Securing your application software supply-chain

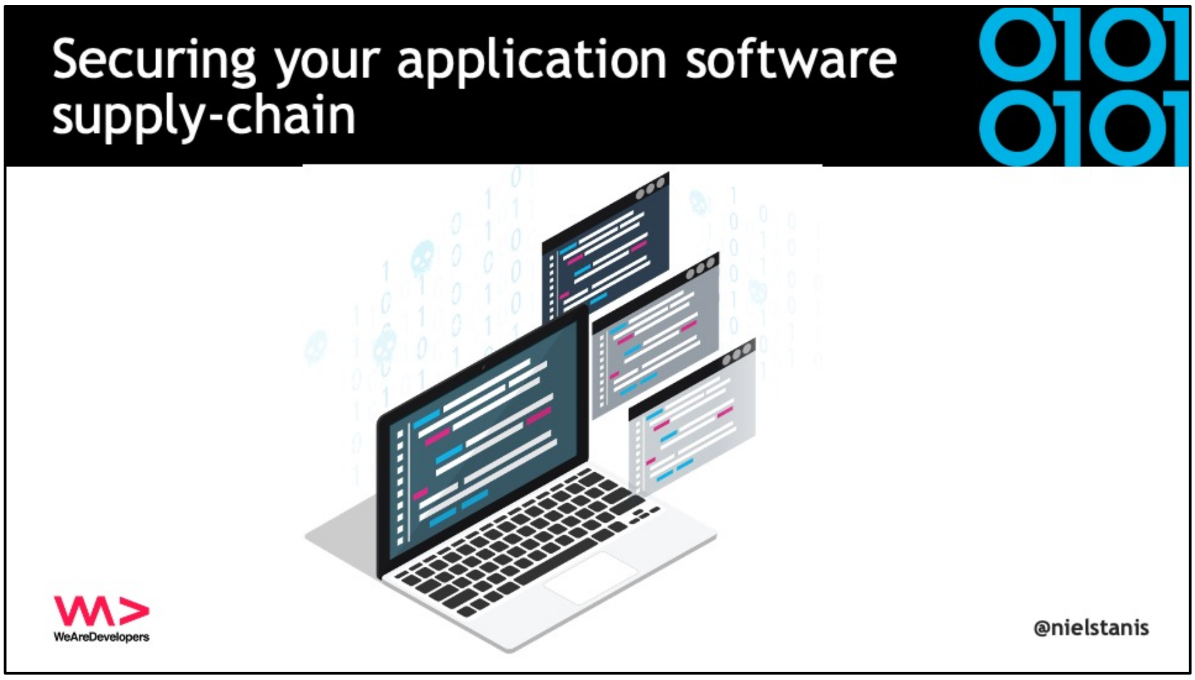
Niels Tanis

Who am I?

- Niels Tanis
- Senior Principal Security Researcher @ Veracode
  - Background .NET Development, Pentesting/ethical hacking, and software security consultancy
  - ISC^2 CSSLP
  - Research on static analysis for .NET apps

WeAreDevelopers

@nielstanis

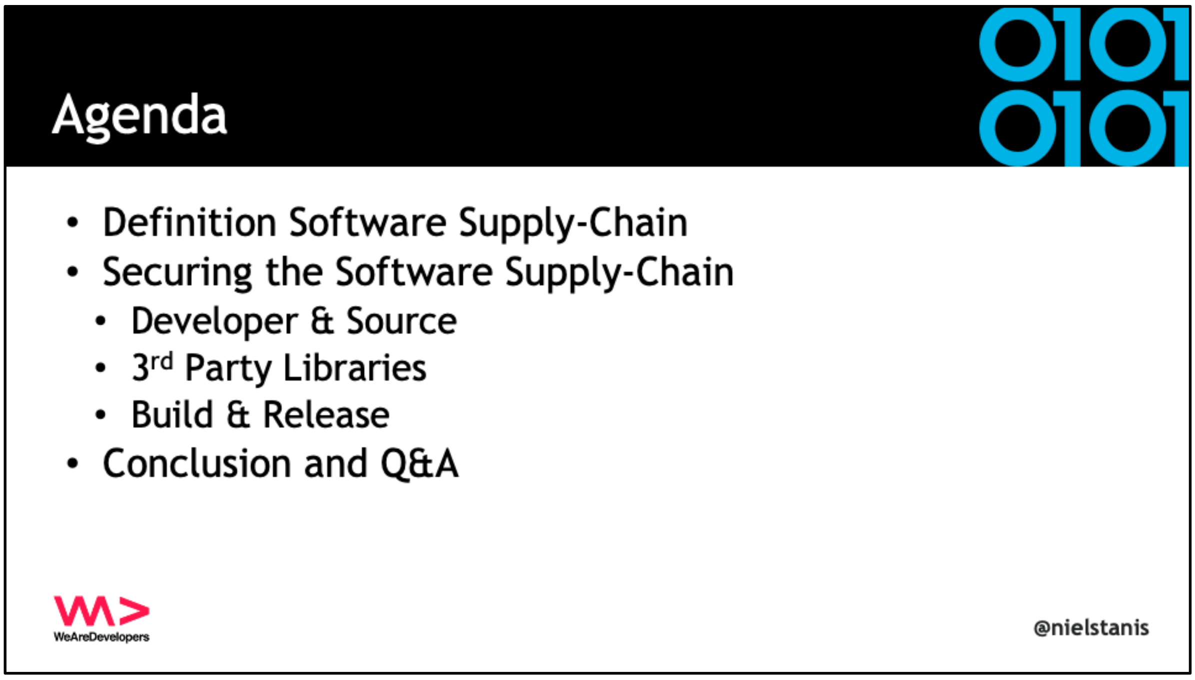


Securing your application software supply-chain

0101
0101
WeAreDevelopers
@nielstanis

Picture is from Veracode report/site:

https://www.veracode.com/sites/default/files/pdf/resources/ipapers/everything-you-need-to-know-open-source-risk/index.html



Agenda

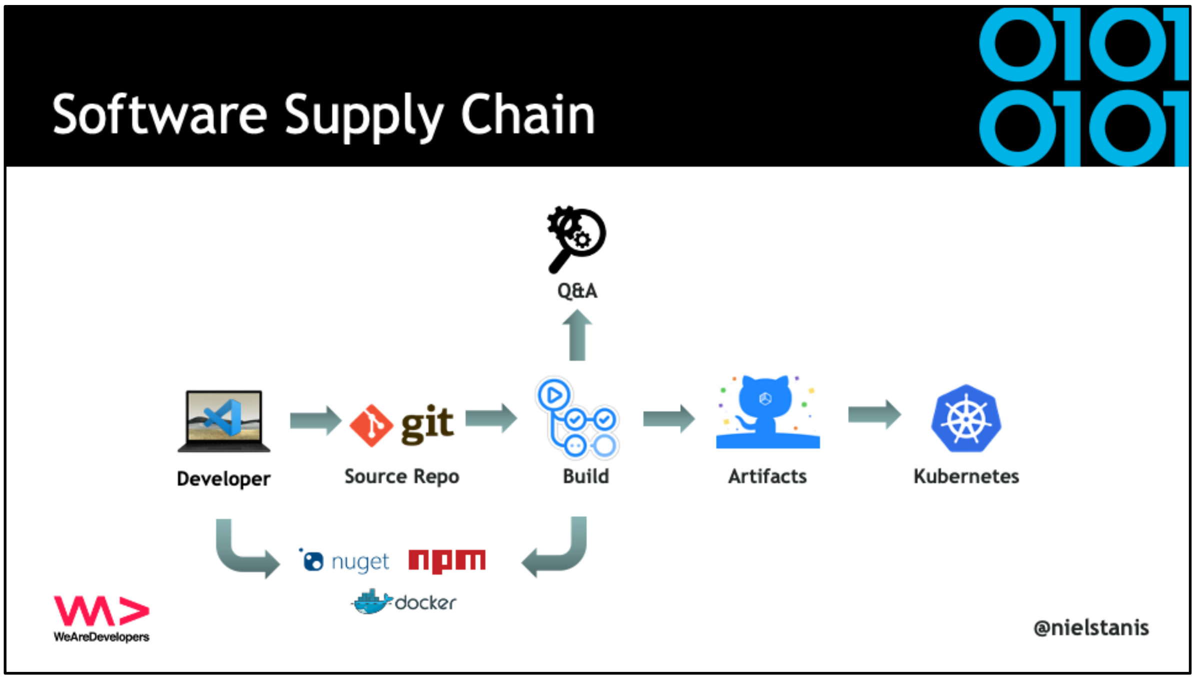
- Definition Software Supply-Chain
- Securing the Software Supply-Chain
  - Developer & Source
  - 3rd Party Libraries
  - Build & Release
- Conclusion and Q&A

WeAreDevelopers

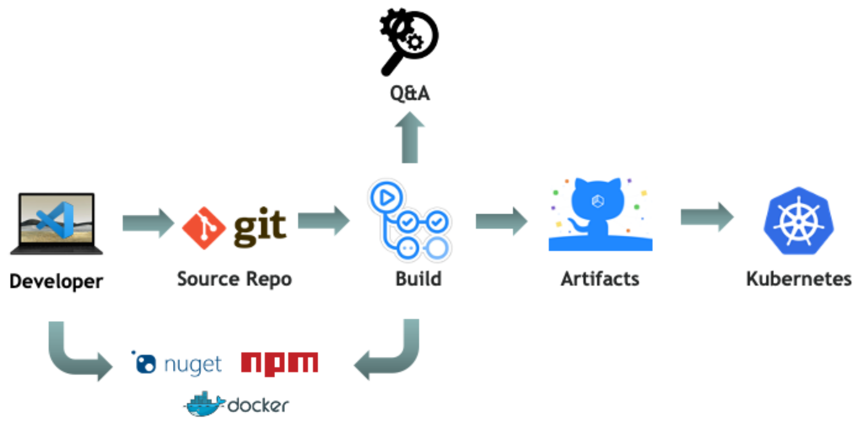
@nielstanis

What is a Supply Chain?

Image source:
https://www.wardsauto.com/sites/wardsauto.com/files/styles/article_featured_retina/public/Renault%20Kadjar%20assembly%20line%20-%20Palencia%20Spain-5_8.jpg?

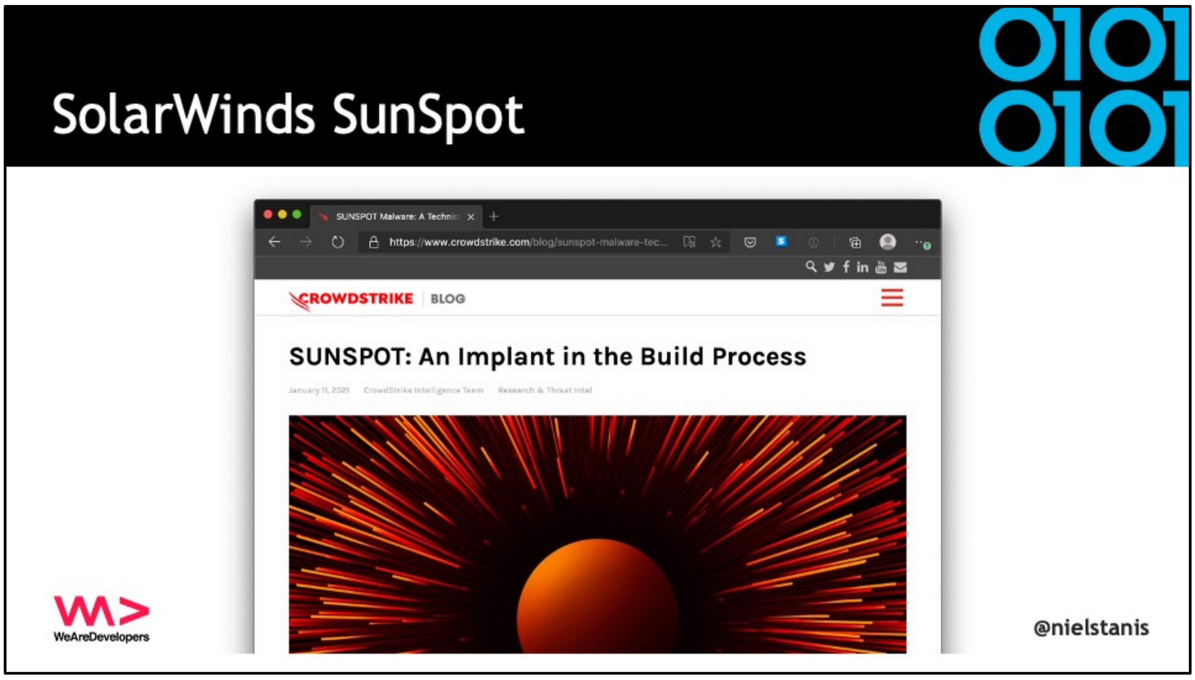


Software Supply Chain



WeAreDevelopers

@nielstanis



SolarWinds SunSpot

SUNSPOT Malware: A Technic

https://www.crowdstrike.com/blog/sunspot-malware-tec...

