
KubeCon
CloudNativeCon
Europe 2023
TiKV

containerd: Project Update and Deep Dive

Maksym Pavlenko, Apple
Samuel Karp, Google
containerd maintainers

containerd's support lifecycle

Three types of releases:

- **Active**
  - At least one year of support
  - Bug fixes and security fixes
- **Extended**
  - After the Active window ends
  - Security fixes only
  - No set window length
- **Long Term Stable**
  - At least three years of support
  - Bug fixes, security fixes, and dependency version updates
  - Should remain usable by current containerd clients

containerd 1.6 - first LTS!

- Supported until February 2025 (**3 years** from release)
  - Longer support window for bug fixes and security patches
  - Expanded scope for backports
    - (and compatibility with current Kubernetes versions)
- Converts to a regular stable release up to 6 months before (August 2024)
- Kubernetes versions
  - Existing versions: 1.24 – 1.27
  - Future versions: 1.28 (2023), 1.29 (2023), 1.30 (2024)

containerd 1.7 - just released!

- New! Sandbox API *(experimental)*
  - Shim-level API to support groups of containers
  - Try it with CRI using `ENABLE_CRI_SANDBOXES=1` environment variable
- New! Transfer Service *(experimental)*
  - Support new workflows with images
- Supported until March 2024 (**1 year** from release)
  - Or 6 months after 2.0 is released
  - This is *before* the EOL of 1.6
- Kubernetes versions
  - Existing versions: 1.24 – 1.27
  - Future versions: 1.28 (2023), 1.29 (2023)
- **Last 1.x release of containerd**

containerd 2.0

- Production-ready Sandbox API (sbserver)
  - Modular sandboxed CRI plugin
  - Legacy CRI server to be removed
- Production-ready Transfer Service
  - Cover more use cases
  - Sandbox API integration
- Container runtime interface (CRI) updates
- Node resource interface (NRI) updates
- Removing deprecated features

Sandbox API

Sandbox API == New API for container groups:

- Controller interface to handle sandbox lifecycle
  - pod-sandbox (extract from CRI)
  - microVM
  - VM
- Shims provide Controller implementation
- CRI invokes Controller

Sandbox API

Ongoing CRI integration:

- CRI server fork to enable integration (`sbserver` directory)
  - Calls sandbox `Controller` interface instead of podsandbox
  - Adding `RemoteController` to call shims
- Default implementation in v2.0
- Try it out with **`ENABLE_CRI_SANDBOXES`** environment variable in v1.7

Transfer service


```
type Transferer interface {
        Transfer(ctx context.Context, source interface{}, destination interface{}, opts ...Opt) error
}
```

Source	Destination	Description	Local Implementation Version
Registry	Image Store	"pull"	1.7
Image Store	Registry	"push"	1.7
Object stream (Archive)	Image Store	"import"	1.7
Image Store	Object stream (Archive)	"export"	1.7 (in progress)
Object stream (Layer)	Mount/Snapshot	"unpack"	Not implemented
Mount/Snapshot	Object stream (Layer)	"diff"	Not implemented
Image Store	Image Store	"tag"	Not implemented
Registry	Registry	mirror registry image	Not implemented

Transfer service

- New use cases and extension points
  - Signing and image validation
  - Credential management
  - Custom pull logic
  - Image decryption
  - Pluggable sources / destinations

- Sandbox API integration in future
  - Confidential computing
  - Custom image handling (skip snapshotter)

Redirect to registry.k8s.io

registry.k8s.io is GA! 🎉

🚨 ❄️ k8s.gcr.io is frozen ❄️ 🚨

More info on
https://k8s.io/image-registry-redirect

Configuring containerd

containerd config file (/etc/containerd/config.toml)

```
version = 2
required_plugins = ["io.containerd.grpc.v1.cri"]

[plugins."io.containerd.grpc.v1.cri"]
sandbox_image = "registry.k8s.io/pause:3.9"
```

Use `registry.k8s.io`, not `k8s.gcr.io` now!

