



# anchore

**Secure Your Supply Chain: Adding a Software Bill of Materials to Your Containers to Improve Vulnerability Scanning**

# Presenter

---



Paul Novarese  
Senior Solutions Architect  
Anchore  
[pvn@anchore.com](mailto:pvn@anchore.com)



Jeff Atwood @codinghorror ...

As an addendum to "delete happy talk" don't spend a lot of time introducing yourself / your topic in presentations. Nobody cares. BEGIN.

3:40 PM · Oct 13, 2016 · Twitter Web Client

---

32 Retweets 3 Quote Tweets 102 Likes

# Agenda

---



- 00 Types of attacks and why software supply chain attacks are different
- 01 Why container visibility is important
- 02 What an SBOM is and what it does
- 03 How to further secure your containers



# Attack Types

Everything old is new again.

anchore

# Story Time



Mike Mearls  
@mikemearls

...

Two things I learned in 2018:

1. Don't ask people what they do for a living. Ask what keeps them busy. That lets people talk about what's really important to them.
2. Don't use analogies to explain things. You won't get far explaining something by making it into something else.

1:15 AM · Jun 1, 2019 · Twitter for iPad



# Story Time



## Douglas Haig

"Bullets have little stopping power against the horse" (1914)

# WESTERN FRONT, 1915

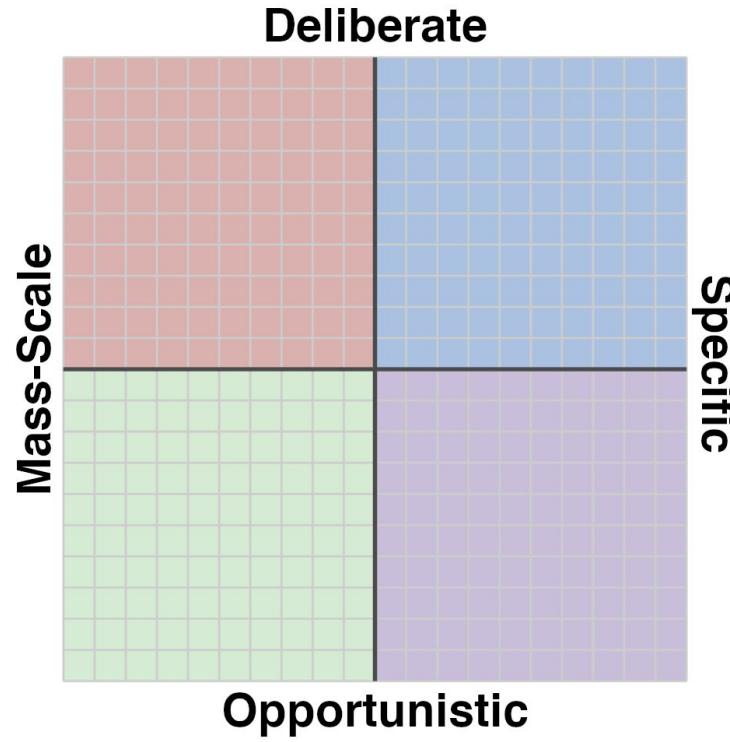


haha machine gun go brrrrr

NOOOOOOO!!!! YOU CAN'T DEPLOY  
MILES AND MILES OF BARBED WIRE  
AND USE EXTREMELY ACCURATE, HIGH-  
POWERED AUTOMATIC RIFLES!!! I HAVE  
INVESTED A TON OF TIME AND EFFORT  
INTO TRAINING THESE CAVALRINOS!!!

