
KubeCon
CloudNativeCon
North America 2022
BUILDING FOR THE ROAD AHEAD
DETROIT 2022
TiKV
TiKV
HELM

KubeCon | CloudNativeCon
North America 2022
BUILDING FOR THE ROAD AHEAD
DETROIT 2022

How SIG Release Cooks Trustworthy Artifacts From Raw Source Code

Carlos Panato, Adolfo García Veytia,
Jeremy Rickard, Sascha Grunert

What we will cover in this session

1. What's new in SIG Release
2. How we turn source code into artifacts
3. How can we trust those artifacts
4. Owning the infrastructure
5. What we plan for the future
6. Getting involved

What’s New?

Kubernetes 1.25

- 40 different enhancements
- Goodbye k8s.gcr.io, hello registry.k8s.io
- kube-proxy images are now distroless

Released: **Tuesday 23rd August 2022**

Kubernetes 1.26

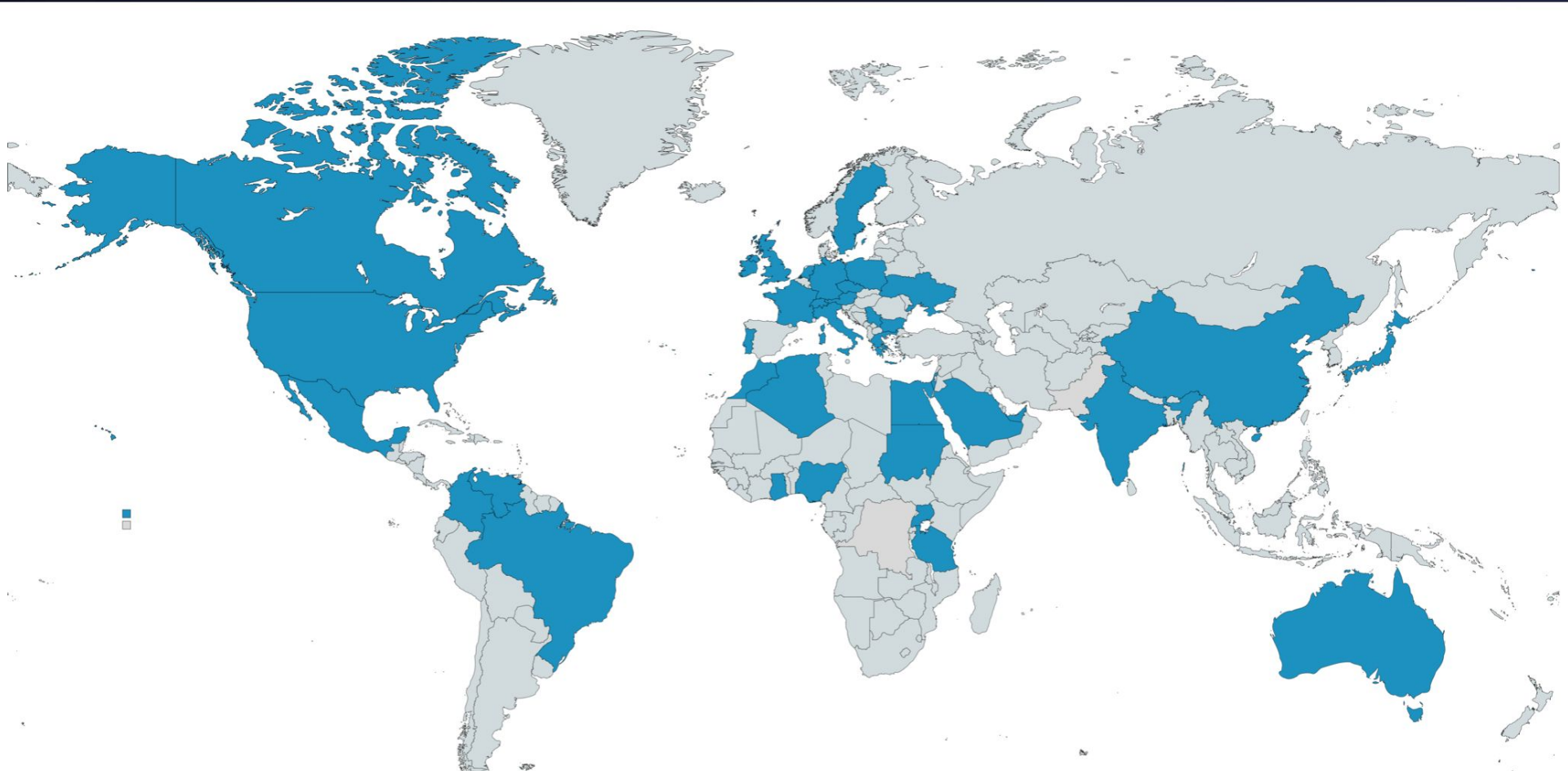
Coming Soon: **Tuesday 6th December 2022**