CROWDSTRIKE | BLOG

SUNSPOT: An Implant in the Build Process

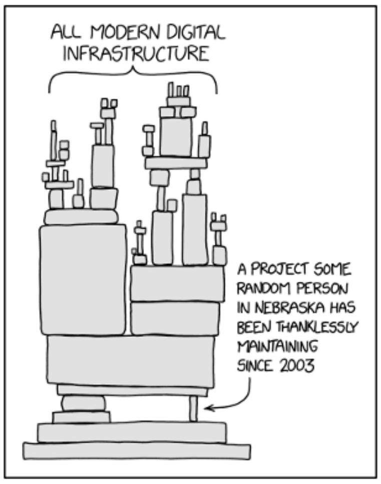
January 11, 2021 CrowdStrike Intelligence Team Research & Threat Intel

WeAreDevelopers

@nielstanis

https://www.crowdstrike.com/blog/sunspot-malware-technical-analysis/
https://www.microsoft.com/security/blog/2020/12/18/analyzing-solorigate-the-compromised-dll-file-that-started-a-sophisticated-cyberattack-and-how-microsoft-defender-helps-protect/

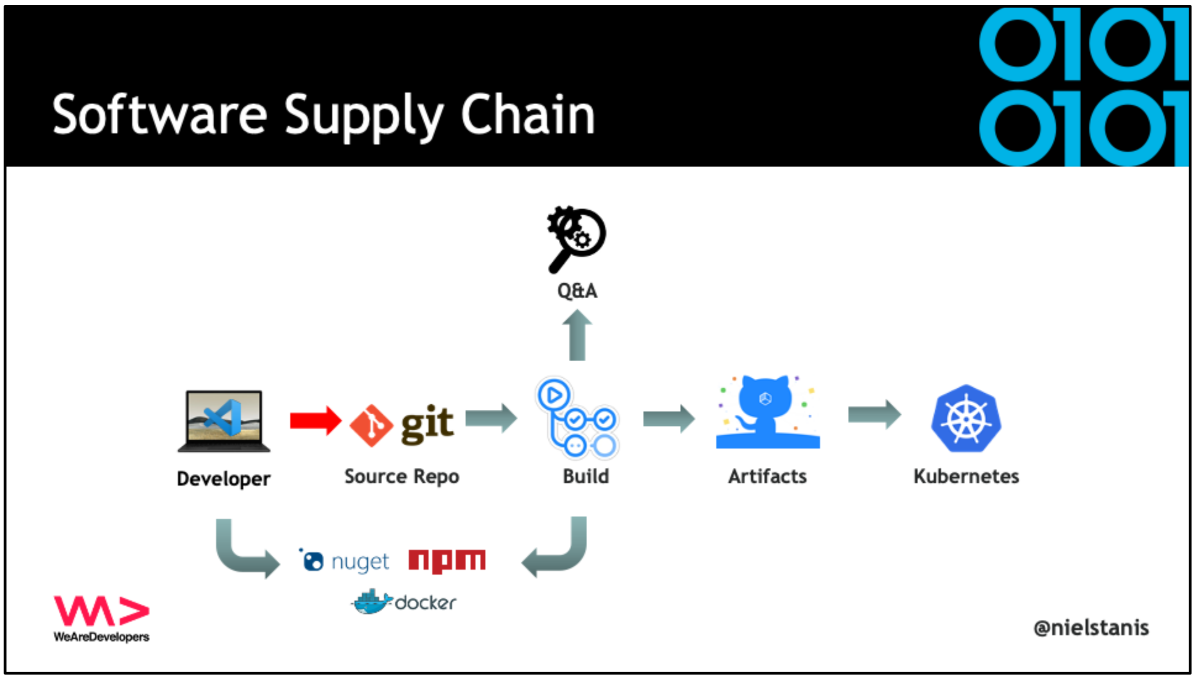
XKDC 2347 - Dependency


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

WeAreDevelopers

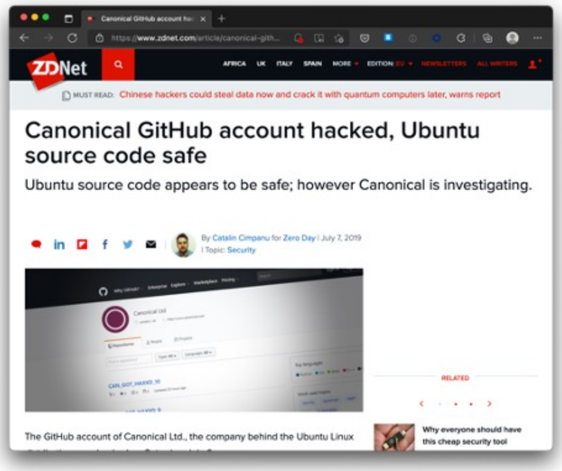
@nielstanis

https://xkcd.com/2347/



Software Supply Chain

GitHub account



Canonical GitHub account ha
https://www.zdnet.com/article/canonical-gith...
ZDNet
AFRICA UK ITALY SPAIN MORE EDITION: EU NEWSLETTERS ALL WRITERS
RELATED

MUST READ: Chinese hackers could steal data now and crack it with quantum computers later, warns report

Canonical GitHub account hacked, Ubuntu source code safe

Ubuntu source code appears to be safe; however Canonical is investigating.

By Catalin Cimpanu for Zero Day | July 7, 2019 | Topic: Security

The GitHub account of Canonical Ltd., the company behind the Ubuntu Linux

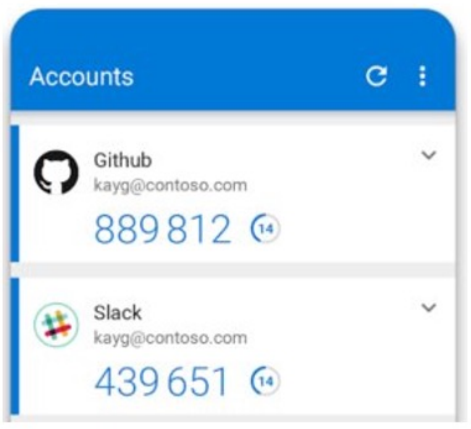
Why everyone should have this cheap security tool

WeAreDevelopers

@nielstanis

https://www.zdnet.com/article/canonical-github-account-hacked-ubuntu-source-code-safe/

Use MFA on source-repository

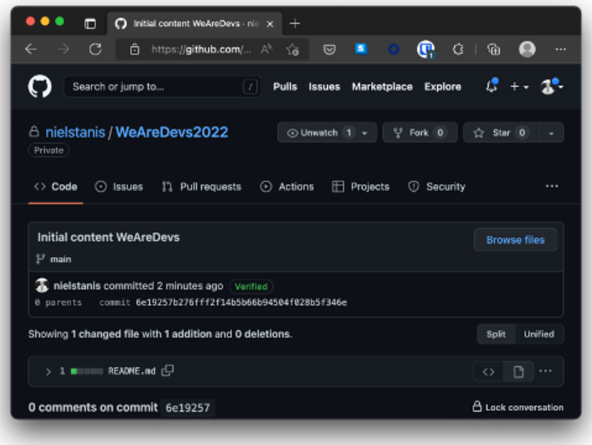


WeAreDevelopers

@nielstanis

https://help.github.com/en/github/authenticating-to-github/configuring-two-factor-authentication

GIT Commit Signing


nielstanis / WeAreDevs2022

Private

Initial content WeAreDevs

main

nielstanis committed 2 minutes ago Verified

```
0 parents    commit 6e19257b276fff2f14b5b66b94504f028b5f346e
```

Showing 1 changed file with 1 addition and 0 deletions.

```
> 1 README.md
```

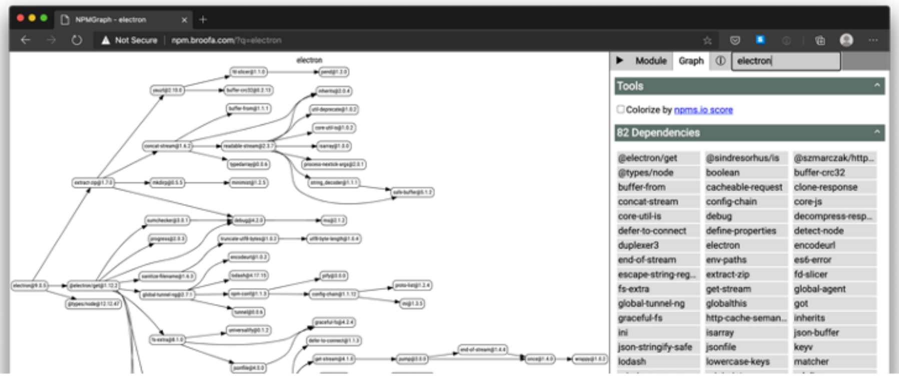
0 comments on commit 6e19257

WeAreDevelopers

@nielstanis

https://www.hanselman.com/blog/HowToSetupSignedGitCommitsWithAYubiKeyNEOAndGPGAndKeybaseOnWindows.aspx

Visual Studio Code



☐ Colorize by npms.io score

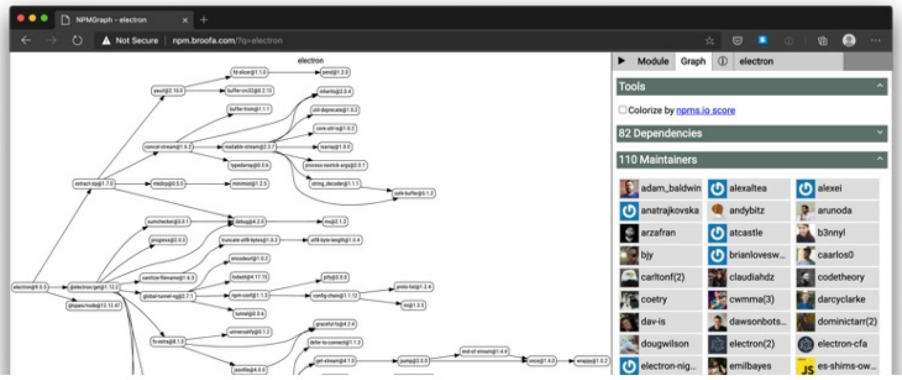
82 Dependencies

@electron/get	@sindresorhus/is	@szmarczak/http...
@types/node	boolean	buffer-crc32
buffer-from	cacheable-request	clone-response
concat-stream	config-chain	core-js
core-util-is	debug	decompress-resp...
defer-to-connect	define-properties	detect-node
duplexer3	electron	encodeurl
end-of-stream	env-paths	es6-error
escape-string-reg...	extract-zip	fd-slicer
fs-extra	get-stream	global-agent
global-tunnel-ng	globalthis	got
graceful-fs	http-cache-seman...	inherits
ini	isarray	json-buffer
json-stringify-safe	jsonfile	keyv
lodash	lowercase-keys	matcher

WeAreDevelopers

@nielstanis

Visual Studio Code



Tools

Colorize by npms.io score

82 Dependencies

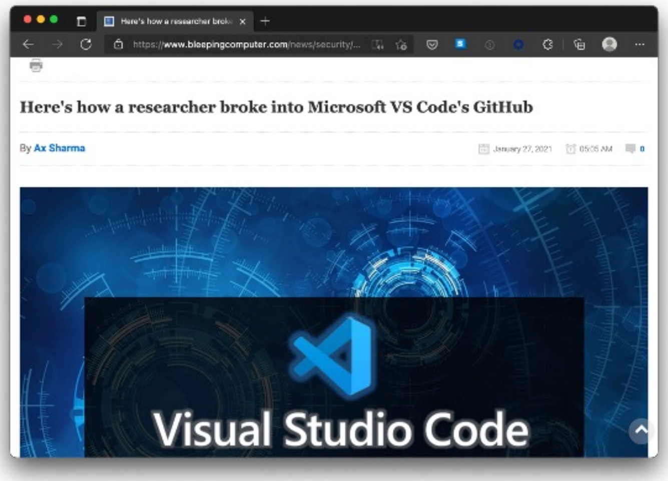
110 Maintainers

adam_baldwin	alexaltea	alexei
anatrajkovska	andybitz	arunoda
arzafran	atcastle	b3nnyl
bjy	brianlovesw...	caarlos0
carltonf(2)	claudiahdz	codetheory
coetry	cwmma(3)	darcyclarke
dav-is	dawsonbots...	dominictarr(2)
dougwilson	electron(2)	electron-cfa
electron-nig...	emilbayes	es-shims-ow...

