

SBOMs in the wider context



HBOM (Hardware)

SBOM (Software)

OBOM (Operations)

SaaSBOM (Software as a Service)

ABOM (Architecture)

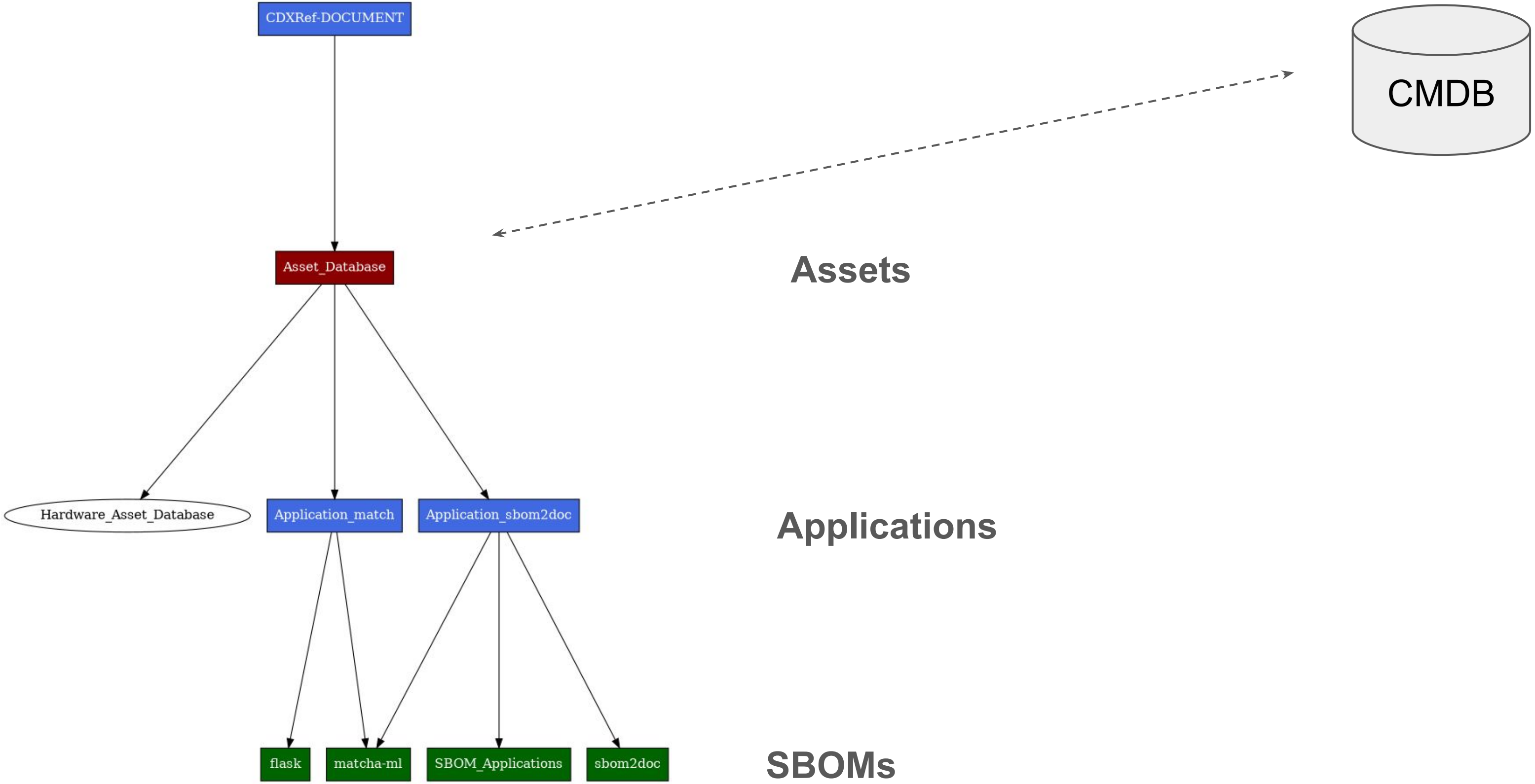
CBOM (Crypto)

MBOM (Manufacturing)

MLBOM (Machine Learning)