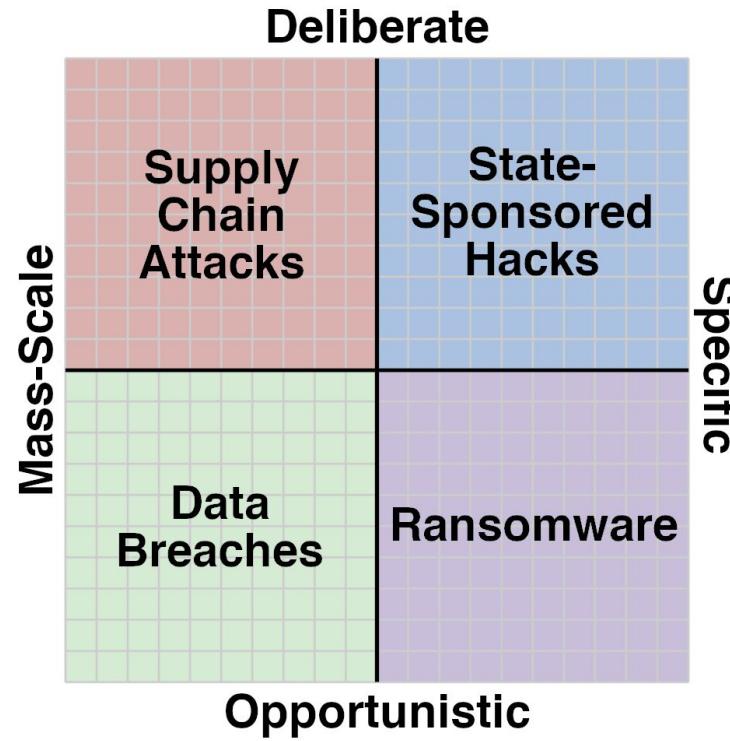


# Types of Attacks



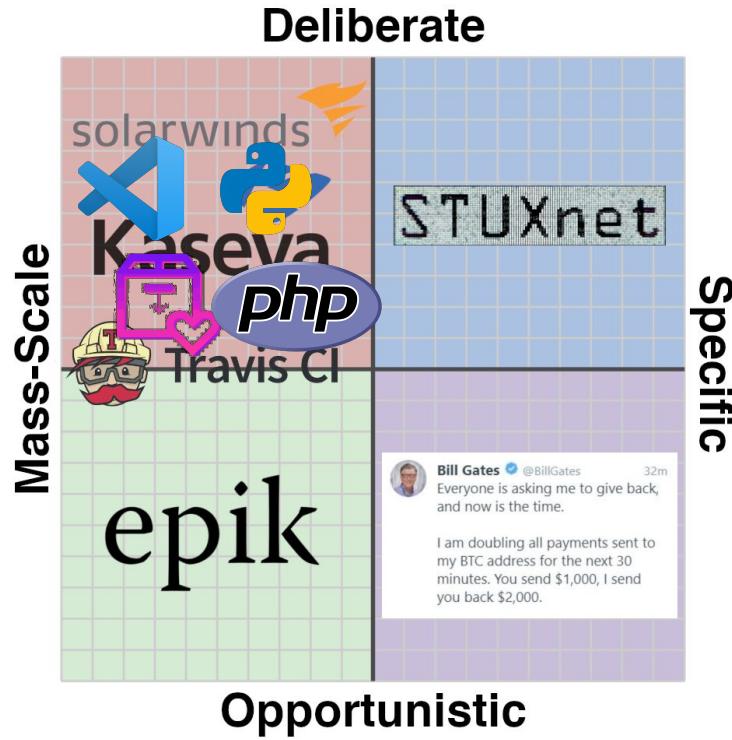


# Types of Attacks

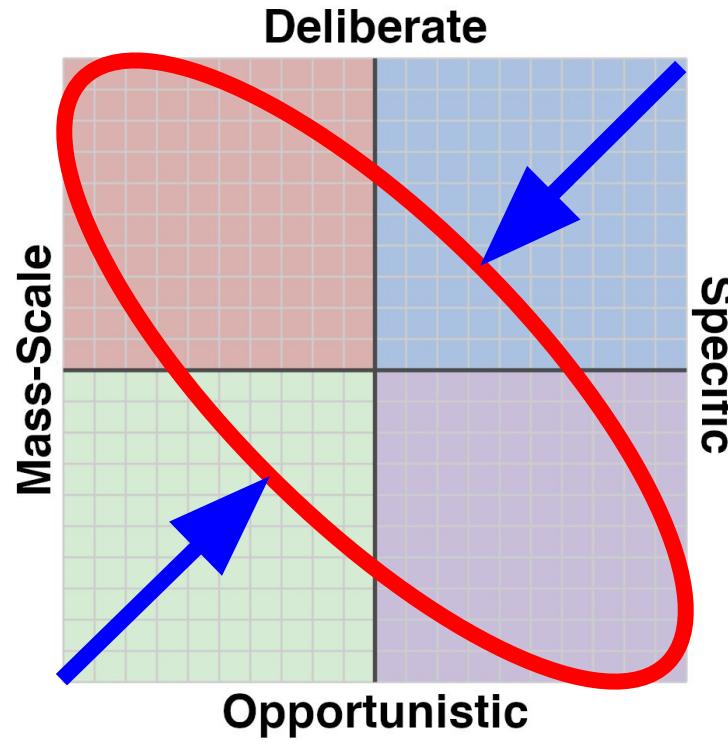




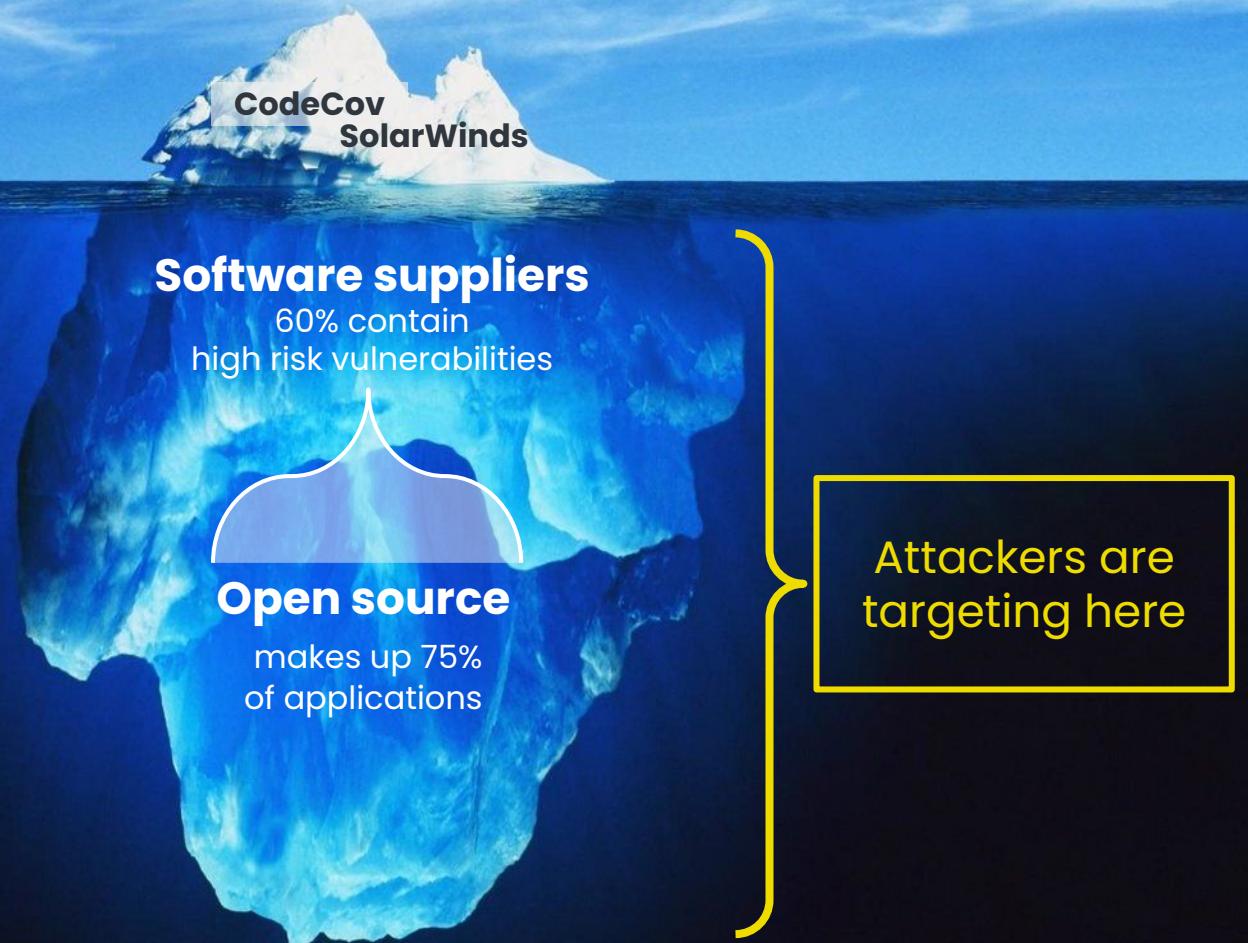
# Types of Attacks: Examples



# "It Won't Happen to Me"



# Risk in the Software Supply Chain



A large, white iceberg is centered in the image, floating in a deep blue ocean under a clear blue sky with wispy clouds. The iceberg is mostly submerged, with only a small portion visible above the water's surface.

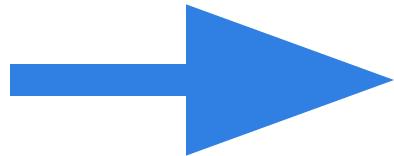
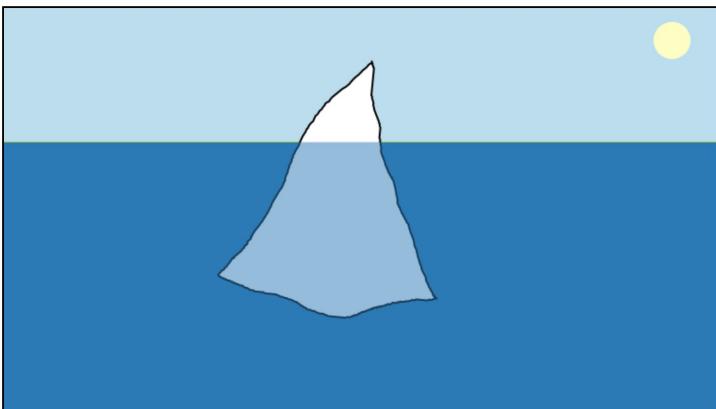
**WRONG**

A large, bold, red "X" is overlaid across the center of the image, covering the entire word "WRONG".

## **Iceberger**

Draw an iceberg and see how it will float.

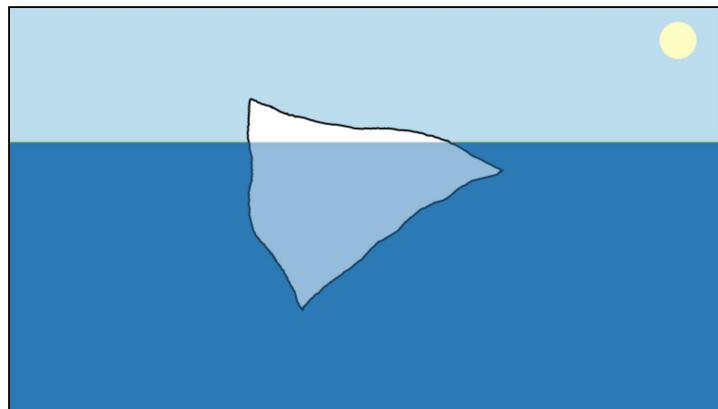
(Inspired by a [tweet by @GlacialMeg](#))



## **Iceberger**

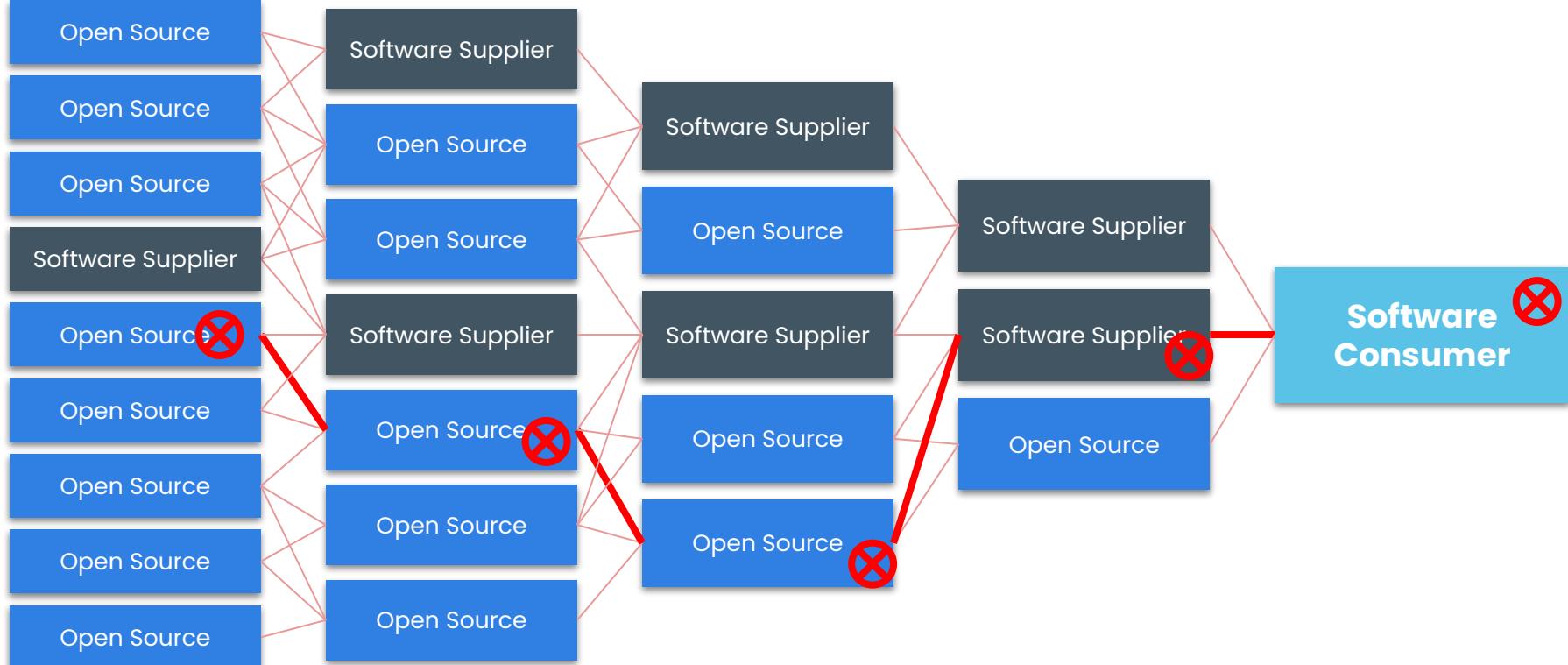
Draw an iceberg and see how it will float.