Release Lead: Leonard Pahlke (@leonardpahlke)

Emeritus Advisor: Nabarun Pal (@palnabarun)

The First Release Lead Entirely Outside Of North America!

The Release Teams Have Been Global



SIG Release Focus Areas 2022

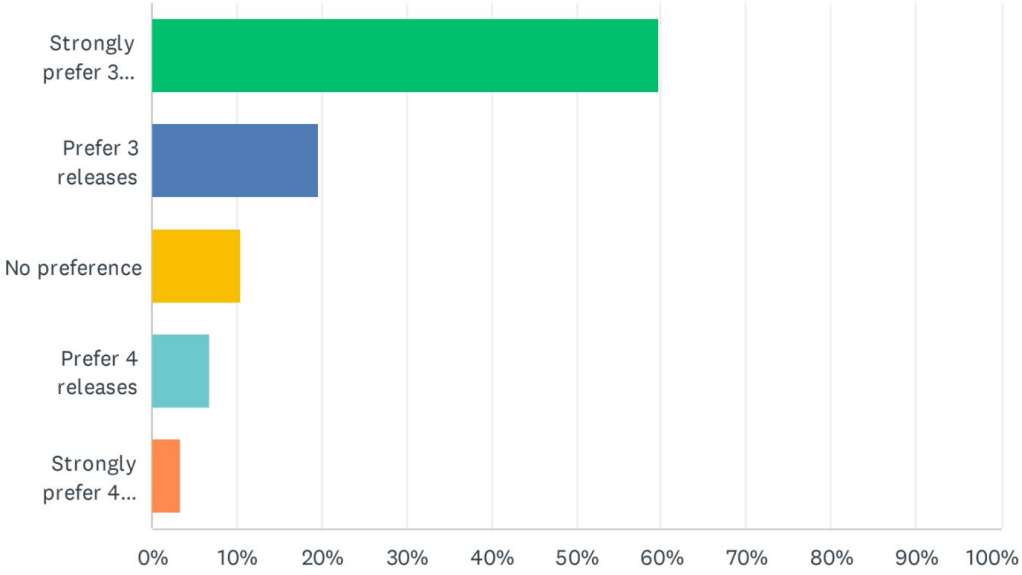
SIG Release Roadmap

1. **Consumable**: Improving the usability of artifacts by making their consumption easier
2. **Introspectable**: Making releases introspectable so consumers know how artifacts are built
3. **Secure:** Ensuring that the artifacts we produce are verified for their integrity.

Introspectable: The Release Cadence Survey

Q9 Instead of 4 minor releases a year, Kubernetes release cadence has been modified to result in 3 minor releases a year. Which release cycle do you prefer between the two? Record your answer below.

Answered: 87 Skipped: 0



Consumable: Distribute Artifacts Between Vendors

kubernetes/enhancements#3055

1. New oci-proxy at registry.k8s.io
2. Release Artifacts Replicated to AWS S3
3. Generally redirect requests from AWS IP to AWS S3
4. Other requests to k8s.gcr.io

Turning Source Code Into Artifacts!