APH10

SBOMs in the wider context



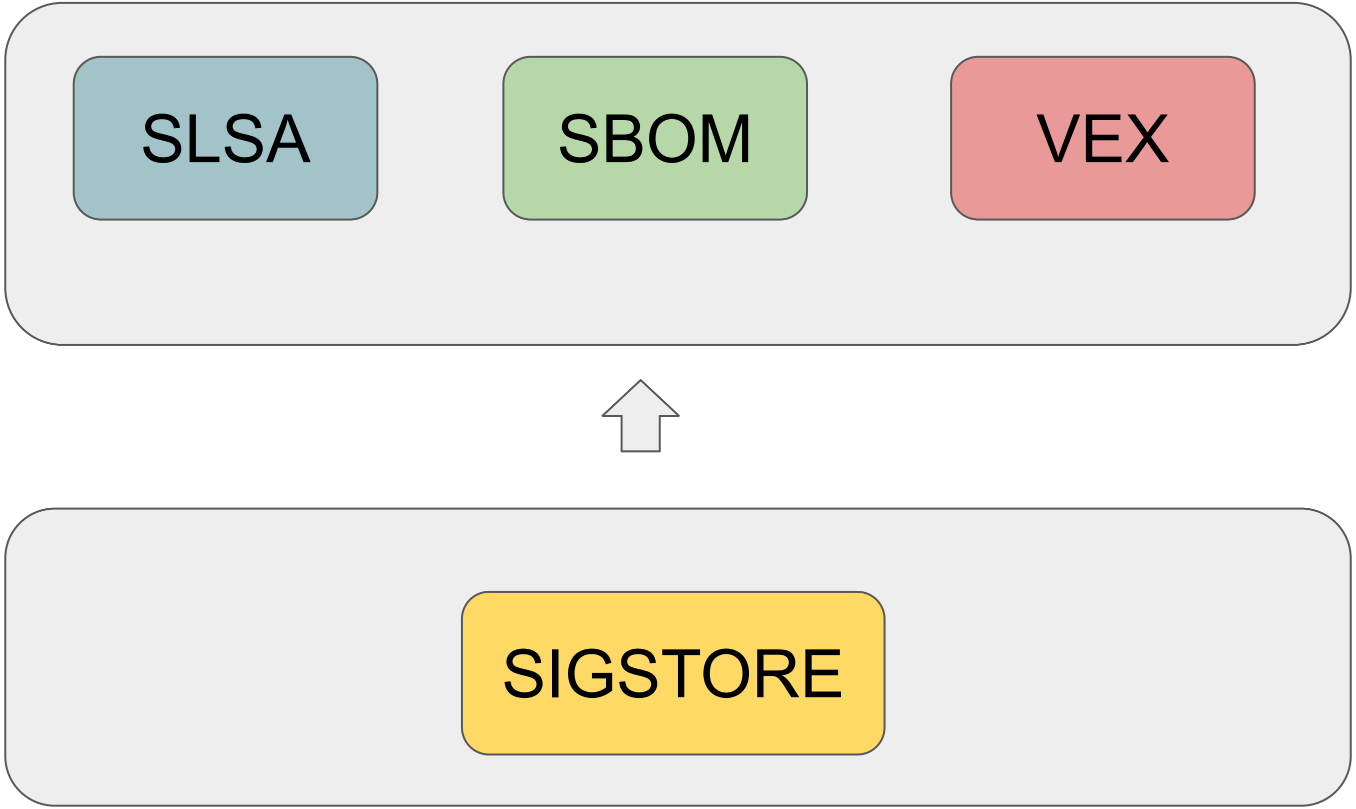
Assets

Applications

SBOMs

APH10

The Bigger Picture for Supply Chain



SLSA

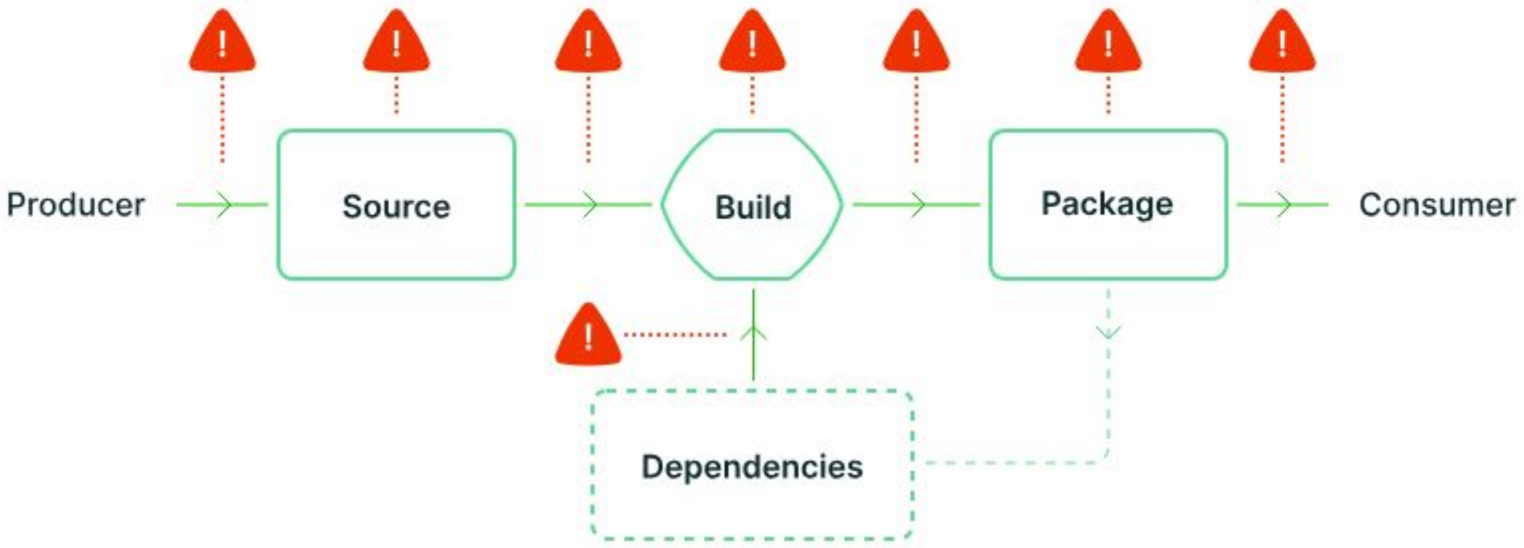
SBOM

VEX

SIGSTORE

APH10

Supply Chain Integrity



Supply Chain Levels for Software Artefacts (SLSA)