(Inspired by a [tweet by @GlacialMeg](#))





# Software Supply Chain: The Problem

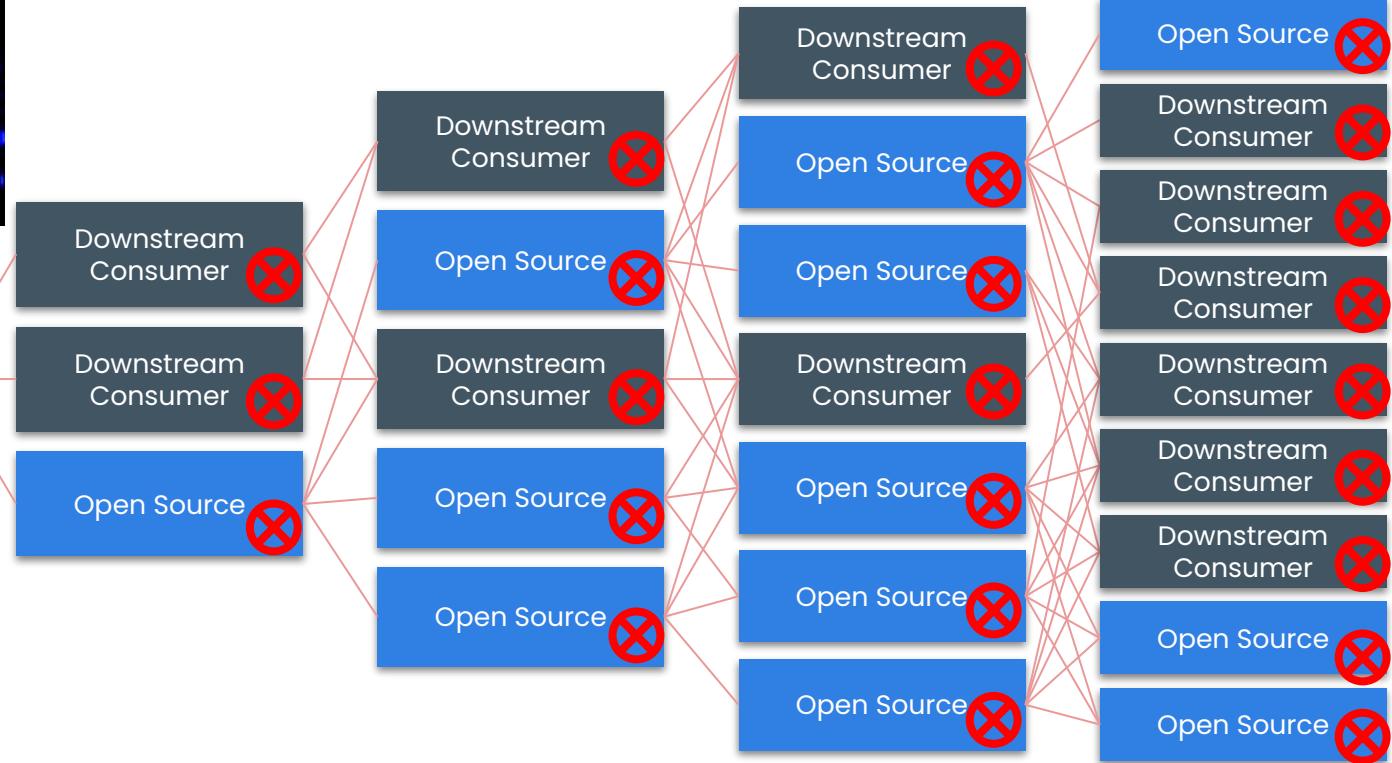




# The Reverse Funnel



Compromised Package





# What's in this Image?

Knowing is half the battle.

anchore



Dan Kaminsky

@dakami

...

Amateurs think about vulnerabilities, professionals  
think about vectors.

4:04 PM · Aug 12, 2017 · [Twitter Web Client](#)

# Beyond CVEs: Overlooked Vectors



01

## Software Vulnerabilities

Known vulnerabilities affecting software components that containers and applications depend on - OS packages, direct application dependencies.

02

## Malware and Trojan Horses

Malicious code injected into regular application executables during the build process.

03

## Software Overrides

Attacks that result in unintentional versions of (typically) dependencies being installed.  
Name-squatting, max version attacks, typosquatting.

04

## Credentials

Unintentional inclusion of dev or prod secrets, keys, or other credentials accidentally included in the container.

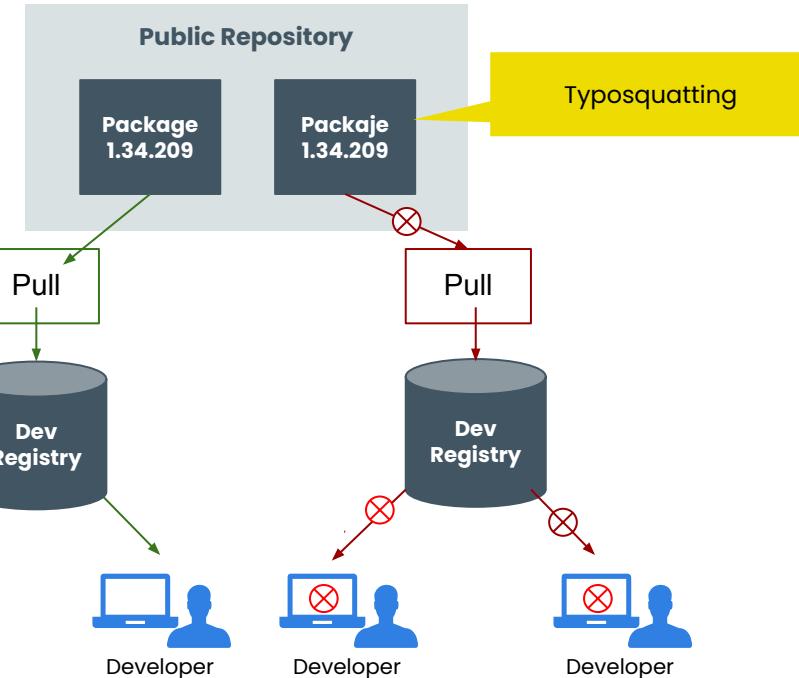
Software vulnerabilities (often reported as CVEs) are critical to detect and report, but many other build-time attack vectors must also be considered.