WeAreDevelopers

@nielstanis

Visual Studio Code

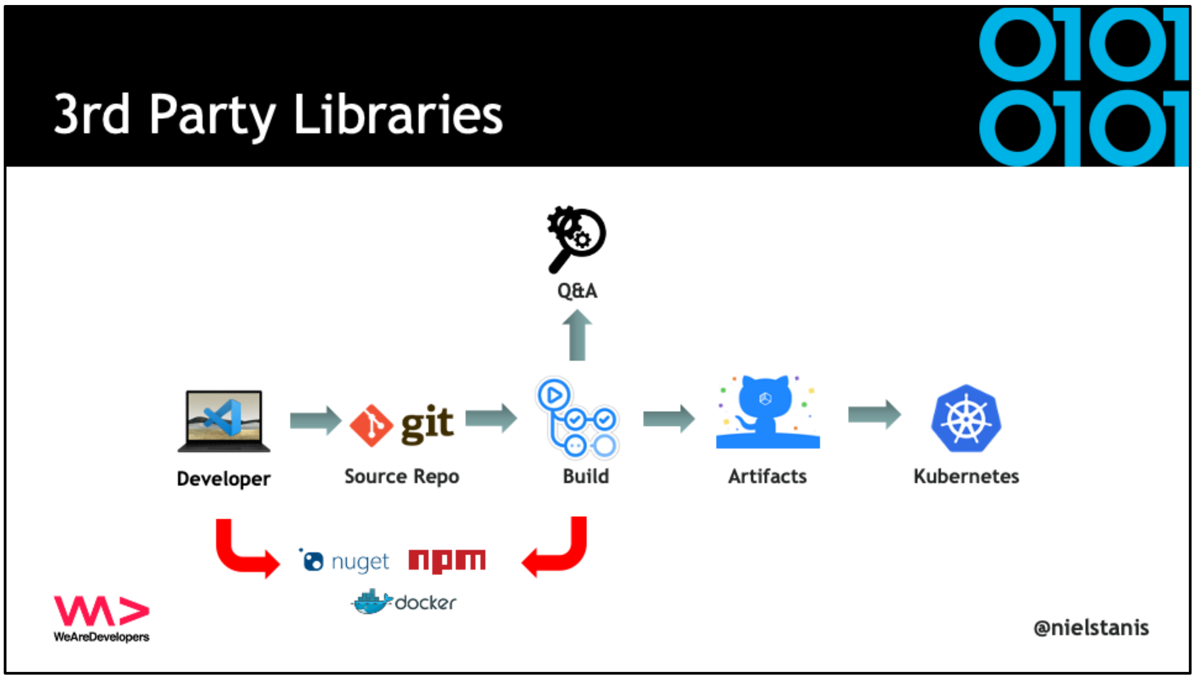

Here's how a researcher broke into Microsoft VS Code's GitHub

By Ax Sharma January 27, 2021 05:05 AM 0

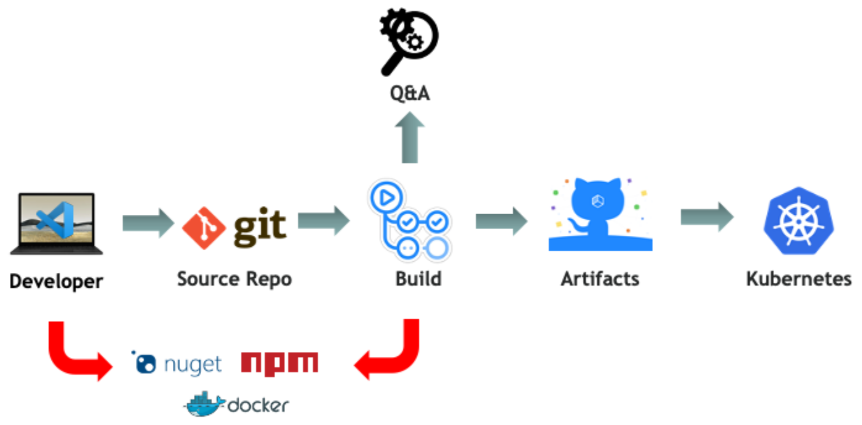
WeAreDevelopers

@nielstanis

https://www.bleepingcomputer.com/news/security/heres-how-a-researcher-broke-into-microsoft-vs-codes-github/

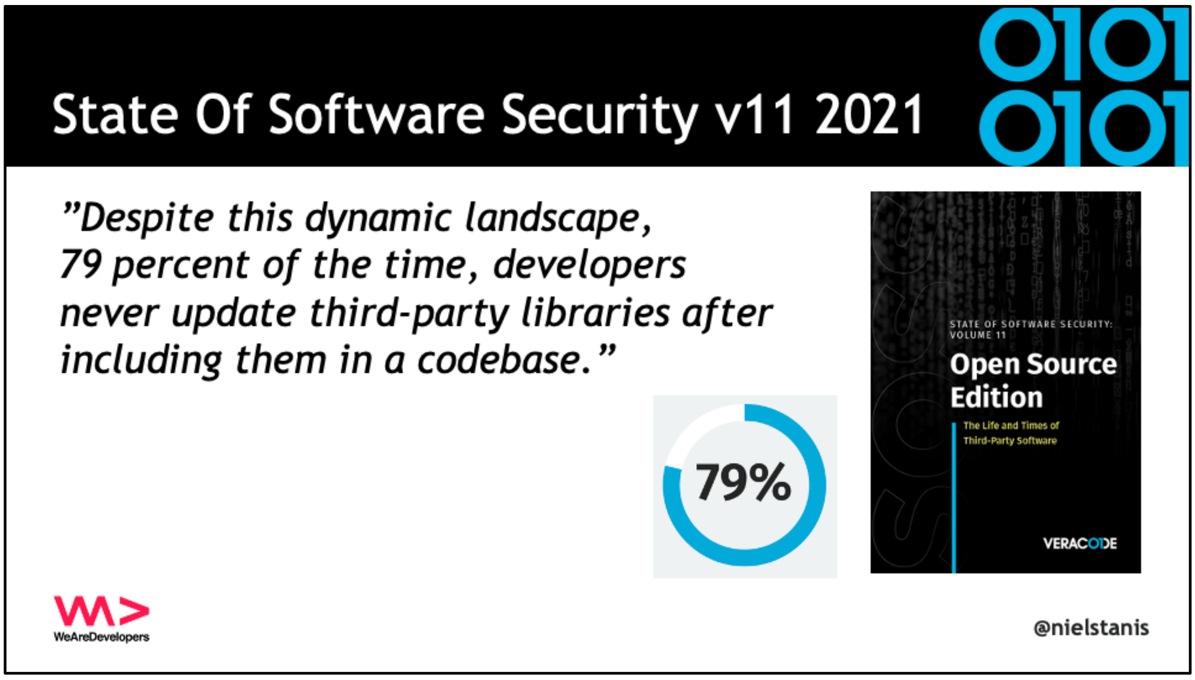


3rd Party Libraries



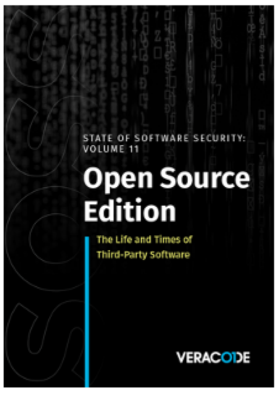
WeAreDevelopers

@nielstanis



State Of Software Security v11 2021

"Despite this dynamic landscape, 79 percent of the time, developers never update third-party libraries after including them in a codebase."

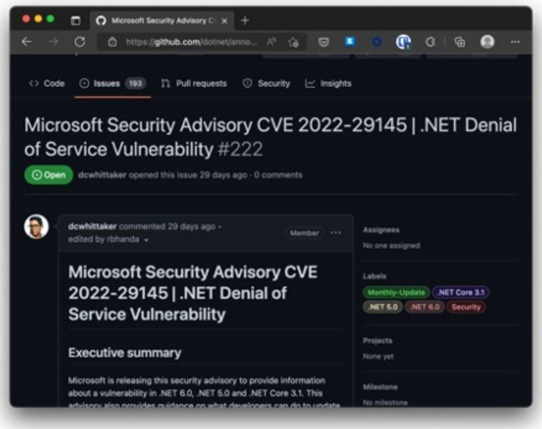

STATE OF SOFTWARE SECURITY: VOLUME 11
Open Source Edition
The Life and Times of Third-Party Software
VERACODE

WeAreDevelopers

@nielstanis

https://info.veracode.com/fy22-state-of-software-security-v11-open-source-edition.html

Vulnerabilities in libraries


Microsoft Security Advisory C
https://github.com/dotnet/anno...
Code
Issues 193
Pull requests
Security
Insights

Microsoft Security Advisory CVE 2022-29145 | .NET Denial of Service Vulnerability #222

Open dcwhittaker opened this issue 29 days ago · 0 comments

dcwhittaker commented 29 days ago · edited by rbhanda Member

Microsoft Security Advisory CVE 2022-29145 | .NET Denial of Service Vulnerability

Executive summary

Microsoft is releasing this security advisory to provide information about a vulnerability in .NET 6.0, .NET 5.0 and .NET Core 3.1. This

Assignees
No one assigned

Labels
Monthly-Update .NET Core 3.1 .NET 5.0 .NET 6.0 Security

Projects
None yet

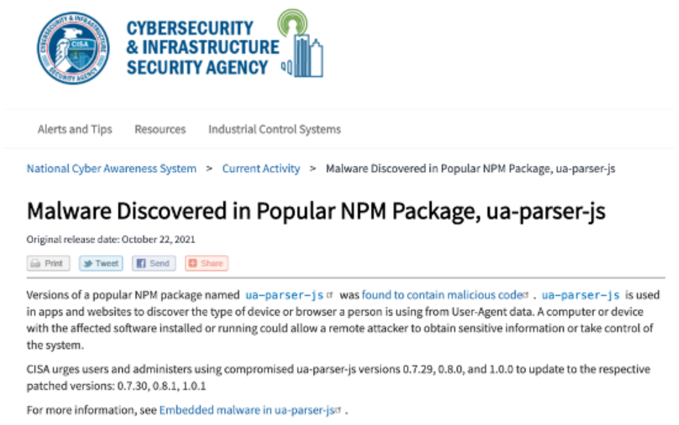
Milestone
No milestone

WeAreDevelopers

@nielstanis

https://github.com/dotnet/announcements/issues/222

Vulnerabilities in libraries


CYBERSECURITY & INFRASTRUCTURE SECURITY AGENCY

Alerts and Tips Resources Industrial Control Systems

National Cyber Awareness System > Current Activity > Malware Discovered in Popular NPM Package, ua-parser-js

Malware Discovered in Popular NPM Package, ua-parser-js

Original release date: October 22, 2021

Print Tweet Send Share

Versions of a popular NPM package named `ua-parser-js` was found to contain malicious code. `ua-parser-js` is used in apps and websites to discover the type of device or browser a person is using from User-Agent data. A computer or device with the affected software installed or running could allow a remote attacker to obtain sensitive information or take control of the system.

CISA urges users and administers using compromised ua-parser-js versions 0.7.29, 0.8.0, and 1.0.0 to update to the respective patched versions: 0.7.30, 0.8.1, 1.0.1

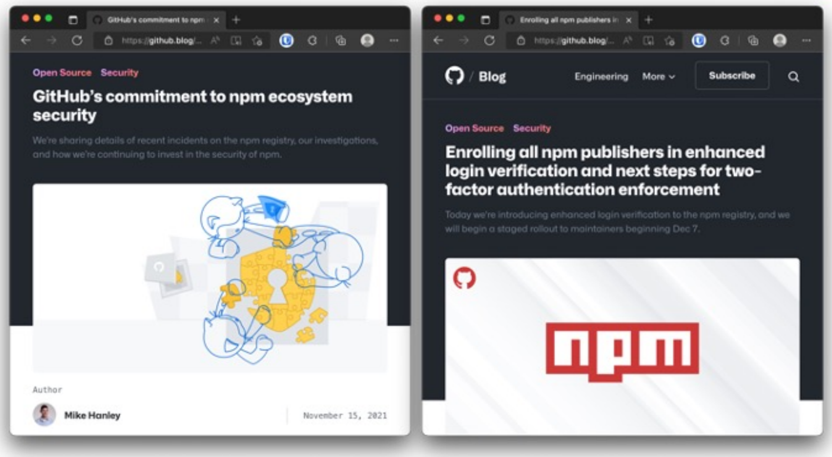
For more information, see Embedded malware in ua-parser-js .

