Storage Subsystem for Hardware TEE Based Confidential Containers

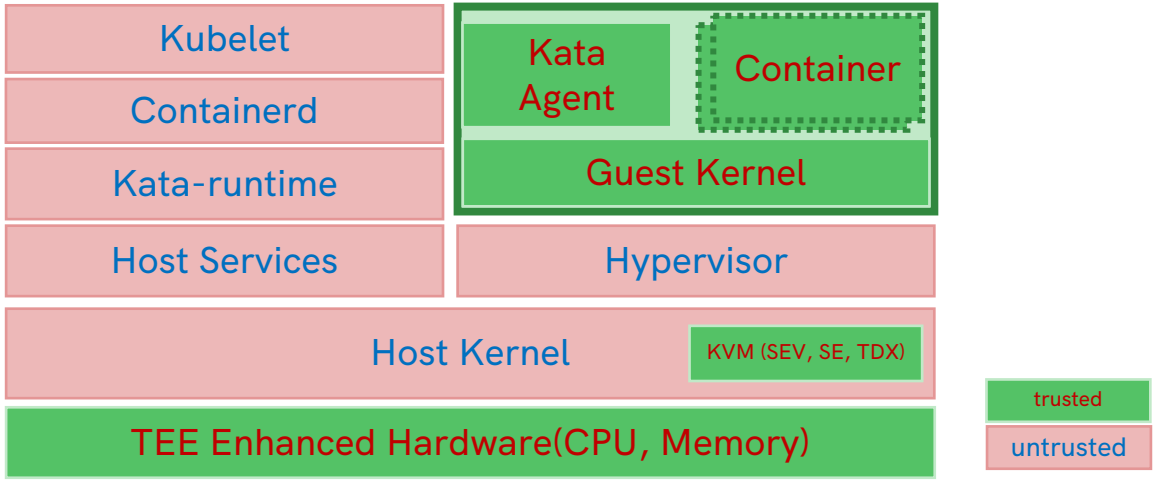
Gerry Liu

Senior Staff Engineer

Alibaba Cloud

Hardware TEE Based Confidential Containers

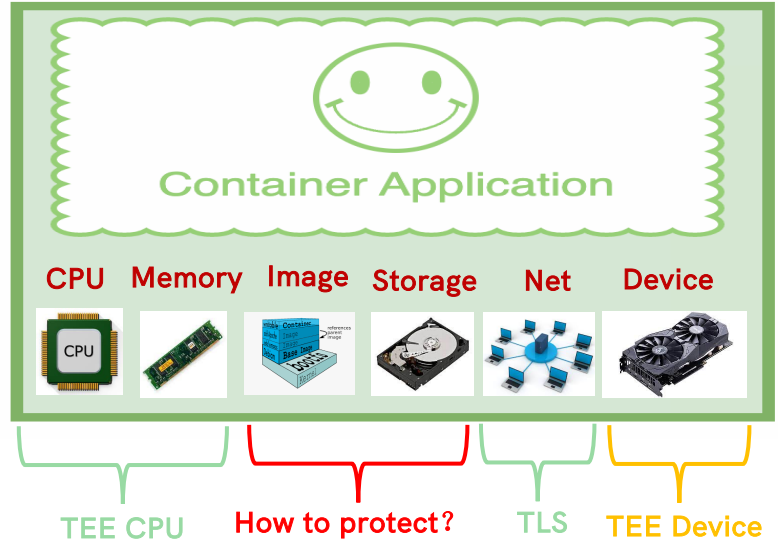
Confidential Containers aims to protect ***confidentiality*** and ***integrity*** for ***container*** workloads with ***hardware TEEs***.



What Resources to Protect?

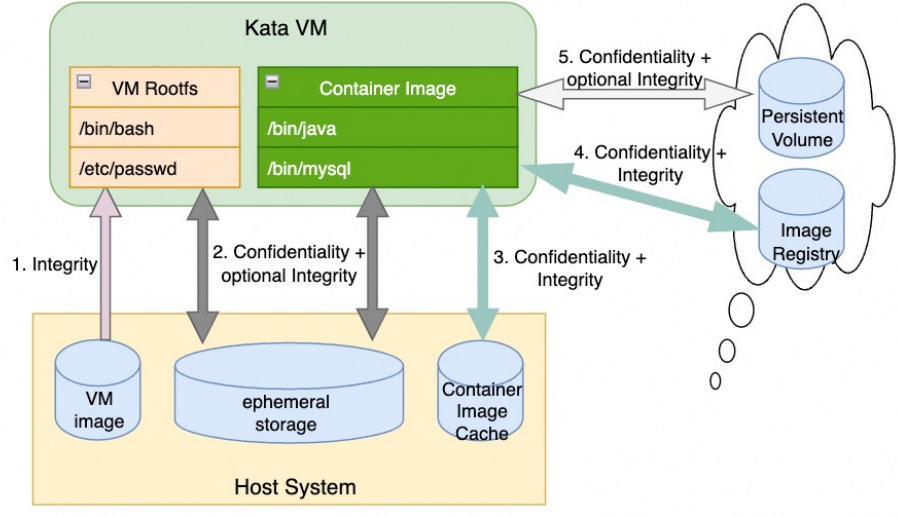
Requirements

- **Confidentiality**
- **Integrity**
- **Consistency**
- **High Performance**
- **Moderate Cost**

