Building a **Trusted and Resilient** *Software Supply Chain*

Paolo Mainardi

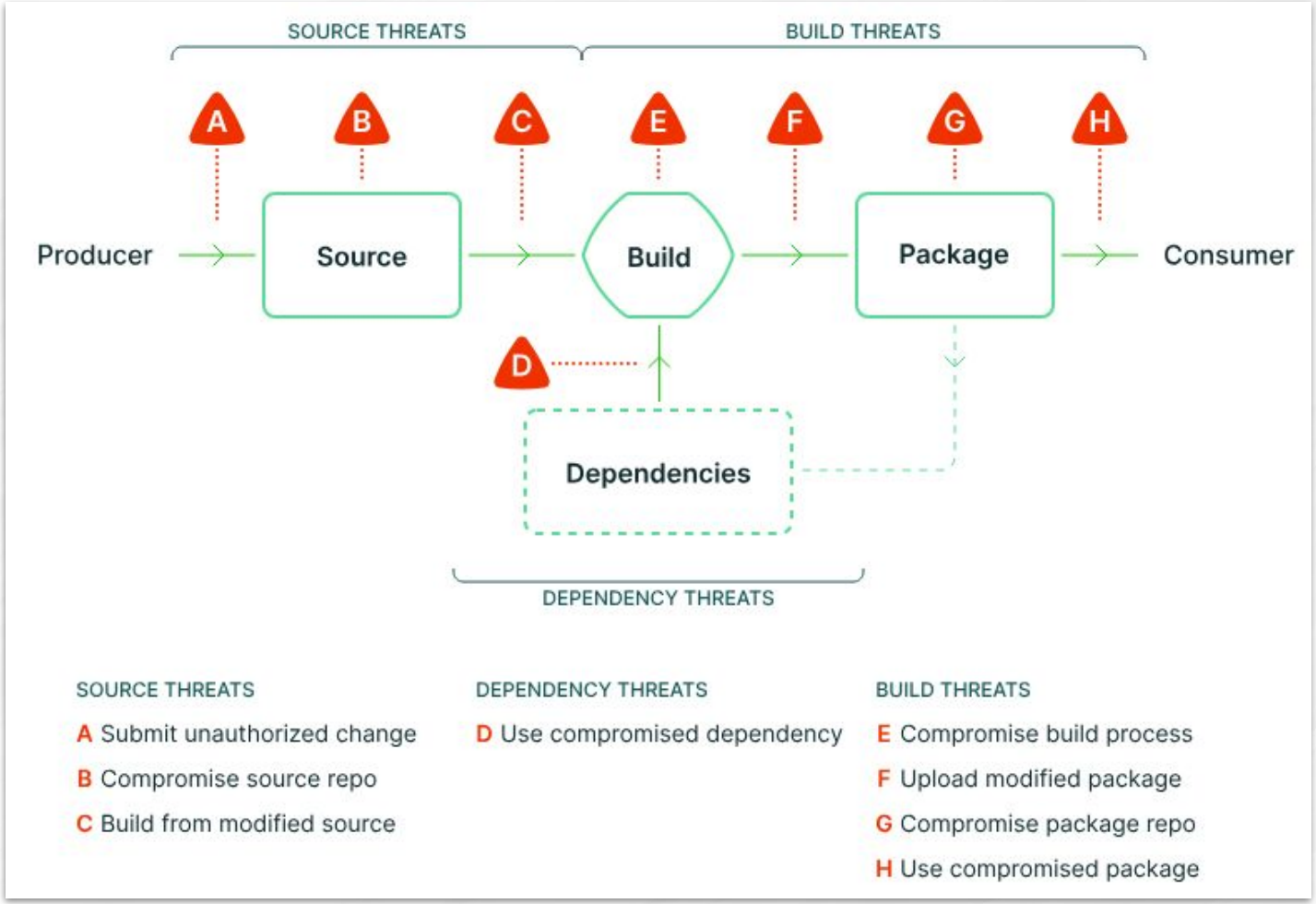
@paolomainardi

- Co-founder and CTO @Sparkfabrik
- Linux Foundation Europe Advisory Member
- Blog: paolomainardi.com
- Podcast: Continuous Delivery
- linkedin.com/in/paolomainardi
- continuousdelivery.social/@paolomainardi
- paolo.mainardi@sparkfabrik.com

THE SESSION

- ➔ What is a **Software Supply Chain**
- ➔ State of the open source ecosystem
- ➔ What are **the threats** and how can be challenged with:
  Sigstore - SBOM - OSV - Scorecard

A **supply chain** is a network of individuals and companies who are involved in creating a product and delivering it to the consumer



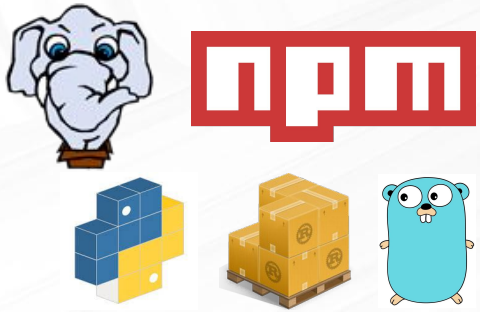
https://slsa.dev/spec/v0.1/#supply-chain-threats