**\*slaps roof of container\* this bad boy  
can fit so many supply chain attacks in it**





# Know What You're Getting



```
"dependencies": {  
    "express": "^4.3.0",  
    "dustjs-helpers": "~1.6.3",  
    "continuation-local-storage": "^3.1.0",  
    "pplogger": "^0.2",  
    "auth-paypal": "^2.0.0",  
    "wurfl-paypal": "^1.0.0",  
    "analytics-paypal": "~1.0.0"  
}
```

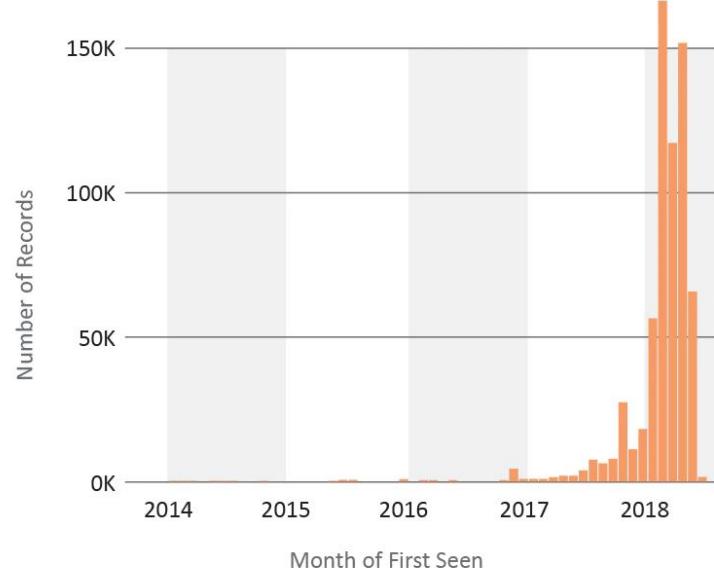
Dependency Confusion

# What Is Cryptomining?



- Malware that infects computers, turning them into zombies used to mine for cryptocurrency without the users knowledge
- Steadily on the rise since 2017
- A prevalent form of attack in misconfigured Kubernetes clusters
- Two well-known attacks:
  - Kubeflow
  - ArgoCD

**Figure 1.** Cryptocurrency Mining Malware Detections from 2014-2018, courtesy of several CTA members





# What is Image Typosquatting?

- Humans make mistakes, there is no spell check when pulling images
  - Ngnixx:latest,
  - Postgress:latest,
  - MYSQLL:latest
- Typosquatting can be combined with cryptomining by hiding miners in a misspelled name or tag
- And yes, this really does happen...

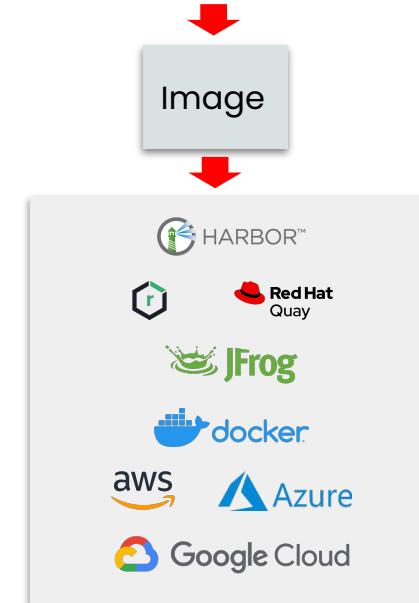
*"Typosquatting and credential stuffing are two of the most common ways attackers try to target companies' container infrastructure and the Docker-image supply chain. Attacks are up nearly 600% in H2 2020 compared with the same period a year ago."*

**DARK**Reading

# What is Registry/Repository Poisoning?



- Registry and/or Repository Poisoning is when an attacker plants a malicious image in the victim's registry or repository
- Registry/Repository Poisoning is particularly effective if the image appears benign. This can then easily proliferate throughout your environment





# Addressing the Problem

---





# What's a SBOM?

And what can it do for me?

anchore



**YOU NEED AN  
ACCURATE  
INVENTORY OF YOUR  
ASSETS BEFORE  
YOU CAN PROTECT THEM**

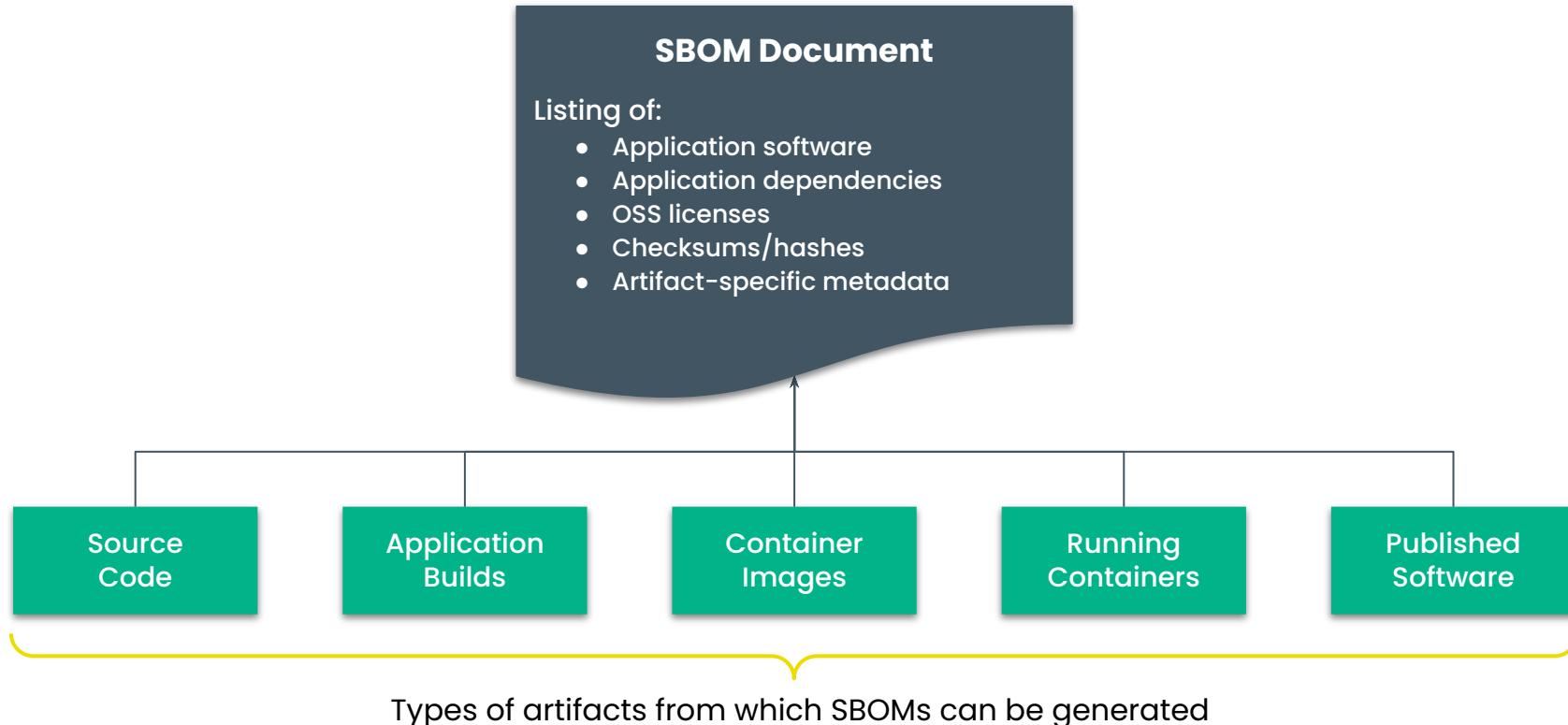


# What is an SBOM?





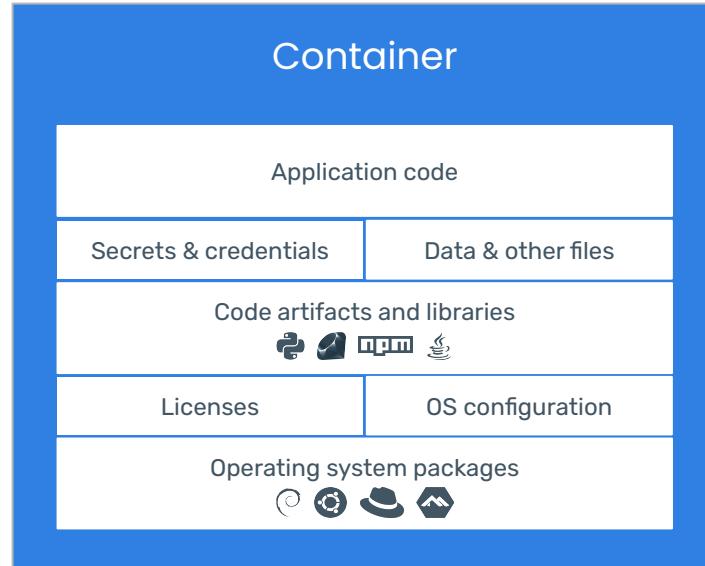
# What is an SBOM?