WeAreDevelopers

@nielstanis

https://us-cert.cisa.gov/ncas/current-activity/2021/10/22/malware-discovered-popular-npm-package-ua-parser-js
https://portswigger.net/daily-swig/popular-npm-package-ua-parser-js-poisoned-with-cryptomining-password-stealing-malware

Vulnerabilities in libraries


Open Source Security

GitHub's commitment to npm ecosystem security

We're sharing details of recent incidents on the npm registry, our investigations, and how we're continuing to invest in the security of npm.

Author

Mike Hanley November 15, 2021

Open Source Security

Enrolling all npm publishers in enhanced login verification and next steps for two-factor authentication enforcement

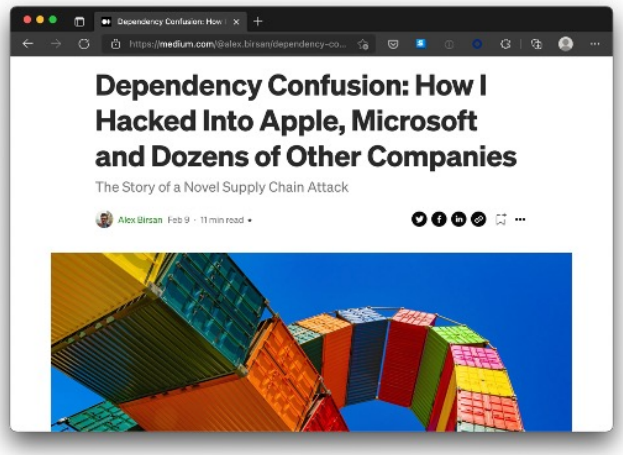
Today we're introducing enhanced login verification to the npm registry, and we will begin a staged rollout to maintainers beginning Dec 7.

WeAreDevelopers

@nielstanis

https://github.blog/2021-11-15-githubs-commitment-to-npm-ecosystem-security/
https://github.blog/2021-12-07-enrolling-npm-publishers-enhanced-login-verification-two-factor-authentication-enforcement/

Dependency Confusion


Dependency Confusion: How I

https://medium.com/@alex.birsan/dependency-co...

Dependency Confusion: How I Hacked Into Apple, Microsoft and Dozens of Other Companies

The Story of a Novel Supply Chain Attack

Alex Birsan · Feb 9 · 11 min read

WeAreDevelopers

@nielstanis

https://medium.com/@alex.birsan/dependency-confusion-4a5d60fec610

Microsoft Whitepaper

3 ways to mitigate risk when using private package feeds

Secure Your Hybrid Software Supply Chain

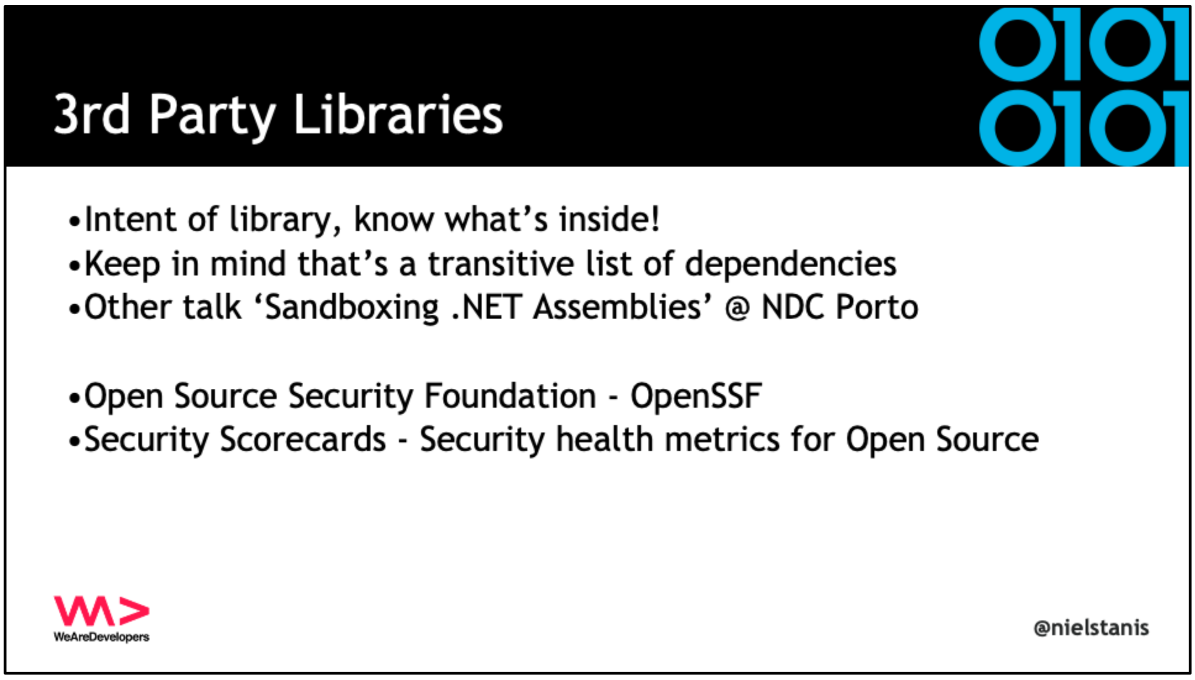
An always-up-to-date version of this whitepaper is located at: https://aka.ms/pkg-sec-wp

WeAreDevelopers

@nielstanis

https://azure.microsoft.com/nl-nl/resources/3-ways-to-mitigate-risk-using-private-package-feeds/

https://azure.microsoft.com/mediahandler/files/resourcefiles/3-ways-to-mitigate-risk-using-private-package-feeds/3%20Ways%20to%20Mitigate%20Risk%20When%20Using%20Private%20Package%20Feeds%20-%20v1.0.pdf



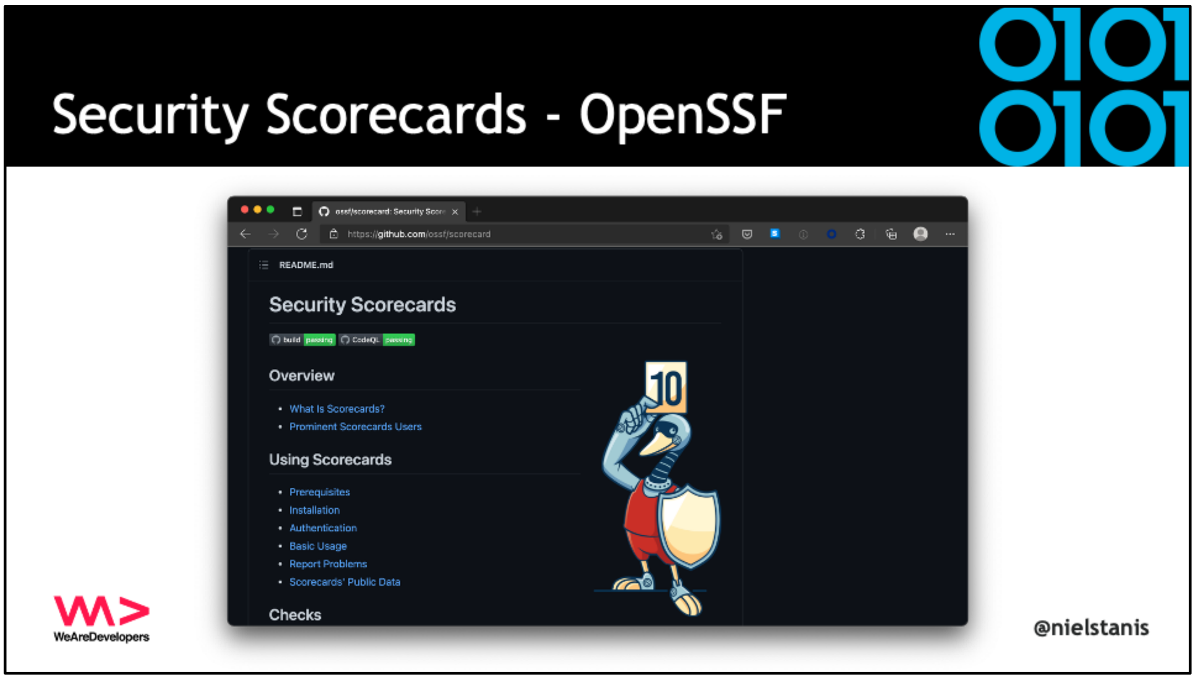
3rd Party Libraries

- Intent of library, know what's inside!
- Keep in mind that's a transitive list of dependencies
- Other talk 'Sandboxing .NET Assemblies' @ NDC Porto

- Open Source Security Foundation - OpenSSF
- Security Scorecards - Security health metrics for Open Source

WeAreDevelopers

@nielstanis



Security Scorecards - OpenSSF

ossf/scorecard: Security Score...

https://github.com/ossf/scorecard

README.md

Security Scorecards

build passing | CodeQL passing

Overview

- What Is Scorecards?
- Prominent Scorecards Users

Using Scorecards

- Prerequisites
- Installation
- Authentication
- Basic Usage
- Report Problems
- Scorecards' Public Data

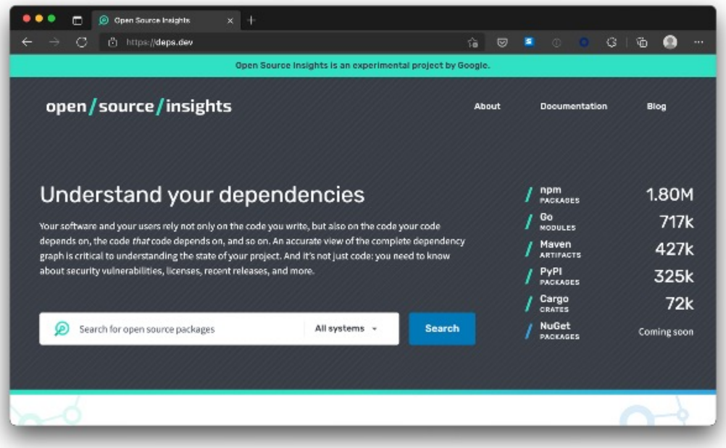
Checks

WeAreDevelopers

@nielstanis

https://github.com/ossf/scorecard

Deps.Dev by Google



Open Source Insights is an experimental project by Google.

open/source/insights

Understand your dependencies

Your software and your users rely not only on the code you write, but also on the code your code depends on, the code *that* code depends on, and so on. An accurate view of the complete dependency graph is critical to understanding the state of your project. And it's not just code: you need to know about security vulnerabilities, licenses, recent releases, and more.

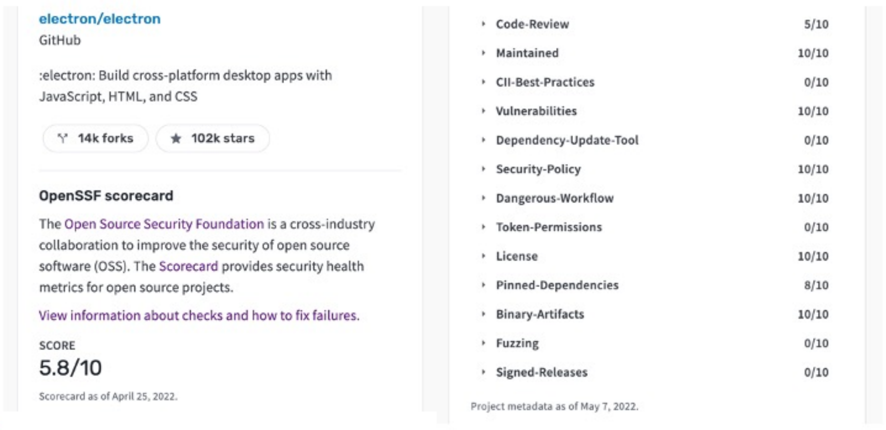
npm PACKAGES	1.80M
Go MODULES	717k
Maven ARTIFACTS	427k
PyPI PACKAGES	325k
Cargo CRATES	72k
NuGet PACKAGES	Coming soon

WeAreDevelopers

@nielstanis

https://deps.dev/

Deps.Dev by Google


electron/electron
GitHub

:electron: Build cross-platform desktop apps with JavaScript, HTML, and CSS

14k forks 102k stars

OpenSSF scorecard

The Open Source Security Foundation is a cross-industry collaboration to improve the security of open source software (OSS). The Scorecard provides security health metrics for open source projects.

