OWASP TRANSPARENCY EXCHANGE API

OWASP Transparency Exchange API Overview

oej FOSDEM 2025-02-02 v2.2
Olle E Johansson - oej@edvina.net

OWASP®

TC 54

https://tc54.org/

CycloneDX

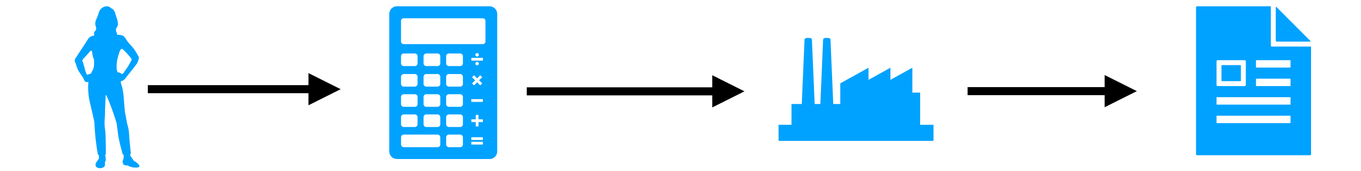
Project Koala

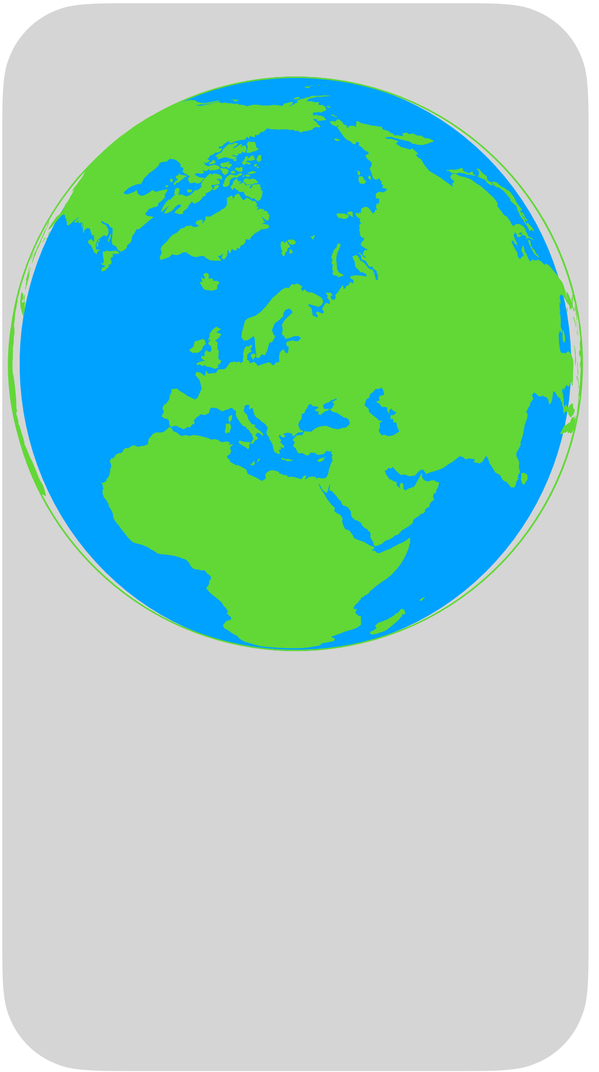
The problem

Many customers have many products from many vendors.

In order to **automatically** or manually be able
to retrieve standardised transparency attestations
(SBOM, VEX and others)
we need to also standardise
discovery, **identification**, **authentication**
and retrieval of these documents.

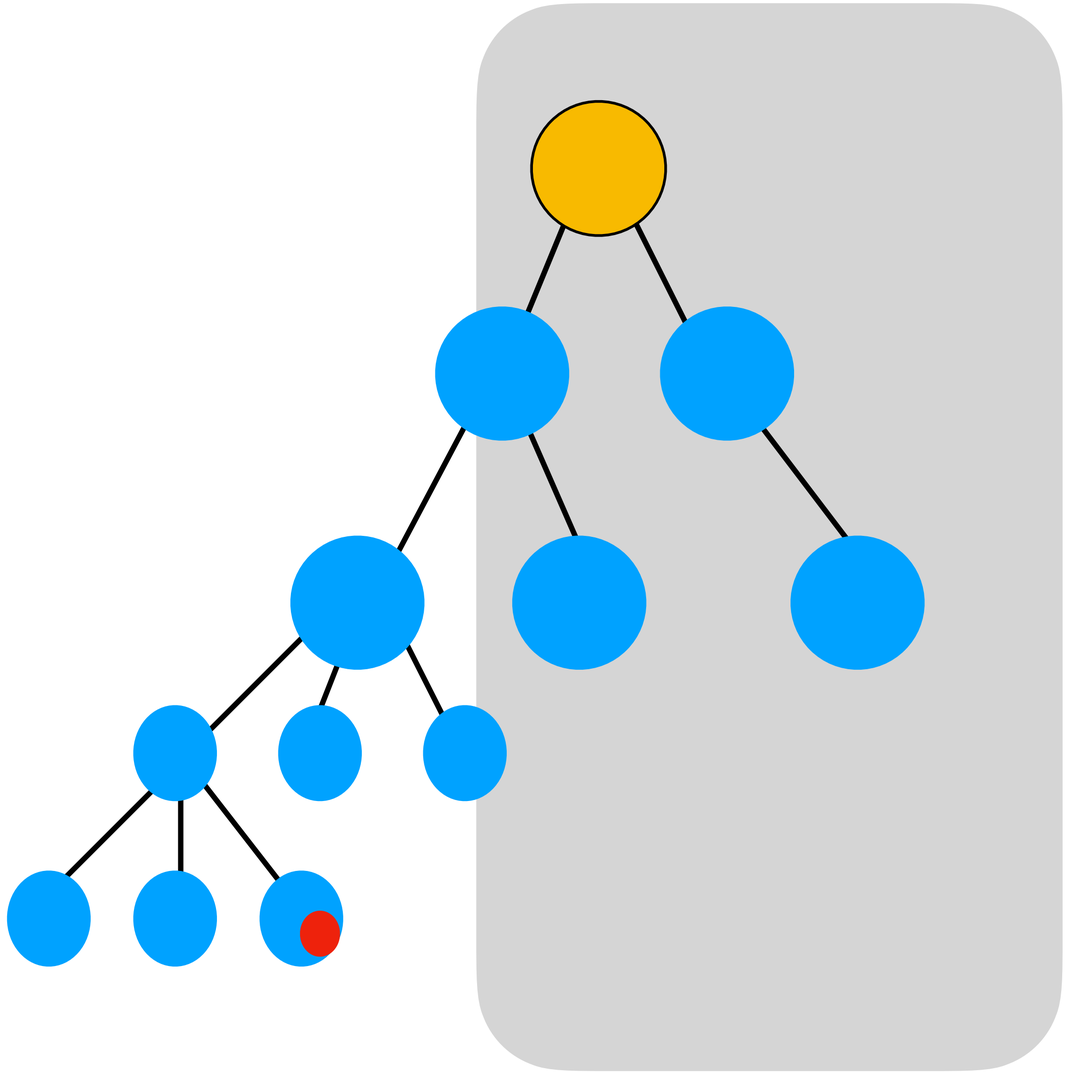
The solution has to scale globally.