# Containers Provide

...an easy way to  
package and deliver

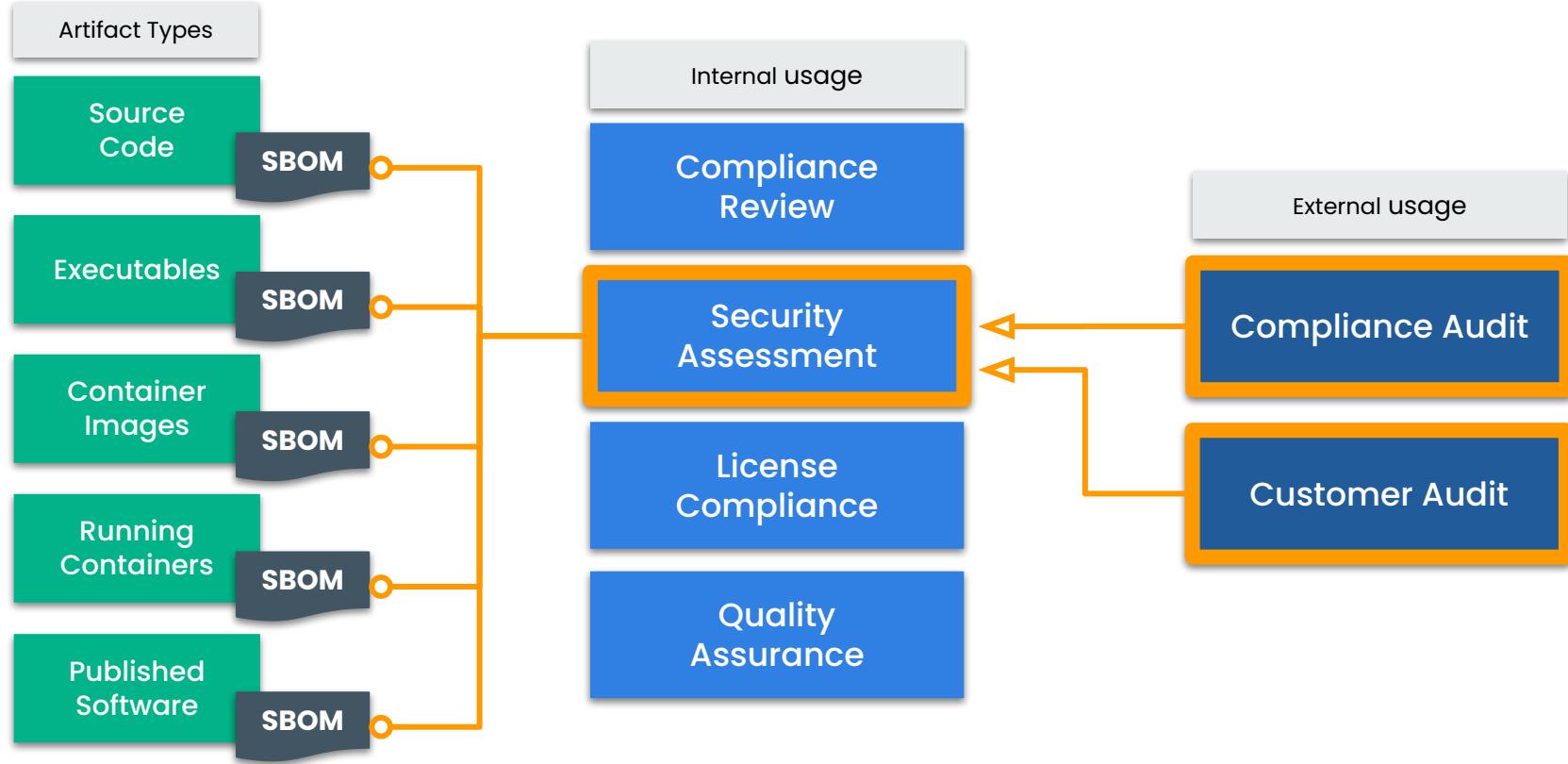


## SBOM Includes

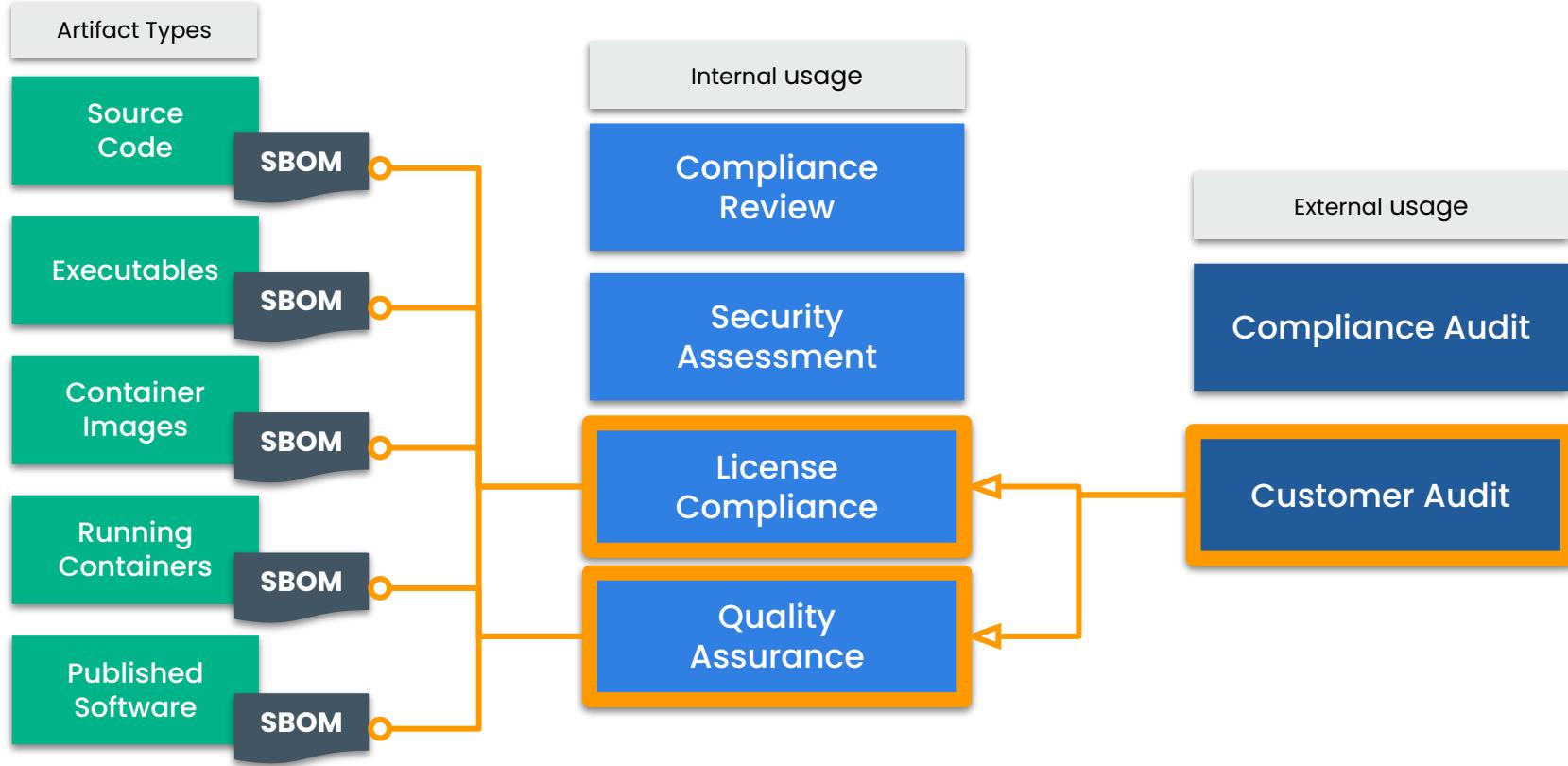
- App. dependencies
- OS Packages
- Licenses
- File data
- Configuration files
- Container metadata



