Containers in HPC – Podman

CHRIM Onboarding

Stefano Ansaloni

University of Manitoba

November 1, 2024

University of Manitoba

Digital Research Alliance of Canada

About me

Stefano Ansaloni

Cloud Computing Specialist at University of Manitoba
(part of the Advanced Research Computing team)

Software Developer and DevOps Specialist since 2017

Linux User/Admin since 2005

University of Manitoba

What is a container

From Wikipedia (Containerization):

> *Containerization is operating-system–level virtualization or application-level virtualization so that softwares can run in isolated user spaces called "containers" in any cloud or non-cloud environment, regardless of type or vendor.*

University of Manitoba

Properties of containers

- Each container is a fully functional and portable computing environment surrounding the application and keeping it independent of other environments running in parallel
- Each container simulates a different software application and runs isolated processes (including configurations, libraries, and dependencies)
- Multiple containers share a common operating system kernel (operative system)

University of Manitoba

Wikipedia [Containerization]

Focusing On Podman

University of Manitoba

What is Podman

From Wikipedia (Podman):

> *Podman (pod manager) is an open source OCI-compliant container management tool from Red Hat used for handling containers, images, volumes, and pods; offering APIs for the lifecycle management of those components (the API is identical to the Docker API).*

University of Manitoba

Why not Docker

Podman aims to provide a more secure and lightweight alternative to Docker:

- Daemonless $\Rightarrow$ Don't rely on a process with root privileges to run containers
- Rootless containers $\Rightarrow$ Run containers as regular users, without interacting with a root-owned daemon
- User namespaces $\Rightarrow$ Careful use of kernel capabilities

---

Red Hat [What is Podman]
Better Stack [Exploring Podman]

University of Manitoba

Compatibility with Docker

For most use cases, Podman can be used as a "drop-in" replacement for Docker:

- Podman CLI syntax is almost the same as Docker's one
- Podman can use the same images as Docker
- Podman can use the same registries as Docker

University of Manitoba

Basic commands

Print version	`podman version`
Pull image	`podman pull <REGISTRY>/<NAME>:<TAG>`
Delete image	`podman image rm <IMG>`
List images	`podman image ls`
Create and start a container	`podman run [OPTS] <IMG> [CMD [ARGS]]`
Execute command inside a running container	`podman exec [OPTS] <CNT> [CMD [ARGS]]`
Stop a running container	`podman stop <CNT>`
Start a stopped container	`podman start <CNT>`
Delete a container	`podman rm <CNT>`
List containers	`podman ps [-a]`

University of Manitoba

Podman on HPC – Disclaimer

Podman is considered an advanced tool to be used only by experienced users when their workloads cannot be run using standard HPC programs/modules.

In any case, Podman must not be used:

- on login nodes
- to execute long-running services (e.g. databases)

University of Manitoba

Podman on HPC

Podman is available as a module on national clusters and UManitoba HPC cluster (Grex).

HPC System	**Command**	**Current version**
National Clusters	`module load StdEnv/2023 podman`	4.9.5
Grex	`module load podman`	5.2.5

University of Manitoba

Podman on Grex

When on Grex, it is important to use the local version of Podman:

- local proxy cache for registries
- better default configuration
- newer version

University of Manitoba

Questions?

University of Manitoba

Thank you

University of Manitoba