View information about checks and how to fix failures.

SCORE
5.8/10

Scorecard as of April 25, 2022.

Code-Review	5/10
Maintained	10/10
CII-Best-Practices	0/10
Vulnerabilities	10/10
Dependency-Update-Tool	0/10
Security-Policy	10/10
Dangerous-Workflow	10/10
Token-Permissions	0/10
License	10/10
Pinned-Dependencies	8/10
Binary-Artifacts	10/10
Fuzzing	0/10
Signed-Releases	0/10

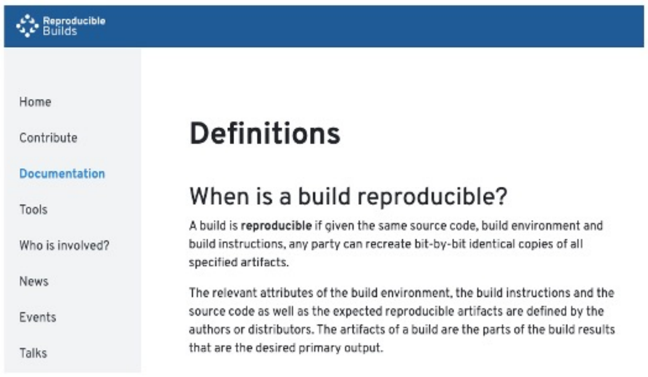
Project metadata as of May 7, 2022.

WeAreDevelopers

@nielstanis

https://deps.dev/npm/electron

Reproducible/Deterministic Builds



Definitions

When is a build reproducible?

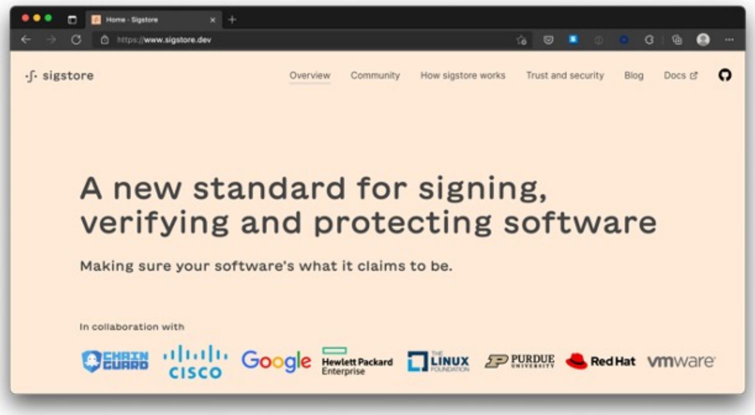
A build is **reproducible** if given the same source code, build environment and build instructions, any party can recreate bit-by-bit identical copies of all specified artifacts.

The relevant attributes of the build environment, the build instructions and the source code as well as the expected reproducible artifacts are defined by the authors or distributors. The artifacts of a build are the parts of the build results that are the desired primary output.

@nielstanis

https://reproducible-builds.org/docs/definition/

Signing artifacts



A new standard for signing, verifying and protecting software

Making sure your software's what it claims to be.

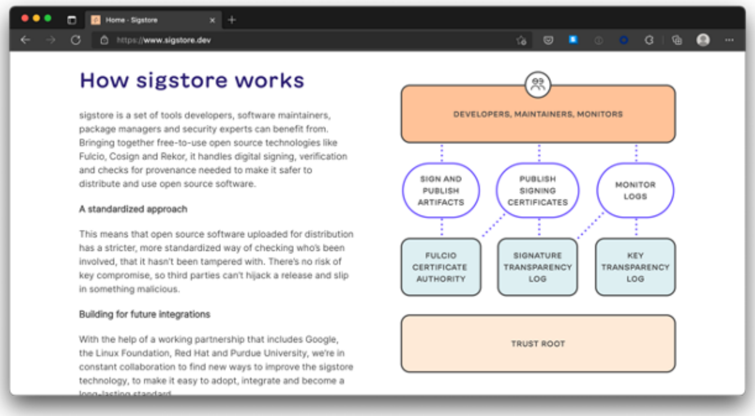
In collaboration with

WeAreDevelopers

@nielstanis

https://sigstore.dev

Signing artifacts


Home - Sigstore
https://www.sigstore.dev
DEVELOPERS, MAINTAINERS, MONITORS
SIGN AND PUBLISH ARTIFACTS
PUBLISH SIGNING CERTIFICATES
MONITOR LOGS
FULCIO CERTIFICATE AUTHORITY
SIGNATURE TRANSPARENCY LOG
KEY TRANSPARENCY LOG
TRUST ROOT

How sigstore works

sigstore is a set of tools developers, software maintainers, package managers and security experts can benefit from. Bringing together free-to-use open source technologies like Fulcio, Cosign and Rekor, it handles digital signing, verification and checks for provenance needed to make it safer to distribute and use open source software.

A standardized approach

This means that open source software uploaded for distribution has a stricter, more standardized way of checking who's been involved, that it hasn't been tampered with. There's no risk of key compromise, so third parties can't hijack a release and slip in something malicious.

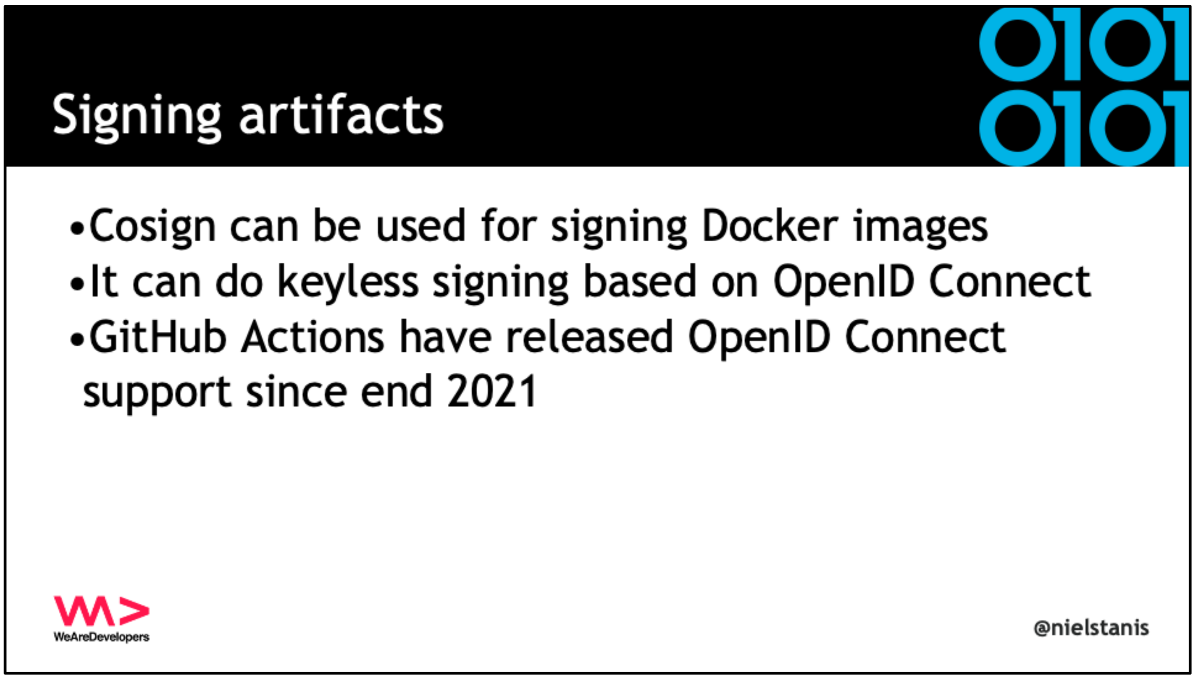
Building for future integrations

With the help of a working partnership that includes Google, the Linux Foundation, Red Hat and Purdue University, we're in constant collaboration to find new ways to improve the sigstore technology, to make it easy to adopt, integrate and become a long-lasting standard.

WeAreDevelopers

@nielstanis

https://sigstore.dev



Signing artifacts

- Cosign can be used for signing Docker images
- It can do keyless signing based on OpenID Connect
- GitHub Actions have released OpenID Connect support since end 2021

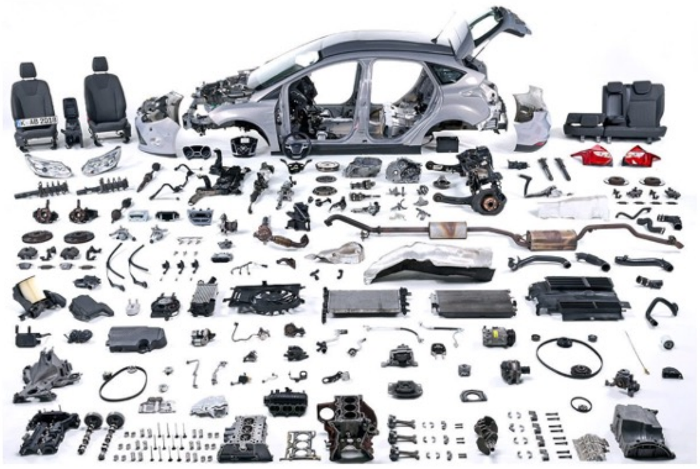
WeAreDevelopers

@nielstanis

https://sigstore.dev

Automotive Industry

0101
0101



WeAreDevelopers

@nielstanis

Car Supply Chain

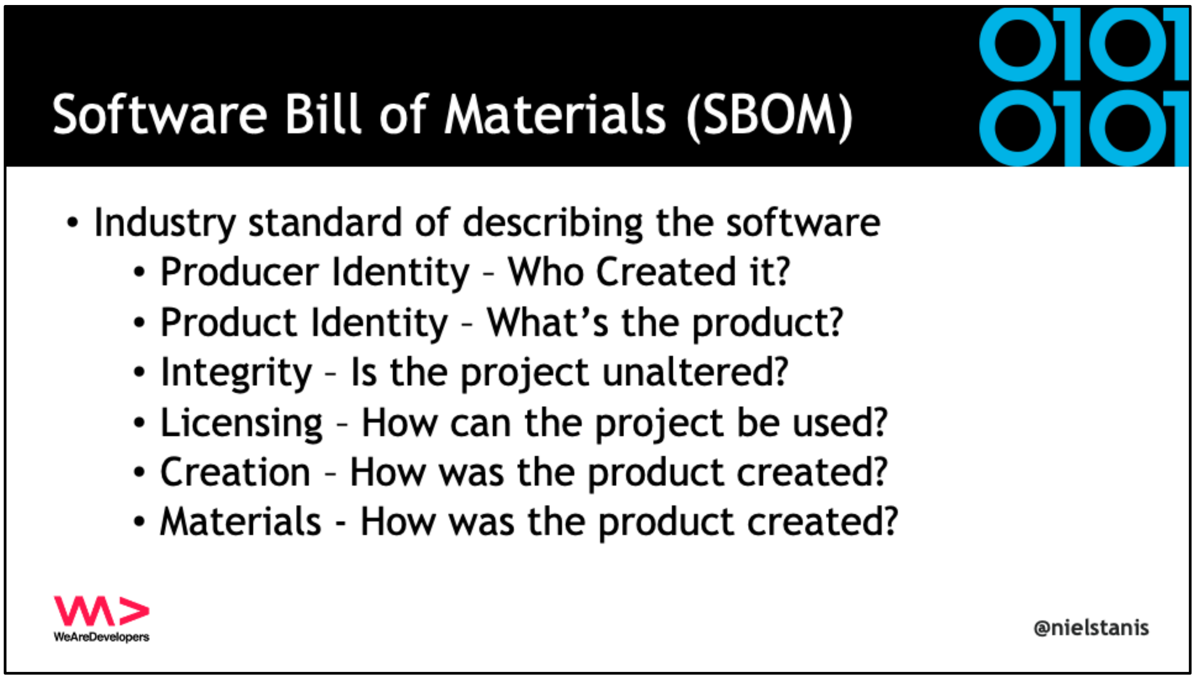
Tata Steel Factory
- Iron Ore from Sweden
- ISO 6892-1 Tested/Certified
- Batch #1234

Bosch Factory
- Steel Batch #1234 Tata
- ECE-R90 Tested/Certified
- Serie #45678
- Used by Ford, Volkswagen and Renault

Renault Manufacturing
- Bosch Disk #45678
- Bosal Exhaust #RE9876
- Goodyear Tires #GY8877
- Kadjar VIN 1234567890

WeAreDevelopers

@nielstanis



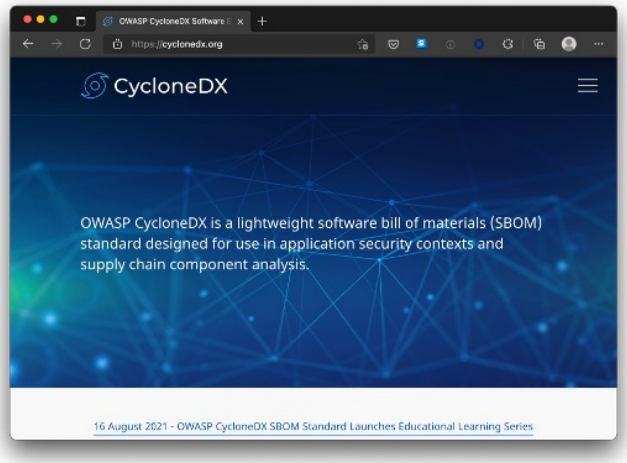
Software Bill of Materials (SBOM)

- Industry standard of describing the software
  - Producer Identity – Who Created it?
  - Product Identity – What's the product?
  - Integrity – Is the project unaltered?
  - Licensing – How can the project be used?
  - Creation – How was the product created?
  - Materials - How was the product created?