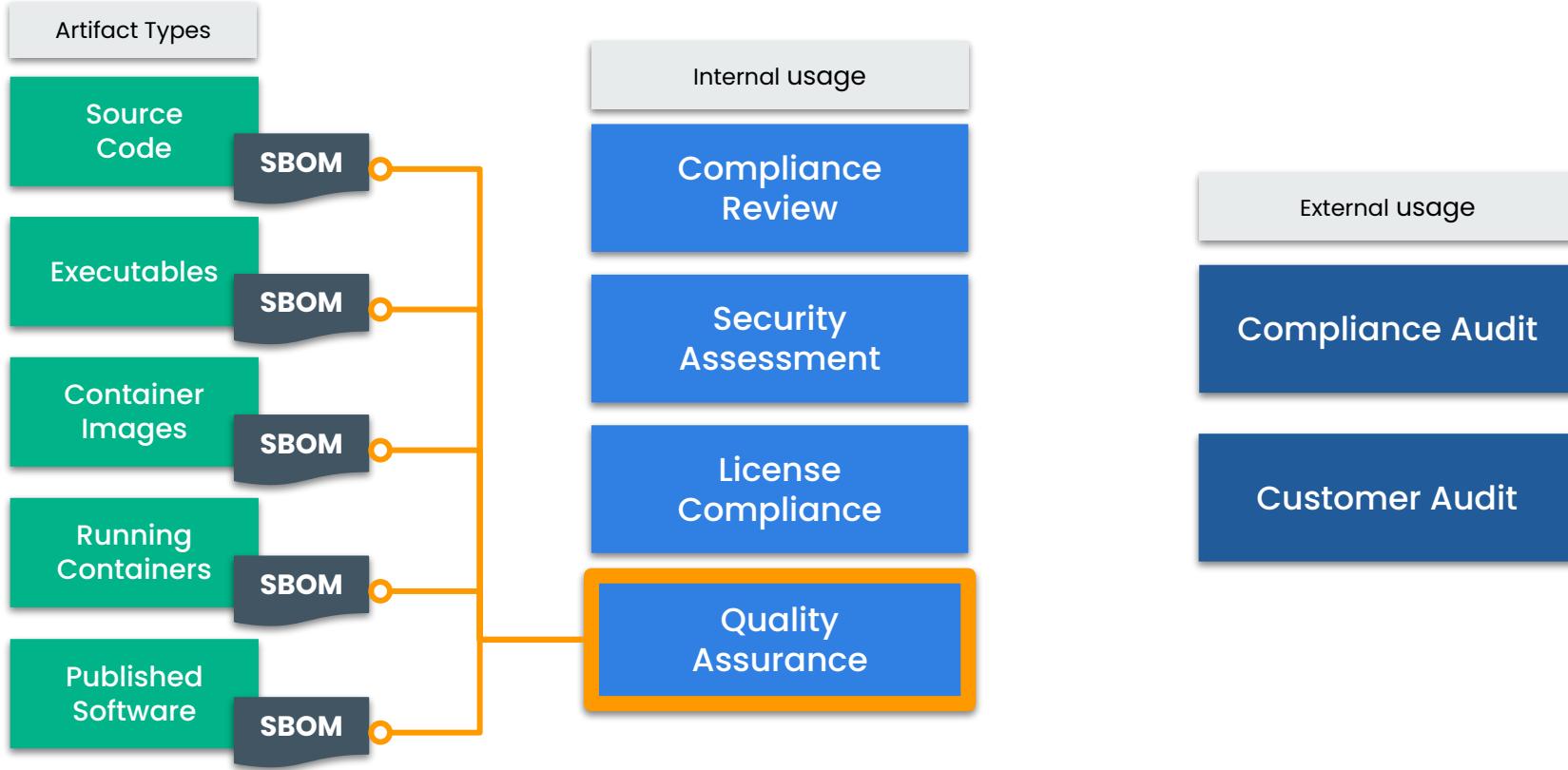
# SBOM Use Case: Security Incidents



# SBOM Use Case: Cross-Org Compliance



# SBOM Use Case: Application Complexity





# Existing SBOM formats

	<b>SPDX</b> "Software Package Data eXchange"	<b>CycloneDX</b>	<b>SWID</b> "Software ID"
<i>Organization</i>	SPDX Workgroup (~20 orgs) under the Linux Foundation	A "meritocratic, consensus-based community project" with a Industry Working Group	ISO/IEC Joint Technical Committee  Trusted Computing Group
<i>Initial Draft</i>	2010	2017	2009
<i>Formats</i>	RDF, XLS, SPDX, YAML, JSON	XML, JSON	XML, CBOR (CoSWID only)
<i>Spec</i>	<a href="https://spdx.github.io/spdx-spec">spdx.github.io/spdx-spec</a>  BS ISO/IEC 5962 - 2020 Draft	<a href="https://github.com/CycloneDX/specification">github.com/CycloneDX/specification</a>	<a href="https://iso.org/standard/65666.html">iso.org/standard/65666.html</a>  ISO/IEC 19770-2:2015



# Existing SBOM formats: Use Cases

	<b>SPDX</b> “Software Package Data eXchange”	<b>CycloneDX</b>	<b>SWID</b> “Software ID”
<i>Original use cases</i>	License management	For use with OWASP Dependency-Track	Inventory and change tracking
<i>Unique Features</i>	Extensive support for expressing license details	Extensible format and integrates SPDX license IDs, pURL, and other external identifiers	Deeply integrated into the build and publishing process for a software component
<i>Use cases of latest format versions</i>	<ul style="list-style-type: none"><li>• Tracking attributes of multiple software components (e.g. vendor, license, version, etc.)</li><li>• Generically describe packages, containers, os distributions, archives, etc</li><li>• Integrity verification of software components and sub-components</li></ul>		



# A “good” SBOM describes...

## What is being catalogued

For example a running system, a machine image, a container image, etc.

## Each item uniquely

Such as each component name, version, UUID, and relationships to other components.

## What did the cataloguing

The tool that generated the document with its configuration.



# A “great” SBOM also includes...

## In scope and out of scope

For example “only these paths were searched” or “only JARS and RPMs are being search for”.

## Exceptional conditions

Such as warnings or errors that occur during processing or missing environmental factors

## Additional metadata

Such as Java pom properties, key-values, additional RPM DB tag entries, and licenses.

# Introducing Syft

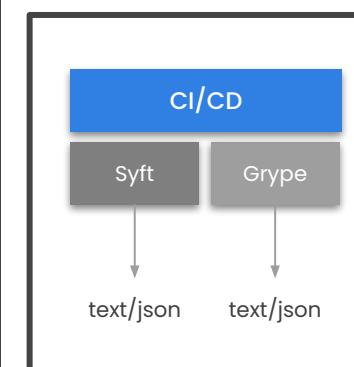
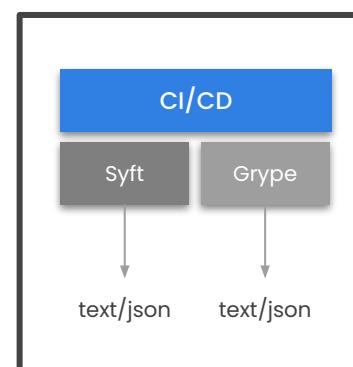
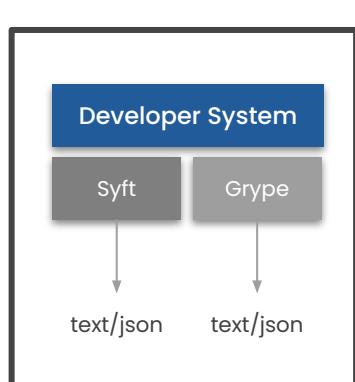


- Syft is an **open source tool that generates SBOMs** from container images and filesystems
- Syft supports **many package ecosystems:**
  - APK, DEB, RPM, Ruby Bundle, Python Wheel/Egg/requirements.txt, JavaScript NPM/Yarn, Java JAR/EAR/WAR, Jenkins plugin JPI/HPI, Go modules, Rust Crate
- Syft also supports **multiple output formats**
  - Summary table
  - JSON details
  - CycloneDX
  - SPDX

# Anchore Open Source



**Open Source** | Stateless, decentralized tools for developers



**Syft**  
Generates an SBOM for a container image

**Grype**  
Generates a list of CVEs using the SBOM

## Open Source

- **Lightweight** tools, written in **Go**
- **API-driven** to run in CI/CD
- **Linux** containers only
- **Local credentials** only
- **Stateless**, no data persistence
- **Siloed**, no centralized control

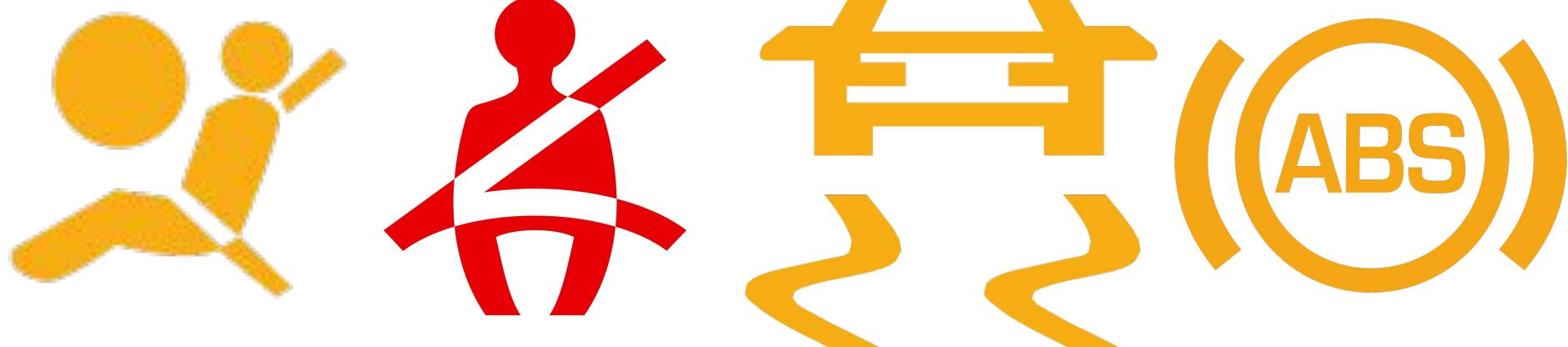


# Now What?

How will this improve my life?