A MODERN APPLICATION

A MODERN APPLICATION

Application

Dependencies

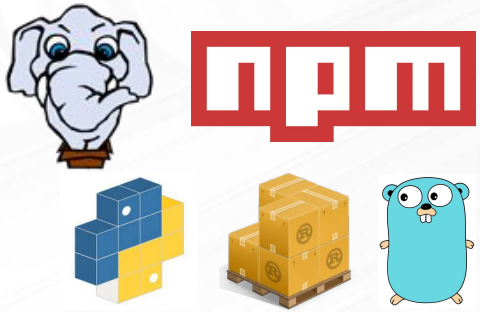


A MODERN APPLICATION

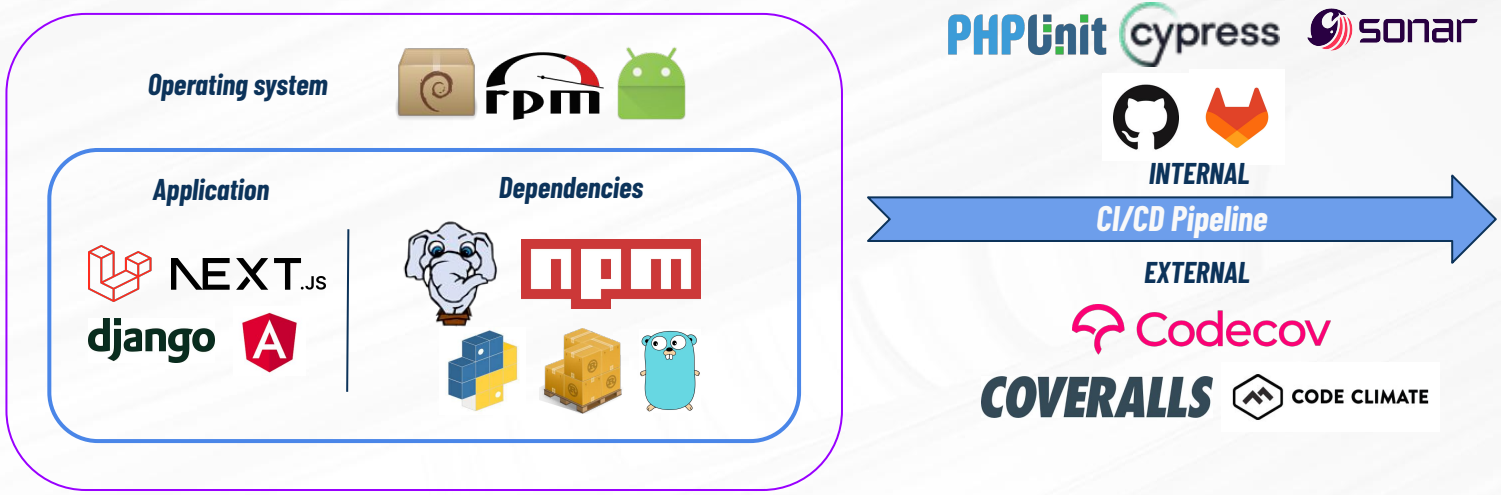
Operating system

Application

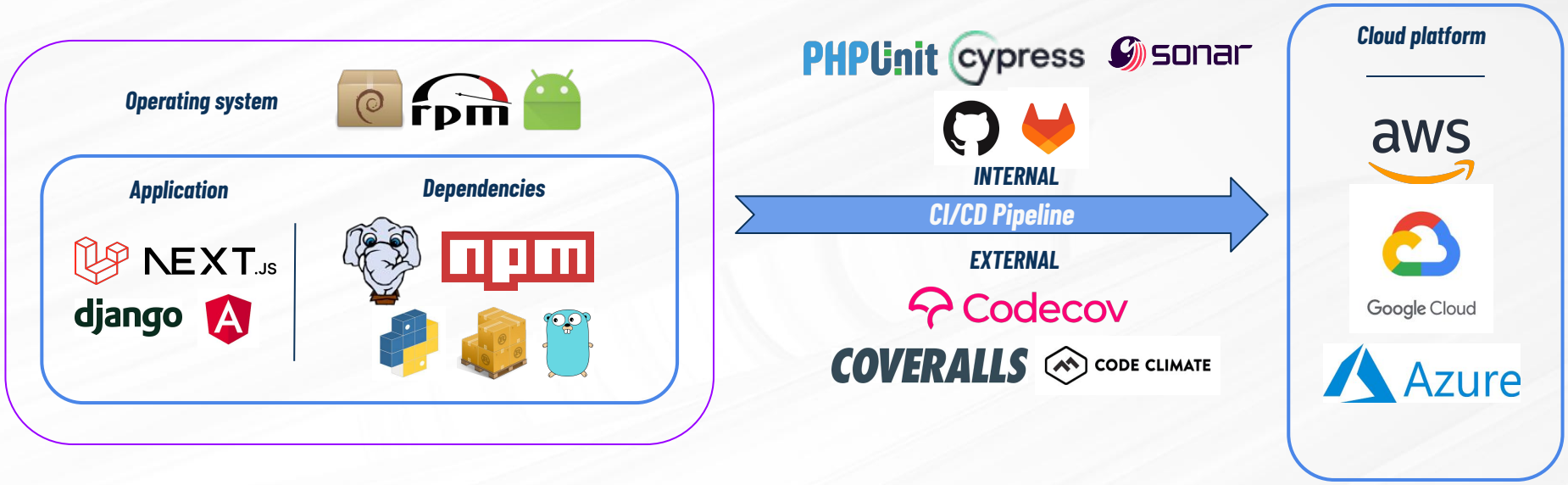
Dependencies



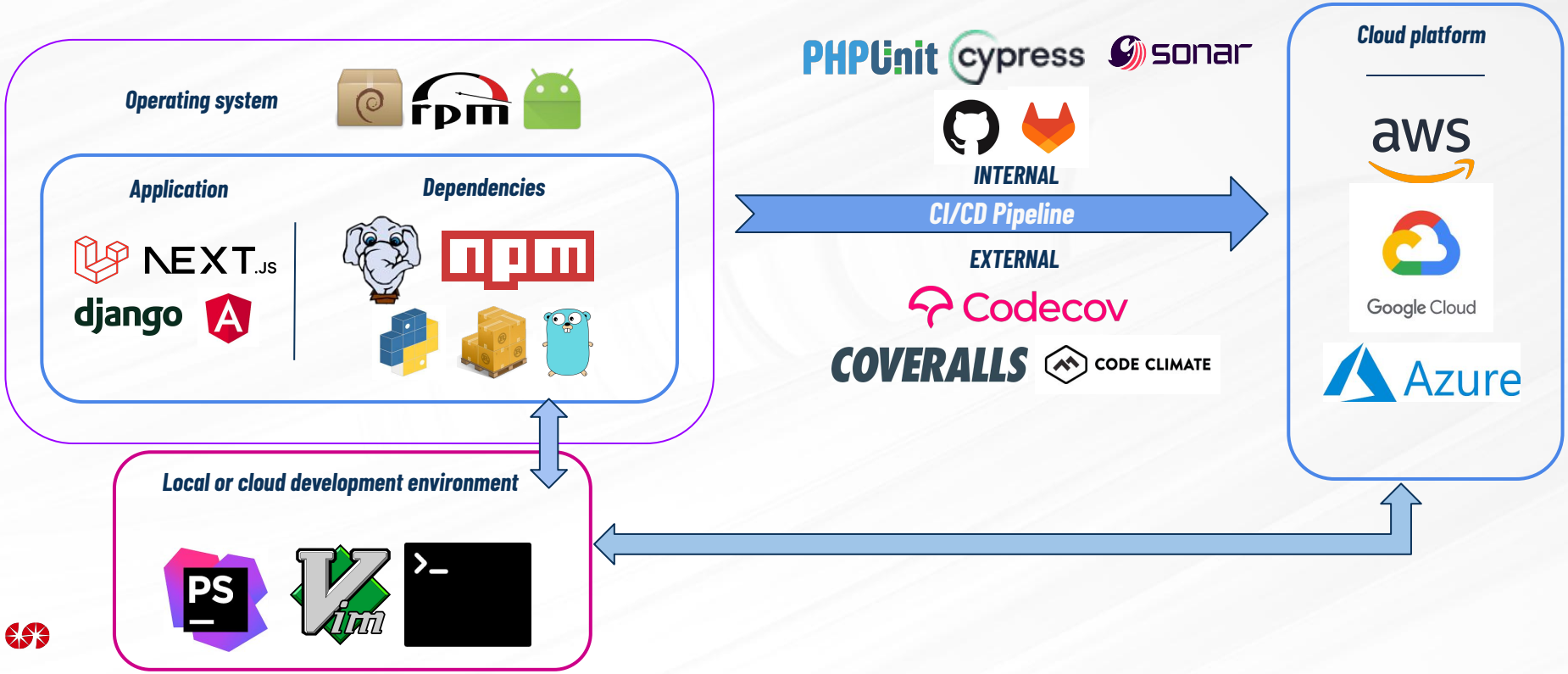
A MODERN APPLICATION



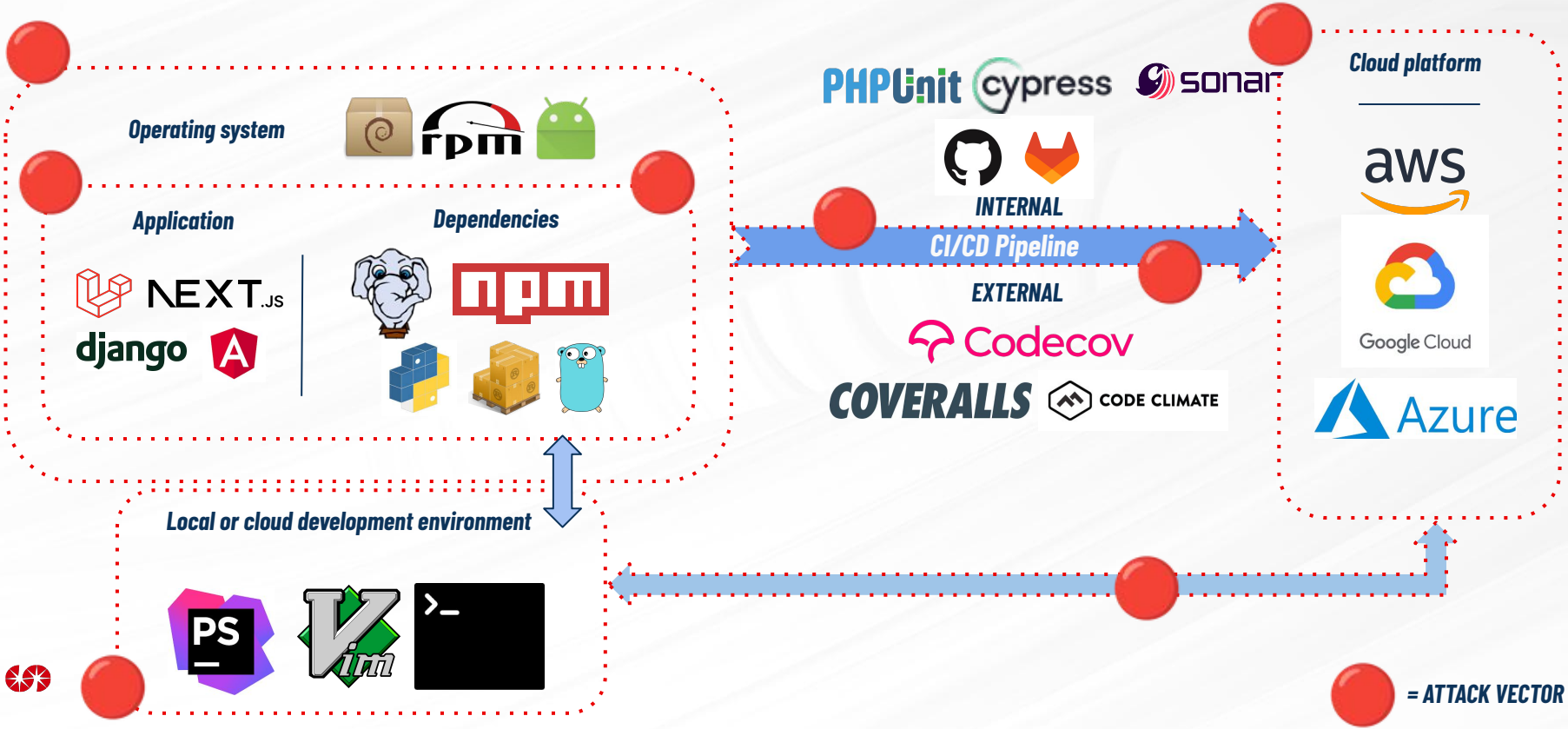
A MODERN APPLICATION

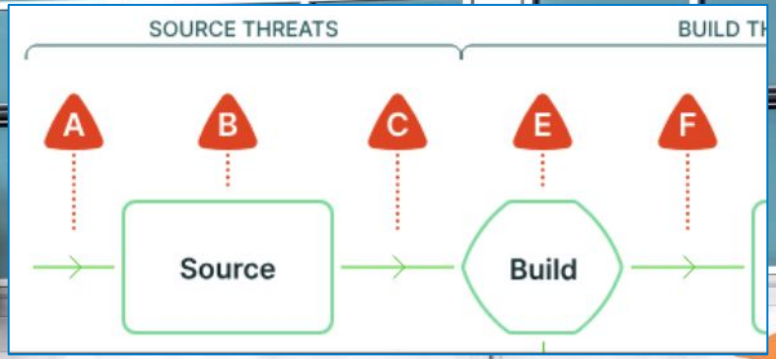


A MODERN APPLICATION



A MODERN APPLICATION





2020

About 18,000 customers of SolarWinds installed the infected updates, including firms like Microsoft (Cisco, Intel, Deloitte) and top government US agencies like Pentagon, Homeland security, National Nuclear Security etc.

2021 - Log4j - Log4shell 2021 - CVE-2021-44228

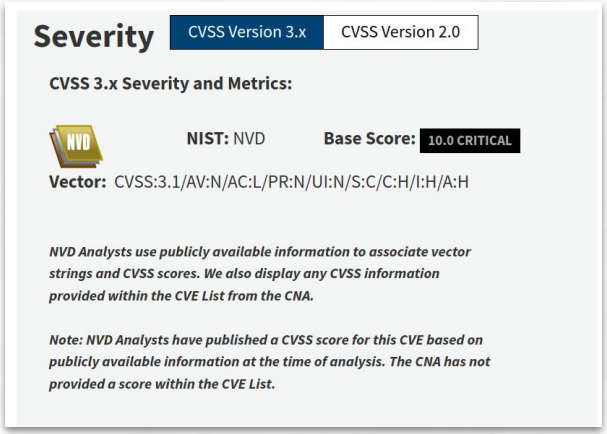
Why Log4Shell could be the worst software vulnerability ever