WeAreDevelopers

@nielstanis

Software Bill of Materials (SBOM)



OWASP CycloneDX is a lightweight software bill of materials (SBOM) standard designed for use in application security contexts and supply chain component analysis.

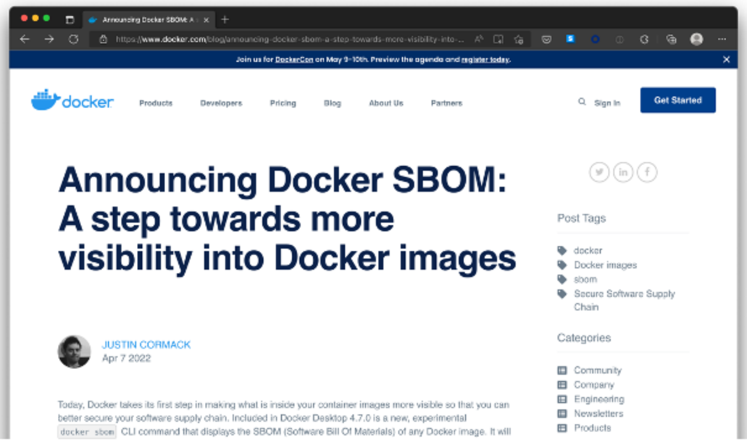
16 August 2021 - OWASP CycloneDX SBOM Standard Launches Educational Learning Series

WeAreDevelopers

@nielstanis

https://cyclonedx.org

Docker SBOM


Announcing Docker SBOM: A
https://www.docker.com/blog/announcing-docker-sbom-a-step-towards-more-visibility-into-...
Join us for DockerCon on May 9-10th. Preview the agenda and register today.
docker
Products
Developers
Pricing
Blog
About Us
Partners
Sign In
Get Started

Announcing Docker SBOM: A step towards more visibility into Docker images

JUSTIN CORMACK
Apr 7 2022

Today, Docker takes its first step in making what is inside your container images more visible so that you can better secure your software supply chain. Included in Docker Desktop 4.7.0 is a new, experimental `docker sbom` CLI command that displays the SBOM (Software Bill Of Materials) of any Docker image. It will

Post Tags

- docker
- Docker images
- sbom
- Secure Software Supply Chain

Categories

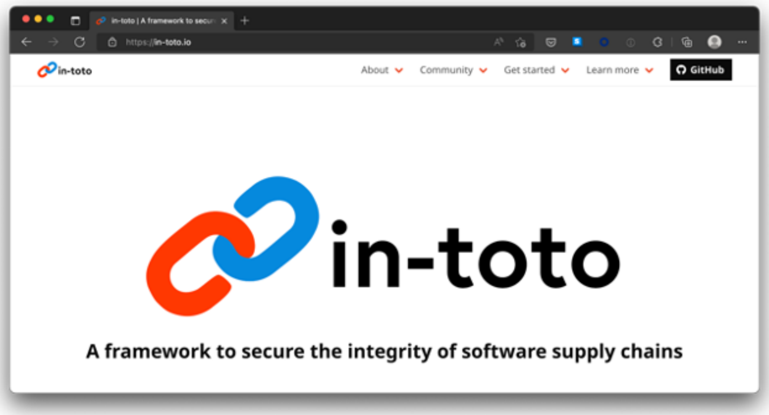
- Community
- Company
- Engineering
- Newsletters
- Products

WeAreDevelopers

@nielstanis

https://www.docker.com/blog/announcing-docker-sbom-a-step-towards-more-visibility-into-docker-images/

In-toto

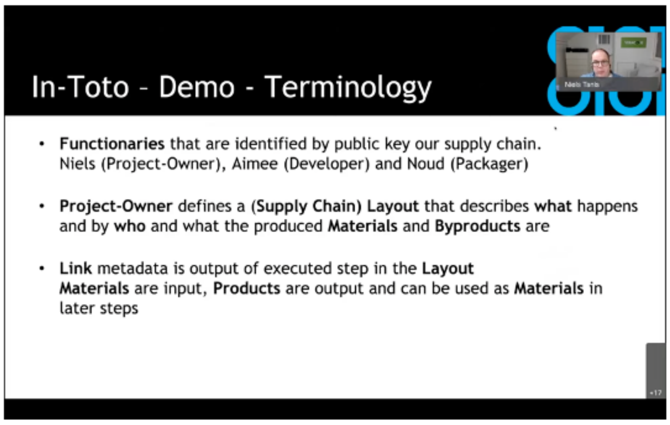


A framework to secure the integrity of software supply chains

WeAreDevelopers

@nielstanis

In-Toto – Demo



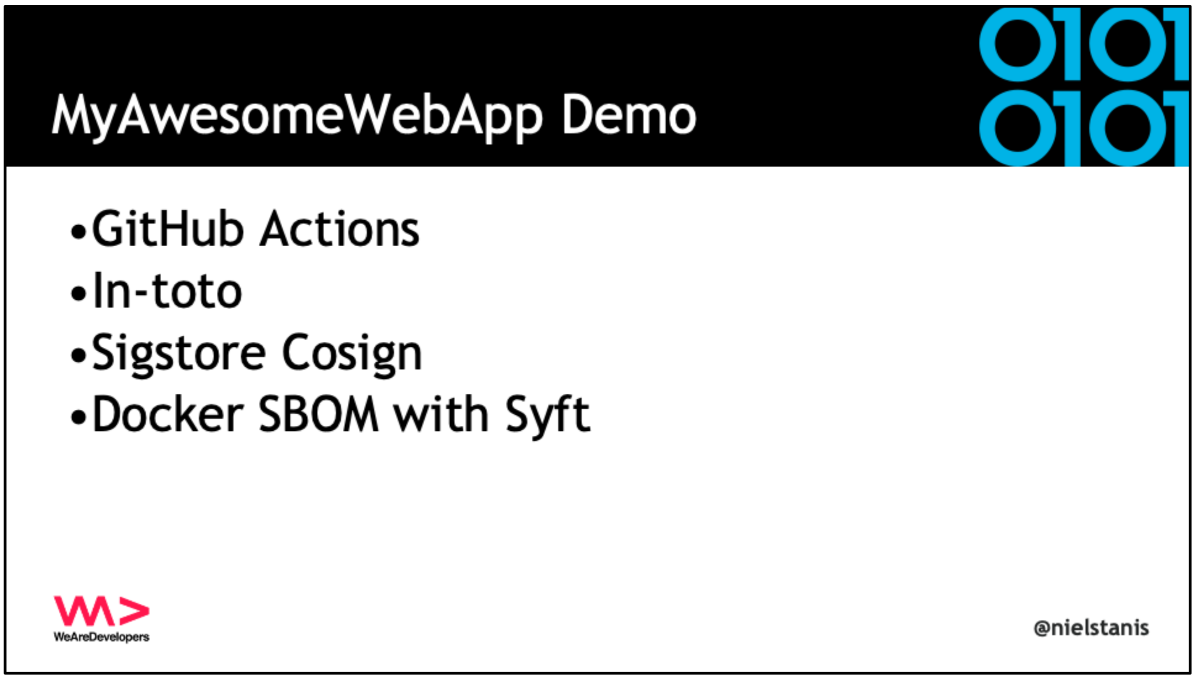
In-Toto - Demo - Terminology

- **Functionaries** that are identified by public key our supply chain.
  Niels (Project-Owner), Aimee (Developer) and Noud (Packager)
- **Project-Owner** defines a **(Supply Chain) Layout** that describes **what** happens and by **who** and what the produced **Materials** and **Byproducts** are
- **Link** metadata is output of executed step in the **Layout**
  Materials are input, **Products** are output and can be used as **Materials** in later steps

WeAreDevelopers

@nielstanis

https://youtu.be/fYCfB7MZPh4?t=2777



MyAwesomeWebApp Demo

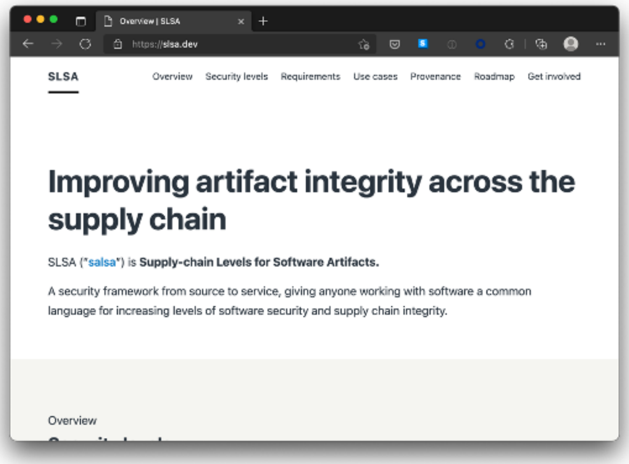
- GitHub Actions
- In-toto
- Sigstore Cosign
- Docker SBOM with Syft

WeAreDevelopers

@nielstanis

https://github.com/nielstanis/myawesome-webapp

Google SLSA



Improving artifact integrity across the supply chain

SLSA ("salsa") is **Supply-chain Levels for Software Artifacts.**

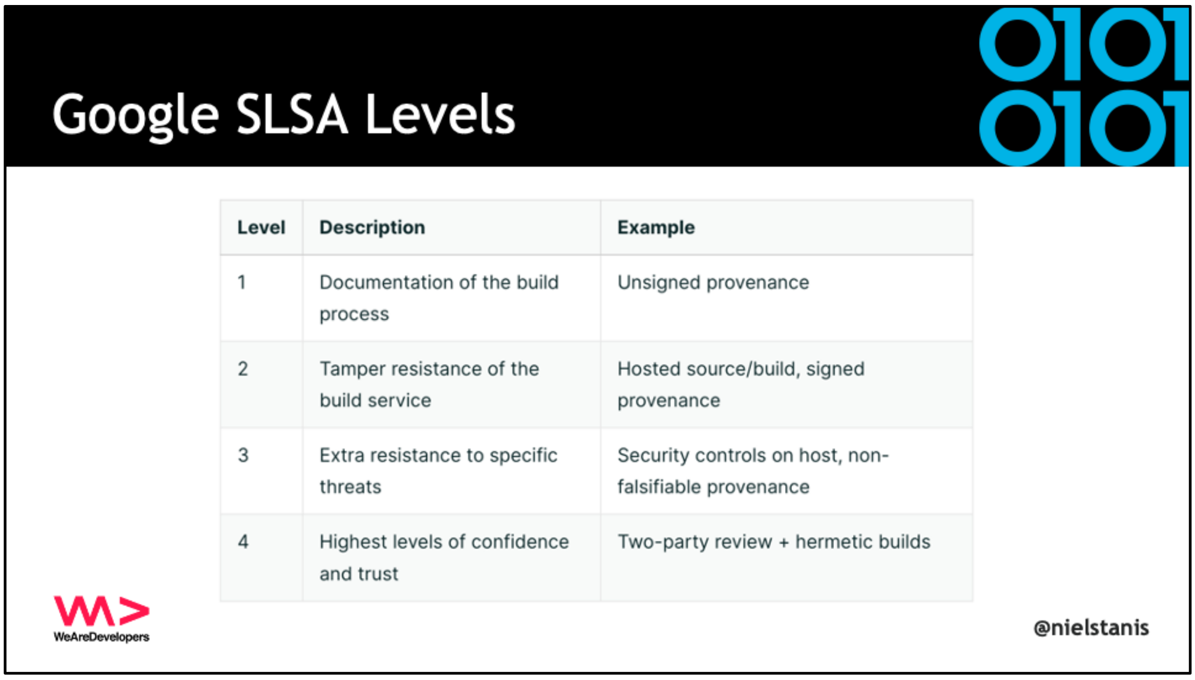
A security framework from source to service, giving anyone working with software a common language for increasing levels of software security and supply chain integrity.