The supply chain

- **As a customer**, I need to get attestations from all my suppliers - both commercial and open source
- **As a manufacturer,** I need to get attestation from upstream suppliers, both commercial and open source
- As an **Open Source project**, we need to make sure that our software with dependencies can always be built without vulnerabilities



CISA SBOM Sharing

Lifecycle phase	Sophistication low	Sophistication medium	Sophistication high
Discovery	• Consumer initiated • Limited or nonexistent guidance given by Author or Distributor	SBOM placed in software source code - Point in time (Singular Version) - Manufacturer Usage Description (MUD) • Known central repository • Website	Automated propagation of available SBOMs • Continuous updates to relevant parties • Publish/Subscribe pattern • Distributed ledger
Access	• No controls in place • Manual controls • Case-by-case	• Authentication required • Limited access control granularity • Private/ broadca st/public channels, roles • Private chains/consens us algorithm	Delegated authentication and access controls • Full access control granularity
Transport	• Human initiated process • Point-to-point • Verbal transmission	• Inconsistent, varied method or documentation • Ad-hoc automation	• Documented • Repeatability • Automated access • Well-known protocols (e.g., REST/RESTful/SO AP API) • Distributed ledger synchronization

https://www.cisa.gov/sites/default/files/2024-05/SBOM%20Sharing%20Primer.pdf

It’s not only about the SBOM

SPDX
SBOM

CycloneDX
SBOM

VEX

SCITT
Statement

IN-TOTO
Attestation

CBOM

HBOM

Certificate
of compliance

Other
Attestation

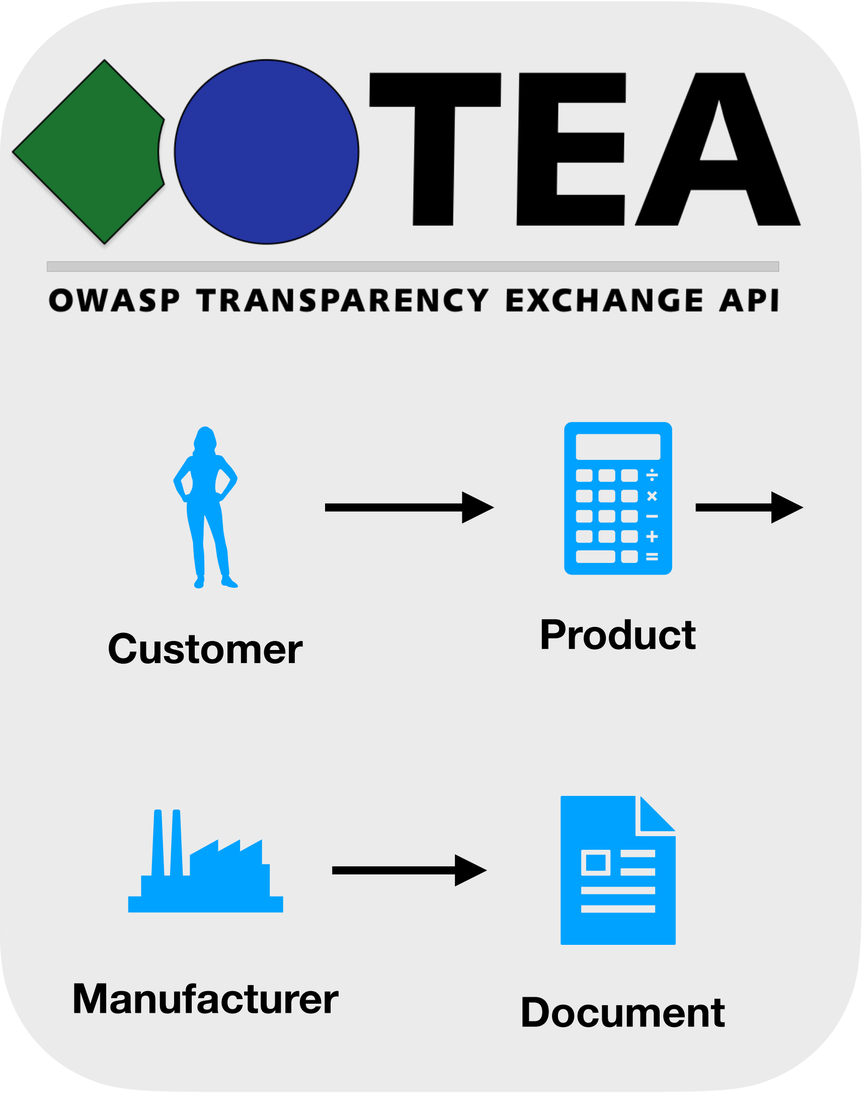
Many different attestations depending on sector, regulation and type of product.

TEA will be a standardised API

- The **Transparency Exchange API** will standardise publication and retrieval of transparency data for software and hardware.
- The API is based on a **discovery scheme** with a URN called Transparency Exchange Identifier
- The URN uniqueness is based on DNS names and existing product identifiers, from EAN barcodes to Package URLs and random UUIDs. The manufacturer defines the TEI.
- While TEA is standardised as part of the OWASP CycloneDX project, it's not specific to the CycloneDX format.
- TEA includes authentication and authorization, controlling who can access what information and who can publish what.

TC 54

ECMA
https://tc54.org/



TEA Components

Product

A product is something sold with software - an app, a server, embedded system, toy, IoT sensor etc

TEX Product index

TPI is a list of parts in a product, called TEA leaves. For each part there is a pointer (TEI or URL) for the TLI for each leaf.

TEX Leaf index

TLI is an index of all software versions for a product with indications of the state of the software version and reference to where a collection can be found.

TEI is a Transparency Exchange Identifier. A unique identifier for a specific product regardless of software version.
TEI is based on existing identities for a product, not replacing them.

TEA Collection

The TEA collection is the repository of current artifacts for a specific version of software in a given product.
The collection includes SBOM, VEX, SLSA attestations and more.