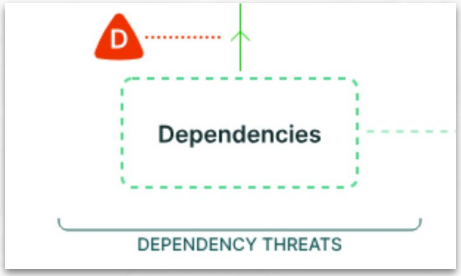
Zbigniew Banach - Mon, 20 Dec 2021 -

Thousands of Java applications across the world are wide open to remote code execution attacks targeting the Log4j library. This post summarizes what we know so far about the Log4Shell vulnerability, how you can mitigate it, how to find it using Invicti, and what it means for cybersecurity here and now.

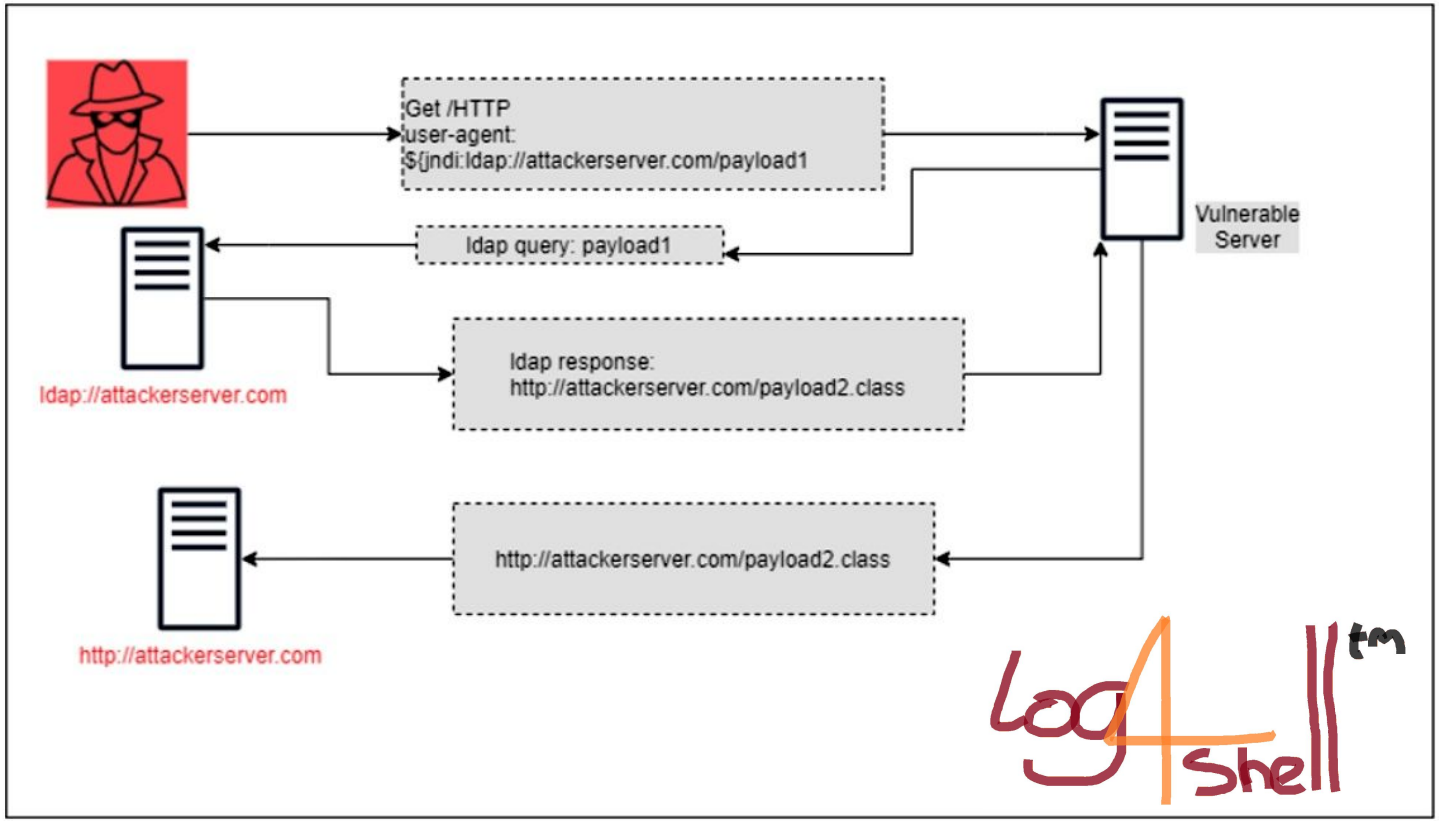
https://www.lunasec.io/docs/blog/log4j-zero-day/

Editor's note (28 Dec 2021 at 7:35 p.m. GMT): *The Log4j team released a new security update that found 2.17.0 to be vulnerable to remote code execution, identified by CVE-2021-44832. We recommend upgrading to the latest version, which at this time is 2.17.1. Please note that the Log4Shell situation is rapidly changing and we are updating our blogs as new information becomes available.*

