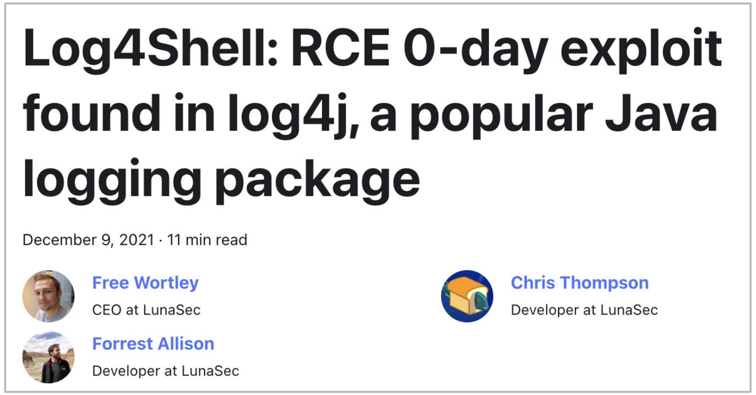
Securing your Software Supply Chain on Kubernetes with Sigstore

Priya Wadhwa
Chainguard

About me

- Eng manager @ Chainguard
- Technical Steering Committee member for Sigstore
- Previously maintained k8s developer tools like minikube, skaffold and kaniko

Chainguard sigstore minikube SKAFFOLD Kaniko



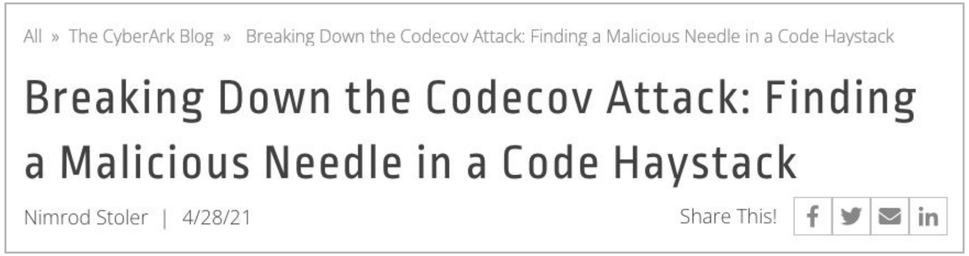
Log4Shell: RCE 0-day exploit found in log4j, a popular Java logging package

December 9, 2021 · 11 min read

Free Wortley
CEO at LunaSec

Chris Thompson
Developer at LunaSec

Forrest Allison
Developer at LunaSec

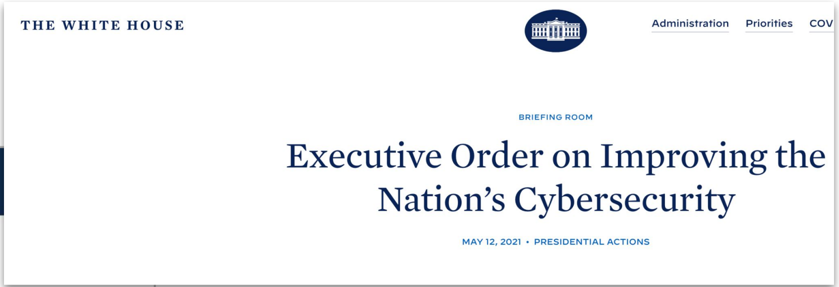


All » The CyberArk Blog » Breaking Down the Codecov Attack: Finding a Malicious Needle in a Code Haystack

Breaking Down the Codecov Attack: Finding a Malicious Needle in a Code Haystack

Nimrod Stoler | 4/28/21

Share This!