TEA
OWASP TRANSPARENCY EXCHANGE API

TEA
OWASP TRANSPARENCY EXCHANGE API

TEA Use cases

https://github.com/CycloneDX/transparency-exchange-api/blob/main/doc/tea-usecases.md

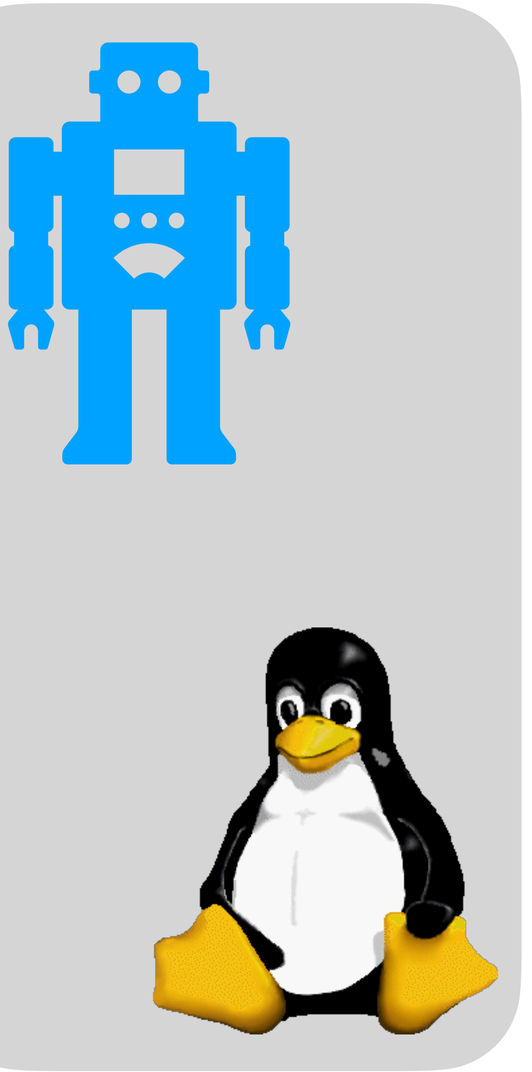
OWASP®

CycloneDX

Project Koala

Use case #1: Retail consumer

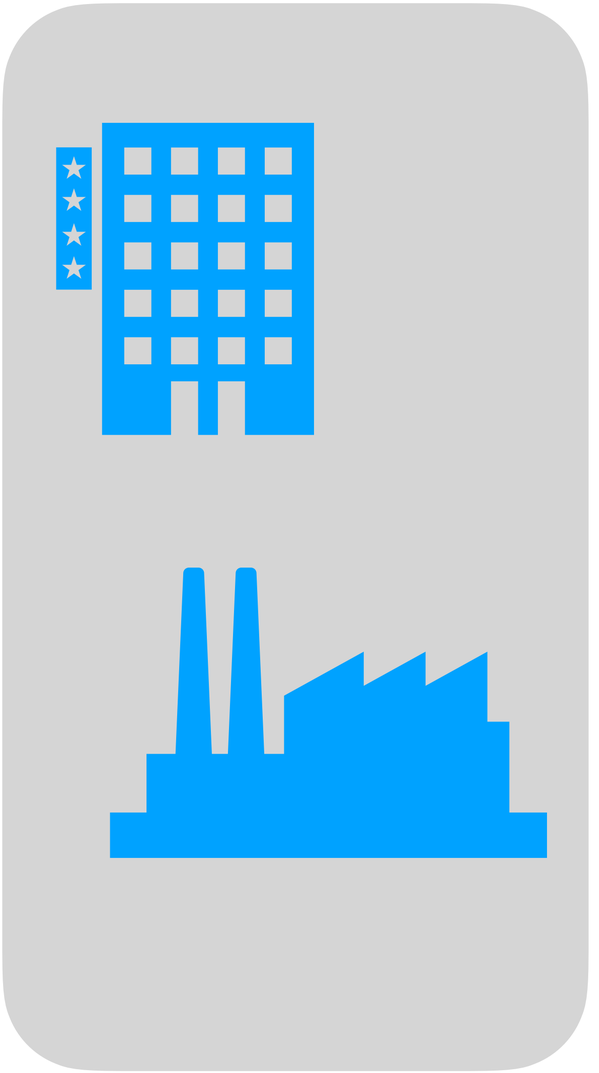
- Alice bought a gadget at the gadget store that contains a full Linux system. Where will she find the SBOM and VEX for the gadget?



Use case #2: Business consumer

- Acme LLC buys 3 000 gadgetrons from Emca LTD to be distributed over a retail chain. Acme runs an in-house vulnerability management system (Dependency Track) to manage SBOMs and download VEX files to check for vulnerabilities. Acme has products from exactly 14.385 vendors in the system.

- How will their systems get continuous access to current and old documents - attestations, SBOM, VEX and other files?



Use case #3: Regulators

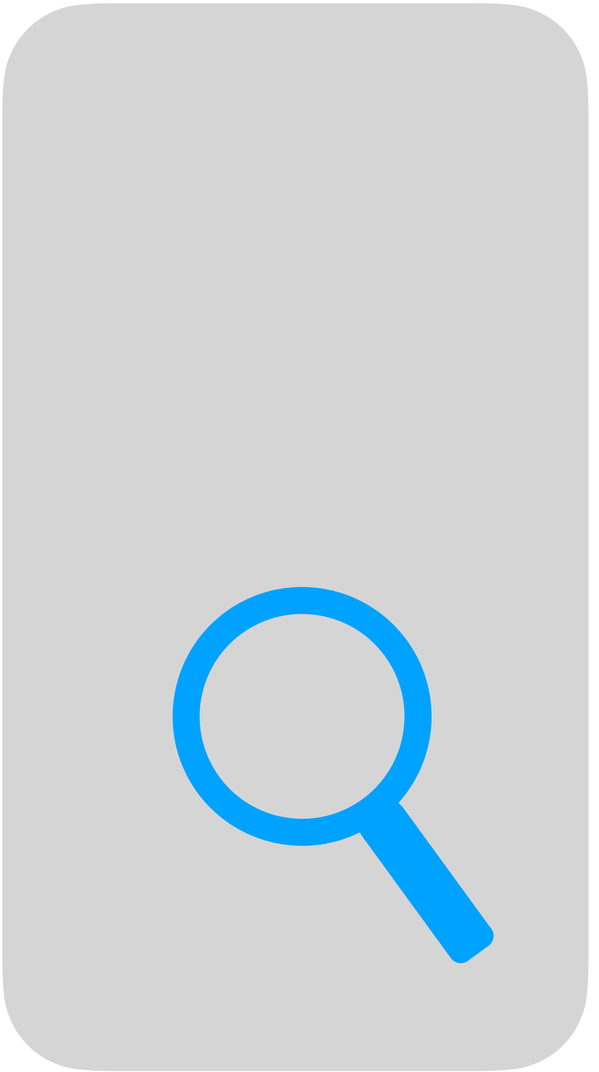
- Alice & Bob Enterprises AB has gotten a EUCC certification to get their Whola Firewall certified for CRA-compatible CE labeling. In order to maintain the certification the certifying body needs access to SBOM and VEX updates from A&BE in an automated way.

Use case #4: Potential customer

- Palme Auditors INC wants to buy the ACME SWISH product from a vendor. They want to examine vulnerability handling and get some insights into the products before making a decision.