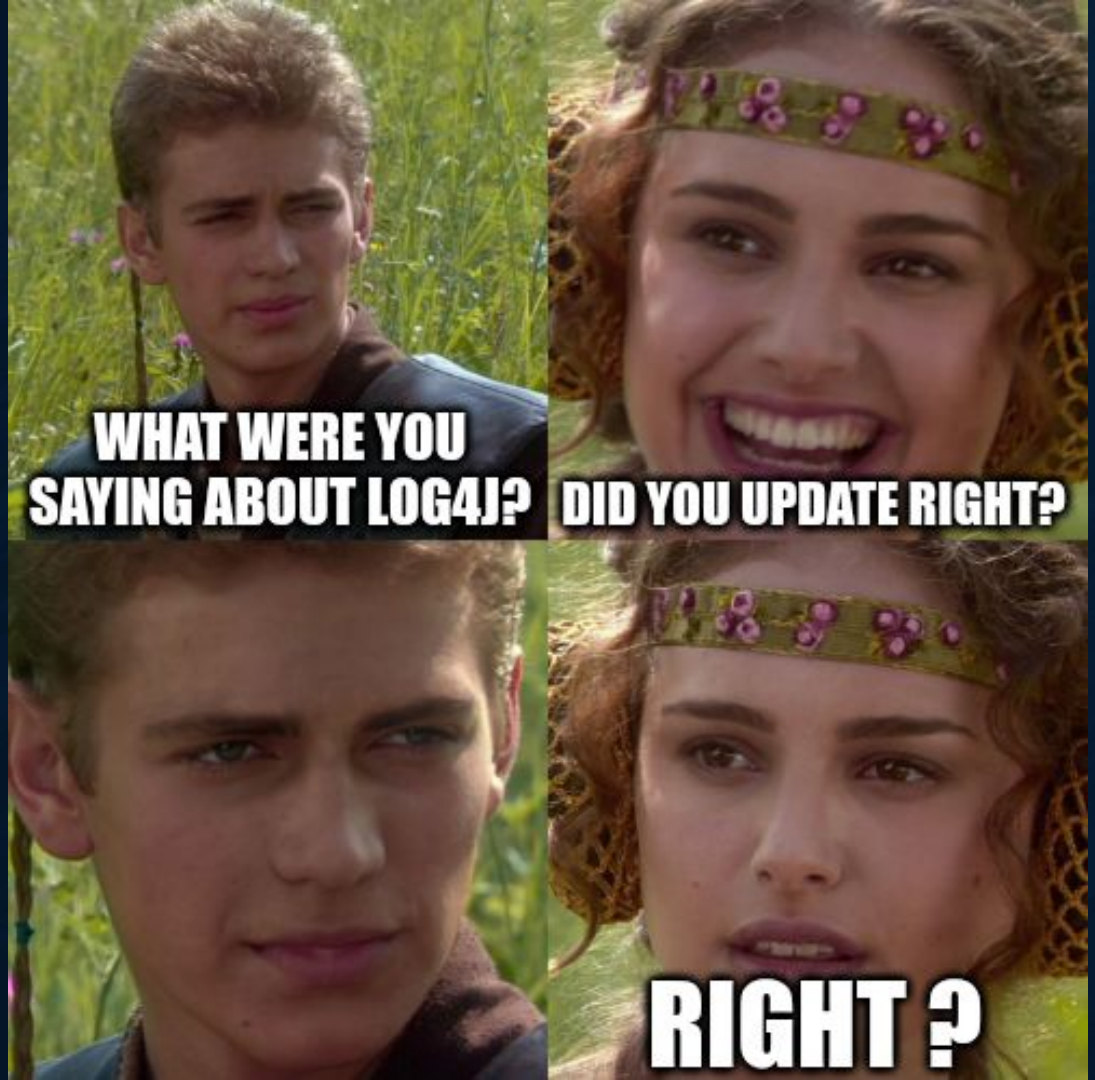




Description of the CVE-2021-44228 vulnerability



© Juniper Networks


WHAT WERE YOU SAYING ABOUT LOG4J?
DID YOU UPDATE RIGHT?
RIGHT ?

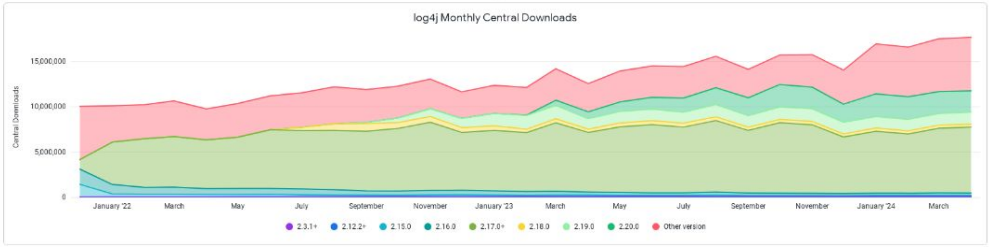
log4j Latest Statistics

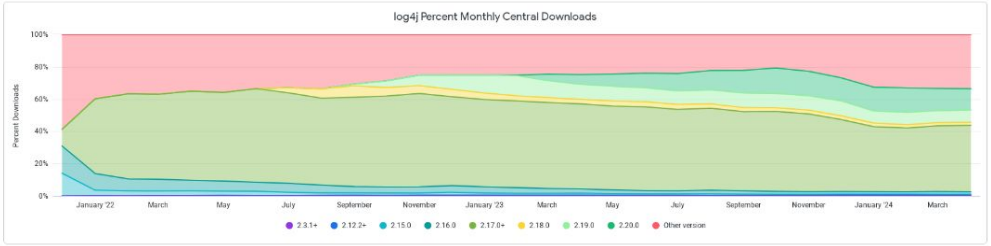
390,498,393

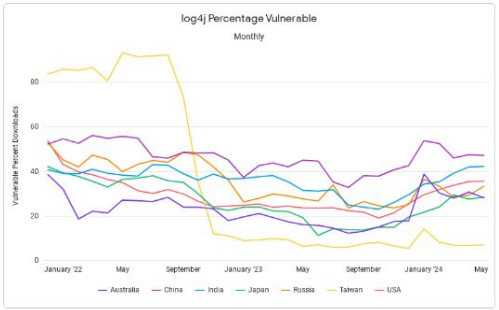
Total Downloads Since Dec 10, 2021
29 % vulnerable

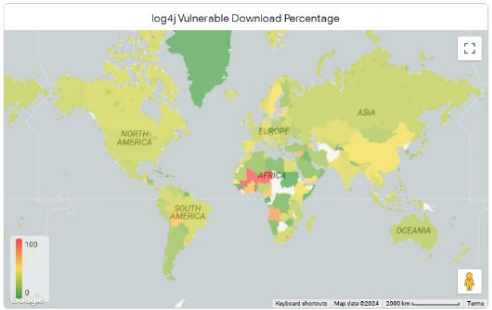
35 %

Vulnerable Downloads Last 7 Days
4,032,600 total downloads









Generated by Looker on May 19, 2024 at 6:36 AM EDT

Source: [Sonatype Log4j exploit update]

April 2024

SSH BACKDOOR VIA LIBLZMA

The story of the XZ failed *(by chance)* attack

https://mastodon.social/@AndresFreundTec/112180083704606941

AndresFreundTec
@AndresFreundTec@mastodon.social

I was doing some micro-benchmarking at the time, needed to quiesce the system to reduce noise. Saw sshd processes were using a surprising amount of CPU, despite immediately failing because of wrong usernames etc. Profiled sshd, showing lots of cpu time in liblzma, with perf unable to attribute it to a symbol. Got suspicious. Recalled that I had seen an odd valgrind complaint in automated testing of postgres, a few weeks earlier, after package updates.

Really required a lot of coincidences.

Mar 29, 2024, 07:32 PM · Web

https://www.openwall.com/lists/oss-security/2024/03/29/4

Openwall
bringing security into open environments

Products | Services | Publications | Resources | What's new

Follow @Openwall on Twitter for new release announcements and other news

[<prev] [next>] [thread-next>] [day] [month] [year] [list]

Date: Fri, 29 Mar 2024 08:51:26 -0700
From: Andres Freund <andres@...razel.de>
To: oss-security@...ts.openwall.com
Subject: backdoor in upstream xz/liblzma leading to ssh server compromise

Hi,

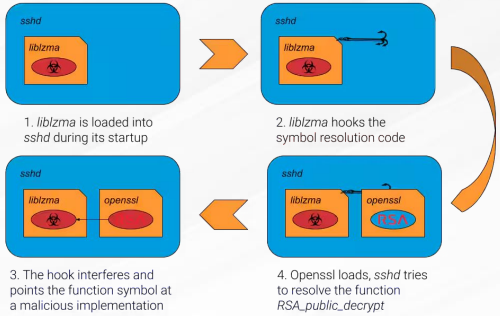
After observing a few odd symptoms around liblzma (part of the xz package) on Debian sid installations over the last weeks (logins with ssh taking a lot of CPU, valgrind errors) I figured out the answer:

The upstream xz repository and the xz tarballs have been backdoored.