NRI updates

- Middleware between CRI and OCI
- Reworked in 1.7
- New API for tracking state changes of containers, pod-sandboxes, and other new sandbox types like micro VMs
- Sandbox API integration in 2.0

Deprecations in 2.0

Component	Deprecation release	Target release for removal	Recommendation
Runtime V1 API and implementation (`io.containerd.runtime.v1.linux`)	containerd v1.4	containerd v2.0 ✅	Use `io.containerd.runc.v2`
Runc V1 implementation of Runtime V2 (`io.containerd.runc.v1`)	containerd v1.4	containerd v2.0 ✅	Use `io.containerd.runc.v2`
config.toml `version = 1`	containerd v1.5	containerd v2.0 ✅	Use config.toml `version = 2`
Built-in `aufs` snapshotter	containerd v1.5	containerd v2.0 ✅	Use `overlayfs` snapshotter
Container label `containerd.io/restart.logpath`	containerd v1.5	containerd v2.0 ✅	Use `containerd.io/restart.loguri` label
`cri-containerd-*.tar.gz` release bundles	containerd v1.6	containerd v2.0	Use `containerd-*.tar.gz` bundles
Pulling Schema 1 images (`application/vnd.docker.distribution.manifest.v1+json`)	containerd v1.7	containerd v2.0	Use Schema 2 or OCI images
CRI `v1alpha2`	containerd v1.7	containerd v2.0 ✅	Use CRI `v1`

containerd's expanded ecosystem

- Built to be extensible
- Lots of places to plug in new functionality!
  - Snapshotters
  - Runtimes
  - Clients
- Plugins/projects that are part of the containerd organization
- Community projects
- Vendor products
- Lots of adopters!

Kubernetes distros adopting containerd

- Amazon Elastic Kubernetes Service
- Azure Kubernetes Service
- Google Kubernetes Engine
- IBM Cloud Kubernetes Service
- Rancher K3s
- VMware Tanzu

Kubelet command-line flag

```
--container-runtime-endpoint=unix:///run/containerd/containerd.sock
```

containerd clients

- **ctr** - command-line development tool
  - typically bundled with containerd
  - core containerd project
- **crictl** - a CLI for CRI
  - Kubernetes project (part of cri-tools)
- **nerdctl** - a Docker-like CLI
  - expanded functionality
    - Lazy-loading images, image encryption, image signing
  - non-core containerd project
- **Colima** - Docker-like experience on MacOS
  - Built in nerdctl and LIMA
  - community project
- **Rancher Desktop** - Docker-like experience on MacOS, Windows, and Linux
  - Built on nerdctl + LIMA
  - Includes a GUI
  - vendor product
- **Finch** - Docker-like CLI on MacOS
  - Built on nerdctl + LIMA + plugins
  - vendor product

Snapshotters

- **Built-in**
  - overlay (Linux)
  - btrfs (Linux)
  - devmapper (Linux)
  - native (Linux, Windows, FreeBSD)
  - lcow (Windows)
  - windows (Windows)
  - zfs (Linux, FreeBSD)
- Extension via **proxy plugins**
- **Remote** (lazy-loading)
  - eStargz (non-core project)
  - Nydus (non-core project)
  - overlayBD (non-core project)
  - SOCI (OSS vendor project)
  - GKE image streaming (vendor product)

Runtimes and shims

- **runc - standard** OCI runtime for Linux containers
- **crun** - alternative OCI runtime for Linux containers
- **runwasi** - OCI runtime for **WASM**
- hcsshim/**runhcs** - containerd shim and OCI runtime for **Windows** containers
- **runj** - experimental OCI runtime for **FreeBSD** jails
- **Kata Containers** - hypervisor-based isolation for pods
- **gVisor/runsc** - independent kernel for isolation
- **firecracker-containerd** - hypervisor-based isolation for containers based on Firecracker

Getting involved

- `#containerd` and `#containerd-dev` channel on CNCF Slack (https://slack.cncf.io)
- Community Meeting on the second Thursday each month
  - See CNCF Calendar for your timezone
  - https://cncf.io/calendar
- Build something in the ecosystem!
- Discussion, issues and pull requests welcome!

  https://github.com/containerd/containerd

Session Q+A

- Virtual attendees may submit questions to speakers through the CNCF Slack channel: **#2-Kubecon-sessions**
- Please create a thread and tag the speaker(s) with questions about their talk.
- Questions will be answered by the speaker and/or other community members after the session concludes.

Please scan the QR Code above
to leave feedback on this session