How we turn source code into artifacts

How releases are being cut

1. We need to plan for the release
    - Per cycle: git.k8s.io/sig-release/releases/release-1.26#timeline
    - For patches: k8s.io/releases/patch-releases
2. We need someone to take the responsibility for cutting the release
    - Cutting releases takes time and dedication
    - Manual steps are required for documentation and verification

How we turn source code into artifacts

Who we have onboard

- Branch Managers and their Shadows
  - Selected per release cycle
  - Have full access to the main Kubernetes repository
  - Plan the pre releases (alpha, beta, rc) together with the Release Team Leads
  - Coordinate between other release team members, for example with the CI Signal subteam

How we turn source code into artifacts

Who we have onboard

- Release Managers and Associates
  - Permanent role within the Release Engineering subproject
  - Coordinating and cutting patch releases
  - Maintaining release branches
  - Actively developing features and maintaining code
  - Working closely together with the Security Response Committee
  - Mentoring the Release Manager Associates group

How we turn source code into artifacts

Steps involved to cut a release

1. Create a release cut issue on GitHub
    - Contains relevant information about the involved steps
    - Links to any blockers or follow-up work
    - References under which conditions the release has been cut (failing informal tests, which tooling versions we used)
    - Lists which Google Cloud Build jobs ran
2. Create a corresponding tracking thread in Slack #release-management

How we turn source code into artifacts

Steps involved to cut a release

3. Contact the Google Build Admins for their availability
    - Help to build, push and sign packages for apt.k8s.io / yum.k8s.io
4. Stage the release by using the Kubernetes Release Toolbox (krel):

    ```
    > krel stage [--type alpha] [--branch master]
    ```

    Krel runs per default in mock (non production) mode

How we turn source code into artifacts

What does *krel stage* do?

- Runs a new predefined Google Cloud Build job
- Automatically checks the prerequisites for the release
- Determines the release versions and tags the repository
- Builds the release
- Generates the changelog and Bill of Materials (SBOM)
- Verifies the built artifacts
- Builds the provenance attestations
- Stages the artifact into a Google Cloud Bucket

How we turn source code into artifacts

Steps involved to cut a release

5. Release the new version by using krel:

```
> krel release --build-version=v1.26.0-alpha.0.687+f0823c0f59d6ea
```

How we turn source code into artifacts

What does *krel release* do?

- Verifies the artifact provenance
- Pushing the artifacts into their final destination
- Pushing the staged git objects (the tag, changelogs)
- Creates the release announcement
- Updates the GitHub release page
- Builds the provenance attestations
- Archives the release

How we turn source code into artifacts

Steps involved to cut a release

6. If everything goes well, we now run *krel stage* in production mode
7. Do the container image promotion by using kpromo:

