FOSDEM 2024

SBOMs that you can trust
the good, the bad, and the ugly

Miguel Martinez Trivino, Chainloop
Daniel Liszka, Chainloop
Hi, we are happy to be here!

Daniel Liszka
co-founder and Chainloop maintainer, previously Engineering at Bitnami and Product at VMware. Dad, previously traveller, biker, and skier ;)

Miguel Martinez
co-founder at Chainloop. 10+ years designing, implementing and operating Software Supply Chain automation at Bitnami/VMware. The IT-crowd fan

☆ if you like what we do, give our GitHub [chainloop-dev/chainloop](https://github.com/chainloop-dev/chainloop) a star :) ☆

[bit.ly/adoc8]
Trustworthy SBOM

● What does it mean?
● Why now?
● How can we achieve it?
● Demo
Yet another SBOM talk
Building the Trust Layer - What’s Trust?

- Can I uniquely identify an SBOM?
- Will it be available when I need it?
- Can I trust that the content has not been tampered with?
- How was it built, from whom or where does it come from?
- Is it complete and consistent?
- Does it even exist?
Building the Trust Layer - Why?

SSC Security bar has been raised and SBOM is just another deliverable

“Any software can introduce vulnerabilities into a supply chain[…] it’s critical to already have checks and best practices in place to guarantee artifact integrity, that the source code you’re relying on is the code you’re actually using[…]”

https://slsa.dev
An SBOMs is yet another artifact as important as the artifact they reference

- They must meet the highest security posture.
- They can get compromised too.
An SBOM that you can’t trust is useless and in fact dangerous...

...we need our SBOMs to be uniquely identifiable, unforgeable, complete and available
Building the Trust Layer - Pattern

Core components

- Decentralized storage
- Content Addressable Storage
- Attestations
- Contracts
“A software attestation is an **authenticated** statement (metadata) about a software artifact or collection of software artifacts ... a generalization of raw artifact/code signing - slsa.dev
Attestations will wrap SBOMs with additional information and a signature to enable integrity and provenance verifications.
Content-Addressable Storage (CAS) is a system that organizes and retrieves data based on the data's content, rather than its location or name, ensuring data integrity and immutability.
Stored SBOMs will be unique, identifiable and integrity verifiable.
Building the Trust Layer - Implementation

SBOMs that you can trust its identity, integrity and origin

- Enforcement: Contracts / Policies
- Provenance / Verification: sigstore, in-toto, SLSA
- Identification: OPEN CONTAINER INITIATIVE, OPEN CONTAINER INITIATIVE
- Integrity: OPEN CONTAINER INITIATIVE
- Availability: Federated Storage, Multiple backends
Chainloop is an Open Source Metadata Vault for your Software Supply Chain metadata, SBOMs, VEX, SARIF files and more

github.com/chainloop-dev/chainloop
**Trusted Supply Chain Metadata**

**Federated storage**

**Federated Content-Addressable Storage (CAS)** works across backends enabling advanced routing for replication, geolocation, retention rules, ...
Contracts are declarative requirements of the pieces of evidence a development team needs to provide.

```json
schemaVersion: v1
materials:
  - type: ARTIFACT
    name: binary
    output: true
  - type: SBOM_CYCLONEDX_JSON
    name: sbom
    runner:
      type: "GITHUB_ACTION"
```

Trusted Supply Chain Metadata Enforcement
SBOMs that you can trust, on identity, integrity and origin. Also storage compliant and enforced.

github.com/chainloop-dev/chainloop
Demo
Demo

1 - Setup, collection and storage
- Collect CycloneDX SBOM from GitHub Action
- Wrap it in in-toto attestation
- Store it in Azure Blob Storage and OCI registry on GCP
- Send it to
  a. Dependency-Track
  b. guacsec/guac

2 - SBOM + VEX use-case

3 - SBOM sharing
The bar has been raised

Metadata compliance and security bar is being raised and **SBOM trust is the next challenge**...

... but you can get a head start with open source security tools today :)
Thank you

Find us in Discord

• https://twitter.com/migmartri

• https://twitter.com/danlishka

• chainloop-dev/chainloop

☆ if you like what we do, give our GitHub chainloop-dev/chainloop a star :) ☆