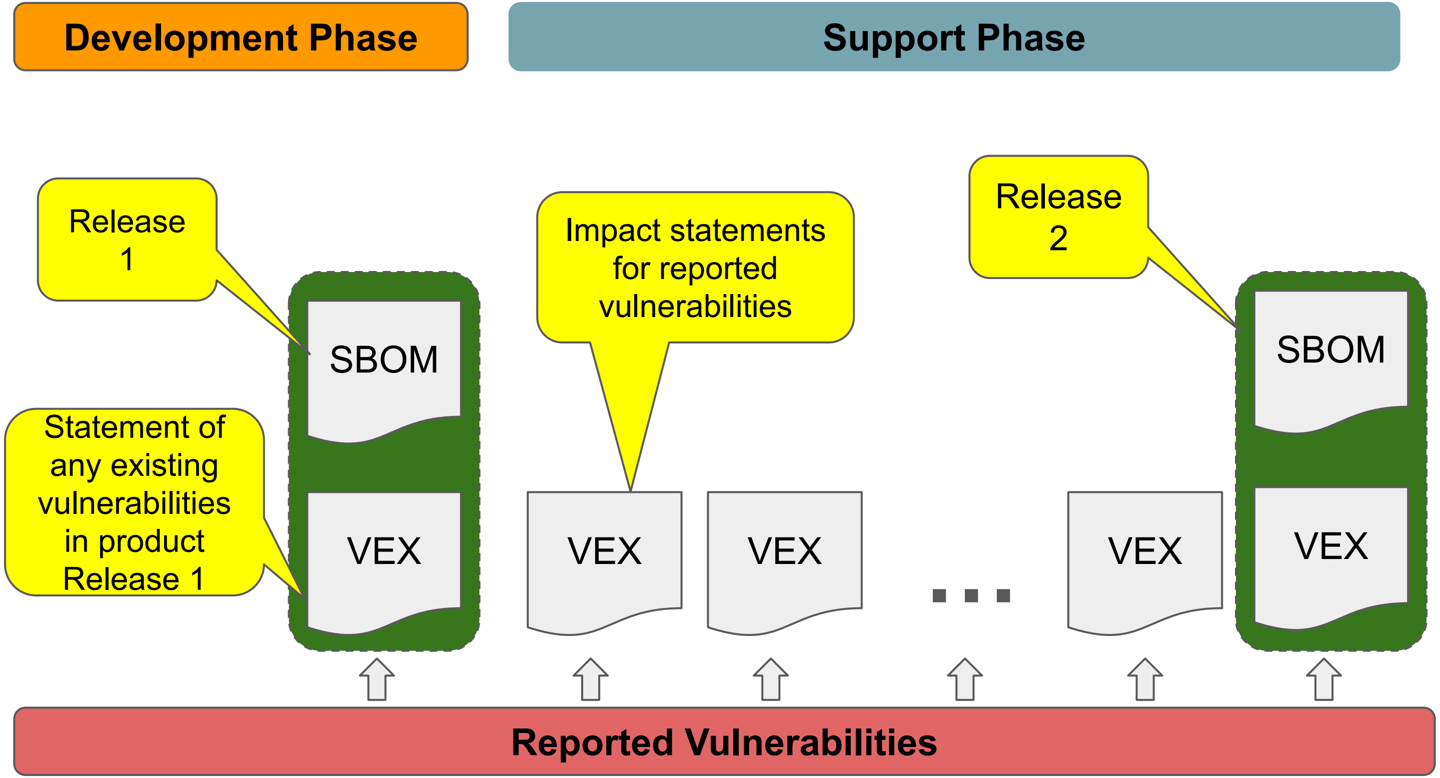
https://slsa.dev/

Sample SBOM Use Cases

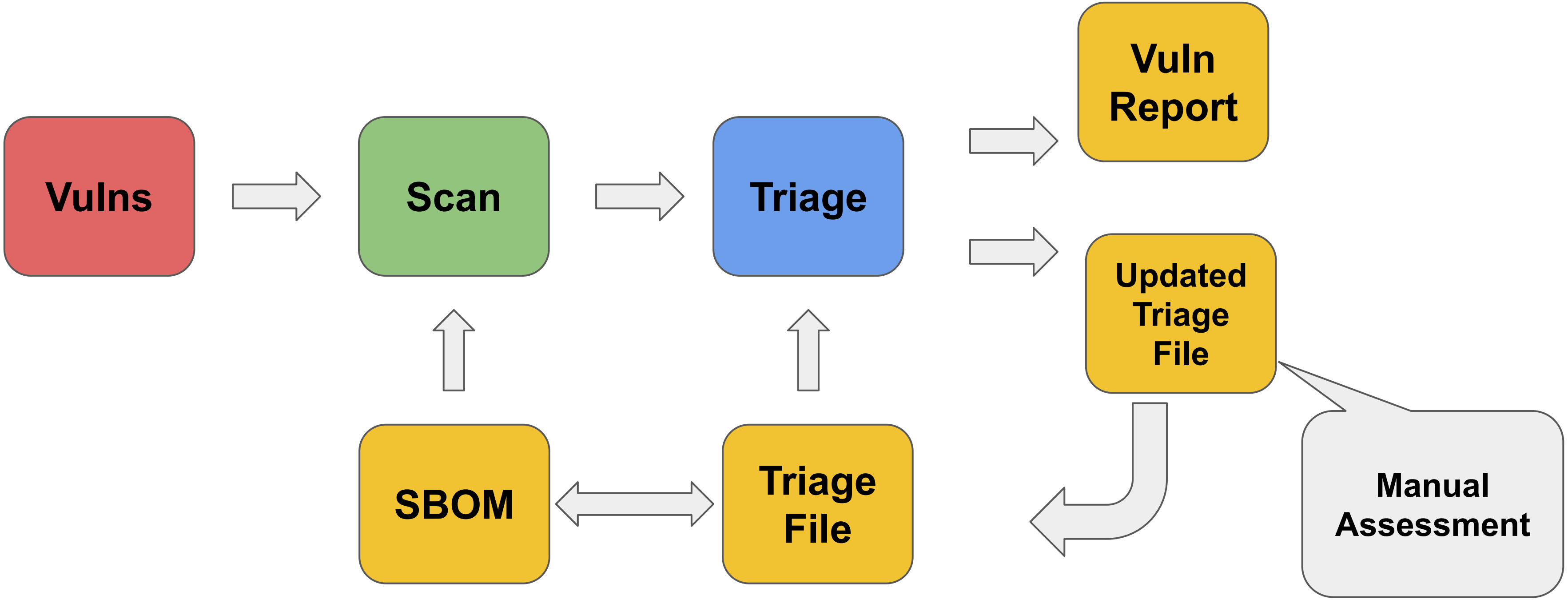
- **Vulnerability Assessment and Management** to support software security and resilience.
- **Licence Compliance** to ensure that 3rd party components are being used in accordance with the requirements of the licence.
- **Support for ESCROW** by providing an inventory of the components used in the production of a software product.
- **Support to M&A** activities by providing transparency into the construction of a software product.
- **Support for Regulatory Compliance** by provision of artefacts required to support cyber security needs.