   ```
   > kpromo pr --fork $GH_USER --interactive --tag v1.26.0-alpha.1
   ```

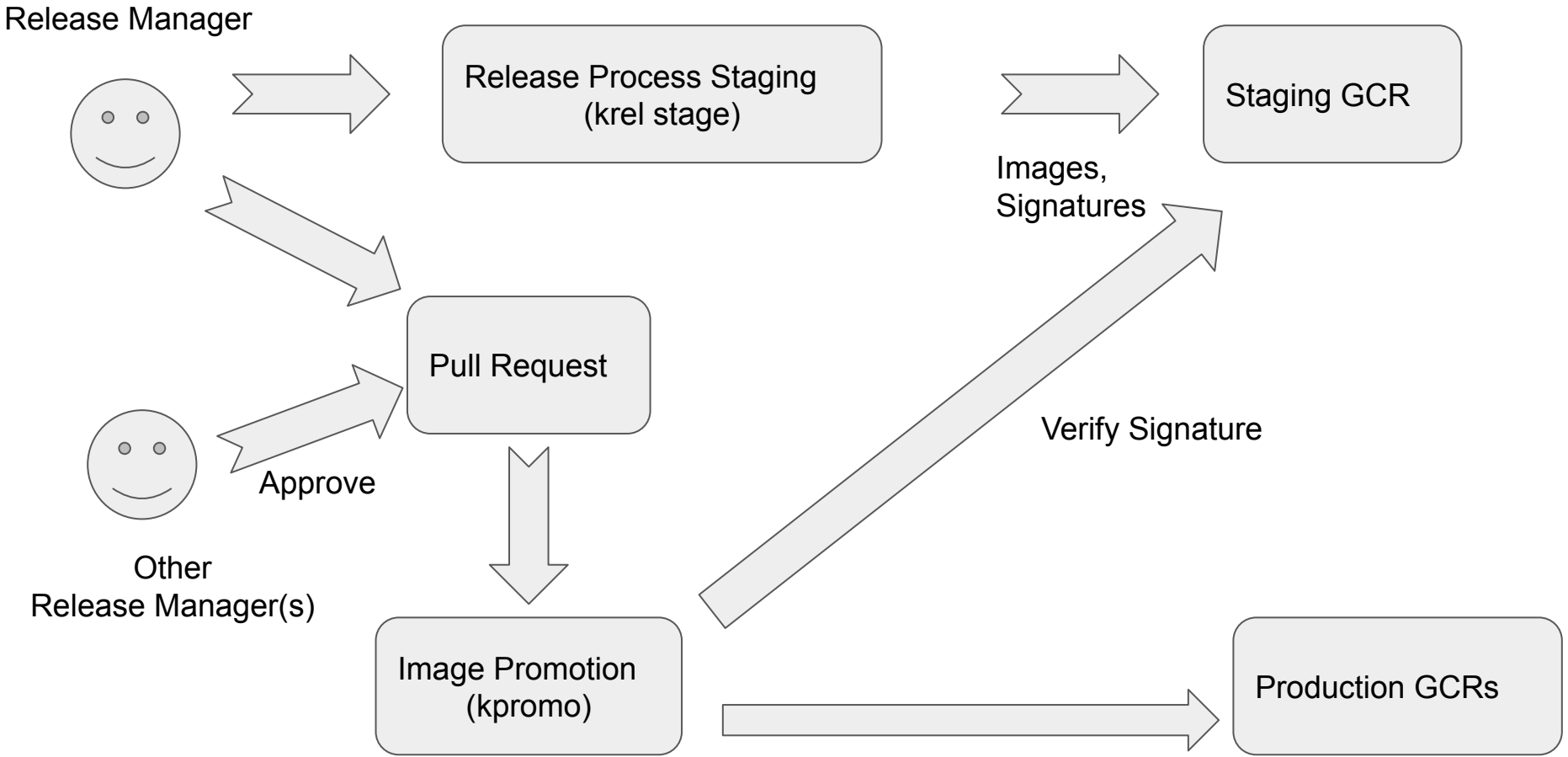
8. If the promotion is done, run *krel release* in production mode
9. Contact the Google Build Admin to cut the deb/rpm packages for us
10. Notify the Slack channel that the release is done
11. Announce the release on the mailing lists by using *krel announce*

How we turn source code into artifacts

What's inside a new release?

- Binary artifacts for Kubernetes server, client and test binaries for every supported platform
- Sigstore signed Container images on k8s.gcr.io for each Kubernetes component on supported platforms
- SLSA provenance metadata for stage and release steps
- An updated Kubernetes repository as well as the staged sources
- Software Bill of Materials (SBOMs) for all of them

Can We Trust Those Artifacts?


Release Manager
Release Process Staging
(krel stage)
Staging GCR
Images,
Signatures
Pull Request
Approve
Other
Release Manager(s)
Verify Signature
Image Promotion
(kpromo)
Production GCRs

How can we trust those artifacts

Verifying container image signatures

```
> cosign verify k8s.gcr.io/kube-apiserver-amd64:v1.26.0-alpha.1
…
Verification for k8s.gcr.io/kube-apiserver-amd64:v1.26.0-alpha.1 --

The following checks were performed on each of these signatures:

  - The cosign claims were validated

  - Existence of the claims in the transparency log was verified offline

  - Any certificates were verified against the Fulcio roots.


[{"critical":{"…"}}]
```

How can we trust those artifacts

Looking at the content of the artifacts

- We produce a Software Package Data Exchange (SPDX) Software Bill of Materials (SBOM)
- Published With Each Release:
  `https://sbom.k8s.io/v1.26.0-alpha.1/release`
- Generated with https://github.com/kubernetes-sigs/bom

