Container
Native FS
Interposer

Introduction
FUSE
CSI
Demo
Plans

Container Native FS Interposer

Julia Hua, Jiawei Xiang, Hilario Gonzalez, Juncheng Cao
Mentors: Vasily Tarasov, Alex Merenstein

September 25, 2024

Introduction

Goal

Introduce a filesystem interposer layer to k8s

- workload tracing
- workload metric collection
- faulty I/O
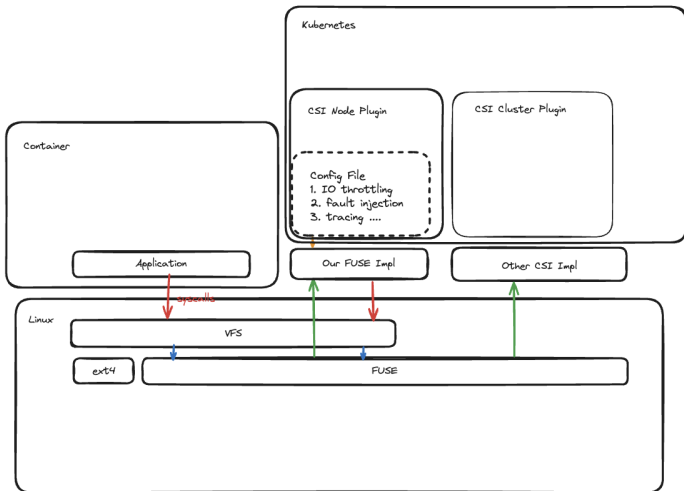- throttle I/O
- fake I/O
- . . .

Architecture

Architecture

FUSE

the mechanisms and API for implementing virtual file systems

CSI

a cloud native way to mount fuse filesystems into containers

What is FUSE

File System in Userspace

Benefits

- Develop virtual file systems in user space
- Can be implemented with any language
- Available in many linux distributions or even other OS

Drawbacks

- Poor performance
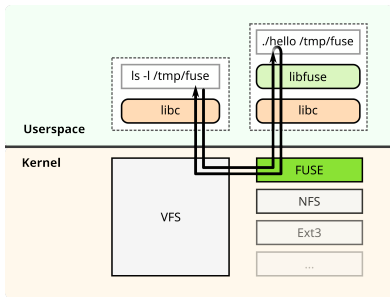- Less reliable

How FUSE works

Container
Native FS
Interposer

Introduction

FUSE

CSI

Demo

Plans

Components

- Kernel module `fuse.ko`
- Userspace library `libfuse`
- Utilities `fusermount`

Why FUSE

- It's hard to ship kernel modules with container
- It's easier and safer
- It's widely used and already have a lot of existing implementations

Container Storage Interface

- Container runtime agnostic way to manage storage
- Supported by Kubernetes, Mesos, Nomad, . . .
- Provisioning and mounting volumes

How CSI works

Container
Native FS
Interposer

Introduction

FUSE

CSI

Demo

Plans

OpenShift API server

Snapshotter container

Resizer container

Attacher container

Provisioner container

Kubelet

Driver container

External CSI controller
(Pod)

Infrastructure node

Driver registrar

Driver container

CSI driver daemon set
(Pod)

Node

Storage backend

Unix domain socket

Any proprietary protocol
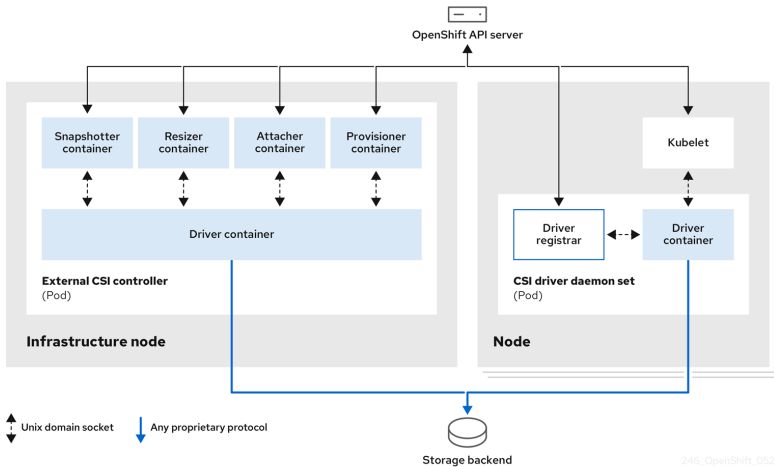
246_OpenShift_0522

Why CSI

- Declarative way to specify desired properties of a volume
- Interoperability with container technologies

- Deploy CSI plugin in local k8s cluster
- Deploy container mounting a volume backed by CSI
- Inspect the content of the mounted CSI volume

Future Plans

Extend FUSE with additional capability

Implement new utility FUSE filesystems for throttling/fault injection/etc.

Integrate the FUSE filesystems with CSI

Make the attributes of the FUSE filesystem configurable via the CSI interface

Burndown

Container
Native FS
Interposer

Introduction

FUSE

CSI

Demo

Plans

Title	Date	Moved to Done Estimated Hours	Moved to Done Actual Hours	Ideal Hours	Burndown Estimated	Burndown Actual	Team #	Hours/Sprint
	Sep 10, 2024	0	0	80	80	80	4	20
	Sep 11, 2024	0	0	74	74	74		
	Sep 12, 2024	0	0	69	69	69		
	Sep 13, 2024	0	0	63	63	63		
	Sep 14, 2024	0	0	57	57	57		
	Sep 15, 2024	0	0	51	51	51		
	Sep 16, 2024	0	0	46	46	46		
Learn FUSE/CSI	Sep 17, 2024	10	10	40	30	30		
	Sep 18, 2024	0	0	34	34	34		
Discussing plans	Sep 19, 2024	5	5	29	24	24		
Scaffolding for CSI plugin	Sep 20, 2024	14	13	23	9	10		
Adapt passthrough FUSE filesystem	Sep 21, 2024	1	1	17	16	16		
	Sep 22, 2024	0	0	11	11	11		
Package and deploy CSI	Sep 23, 2024	5	5	6	1	1		
Prepare for presentation and demo	Sep 24, 2024	6	6	0	-6	-6		

Moved to Done Estimated Hours, Actual Hours, Ideal Hours, Running Total Estimate and Running Total Actual

— Ideal Hours — Burndown Estimated — Burndown Actual