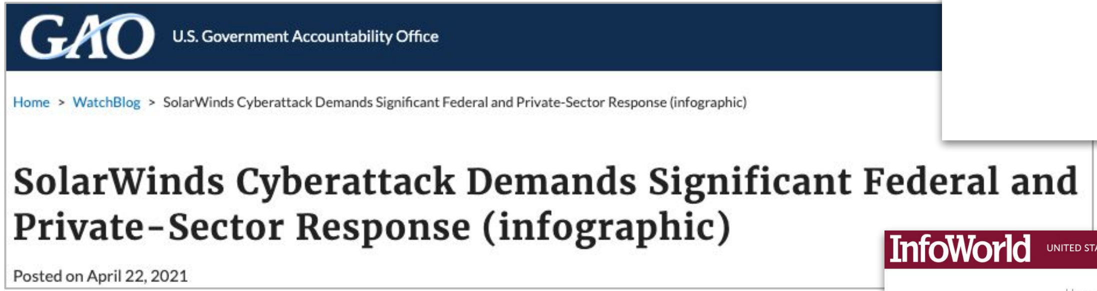
THE WHITE HOUSE

Administration Priorities COV

BRIEFING ROOM

Executive Order on Improving the Nation's Cybersecurity

MAY 12, 2021 • PRESIDENTIAL ACTIONS

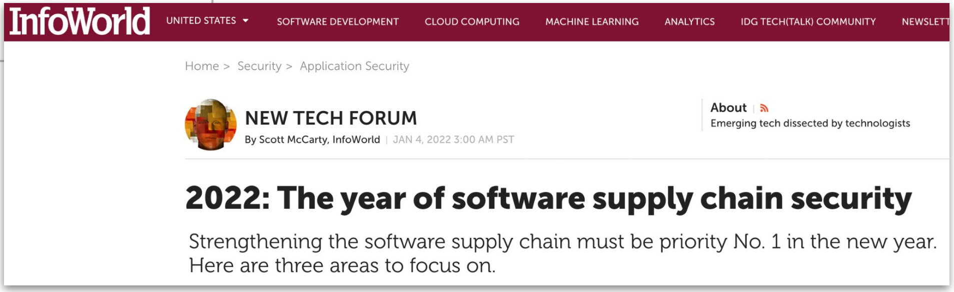


GAO U.S. Government Accountability Office

Home > WatchBlog > SolarWinds Cyberattack Demands Significant Federal and Private-Sector Response (infographic)

SolarWinds Cyberattack Demands Significant Federal and Private-Sector Response (infographic)

Posted on April 22, 2021



InfoWorld UNITED STATES SOFTWARE DEVELOPMENT CLOUD COMPUTING MACHINE LEARNING ANALYTICS IDG TECH(TALK) COMMUNITY NEWSLETT