At first I thought this was a compromise of debian's package, but it turns out to be upstream.

== Compromised Release Tarball ==

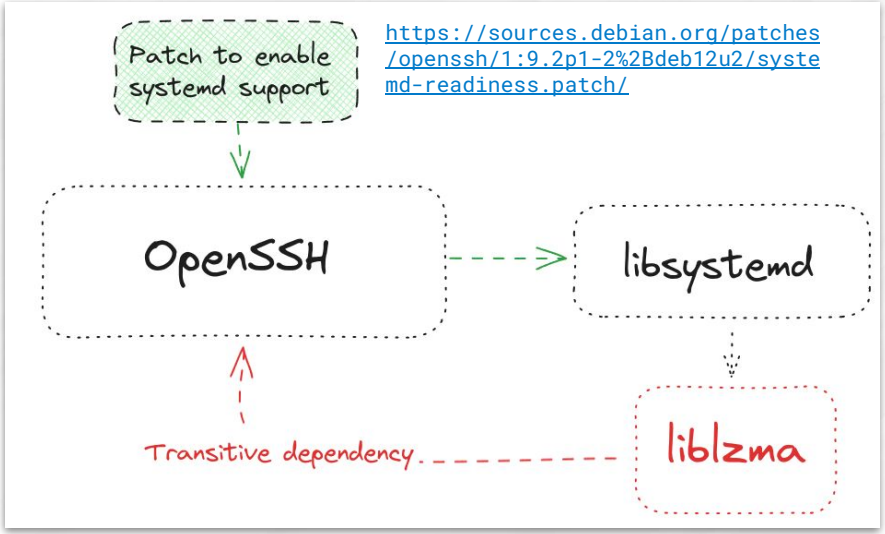
One portion of the backdoor is *solely in the distributed tarballs*. For easier reference, here's a link to debian's import of the tarball, but it is also present in the tarballs for 5.6.0 and 5.6.1:



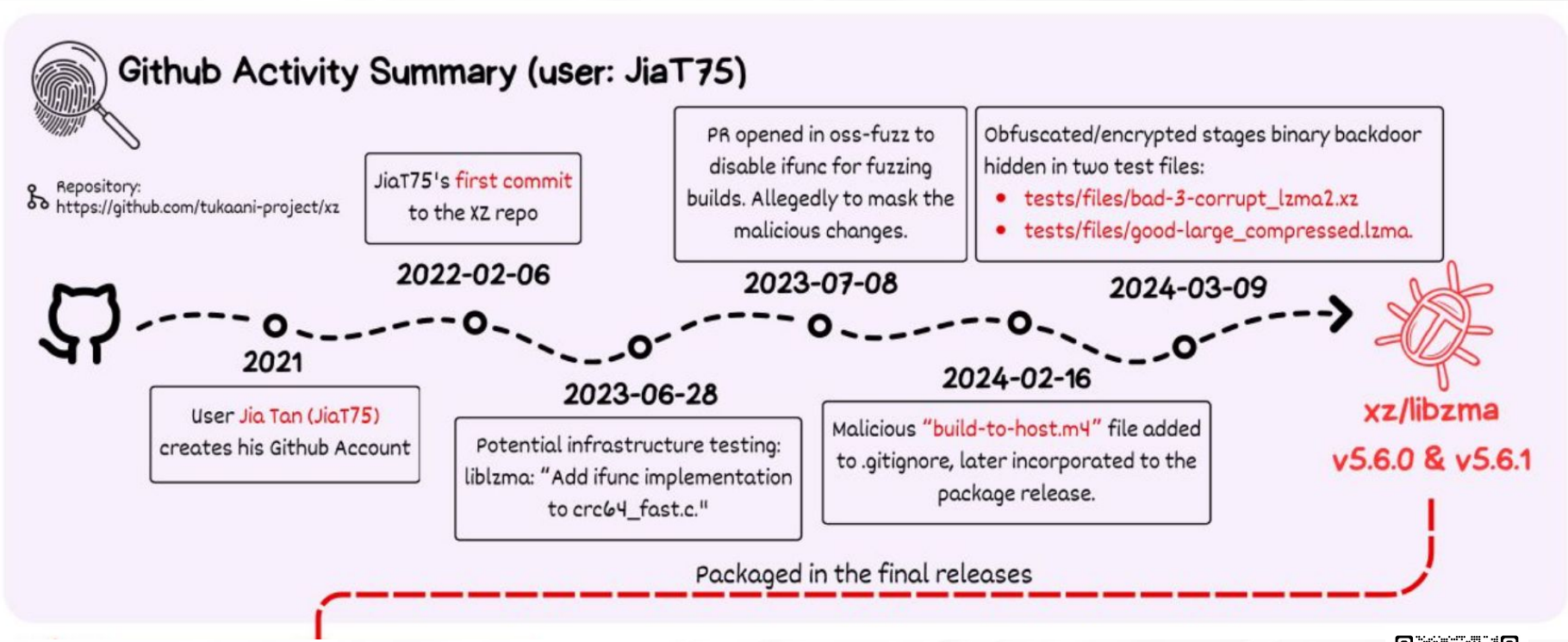
What is XZ and why SSH has been impacted ?

XZ Utils and its underlying library, liblzma, are **open-source projects that implement LZMA compression** and decompression. They are **included in many Linux distributions** out of the box.

OpenSSH does not depend on XZ, however Debian and several other distributions - patch it to support systemd notification, and eventually, **libsystemd does depend on liblzma**.



XZ attack timeline



Source: https://www.linkedin.com/posts/thomas-roccia_infosec-xz-cybersecurity-activity-7180110597139697664-nzJL

Open source has won

It won yes, but is it still sustainable ?