Responding to a Vulnerability

- Not my job….
- Ignore it
- Park it
- Triage it and then park it
- **Analyse it, communicate analysis and remediate it (if required)**


Development Phase
Support Phase
Release 1
Impact statements for reported vulnerabilities
Release 2
SBOM
Statement of any existing vulnerabilities in product Release 1
VEX
VEX
VEX
...
VEX
SBOM
VEX
Reported Vulnerabilities

Triage Process



APH10

VEX

- VEX stands for **"Vulnerability Exploitability eXchange"**
- A form of a security advisory that indicates whether a product or products are affected by a known vulnerability or vulnerabilities
- A document format used to communicate whether a specific vulnerability is actually exploitable in a given software context
- It is NOT just a list of vulnerabilities

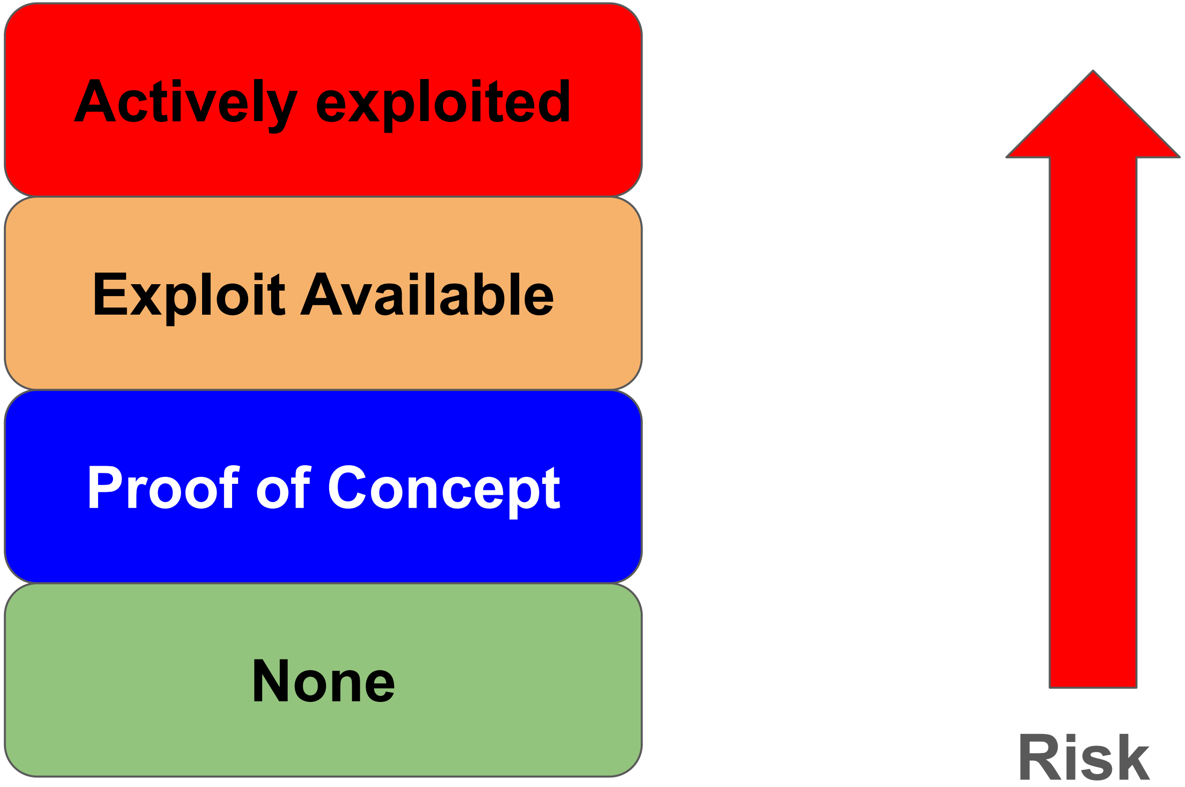
Exploitable Vulnerabilities

What does it mean?

How to detect?

Who determines?