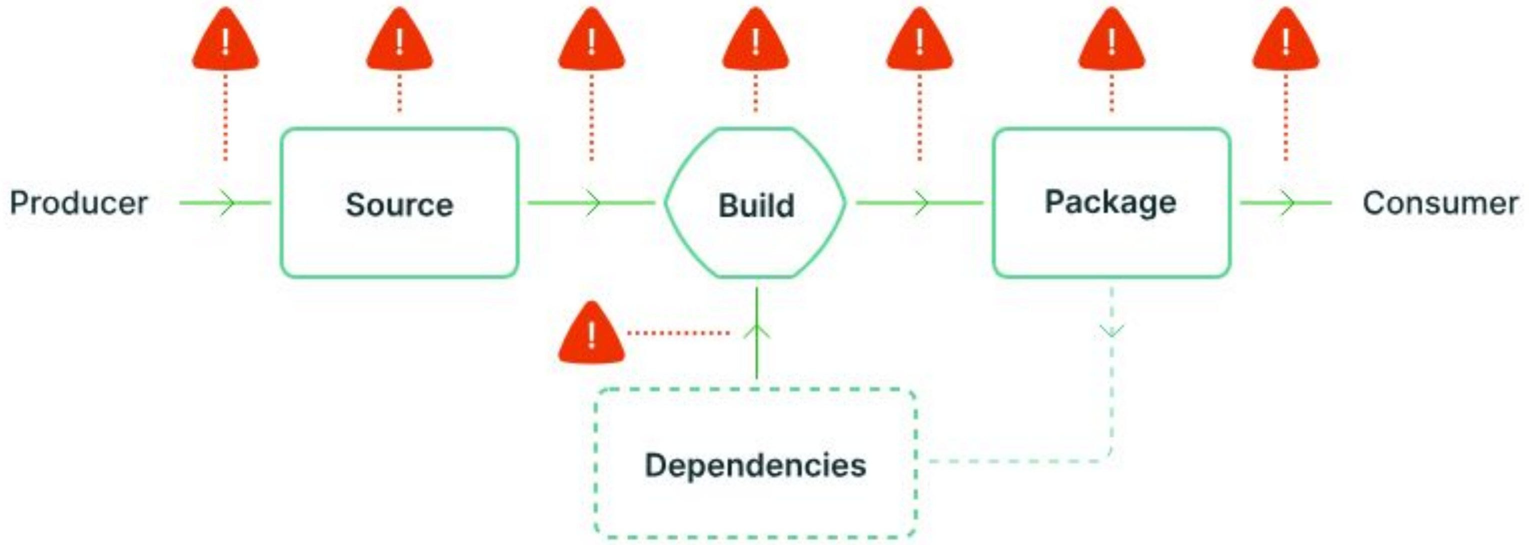
Home > Security > Application Security

NEW TECH FORUM
By Scott McCarty, InfoWorld | JAN 4, 2022 3:00 AM PST

About
Emerging tech dissected by technologists

2022: The year of software supply chain security

Strengthening the software supply chain must be priority No. 1 in the new year. Here are three areas to focus on.



https://slsa.dev/

Signing software…

… a piece of the puzzle

Who built the images in my cluster?

Where were the images running in my cluster built?

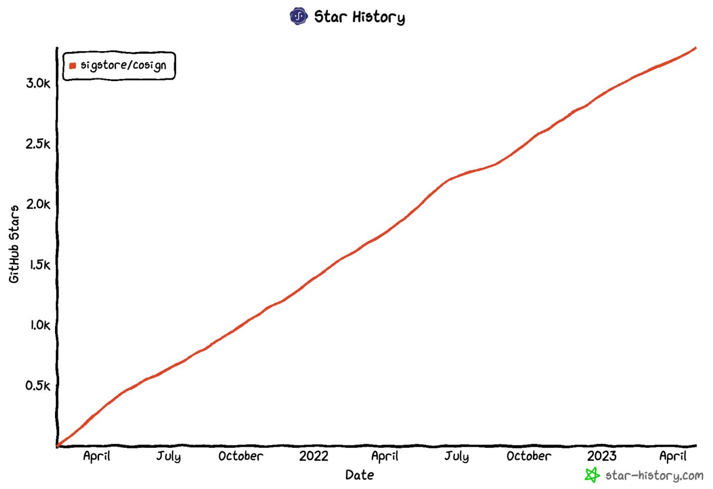
What source code went in to my images?

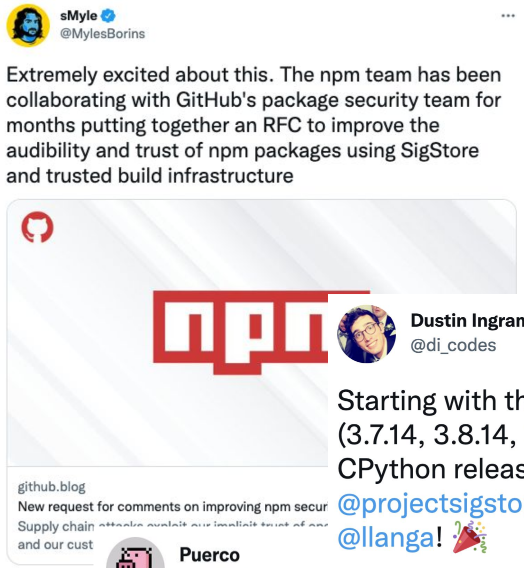
What vulnerabilities are running in my cluster?

Sigstore

- Makes signing and verifying software easy
- No need to manage long lived keys
- Transparent and auditable
- Sigstore components
  - Fulcio: Certificate Authority
  - Rekor: Transparency Log
  - Cosign: CLI tool for interacting with Sigstore