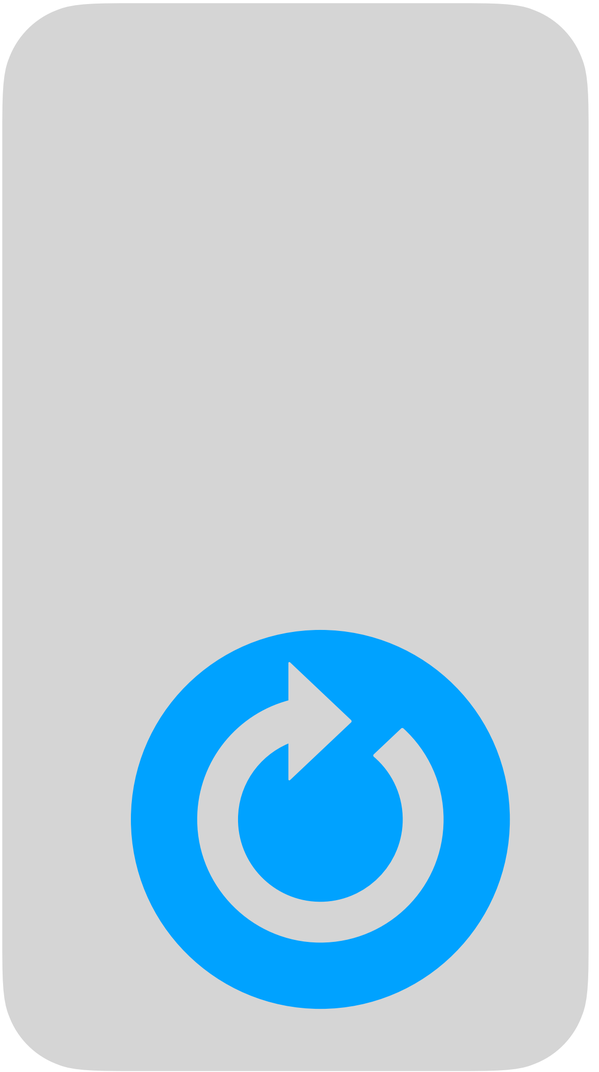
Use case #5: Open Source user

- Palme Inc considers using the asterisk.org open source telephony PBX.
- They need to make an assessment before starting tests and possible production use.
- The can either use the Debian package, the Alpine Linux Package or build a binary themselves from source code.
- How can they find the transparency exchange data sets?



#6: Components for inclusion in products by developers

- A **developer** needs a component - software, library (open source and/or proprietary) to include in a product - publicly available (may be commercial) or in an internal system
- Developer can be individual, company or public sector
- They need insight into the library



TEA
OWASP TRANSPARENCY EXCHANGE API

The way forward

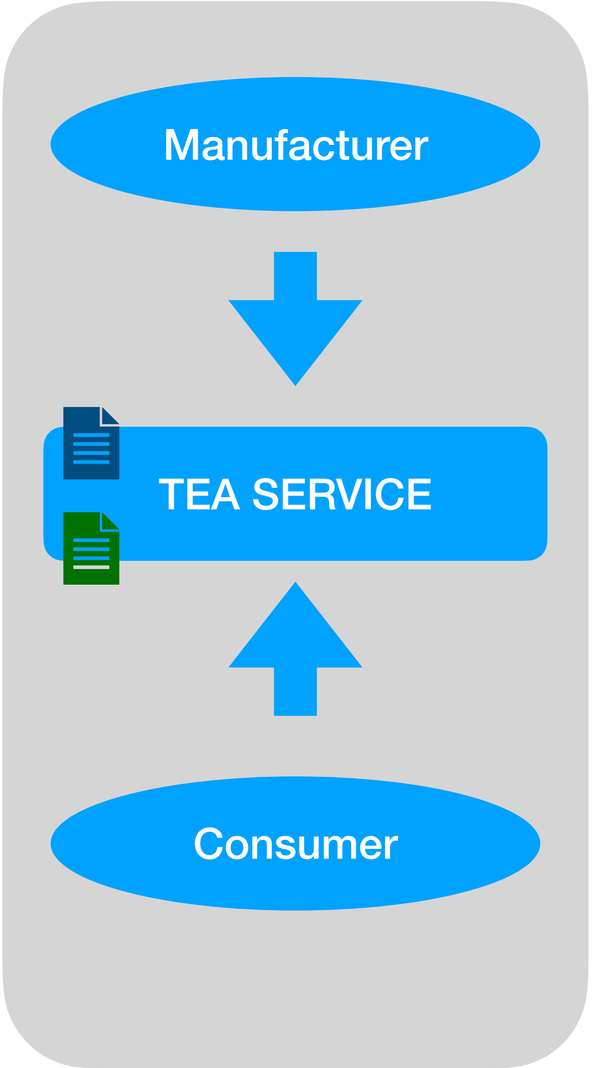
OWASP®

CycloneDX

Project Koala

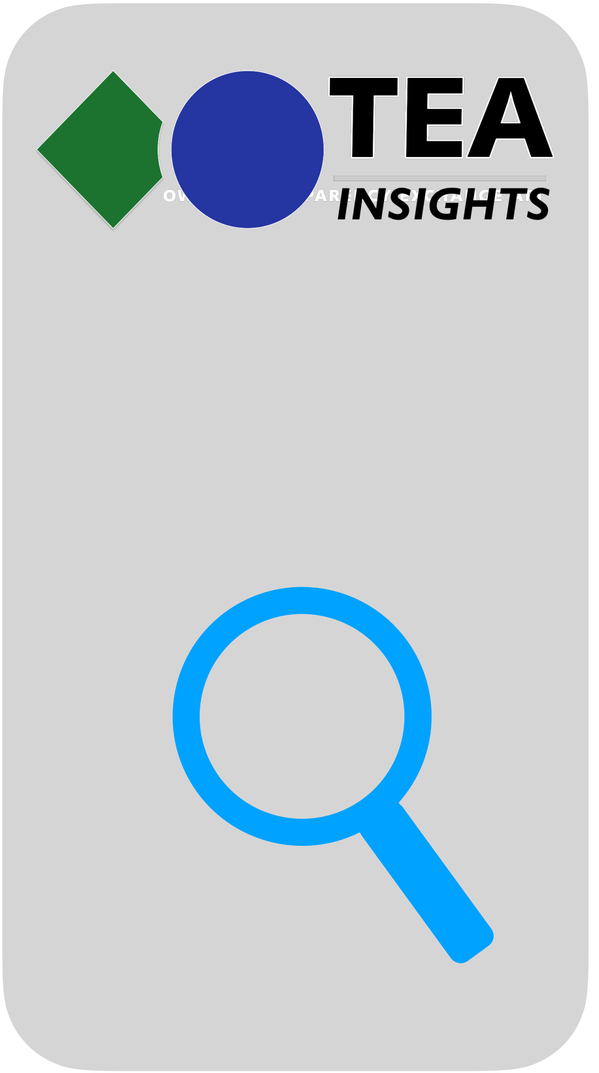
Step 1: Artifact retrieval

- The first focus is to get the basic API done for distribution of various kinds of artifacts
- Publication API for manufacturers
- Consumer API for customers
- Discovery mechanism to find the API servers on the net
- Getting information about new versions and life cycle events (CLE)
- Support for hosted solutions (multi-tenant)



Step 2: Insights

- In a coming release we'll add insights
- Insights allows for a "limited transparency" that can be handled within the API using an expression language that can be tightly scoped or outcome-driven.
- Insights allow customers to query like:
  - Do any of my licensed produces from use Apache Struts?
  - Are any products vulnerable for log4shell - is there any action I need to take?