Exploited Vulnerabilities



Actively exploited

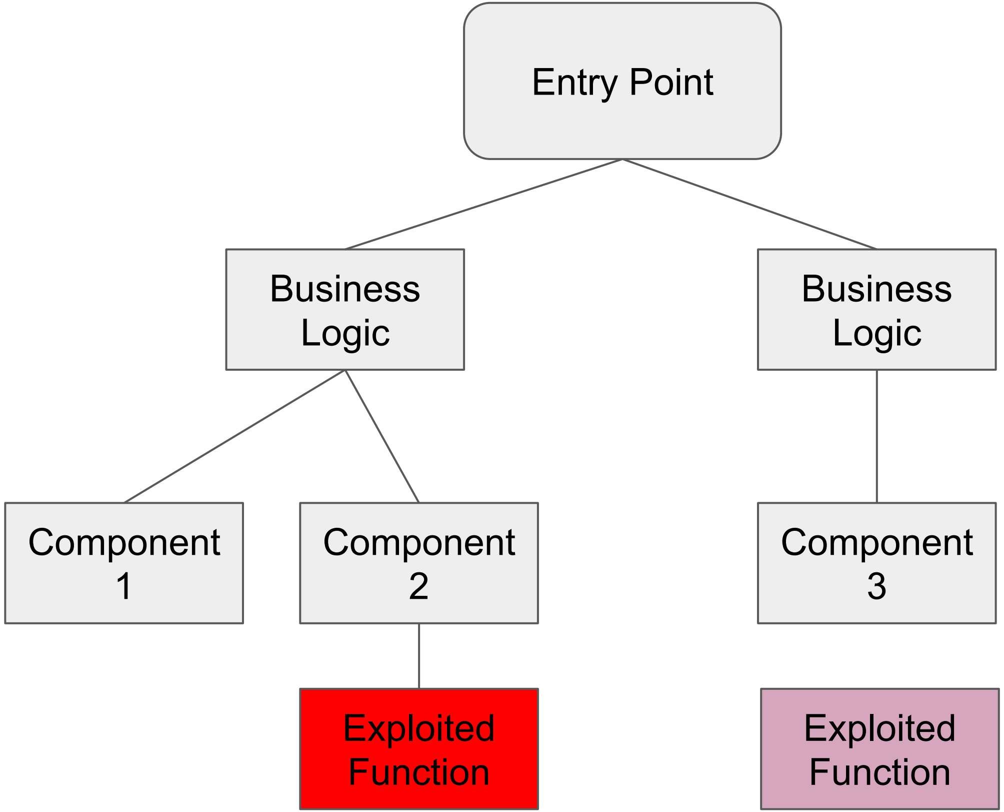
Exploit Available

Proof of Concept

None

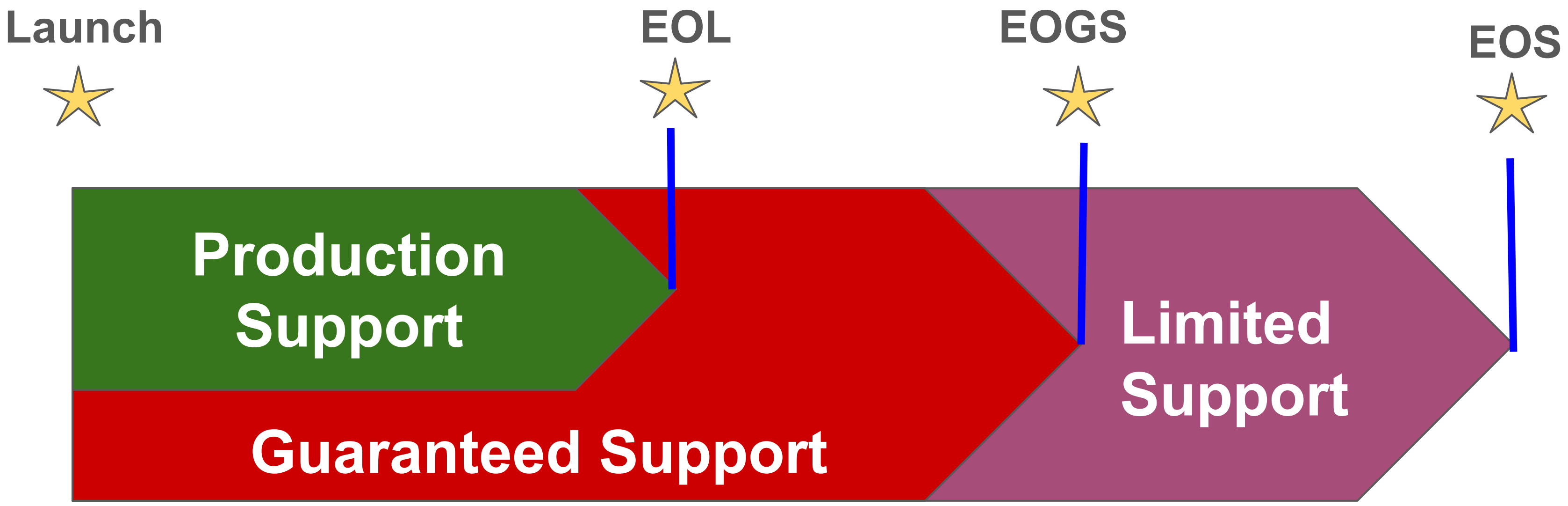
Risk

But is it exploitable in MY product?