anchore

# Story Time (again)





# How Do SBOMs Actually Help?

---

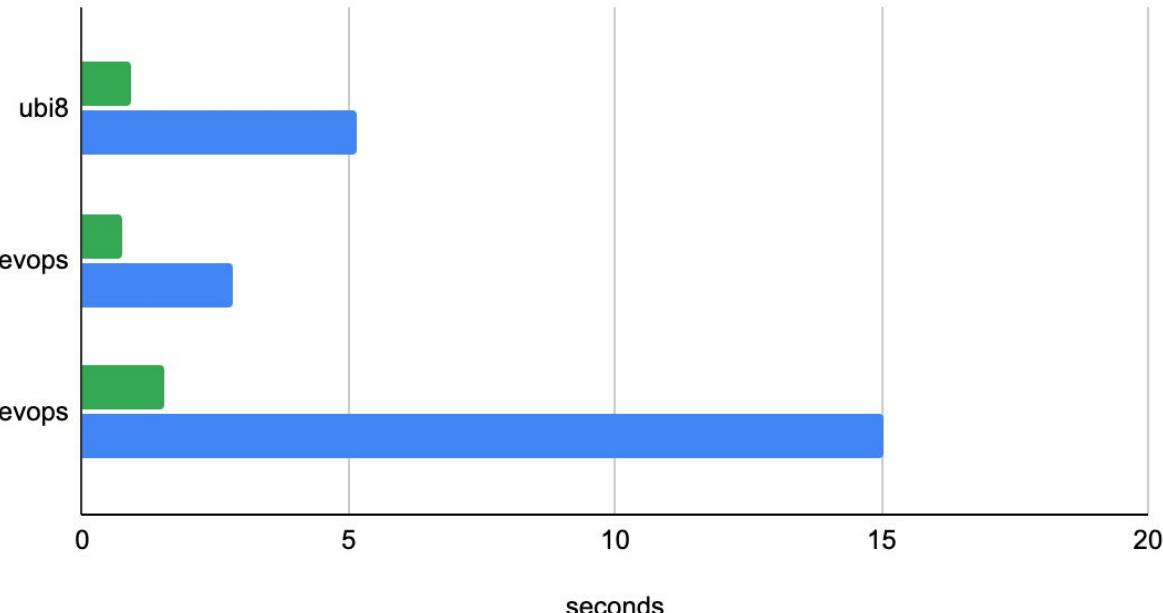




# Grype Scan Timing

grype vulnerability check

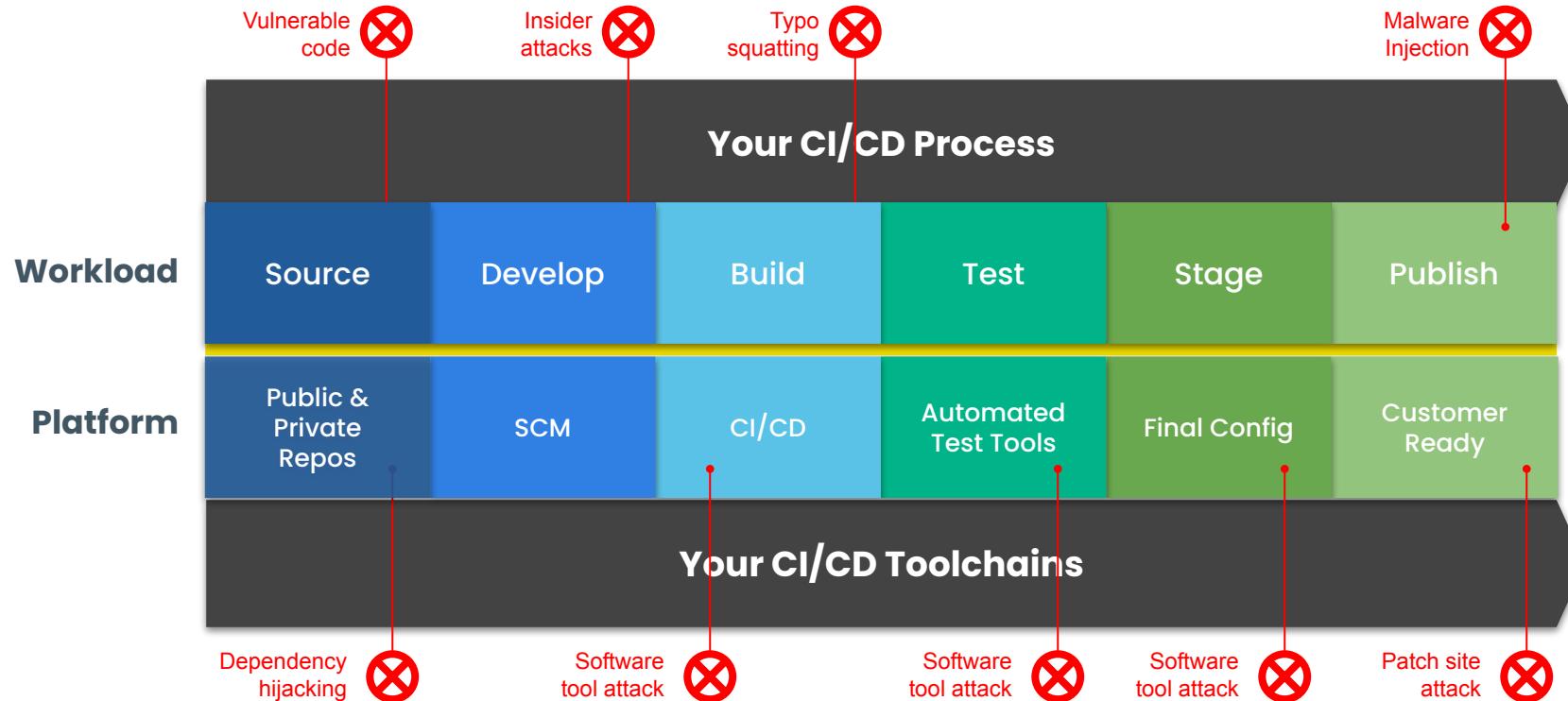
sbom    image



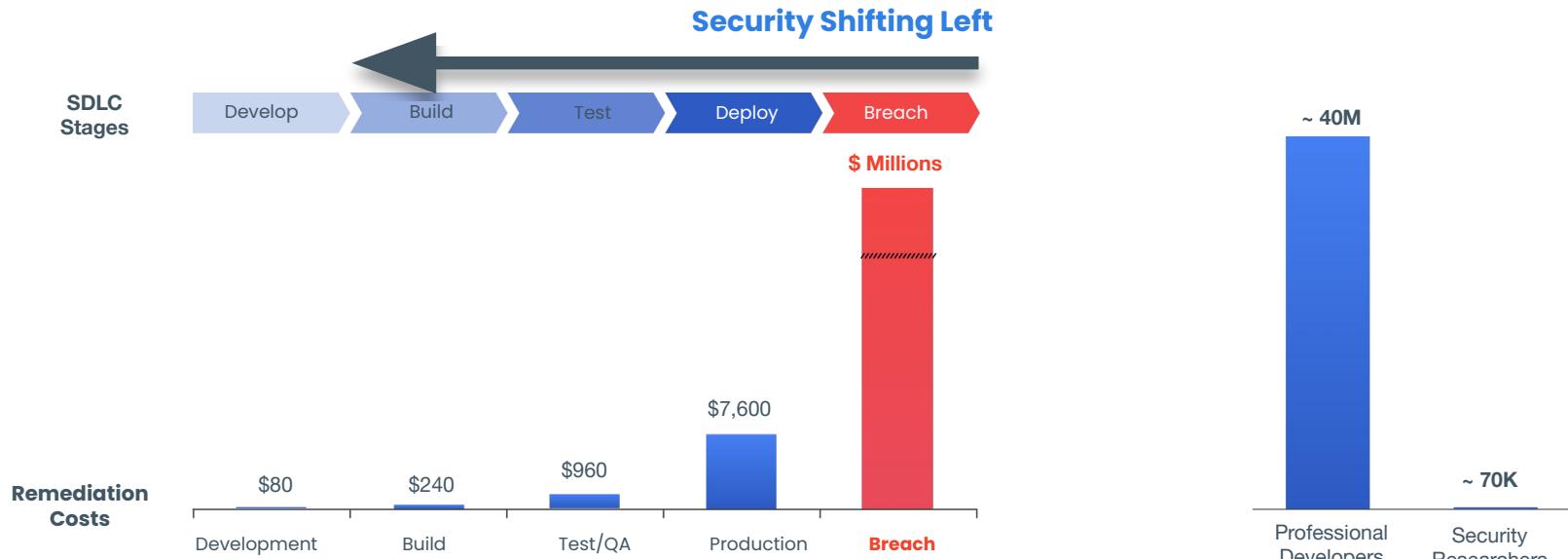
	ubi8	github-devops	gitlab-devops
sbom	0.917	0.751	1.55
image	5.159	2.839	15.031



# SBOMs Enable Continuous Evaluation



# Shift Security Left



Vastly more cost effective to remediate during development

**570X** more developers than security researchers

**STOP TALKING**



**DEMO TIME**

memegenerator.net



**DEMO TIME**

memegenerator.net

**LET'S DO**

**A DEMO**

memegenerator.net

**WHAT COULD GO WRONG?**

**I CAN HAS DEMO?**

**DEMO?**

memegenerator.net





Paul V. Novarese

@pvn

...