OWASP

TEA
OWASP TRANSPARENCY EXCHANGE API

TEA discovery: TEI
Transparency Exchange Identifier

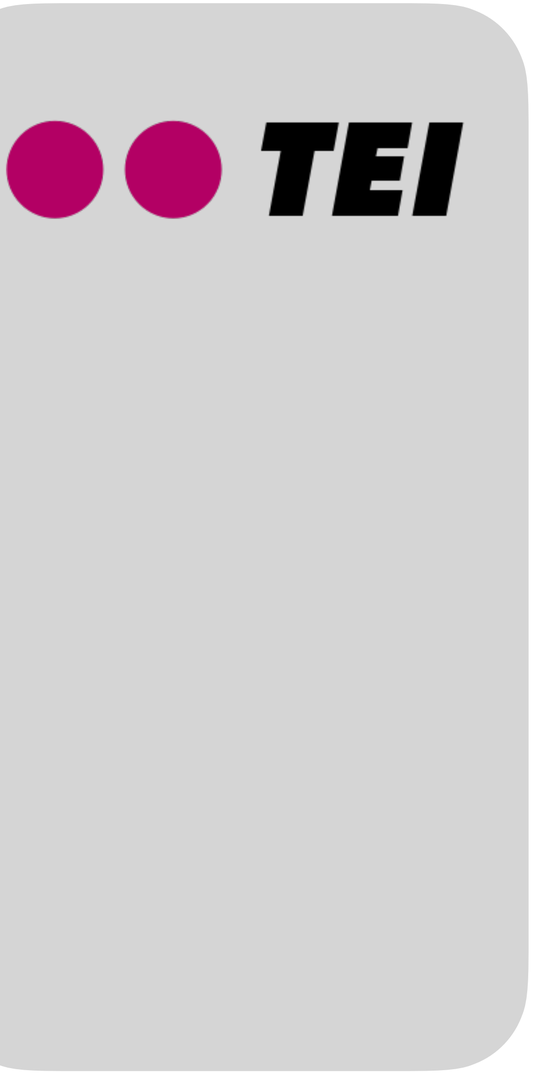
OWASP®

CycloneDX

Project Koala

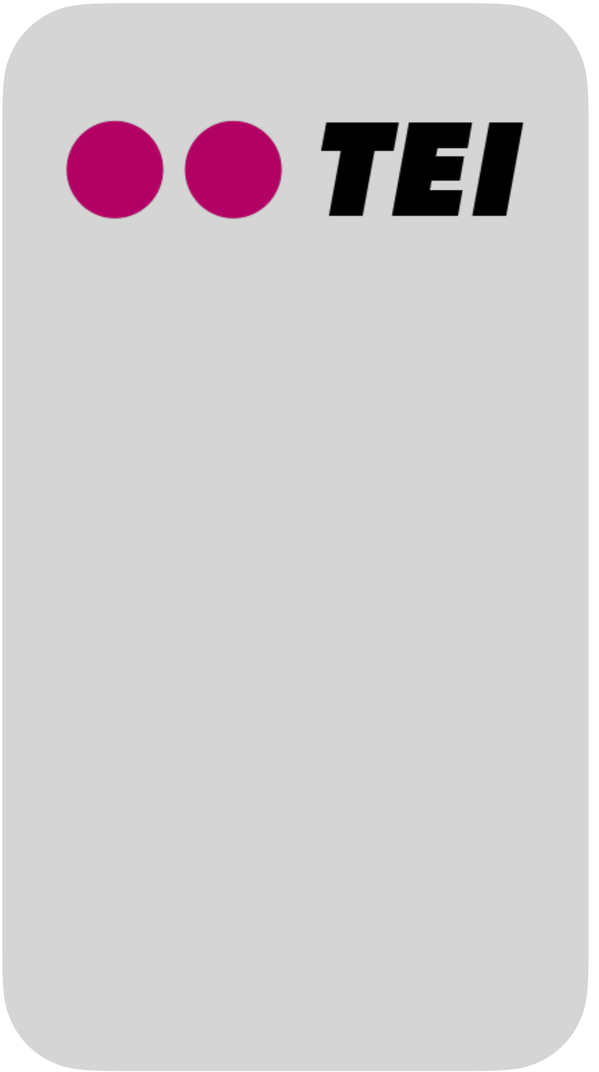
TEI: Transparency Exchange Identifier

- A **unique identifier** based on existing product identifiers
- Base for the **discovery** process
- An identifier that can be shown in the software itself, in MUD or be retrieved by other means
- This identifier needs to be a persistent identifier for a specific **product**, regardless of version
- The identifier needs to be globally unique - which is easiest to base on a **registered domain name**. Other solutions require a registry, a registrar and a registration process.