How can we trust those artifacts

KubeCon CloudNativeCon North America 2022

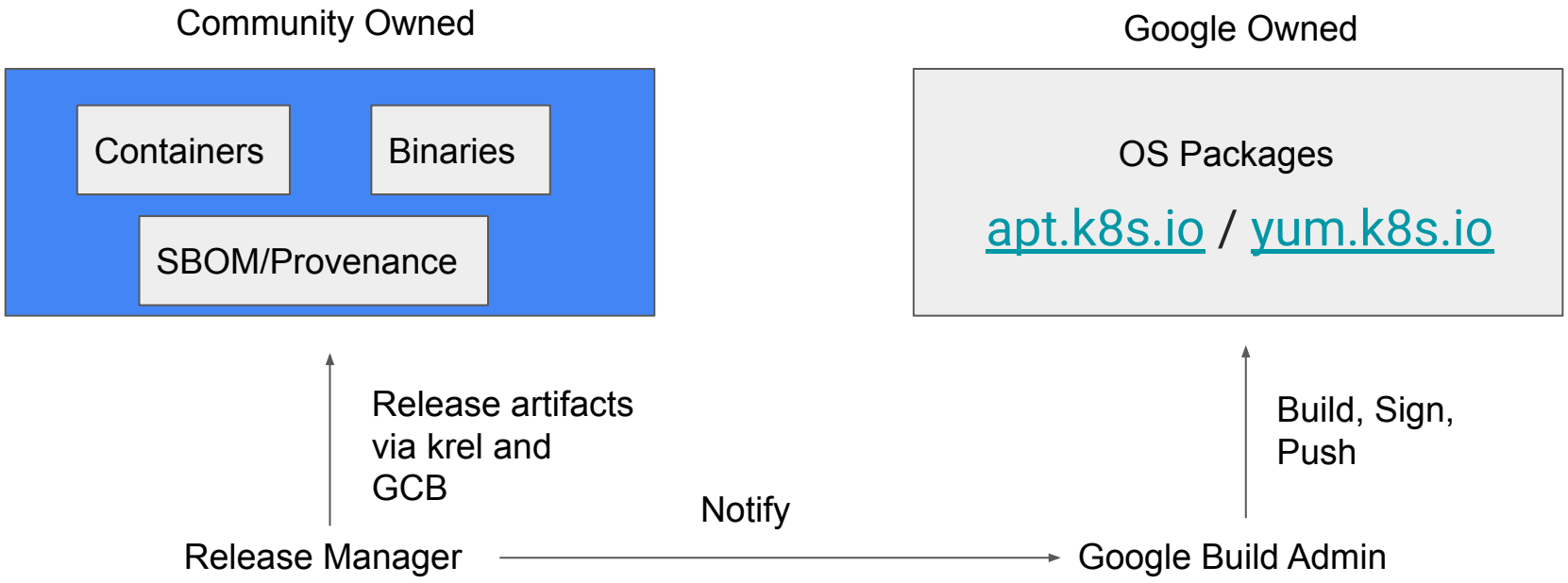
Verifying SBOMs

```
> bom document outline https://sbom.k8s.io/v1.26.0-alpha.1/release
…
 ___ _ __   ___| |_ __
/ __| '_ \ / _ \ \/ /
\__ \ |_) | (_| |>  <
|___/ .__/ \__,_/_/\_\
    |_|

  📂 SPDX Document Kubernetes Release v1.26.0-alpha.1
  │
  │  📦 DESCRIBES 25 Packages
  │
  ├ registry.k8s.io/kube-controller-manager-arm:v1.26.0-alpha.1
  │  │ 🔗 4 Relationships
  │  ├ CONTAINS PACKAGE 34aa560c5ee07db2b796103caa5ffe1400880fcb630ad5c0fed13023fde4722c/layer.tar
  │  ├ CONTAINS PACKAGE 43cb326dea3f2dc92a622b91e271cad33e6d0ec952bc949920bdaf298ba4eace/layer.tar
  │  ├ CONTAINS PACKAGE 38961665f9ab8d0f3095ab1222864295edbe505f5ff5eadd261a1c325e8420c7/layer.tar
  │  └ GENERATED_FROM DocumentRef-kubernetes-v1.26.0-alpha.1 (external)
  │
  ├ registry.k8s.io/kube-apiserver-arm:v1.26.0-alpha.1
  │  │ 🔗 4 Relationships
  │  ├ CONTAINS PACKAGE 34aa560c5ee07db2b796103caa5ffe1400880fcb630ad5c0fed13023fde4722c/layer.tar
  │  ├ CONTAINS PACKAGE 43cb326dea3f2dc92a622b91e271cad33e6d0ec952bc949920bdaf298ba4eace/layer.tar
  │  ├ CONTAINS PACKAGE b8cdb1024938ff720df83dd13cdc366927c5fce156f137b2489fed97fde25579/layer.tar
  │  └ GENERATED_FROM DocumentRef-kubernetes-v1.26.0-alpha.1 (external)
```