Sigstore Adoption

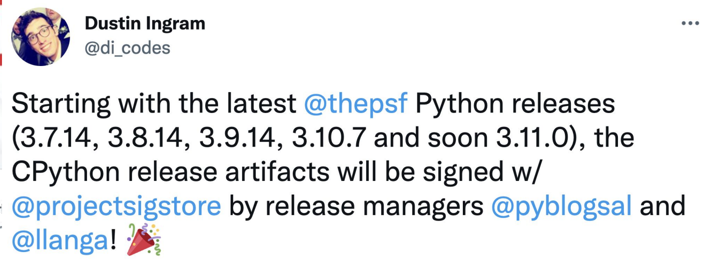




sMyle
@MylesBorins

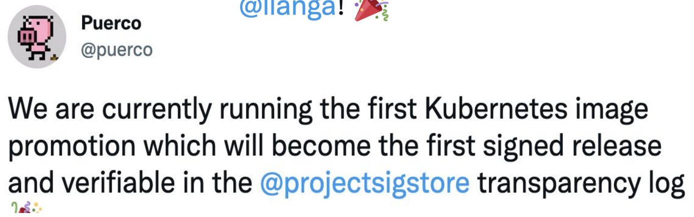
Extremely excited about this. The npm team has been collaborating with GitHub's package security team for months putting together an RFC to improve the audibility and trust of npm packages using SigStore and trusted build infrastructure

github.blog
New request for comments on improving npm secur
Supply chain attacks exploit our implicit trust of ope
and our cust



Dustin Ingram
@di_codes

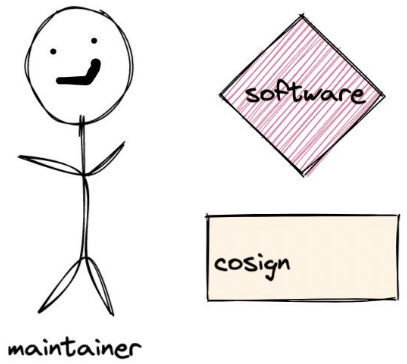
Starting with the latest @thepsf Python releases (3.7.14, 3.8.14, 3.9.14, 3.10.7 and soon 3.11.0), the CPython release artifacts will be signed w/ @projectsigstore by release managers @pyblogsal and @llanga! 🎉



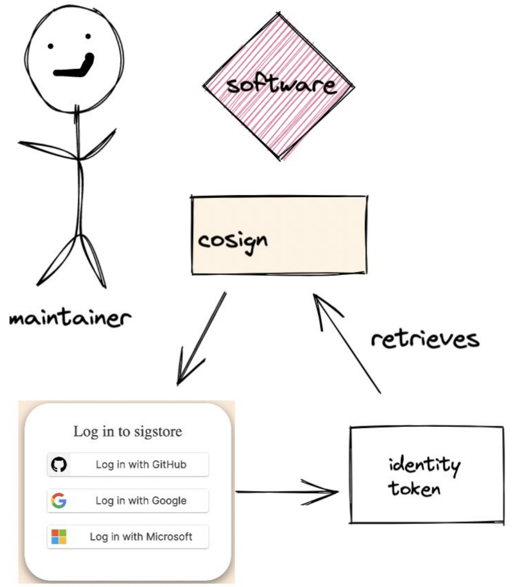
Puerco
@puerco

We are currently running the first Kubernetes image promotion which will become the first signed release and verifiable in the the @projectsigstore transparency log 🎉

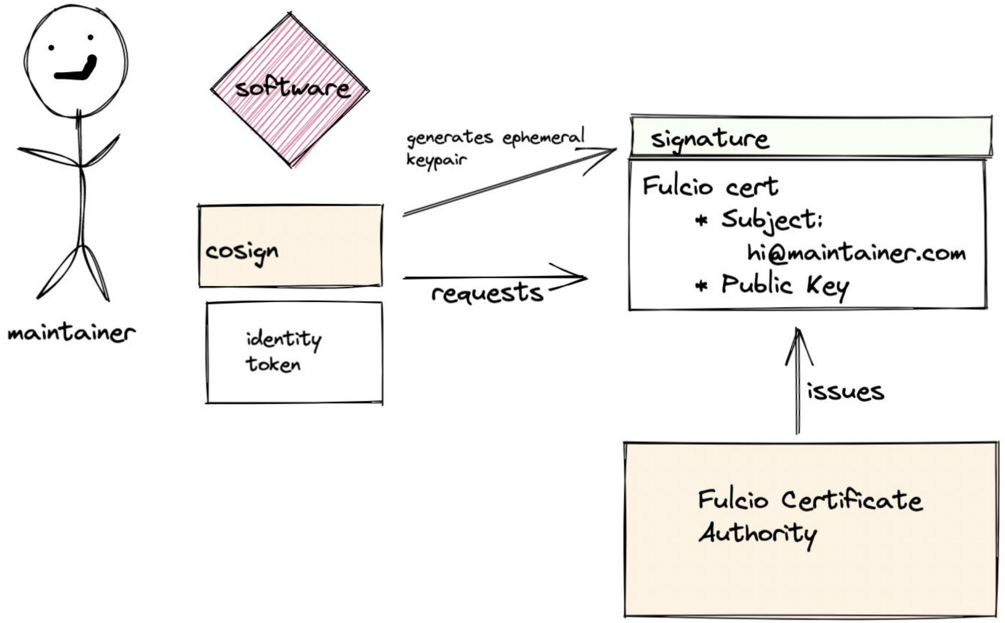
How does this work?