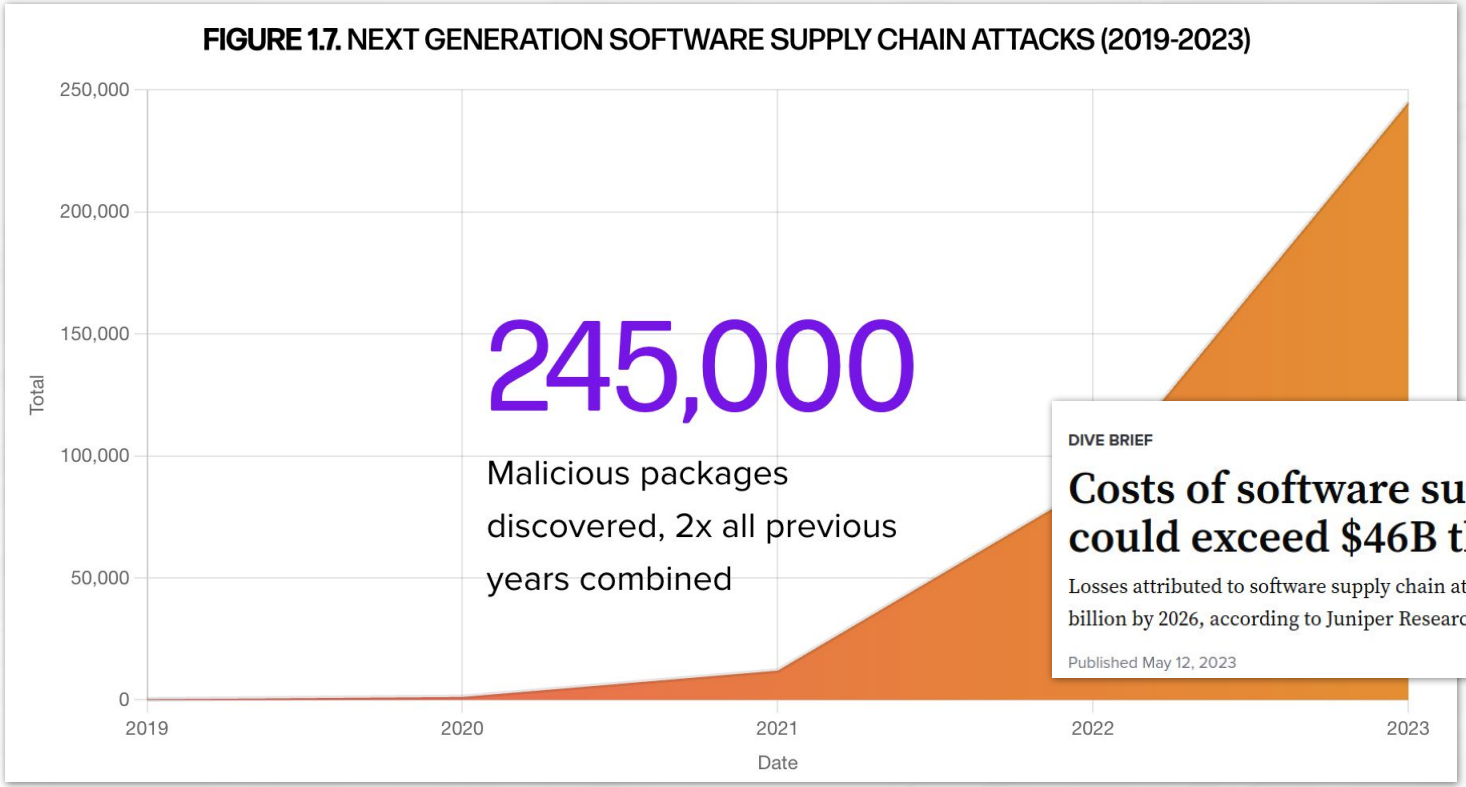
Re: [xz-devel] XZ for Java

Lasse Collin | Wed, 08 Jun 2022 03:28:08 -0700

On 2022-06-07 Jigar Kumar wrote:
> Progress will not happen until there is new maintainer. XZ for C has
> sparse commit log too. Dennis you are better off waiting until new
> maintainer happens or fork yourself. Submitting patches here has no
> purpose these days. The current maintainer lost interest or doesn't
> care to maintain anymore. It is sad to see for a repo like this.

I haven't lost interest but my ability to care has been fairly limited mostly due to longterm mental health issues but also due to some other things. Recently I've worked off-list a bit with Jia Tan on XZ Utils and perhaps he will have a bigger role in the future, we'll see.

It's also good to keep in mind that this is an unpaid hobby project.



DIVE BRIEF

Costs of software supply chain attacks could exceed $46B this year

Losses attributed to software supply chain attacks will jump 76%, reaching almost $81 billion by 2026, according to Juniper Research.

Published May 12, 2023

https://www.sonatype.com/state-of-the-software-supply-chain/introduction

National Cybersecurity Strategy

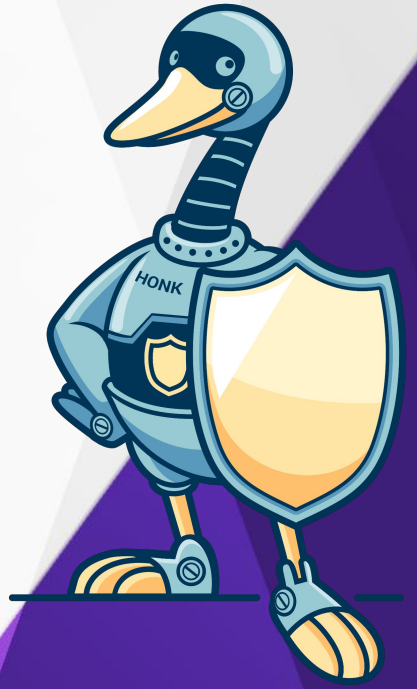
MARCH 2023

https://linuxfoundation.eu/cyber-resilience-act

OpenSSF
OPEN SOURCE SECURITY FOUNDATION https://openssf.org

About OpenSSF

- Formed in 2020
- Part of the Linux Foundation and not for profit
- Focus on Open Source Security
- Security newsletters and blog
- Tools and standards: SLSA, OpenVex, Sigstore, Scorecard, OSV, GUAC
- Free courses:
  - Secure Software Development Fundamentals
  - Securing Your Software Supply Chain with Sigstore
  - Securing Projects with OpenSSF Scorecard



INTEGRITY, TRUST AND DEPENDENCIES

What is the ***trusting model*** between
us and ***digital artifacts***?

How can i be sure that ***what I'm running***
is coming from a ***trusted source***?

1984

Reflections on Trusting Trust

To what extent should one trust a statement that a program is free of Trojan horses? Perhaps it is more important to trust the people who wrote the software.

MORAL

The moral is obvious. You can't trust code that you did not totally create yourself. (Especially code from companies that employ people like me.) No amount of source-level verification or scrutiny will protect you from using untrusted code. 1

KEN THOMPSON

SECURE SOFTWARE SUPPLY CHAIN CHECKLIST

✅ Who built it, when and how
(***Signatures*** and ***Provenance Attestations***)

✅ The list of things who made the artifact
(***SBOM - Software Bill of Material***)

DIGITAL SIGNATURES 101

Integrity

Ensure the data signed was not altered.

Authenticity

Attest that the data was sent by the signer.

Non-repudiation

Ensure that the signer cannot deny the authenticity of the signature.

Managing keys is hard

Distribution, Storage, Compromise

DIGITAL SIGNATURES - SIGSTORE

Sigstore is an OSS project under the umbrella of ***OpenSSF*** foundation.

Fast growing community and mainstream adopted

Used in ***Kubernetes*** and many other big vendors

(Github, Rubygems, Arch Linux etc..)

https://openssf.org/community/sigstore

In collaboration with

OpenSSF Chainguard cisco Google Hewlett Packard Enterprise PURDUE UNIVERSITY Red Hat vmware

DIGITAL SIGNATURES - SIGSTORE

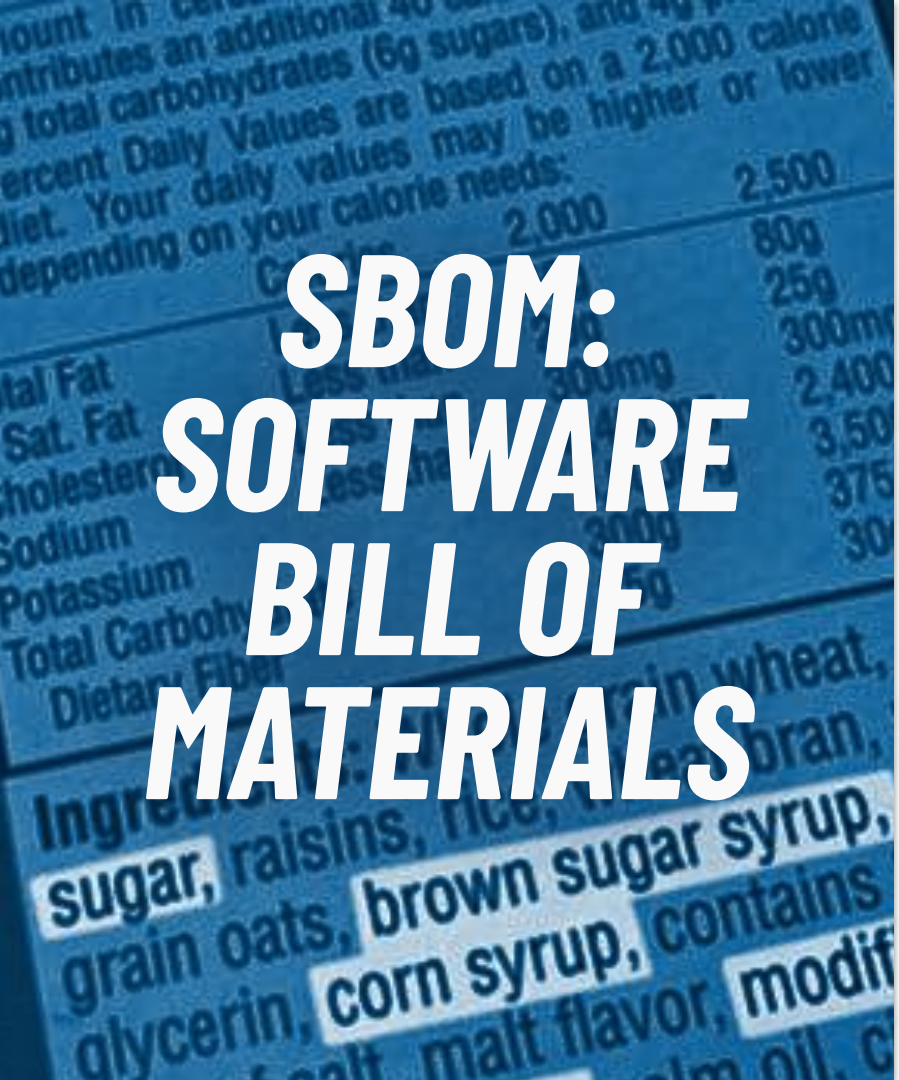
Keyless signing of any software artifact