Owning The Infrastructure

Who owns what parts of the process?



Owning The Rest Of The Infrastructure

What's left?

- Migrate away from Google built packages (and keys)
- Currently investigating Open Build Service
- Join us at #release-packages-poc (notes)

What we plan for the future

The Future: Consumable

What's Coming Up?

- Reimagine Image Promotion Process (from KEP 3055)
  - Tighter colab with SIG K8s Infra
  - Rewrite large kpromo chunks
- System Package Migration
- General Release Guidelines for Tiered Repositories
- GitHub Actions Repo! new new new

The Future: Introspectable

What's Coming Up?

- Provenance Metadata
  - Tejolote SLSA Attester is out
  - Attesting Image Promotion
  - .. and now done properly :)

The Future: Secure

What's Coming Up?

- File Signing for
  a. Binaries
  b. SBOMs
  c. Provenance
- Push to the new SLSA 3 Levels
- Artifact Verification in `publish-release`
- Multi Language Support for bom 🎉
- *The SIG Release Guide to Secure Releases*

SPDX

SLSA

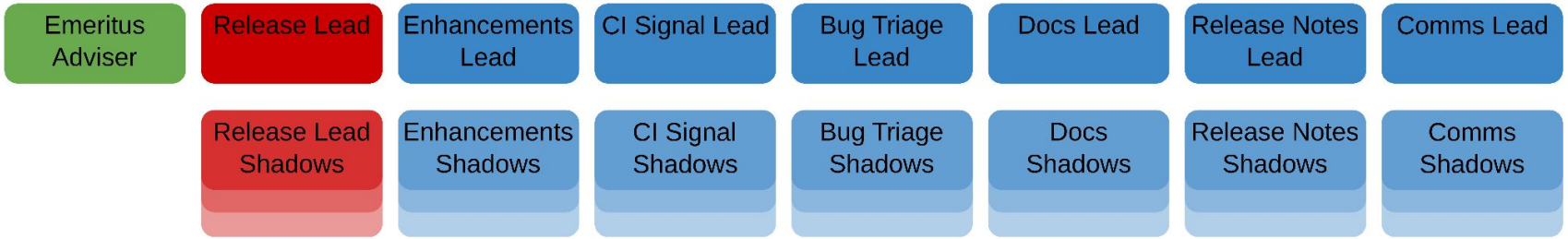
sigstore

Getting involved

Getting involved

Apply as a Release Team Shadow for the upcoming 1.27 cycle

- Watch for the `[ANNOUNCEMENT]` on dev@kubernetes.io
- Learn more about the roles: git.k8s.io/sig-release/release-team/README.md
- Choose the area of your particular interest:



Getting involved

Interested in the more technical aspects of a release?

Show interest in becoming a Release Manager Associate, by:

- Joining the Release Team as Shadow and later lead a technical role
- Starting to contribute consistently to our repositories
- Being active in our technical discussions in #release-management or #sig-release
- Identifying working areas and helping with our recurring work
- Connect to existing Release Managers to find areas to contribute

Getting involved

Reach out to us, we're happy to help!

- The Release Team Shadowing Program provides one of the best onboarding experiences in the community
- SIG Release is always welcoming to discuss process or technical enhancements into any direction
- Being most inclusive for everyone is our goal to lower the barrier for new and existing community members

BUILDING FOR THE ROAD AHEAD

DETROIT 2022

Thank you!