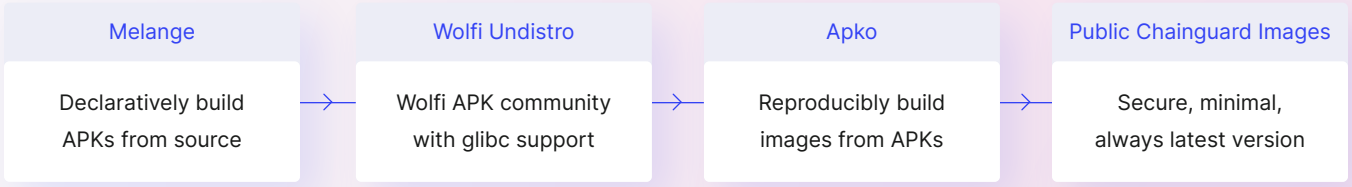
Chainguard Images from Source to Prod

Chainguard is setting the standard for lightweight, hardened base images that deliver faster builds and deploys using less resources and a reduced attack surface aiming for 0-known vulnerabilities. This document outlines our internal security measures for how Chainguard Images are built and distributed to our customers.



As a result of this architecture the production of images involves development and release across a number of git repositories and delivery pipelines.

Development practices

All of the projects that feed into Chainguard Images enforce the following development practices:

Source

- **Version Controlled** - Project source is version controlled using Git and served by Github
- **Restricted Approvers** - Our projects have identified a restricted set of trusted parties as approvers
- **2 Person Review** - All source changes to a project are approved by at least 2 trusted parties
- **Authenticated** - For chainguard-dev repositories the authenticity of actors are enforced by hardware key based two factor authentication.
- **Commit Signed** - For chainguard-dev repositories we enforce commit signing

Build

- **Build as Code** - Our builds are fully described within the source control of the repository being built
- **Service based** - Build artifacts are produced within a restricted and controlled build service. Most of our builds are GHA based, but our APK packages are in-part built using our own customer build service
- **Ephemeral** - Our build environments are not reused between builds
- **Parameterless** - The change author is unable to parameterize the configuration of the build

Test

- **Extensive unit and e2e tests**
- **Merge Status Checks** - Code changes are blocked on completion of all status checks

Repositories

Public

- Apko - declarative OCI container image builder
- Melange - declarative APK package builder
- Wolfi - public Chainguard APK package manager project
- Chainguard Images - public Chainguard suite of images

Private

- **Chainguard Enterprise Packages** - source and build pipeline for producing our 'Chainguard' paid APK packages.
- **Chainguard Private Images** - source and build pipeline for customer specific images

Packages

Every Chainguard Image is an assembly of APKs composed together to produce a functioning Linux filesystem. Wolfi is Chainguard's public open-source repository of these APK packages. We also host a private repo and registry for packages only available through paid support agreements.

In addition to previously covered development practices, Chainguard and Wolfi packages are:

- CI tested to detect common packaging errors
- Automatically monitored and updated for upstream releases
- Checked for known CVE
- Verified to not break ABI compatibility guarantees
- SBOMs are generated at build time and packaged with the APKs
- Signed using Wolfi and Chainguard specific private RSA 4096 bit signing keys
- Deployed to packages.wolfi.dev automatically from our build pipeline

Images

The production of new Chainguard Images is fully automated and managed through declarative configuration. New images are produced according to our documented containerization best practices.
In addition to our standard development approach, our images development process includes the following:

- Images are tested functionally and UX is evaluated against benchmark image references
- Our build pipeline:
  - Rebuilds and tests all images in the catalog on a nightly basis
  - APKO image builds verify all package signatures for authenticity
  - Produces a signed SBOM attestation of the image contents
  - Scans images for known CVEs and results are published in a vulnerability attestation
  - Sigstore signs produced images with the build pipeline's OIDC identity
- Securely publishes images to cgr.dev