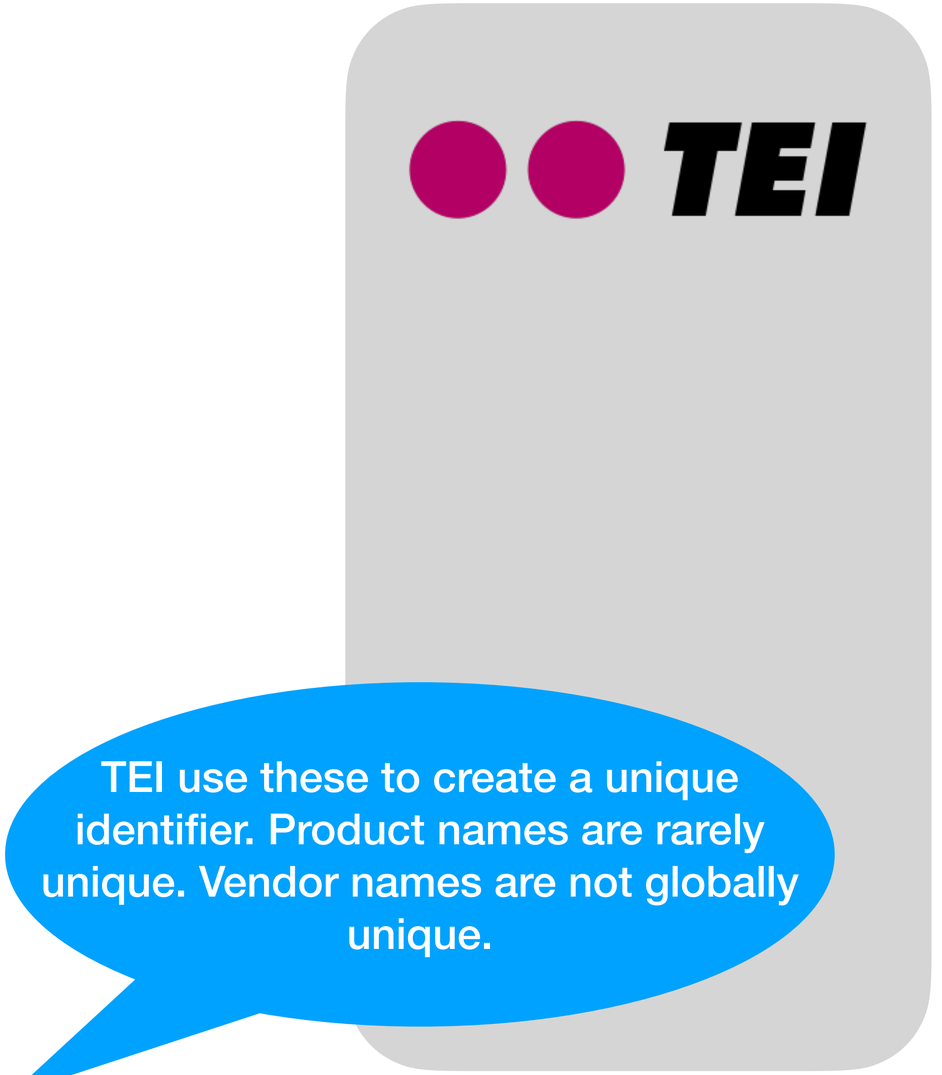
TEI basics

- It’s an identifier, not a locator
- Using standard DNS resolution, can point to multiple locations (URLs)
- The manufacturer defines the TEI for a product being sold
- One product can have multiple TEIs



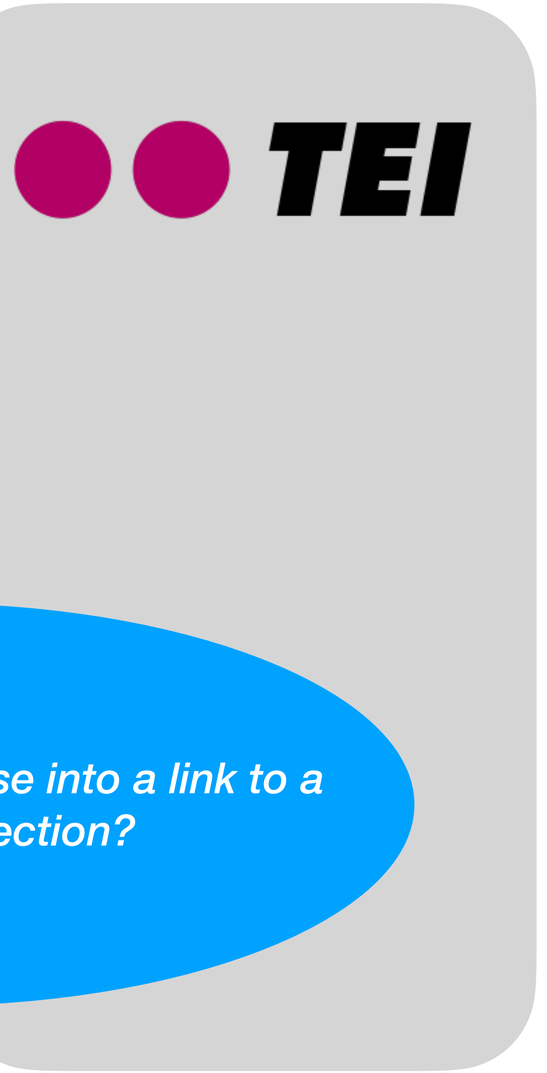
Existing product identifiers

- EAN code on marking
- App store vendor name and product name
- Product name and vendor
- SWID tag
- In app QR code (if possible) or identifier
- SKU - Stock Keeping Unit



Built on existing identifiers

- PURL - Package URL (soon an ECMA standard)
  - urn:tei:purl:
- EAN (Bar code identifier)
  - urn:tei:ean:
- Hash values
  - urn:tei:hash:



Example: URN TEI name space: UUID

- **urn:tei:uuid:** for a company specific name and product identifier as UUID
  - `urn:tei:uuid:products.example.com:d4d9f54a-abcf-11ee-ac79-1a52914d44b1`
  - `urn:tei:uuid:<name based on domain>:<unique identifier>`
  - `urn:tei:uuid:products.example.com:d4d9f54a-abcf-11ee-ac79-1a52914d44b1?version=123.32`
- The uniqueness of the name is the domain part that has to be registred at creation of the TEI.
- The unique identifier has to be unique within the domain. It can be an UUID or any other product identifier.
- A TEI belongs to a single product. A product can have multiple TEIs.
  A product can consist of multiple parts, each with a single TEA index.

How to get the TEI

QR CODE on package

On the invoice / delivery note

In the “about” box of a GUI

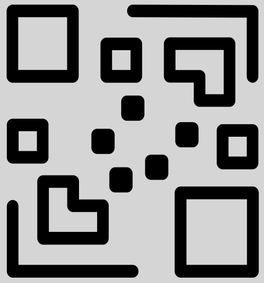
In the support portal

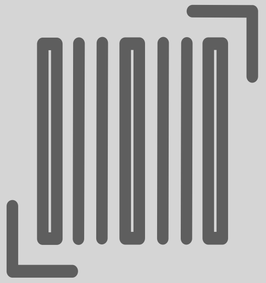
One product

TEI

TEI

TEI





TEI

Finegrained authorization

- Authentication and authorization are used in order to protect access
- Something easy to manage and persistent, we don’t have to spend a lot of time copy/pasting tokens or credentials between systems.
- A HTTP **bearer token is used**. How the token is aquired is out of scope. We may want to standardize how a system tells the API client that the token expired (if not already standardized)
- If the token is a signed JWT token (like in OpenIDconnect) authorization could be included in the token. **The TEA API doesn’t need to specify the token content.**
- Scenario: I log in to Edvina support portal. The Edvina portal knows which products I have. I request an **TEA API access token**. Install that token in Dependency Track by copy/paste. The token is time limited.