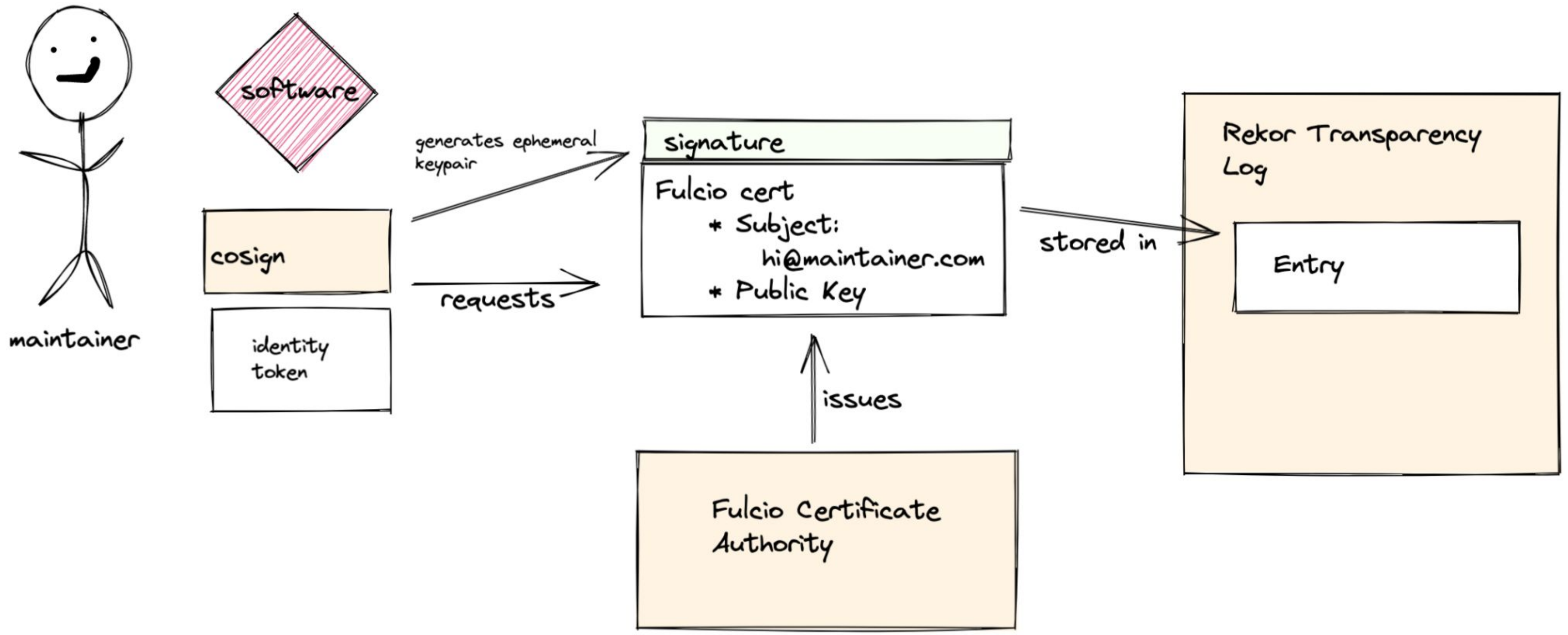
How does this work?



How does this work?



How does this work?