APH10

EOX



EOL - Product no longer sold

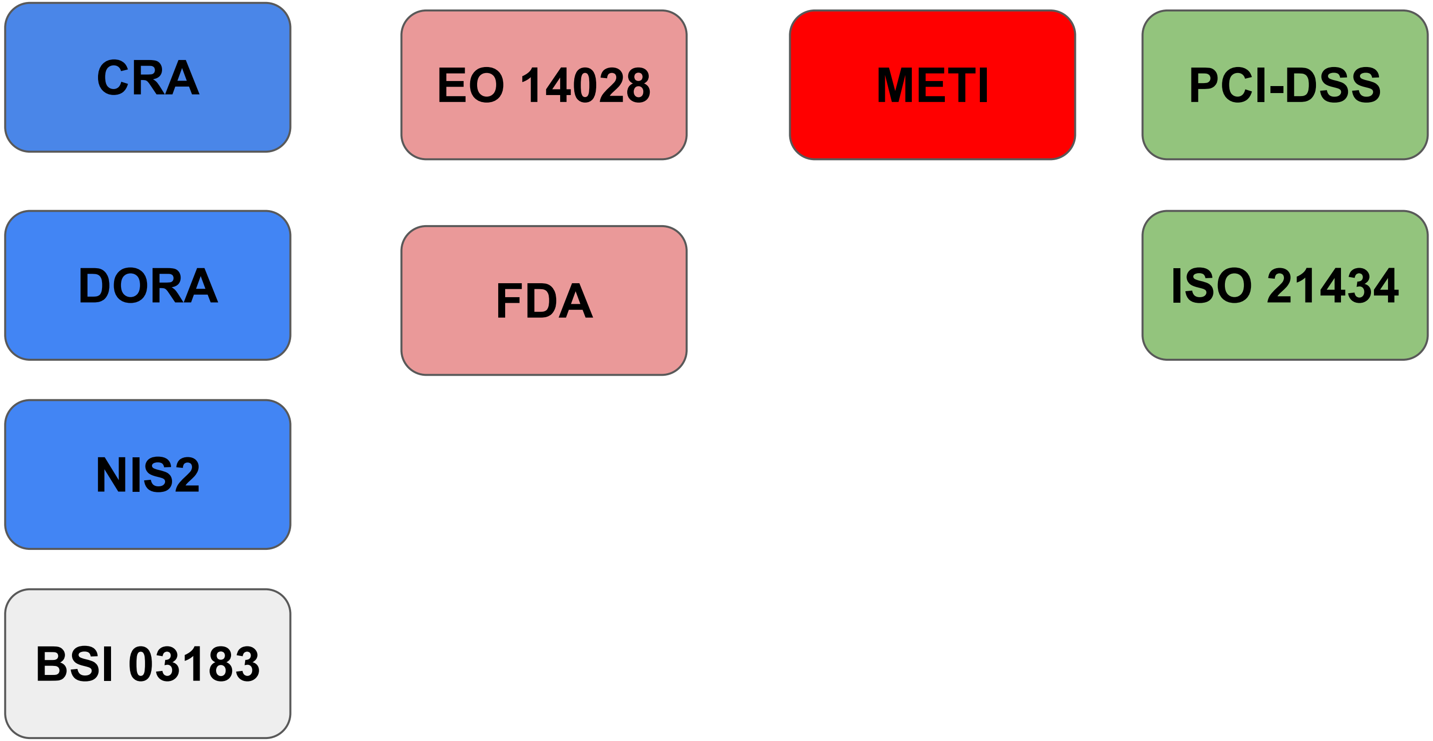
EOS - Product no longer supported

EOGS - Product no longer guaranteed support

Who are potential users of the SBOMs?