TEA
OWASP TRANSPARENCY EXCHANGE API

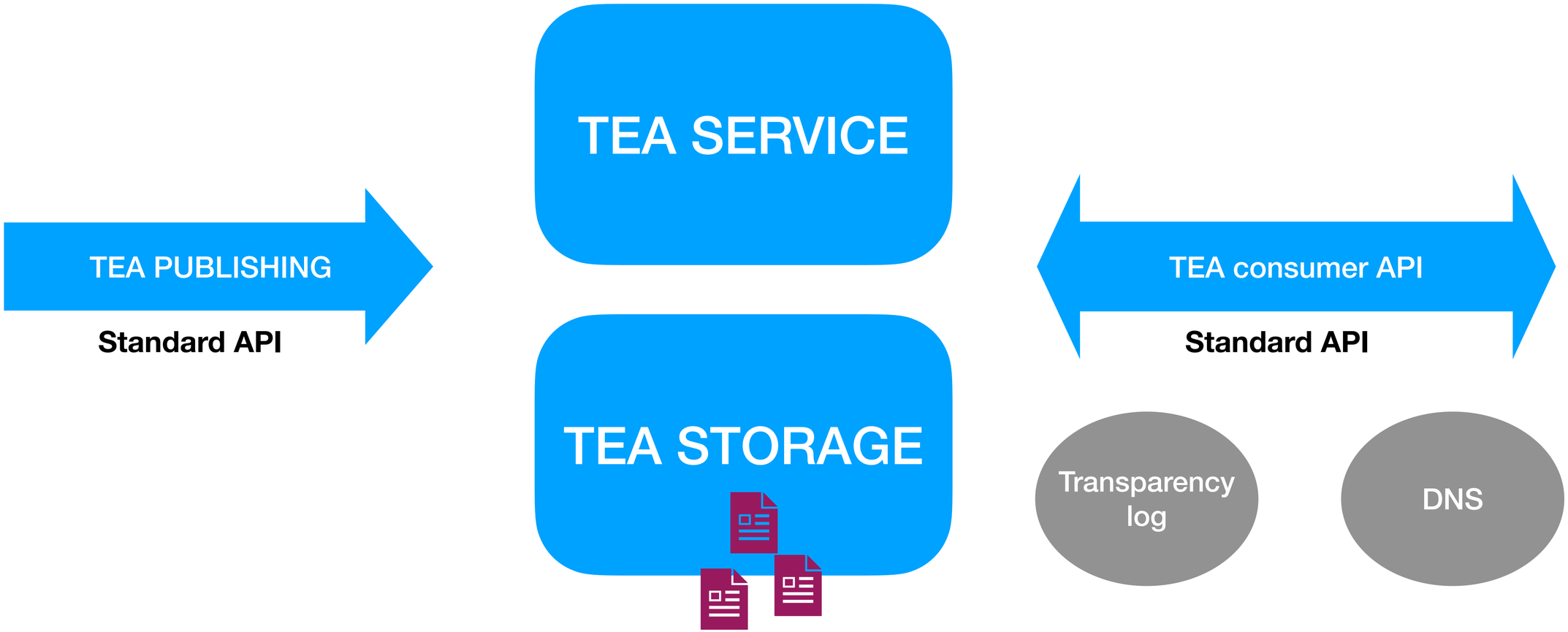
The TEA platform

OWASP®

CycloneDX

Project Koala

A TEA platform



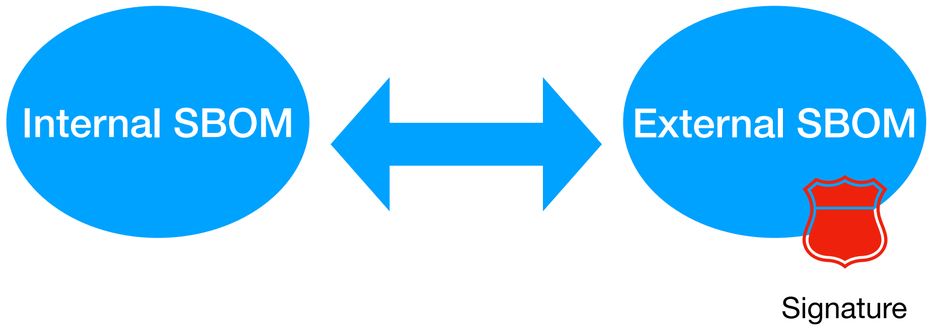
Types of “TEA-pots”

Internal compliance TEA SERVICE

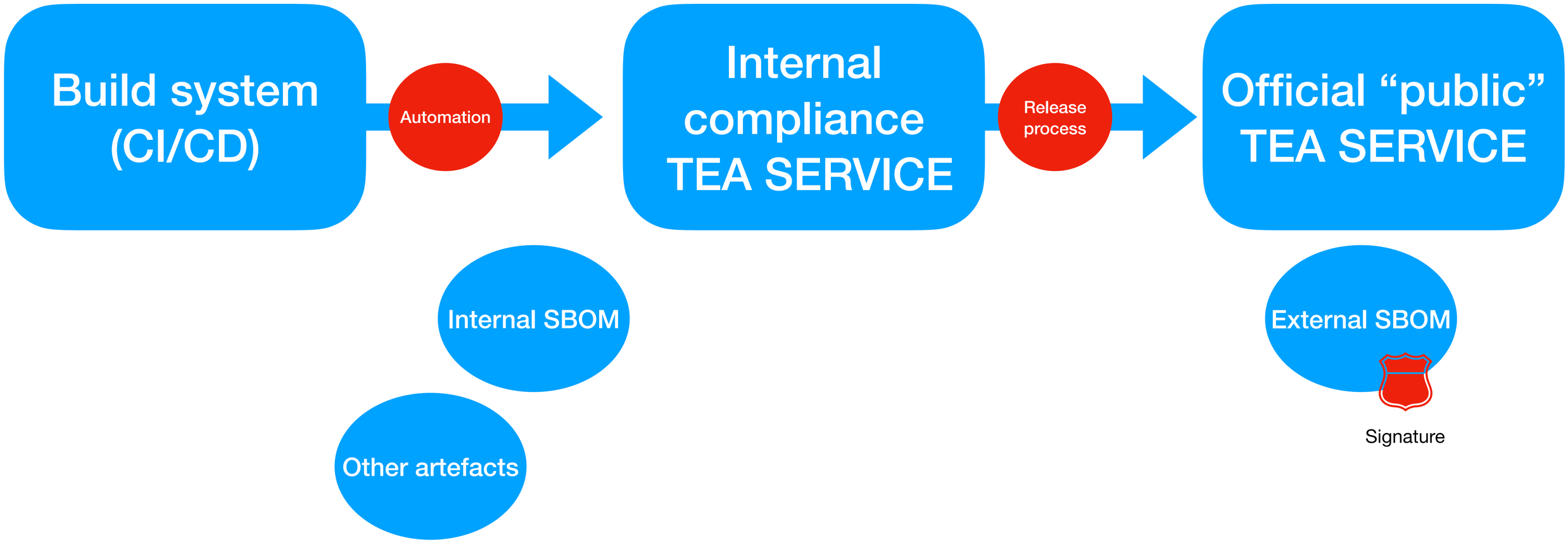
Official customer TEA SERVICE

3rd party TEA Service

Hosted service



Example of a TEA Workflow



The publication process

Artefact generation

- SBOM generator
- Manual update of SBOMs
- VEX file generator
- VEX file triage
- ??? publication
- Other artefacts (Certificate of compliance)
- Automatic attestations (In-Toto, cycloneDX etc)

Artefact publication

- CLI client for CI/CD tool chains
- Web GUI
- “SBOM” Platform
- Integration in existing platforms