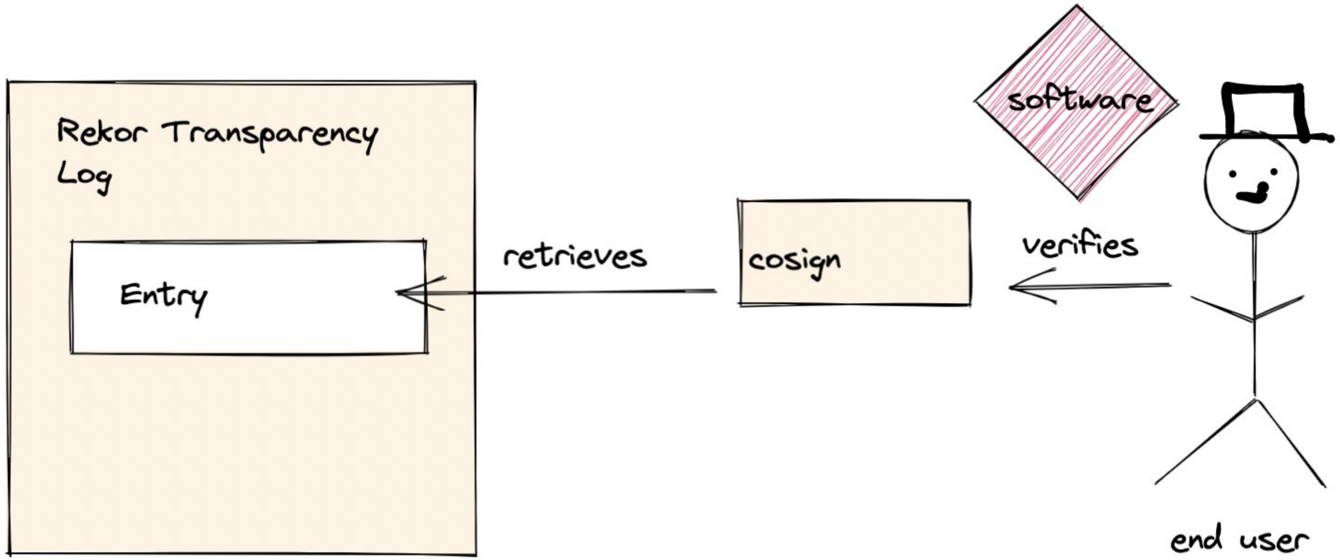


Certificate Example

```
Issuer: O=sigstore.dev, CN=sigstore-intermediate
Validity:
  Not Before: a month ago (2023-03-29T12:27:53-07:00)
  Not After: a month ago (2023-03-29T12:37:53-07:00)
Key Usage (critical):
- Digital Signature
Extended Key Usage:
- Code Signing
Subject Alternative Name (critical):
  email:
  - priya@chainguard.dev
OIDC Issuer: https://accounts.google.com
```

Signatures are great, but we have to verify them!

Verifying signatures with Sigstore



Who built the images in my cluster?

Where were the images running in my cluster built?

What source code went in to my images?

What vulnerabilities are running in my cluster?

People and systems can request Fulcio certificates

- People can sign in with an email address
- Workloads can use the SPIFFE SVID specification
- Kubernetes Service Account
- Github Actions invocations

```
$ cosign verify distroless.dev/go


...
 "Issuer": "https://token.actions.githubusercontent.com",
 "Subject": "https://github.com/distroless/go/.github/workflows/release.yaml@refs/heads/main",
  "run_attempt": "1",
  "run_id": "3041962558",
  "sha": "ea25a8792441eb38ffa80373877dd60e90a8e180",
```

Certificate Example - GitHub Actions

```
Issuer: O=sigstore.dev, CN=sigstore-intermediate
Validity:
  Not Before: 8 months ago (2022-09-07T18:28:12-04:00)
  Not After: 8 months ago (2022-09-07T18:38:12-04:00)
Key Usage (critical):
- Digital Signature
Extended Key Usage:
- Code Signing
Subject Alternative Name (critical):
  url:
  - https://github.com/priyawadhwa/test/.github/workflows/push.yaml@refs/heads/master
OIDC Issuer: https://token.actions.githubusercontent.com
GitHub Workflow Trigger: push
GitHub Workflow SHA: 000d8be59c776298c6790d9d4233e4311a2a8c7c
GitHub Workflow Name: Signed Commit
GitHub Workflow Repository: priyawadhwa/test
GitHub Workflow Ref: refs/heads/master
```

Who built the images in my cluster?