Group	Need
Customers	Understand the risk and the benefits Assess the business impact
Governance Team	Understand the risk in the product Assess the business impact on customers
Product Owners	Prioritise the remediation of risks
Product Developers	Manage dependencies Understand risks in implementation

Legislation and Regulations



CRA

DORA

NIS2

BSI 03183

EO 14028

FDA

METI

PCI-DSS

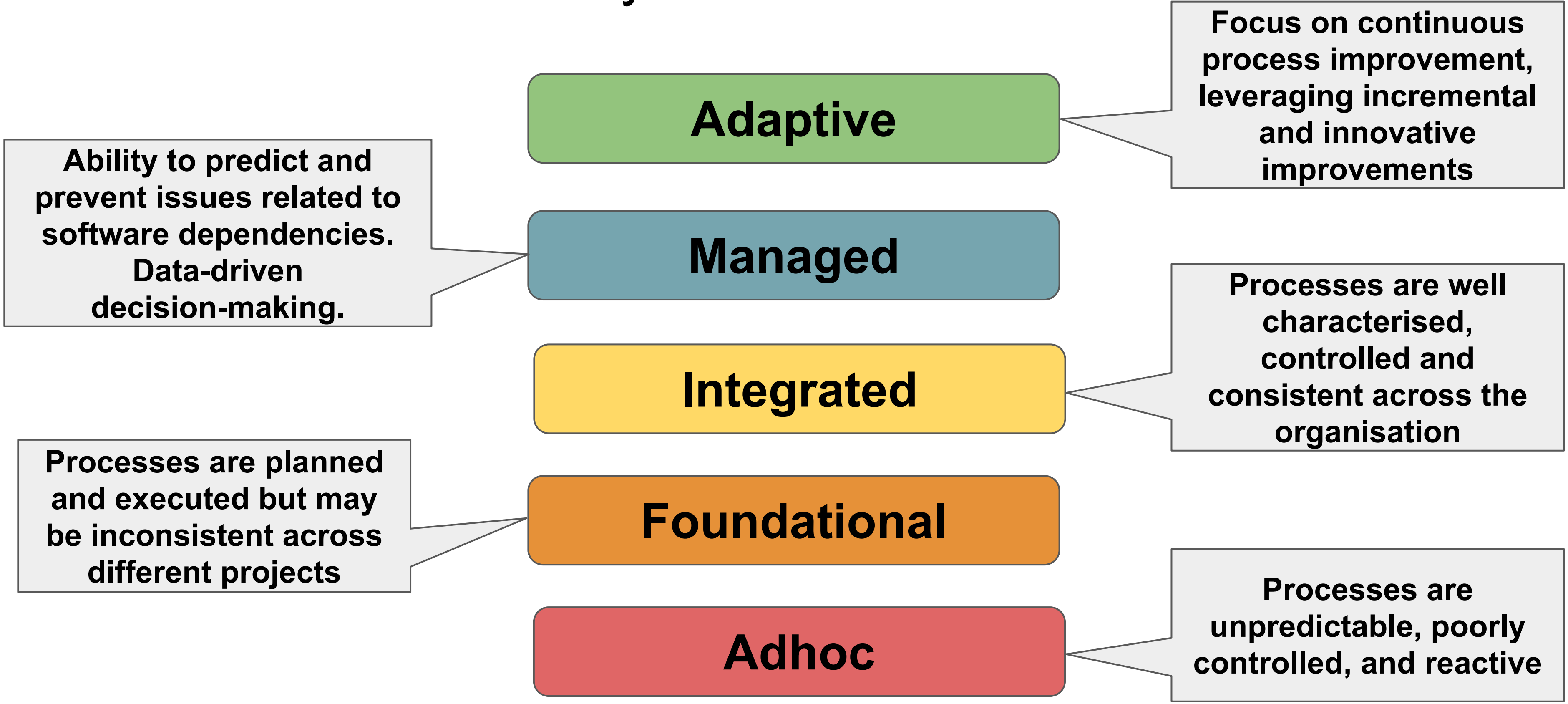
ISO 21434

CRA Requirements

Manufacturers of products with digital elements shall:

*“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**”* (CRA Annex I Part II 1)

SBOM Business Maturity Levels



Focus on continuous process improvement, leveraging incremental and innovative improvements

Adaptive

Ability to predict and prevent issues related to software dependencies. Data-driven decision-making.

Managed

Processes are well characterised, controlled and consistent across the organisation

Integrated

Processes are planned and executed but may be inconsistent across different projects

Foundational

Processes are unpredictable, poorly controlled, and reactive

Adhoc

APH10

Summary

Appsec teams now need an understanding of the software supply chain for any digital product

- Software Bill of Materials provide valuable insights
- Quality of data is crucial to maintaining a secure product

Vulnerability Management needs continual assessment of the risk

- Sharing data upstream/downstream
- Applies throughout the complete development lifecycle

Organisation processes need to incorporate SBOMs

Contact Details - Anthony Harrison

LinkedIn

GitHub

https://www.linkedin.com/company/aph10/
https://www.linkedin.com/in/anthonypharrison/

Email: anthony@aph10.com

https://www.aph10.com
https://www.sbomeurope.eu