Signatures metadata are stored in a ***public tamper-resistant log***

Signatures are ***stored*** alongside images in ***OCI registry***

In collaboration with

OpenSSF Chainguard cisco Google Hewlett Packard Enterprise PURDUE UNIVERSITY Red Hat vmware

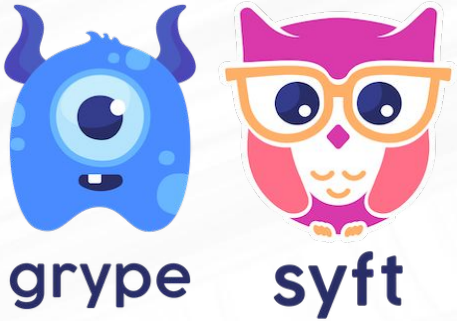


A list of "ingredients" for a software artifact

Can be used for:

- ➔ Vulnerability scanning
- ➔ Software transparency
- ➔ License policy
- ➔ Find abandoned dependencies

SBOM - Tools

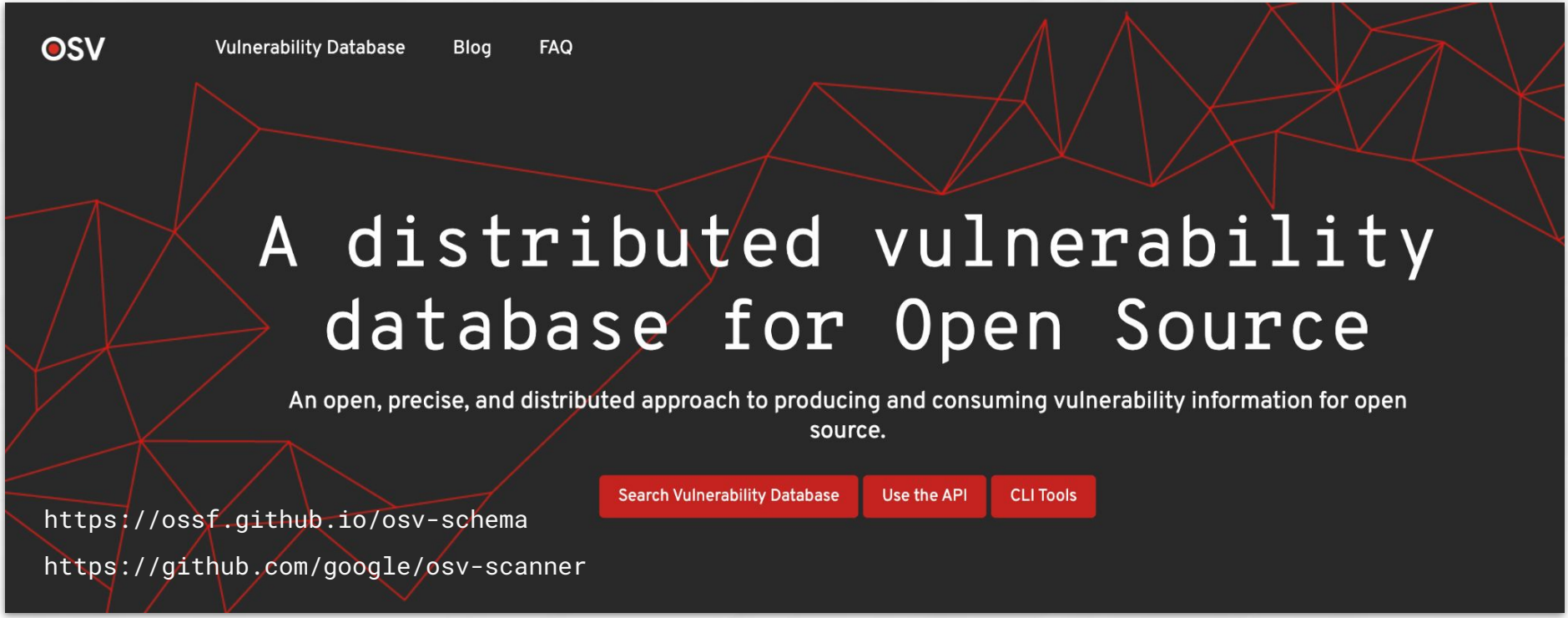


SBOM + Vulnerabilities

KNOW YOUR DEPENDENCIES

OSV

https://osv.dev



OpenSSF Scorecard

https://scorecard.dev

Checks: Code vulnerabilities, Maintenance, Continuous testing, Source risk assessment, Build risk assessment

Open Source Insights

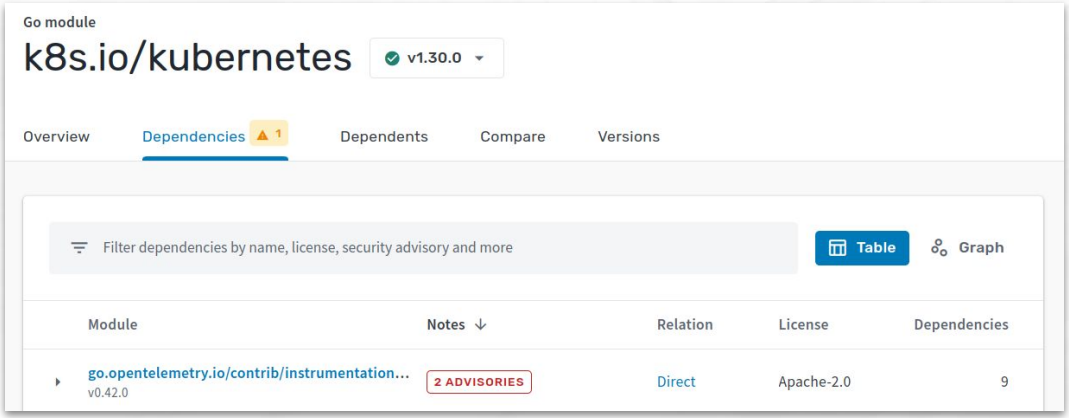
https://deps.dev

An accurate view of the complete dependency graph, licenses, dependencies and security advisories.
Free API access.