TEA PUBLISHING

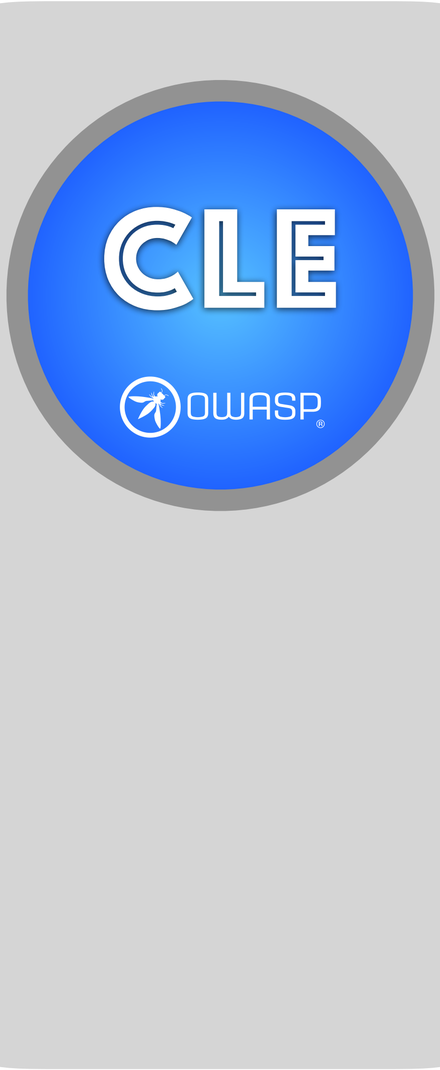
Standard API

- TEA SERVICE
- TEA STORAGE

OWASP Common Lifecycle Enumeration

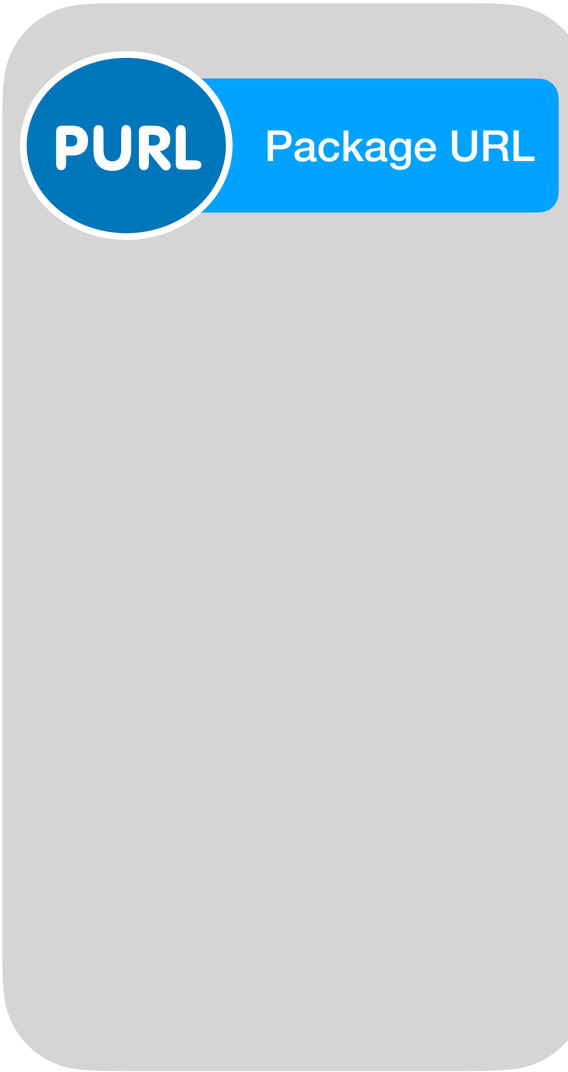
- Will be part of the TEA API
- Mapping status of a product or component
- Is it still supported, still managed?
- “End of life”, “End of support”, “End of security fixes”, “Replaced”
- Possibly part of SBOM, part of TEA

https://owasp.org/www-project-common-lifecycle-enumeration/



What about PURL

- **PURL, Package URL** is an identifier used for vulnerability management
- TEI is a discovery mechanism, which can include a PURL as a product identifier
- PURL is not primarily built for discovery
- TEA will make sure the user gets the correct identifier for a product
- PURL is part of the CycloneDX ECMA TC54 standardisation effort



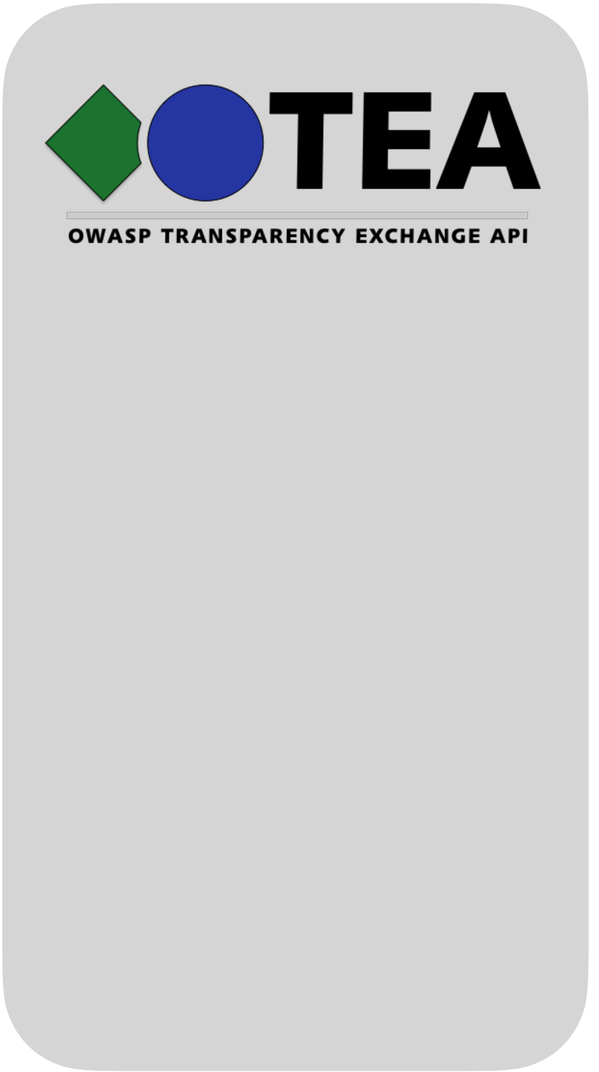
https://github.com/package-url/purl-spec

Status update

Generic modules	Consumer	Publisher
Discovery: Stable	Open API spec: Alfa	Workflow: Started
Object model: Proposal	Security arch: Started	Open API spec: Alfa
Use cases: Done	Authentication, authoritization Not documented in Github	Security arch: Not yet
	Implementation: Not yet	Implementation: Not yet

Summary

- OWASP TEA is going to be the standard for transparency exchange
- Fits nicely to worldwide legislations for software supply chain security
- One API for publication, another for consuming artefacts
- Hackathon end of Q1 2025 to test implementations
- Meet us at OWASP Global AppSec Eu in Barcelona! May 26-28.



Join the work!

We are working on writing specifications for the API and the various formats.

Join the OWASP CycloneDX Transparency Exchange API working group today to participate. We have a channel in the CycloneDX slack space to communicate.

`https://github.com/CycloneDX/transparency-exchange-api`

`https://cyclonedx.org/about/participate/`

OWASP

TEA
OWASP TRANSPARENCY EXCHANGE API

TC 54
https://tc54.org/

CycloneDX

Project Koala