Overview

WeAreDevelopers

@nielstanis

https://slsa.dev



Google SLSA Levels

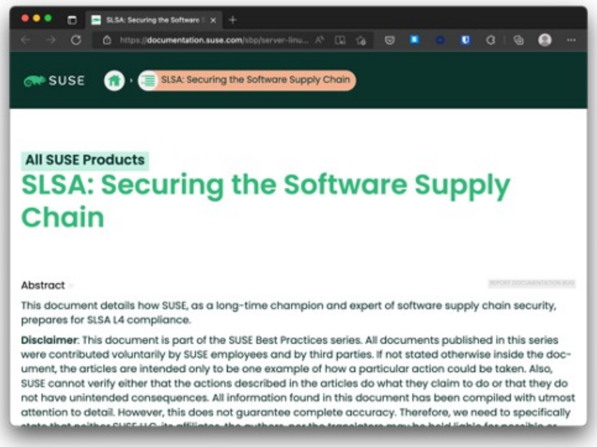
Level	Description	Example
1	Documentation of the build process	Unsigned provenance
2	Tamper resistance of the build service	Hosted source/build, signed provenance
3	Extra resistance to specific threats	Security controls on host, non-falsifiable provenance
4	Highest levels of confidence and trust	Two-party review + hermetic builds

WeAreDevelopers

@nielstanis

https://slsa.dev

SUSE SLSA Level 4


SLSA: Securing the Software
https://documentation.suse.com/sbp/server-linu...
SUSE
SLSA: Securing the Software Supply Chain

All SUSE Products

SLSA: Securing the Software Supply Chain

Abstract

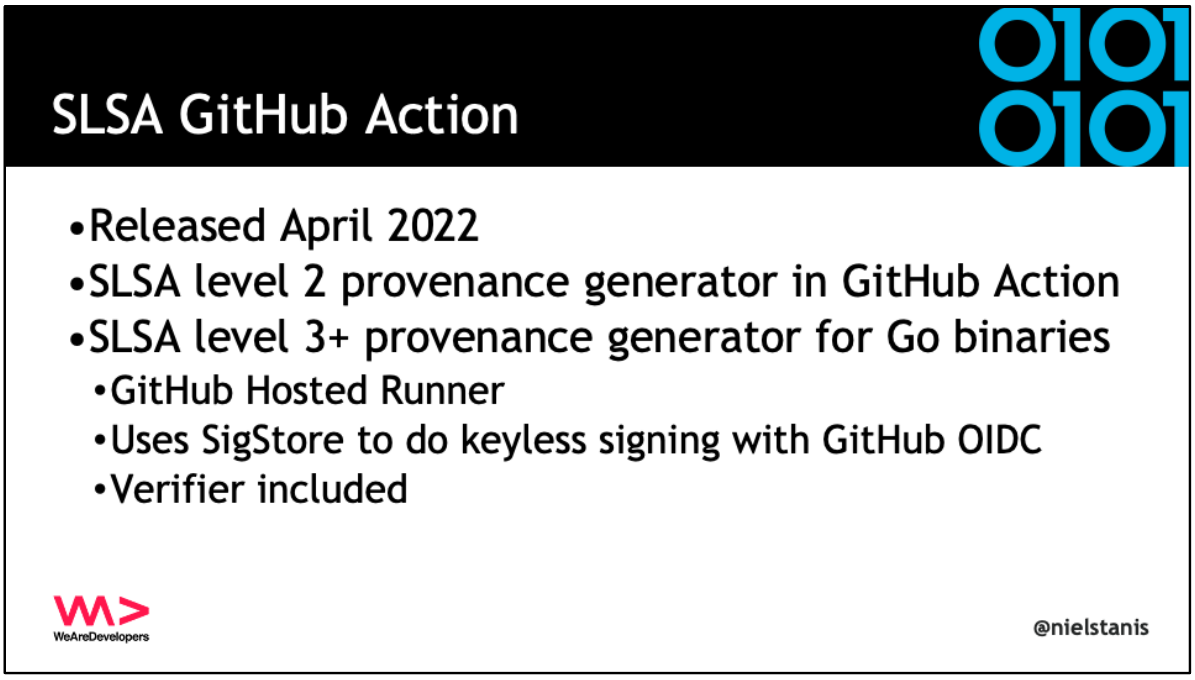
This document details how SUSE, as a long-time champion and expert of software supply chain security, prepares for SLSA L4 compliance.

Disclaimer: This document is part of the SUSE Best Practices series. All documents published in this series were contributed voluntarily by SUSE employees and by third parties. If not stated otherwise inside the document, the articles are intended only to be one example of how a particular action could be taken. Also, SUSE cannot verify either that the actions described in the articles do what they claim to do or that they do not have unintended consequences. All information found in this document has been compiled with utmost attention to detail. However, this does not guarantee complete accuracy. Therefore, we need to specifically

WeAreDevelopers

@nielstanis

https://documentation.suse.com/sbp/server-linux/html/SBP-SLSA4/index.html



SLSA GitHub Action

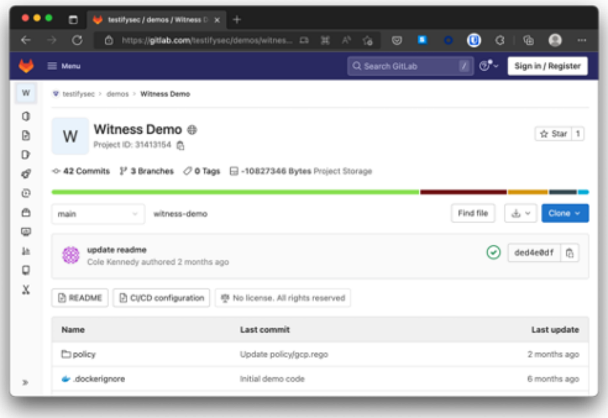
- Released April 2022
- SLSA level 2 provenance generator in GitHub Action
- SLSA level 3+ provenance generator for Go binaries
  - GitHub Hosted Runner
  - Uses SigStore to do keyless signing with GitHub OIDC
  - Verifier included

WeAreDevelopers

@nielstanis

https://security.googleblog.com/2022/04/improving-software-supply-chain.html

Witness & GitLab Attestator


testifysec / demos / Witness D

https://gitlab.com/testifysec/demos/witnes...

Menu · Search GitLab · Sign in / Register

testifysec > demos > Witness Demo

Witness Demo
Project ID: 31413154

42 Commits · 3 Branches · 0 Tags · -10827346 Bytes Project Storage

main · witness-demo · Find file · Clone

update readme
Cole Kennedy authored 2 months ago
ded4e0df

README · CI/CD configuration · No license. All rights reserved

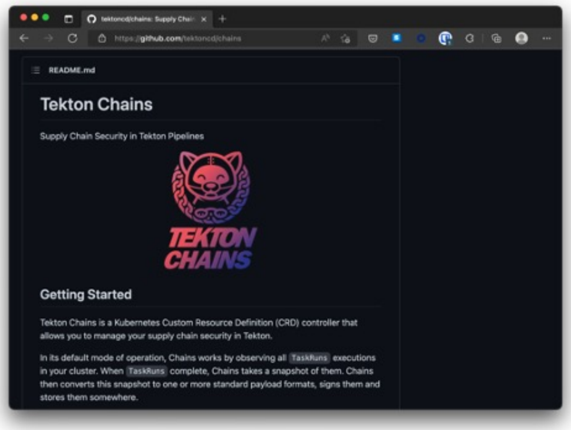
Name	Last commit	Last update
policy	Update policy/gcp.rego	2 months ago
.dockerignore	Initial demo code	6 months ago

WeAreDevelopers

@nielstanis

https://gitlab.com/testifysec/demos/witness-demo
https://github.com/testifysec/witness

Open Shift - Tekton - Tekton Chains


tektoncd/chains: Supply Chai
https://github.com/tektoncd/chains
README.md
TEKTON CHAINS

Tekton Chains

Supply Chain Security in Tekton Pipelines

Getting Started

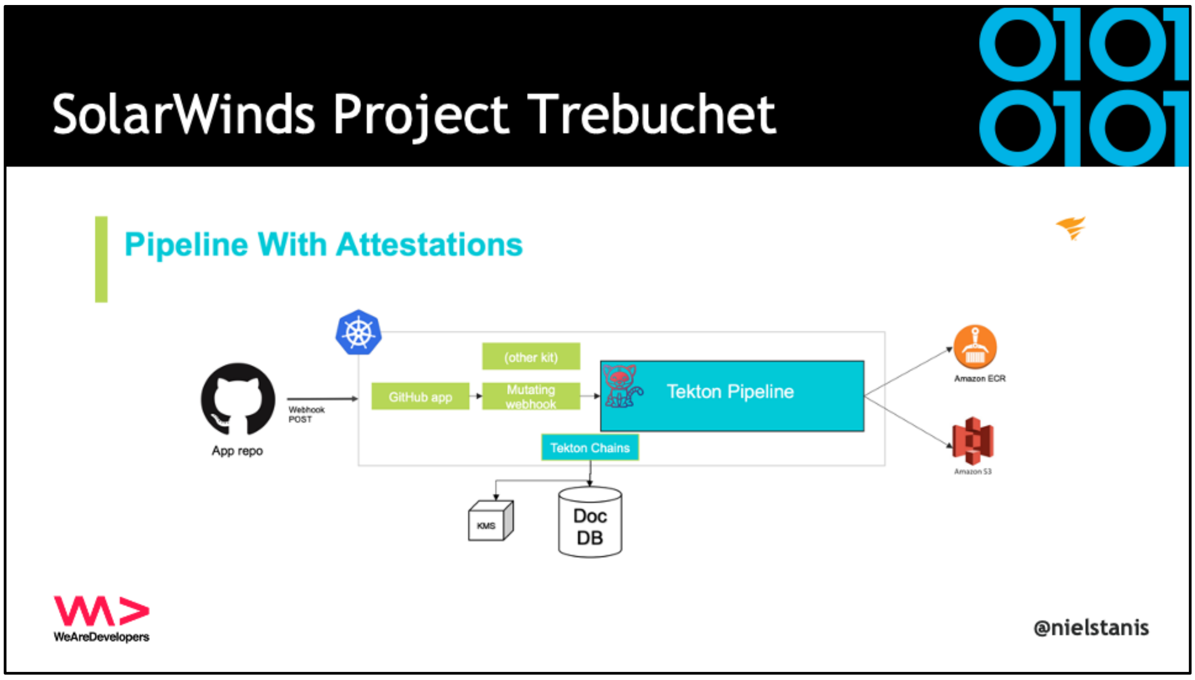
Tekton Chains is a Kubernetes Custom Resource Definition (CRD) controller that allows you to manage your supply chain security in Tekton.

In its default mode of operation, Chains works by observing all `TaskRuns` executions in your cluster. When `TaskRuns` complete, Chains takes a snapshot of them. Chains then converts this snapshot to one or more standard payload formats, signs them and stores them somewhere.