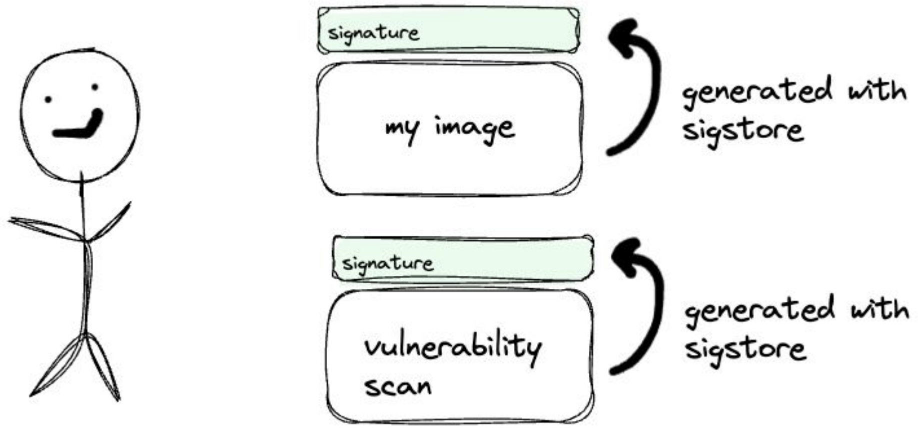
Where were the images running in my cluster built?

What source code went in to my images?

What vulnerabilities are running in my cluster?

Signing Attestations

- In addition to signing your images, Sigstore allows you to sign related artifacts like vulnerability attestations
- We can use tools like trivy or grype to generate vulnerability scans



Demo!

Public Instance

- Free for anyone to use
  - fulcio.sigstore.dev
  - rekor.sigstore.dev
  - search.sigstore.dev
- 24/7 oncall staffed by Sigstore maintainers

sigstore

public service

How can we use this to secure our Kubernetes cluster?

Sigstore Policy Controller

- Admission webhook for Kubernetes
- Supports CUE/Rego
- Install via helm: github.com/sigstore/helm-charts
- Enforce policies based on supply-chain metadata
    - “I only want to run containers from specific registries”
    - “I only want to run containers built in my CI/CD system”
    - “I don’t want any critical vulnerabilities in my cluster”
    - “I don’t want log4j running in my cluster”

https://github.com/sigstore/policy-controller

sigstore policy-controller

ClusterImagePolicy

```
apiVersion: policy.sigstore.dev/v1beta1
kind: ClusterImagePolicy
metadata:
  name: verify-k8s
images:
- glob: registry.k8s.io/**
authorities:
- keyless:
    identities:
    - issuer: https://accounts.google.com
      subject: krel-trust@k8s-releng-prod.iam.gserviceaccount.com
```

https://github.com/sigstore/policy-controller

kubernetes

Policy Catalog

- An open source catalog of common policies designed to work with Sigstore and policy-controller
- Easier than learning Cue or Rego on your own!

https://github.com/chainguard-dev/policy-catalog

ClusterImagePolicy from the Catalog!

```
apiVersion: policy.sigstore.dev/v1beta1
kind: ClusterImagePolicy
metadata:
  name: keyless-signature
  annotations:
    catalog.chainguard.dev/title: Signature policy
    catalog.chainguard.dev/description: Enforce images are signed
    catalog.chainguard.dev/labels: popular, oidc
spec:
  images:
    - glob: **
  authorities:
    - keyless:
          - issuer: https://token.actions.githubusercontent.com
            subjectRegExp: "https://github.com/priyawadhwa/kcd-zurich-demo/*"
        url: https://fulcio.sigstore.dev
      ctlog:
        url: https://rekor.sigstore.dev
```

https://github.com/sigstore/policy-controller

Demo!

What you can do next

- For your own software - start signing with Sigstore
  - Free course provided by the LF
- Try out policy-controller for verifying images in your cluster
  - Verifying k8s images is an easy start!
  - github.com/sigstore/policy-controller
  - Tutorial: https://www.chainguard.dev/unchained/policy-controller-101

https://www.chainguard.dev/unchained/policy-controller-101

Thank you!

sigstore.dev
Sigstore Slack channel

Priya Wadhwa
Twitter: priyawadhwa16@
Github: priyawadhwa@