When I tweak the demo 10 minutes before my presentation



8:41 PM · Aug 26, 2018 · Twitter for iPhone

# DIY Demos

---



- Create a SBOM in Jenkins:  
<https://github.com/pvnovarese/oss-2021-syft-sbom-demo>
  - Includes instructions on deploying a disposable Jenkins container
  - Difficulty: easy
- Complete workflow example in Github Actions with syft, grype, and cosign:  
<https://github.com/pvnovarese/oss-2021-sbom-complete-workflow-demo>
  - Includes Github Actions
  - Build an image, sign it, scan for vulns, and sign the scan results
  - Difficulty: intermediate
  - More info: [Anchore Blog](#)
- Brand new GitHub Anchore SBOM Action:  
<https://github.com/marketplace/actions/anchore-sbom-action>



# Example SBOM Creation

## Pipeline `syft-sbom-demo`

Full project name: OSS-2021/syft-sbom-demo

[add description](#) [Disable Project](#)

 **Last Successful Artifacts**  
[sbom-spdx-1.json](#) 25.57 KB 

 **Recent Changes**

### Stage View

	Declarative: Checkout SCM	Checkout SCM	Build image and tag with build number	Generate SBOM with syft	Clean up
Average stage times: (Average full run time: ~6s)	357ms	289ms	2s	934ms	325ms
#1 Sep 18 05:15 No Changes	357ms	289ms	2s	934ms	325ms

# Example Workflow



✓ .github/workflows/build-and-publish.yaml .github/workflows/build-and-publish.yaml #14 Re-run jobs ⋮

Summary

Manually triggered 2 hours ago	Status	Total duration	Artifacts
pvnovarese -○ 8967565	Success	2m 52s	2

Jobs

- ✓ build\_image
- ✓ sign\_image
- ✓ create\_image\_sbom
- ✓ scan\_for\_vulnerabilities
- ✓ create\_sbom\_attestation
- ✓ create\_vulnerability\_scan\_attesta...

**build-and-publish.yaml**  
on: workflow\_dispatch

```
graph LR; A[build_image] -- 37s --> B[create_image_sbom]; B -- 39s --> C[scan_for_vulnerabilities]; C -- 30s --> D[create_vulnerability_scan_attestation]; D -- 26s --> E[create_sbom_attestation]; E -- 25s --> F[sign_image]; F -- 20s --> G[create_image_sbom]
```

[+/-]

# Best Practices for Securing the Software Supply Chain

---



01

Centralized, secure CI/CD process for all software

02

Build images from trusted sources

03

Automate security testing and policy enforcement

04

Deploy only trusted images into production



Q&A

**Download Syft**

<https://github.com/anchore/syft>

**Download Grype**

<https://github.com/anchore/grype>

Let us know if you like it by giving us a star on GitHub

Get an invite to our open source community Slack at  
<https://anchore.com/slack/>

anchore



Signal ✅

@signalapp

...

Replying to [@signalapp](#)

Ubiquitous e2e encryption is pushing intelligence agencies from undetectable mass surveillance to expensive, high-risk, targeted attacks.

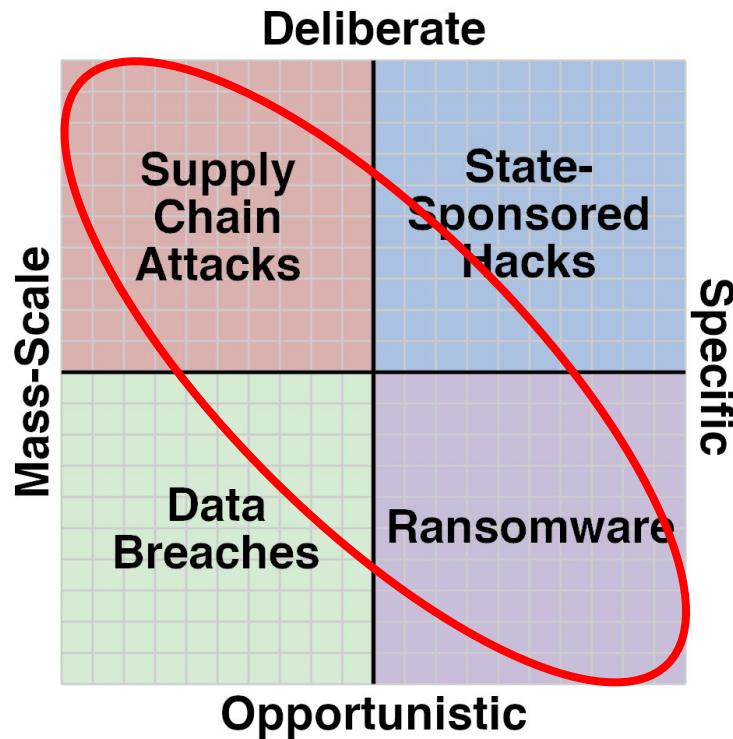
2:03 PM · Mar 7, 2017 · Twitter Web Client

---

**1,164** Retweets   **83** Quote Tweets   **1,397** Likes



# Takeaways





# One Last Thing



Douglas Haig

"Bullets have little stopping power against the horse" (1914)



**Jake Williams**  
@MalwareJake

...

Hey infosec: remember that your job is risk reduction, not risk elimination. There's a BIG difference.

9:31 PM · Aug 29, 2021 · Twitter for Android

---

**258** Retweets   **26** Quote Tweets   **1,677** Likes



# Thanks!

**Download Syft**

<https://github.com/anchore/syft>

**Download Grype**

<https://github.com/anchore/grype>

Let us know if you like it by giving us a star on GitHub

Get an invite to our open source community Slack at  
<https://anchore.com/slack/>

anchore

# Reading List



Reflections on Trusting Trust: [https://www.cs.cmu.edu/~rdriley/487/papers/Thompson\\_1984\\_ReflectionsOnTrustingTrust.pdf](https://www.cs.cmu.edu/~rdriley/487/papers/Thompson_1984_ReflectionsOnTrustingTrust.pdf)

Generate sboms with synt and jenkins: [https://www.youtube.com/watch?v=nMLveJ\\_TxAs](https://www.youtube.com/watch?v=nMLveJ_TxAs)

Solar Winds post mortem: <https://www.lawfareblog.com/solarwinds-and-holiday-bear-campaign-case-study-classroom>

SPDX becomes sbom standard:

<https://www.linuxfoundation.org/press-release/spdx-becomes-internationally-recognized-standard-for-software-bill-of-materials>

Profound Podcast - Episode 10 (John Willis and Josh Corman):

<https://www.buzzsprout.com/1758599/8761108-profound-dr-deming-episode-10-josh-corman-captain-america>

Creating a trusted container supply chain: <https://thenewstack.io/creating-a-trusted-container-supply-chain/>

Github SBOM action: <https://github.com/marketplace/actions/anchore-sbom-action>



# Footnotes

---

Twitter Links/Credits:

<https://twitter.com/codinghorror/status/786667942142435329>  
<https://twitter.com/dakami/status/896477575475642368>  
<https://twitter.com/mikemearls/status/1079999471109337088>

Other notes:

[https://en.wikipedia.org/wiki/Douglas\\_Hair,\\_1st\\_Earl\\_Hair](https://en.wikipedia.org/wiki/Douglas_Hair,_1st_Earl_Hair)  
<https://en.wikipedia.org/wiki/USA-247>  
<https://joshdata.me/iceberger.html>

Images used for SBOM generation timing benchmarks:

registry.access.redhat.com/ubi8:latest

[https://gitlab.com/pvn\\_test\\_images/devops-supply-chain](https://gitlab.com/pvn_test_images/devops-supply-chain)  
<https://github.com/pvnovarese/devops-supply-chain-demo>

Accuracy and Precision: <https://wps.prenhall.com/wps/media/objects/3310/3390101/blb0105.html>

Integration of cosign with syft: <https://github.com/anchore/syft/issues/510>

Add support for Hints in syft: <https://github.com/anchore/syft/issues/31>