WeAreDevelopers

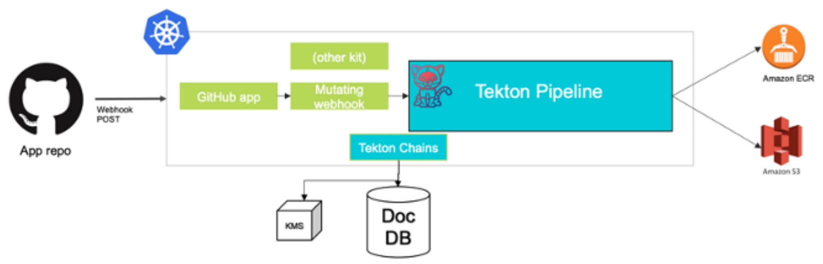
@nielstanis

https://github.com/tektoncd/chains



SolarWinds Project Trebuchet

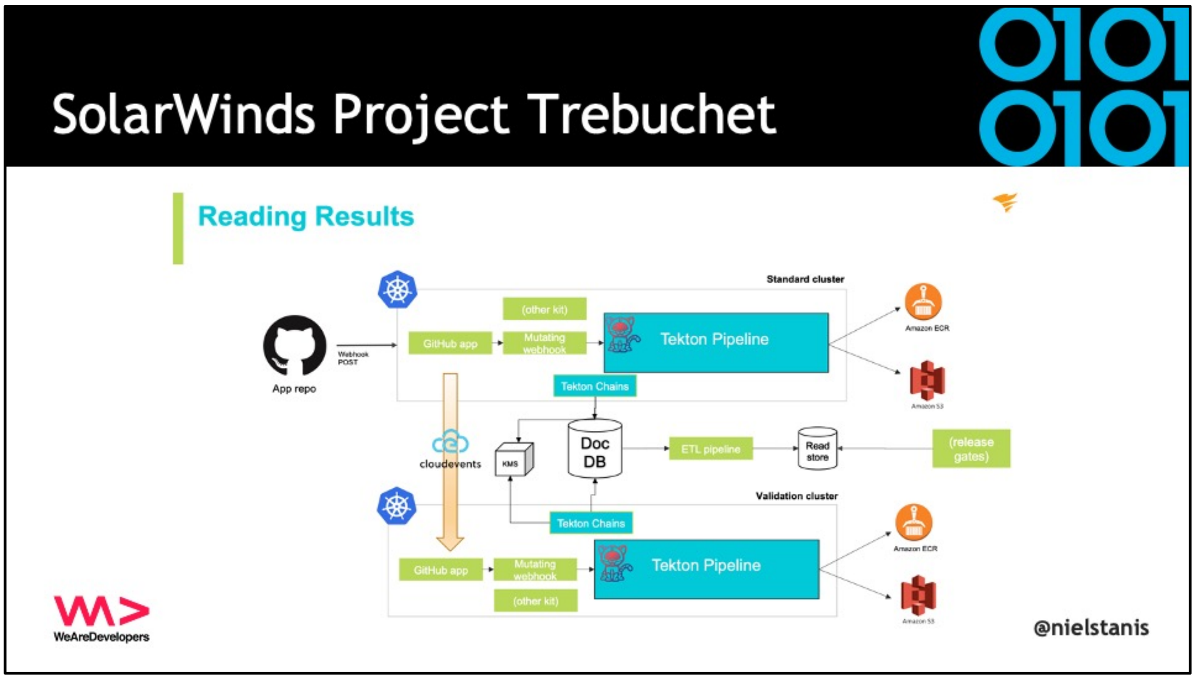
Pipeline With Attestations



WeAreDevelopers

@nielstanis

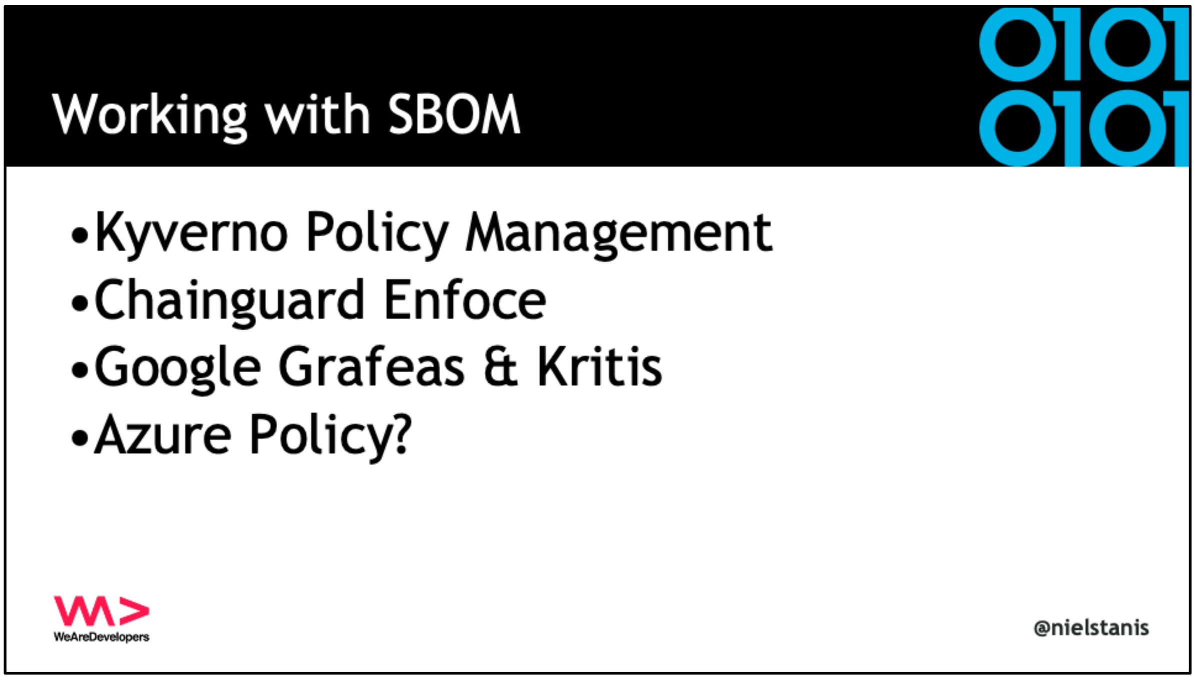
https://static.sched.com/hosted_files/supplychainsecurityconna21/df/SupplyChainCon-TrevorRosen-Keynote.pdf
https://www.youtube.com/watch?v=1-tMRxqMwTQ



SolarWinds Project Trebuchet

Reading Results

https://static.sched.com/hosted_files/supplychainsecurityconna21/df/SupplyChainCon-TrevorRosen-Keynote.pdf
https://www.youtube.com/watch?v=1-tMRxqMwTQ



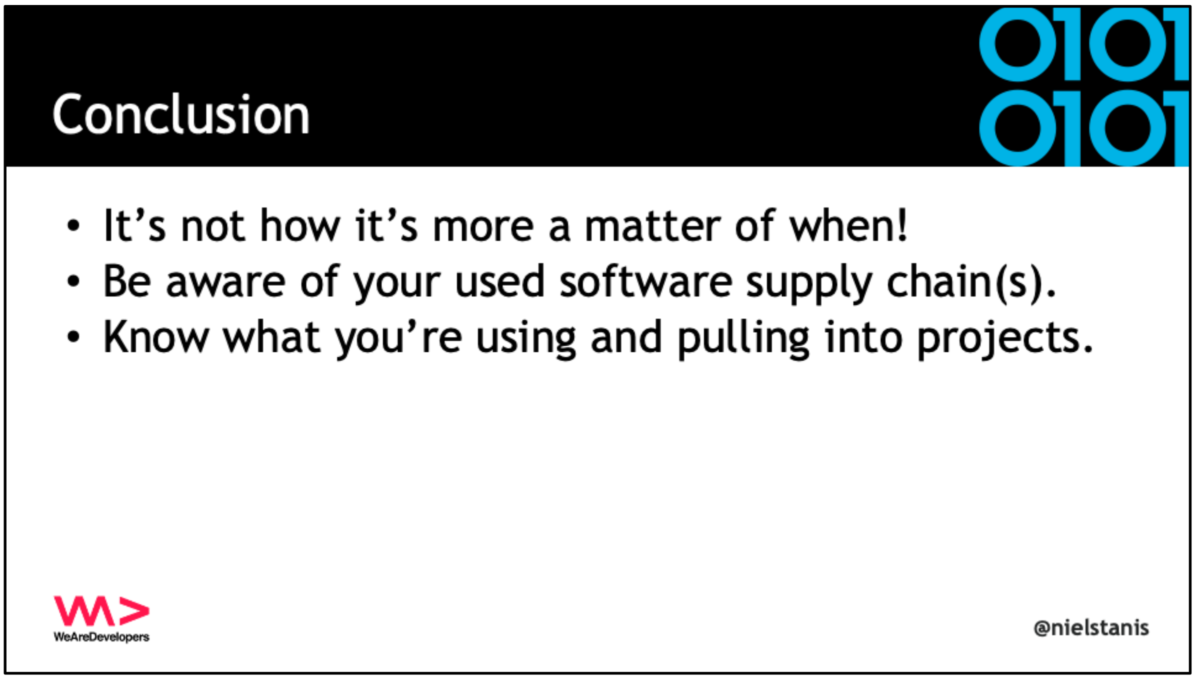
Working with SBOM

- Kyverno Policy Management
- Chainguard Enfoce
- Google Grafeas & Kritis
- Azure Policy?

WeAreDevelopers

@nielstanis

https://www.chainguard.dev/chainguard-enforce
https://thenewstack.io/chainguard-enforce-software-supply-chain-security-for-k8s/

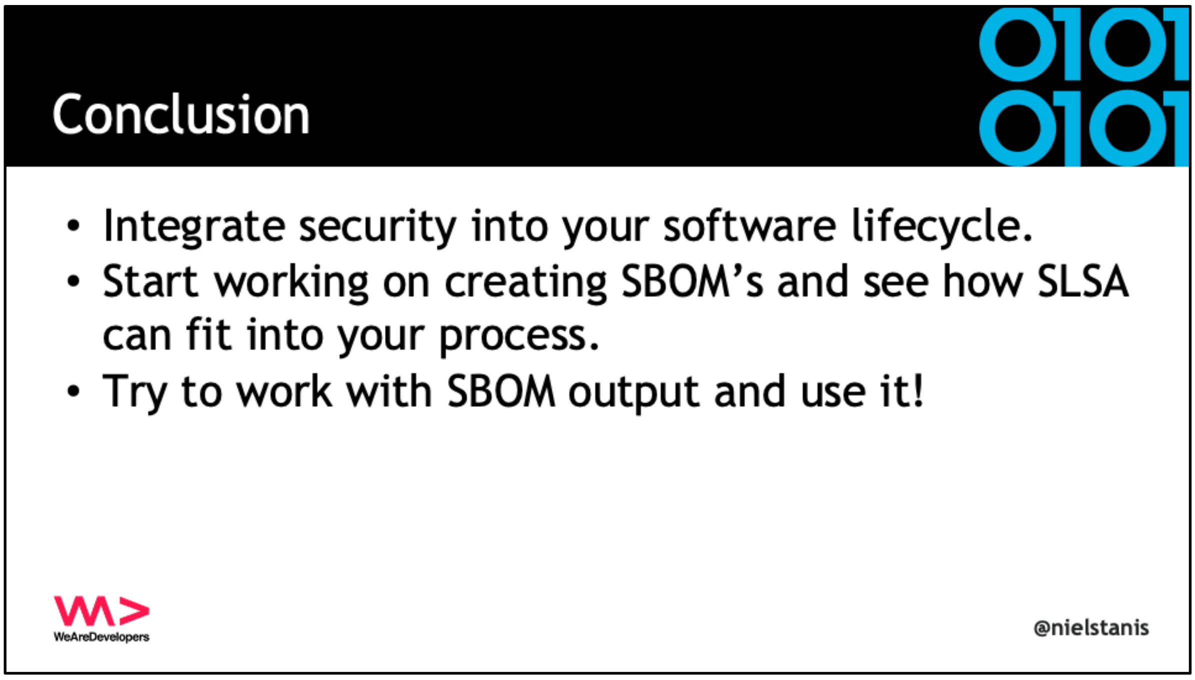


Conclusion

- It’s not how it’s more a matter of when!
- Be aware of your used software supply chain(s).
- Know what you’re using and pulling into projects.

WeAreDevelopers

@nielstanis



Conclusion

- Integrate security into your software lifecycle.
- Start working on creating SBOM's and see how SLSA can fit into your process.
- Try to work with SBOM output and use it!

WeAreDevelopers

@nielstanis

Thanks! Questions?

https://github.com/nielstanis/wearedevs2022
ntanis at veracode.com
@nielstanis on Twitter