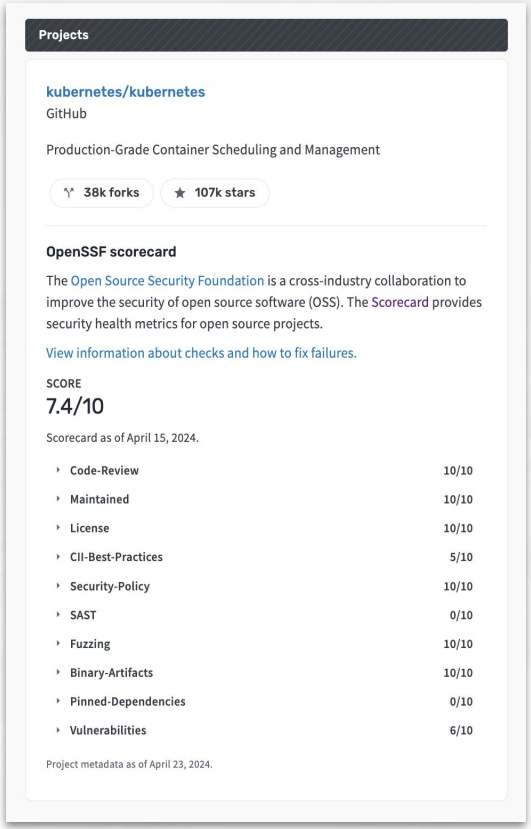
Licenses

Security

Dependencies



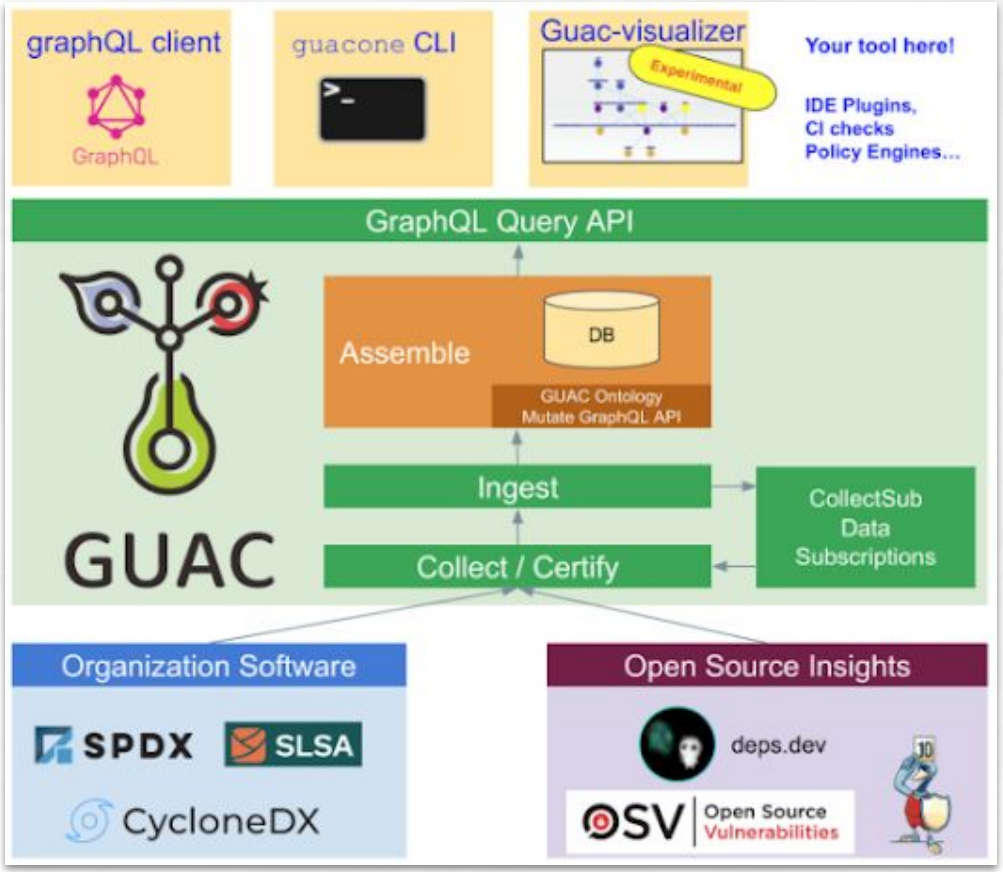
OpenSSF ScoreCard



GUAC - Graph For Understanding Artifact Composition

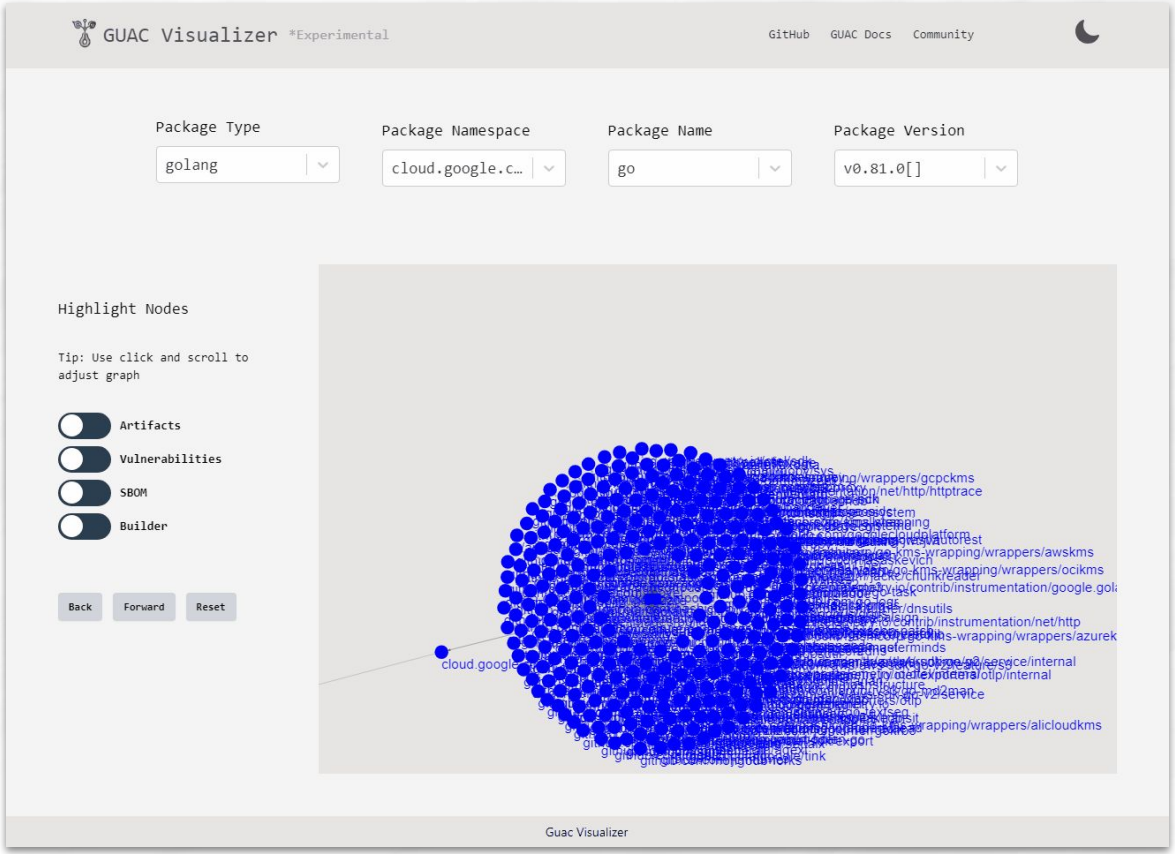
GUAC

https://guac.sh



GUAC - Graph For Understanding Artifact Composition

https://github.com/guacsec/guac-visualizer

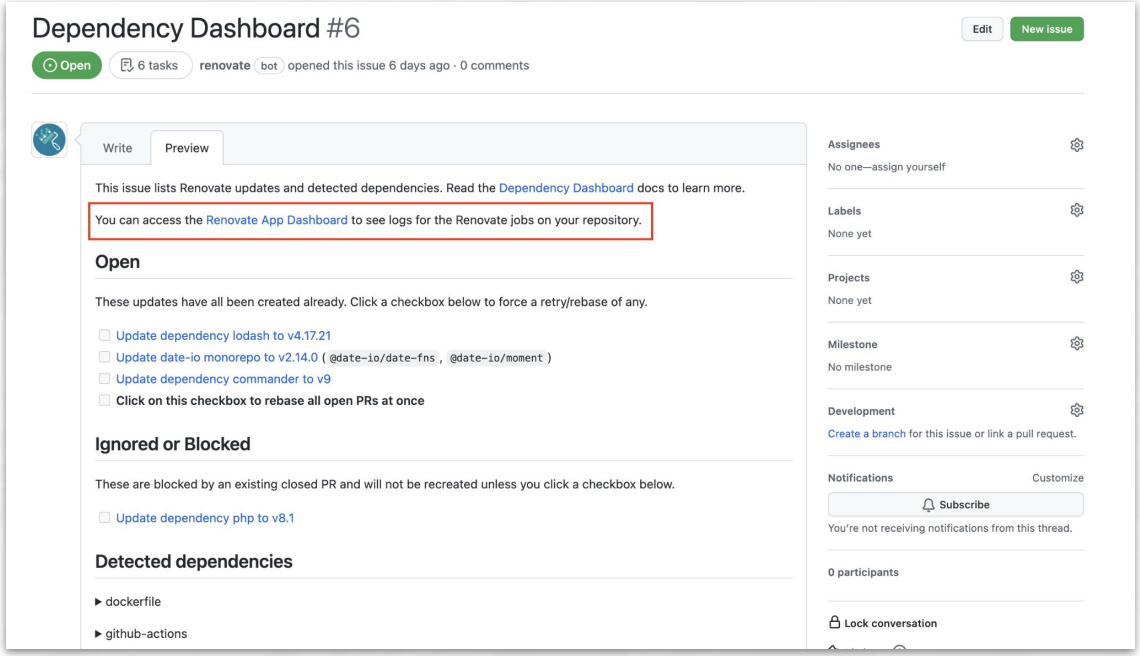


Automated dependencies management

https://github.com/renovatebot/renovate - https://github.com/dependabot

Renovate

Dependabot




Log in to sigstore

Takeaways

- ➔ Digital Signatures with Sigstore and Software Bills of materials.
- ➔ More informed choice of external dependencies with OSV, OpenSSF Scorecard and deps.dev
- ➔ Automate your dependencies management with Github DependaBot or Renovate for all other platforms.
- ➔ Demo of Sigstore, Syft, Grype, Chainguard images: https://www.youtube.com/watch?v=8osHp_h9bYU

Demo

THANKS