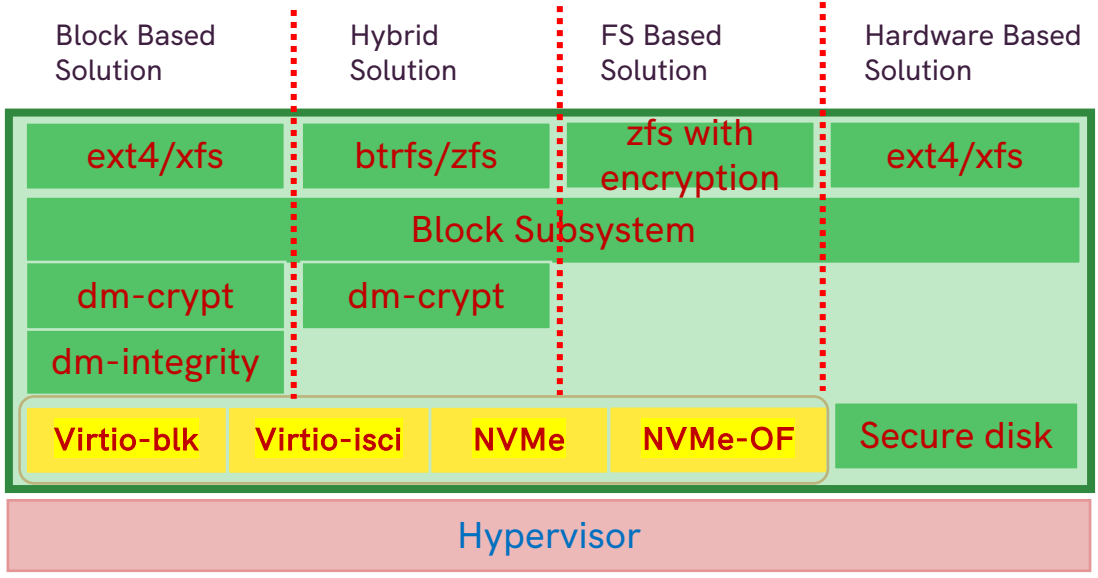


Usage Scenarios and Requirements

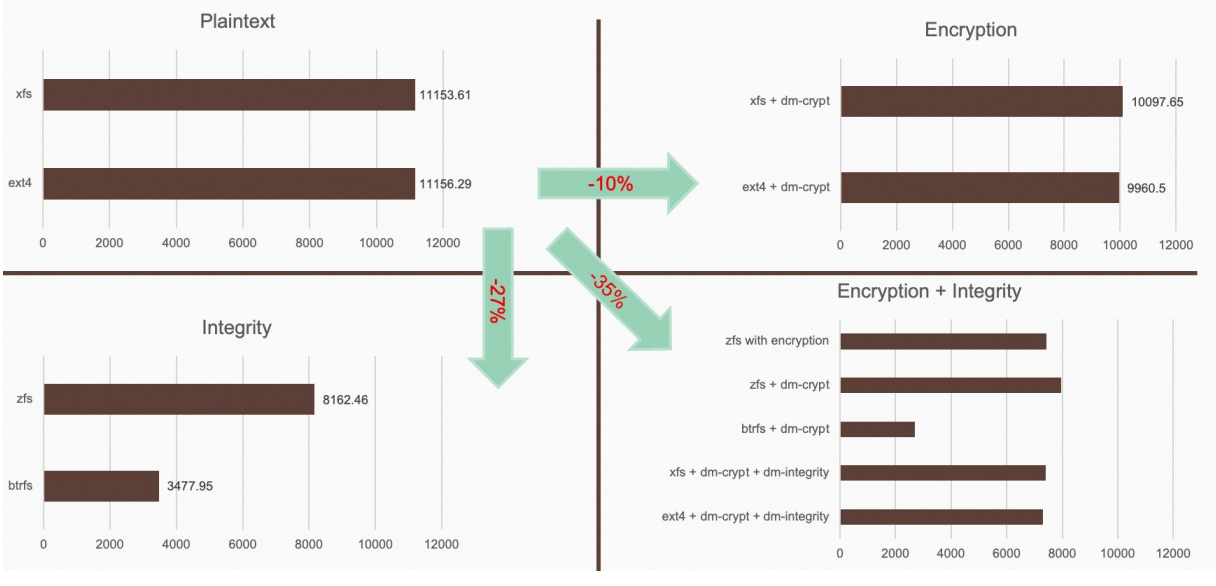
- Guest Rootfs
  Integrity
- Container Image
  Integrity
  Integrity + Confidentiality
- Ephemeral Storage
  Integrity + Confidentiality
- Persistent Storage
  Integrity + Confidentiality



Ephemeral and Persistent Storage
--Available Technical Stacks



Ephemeral and Persistent Storage
-- Cost of Current Technical Stacks

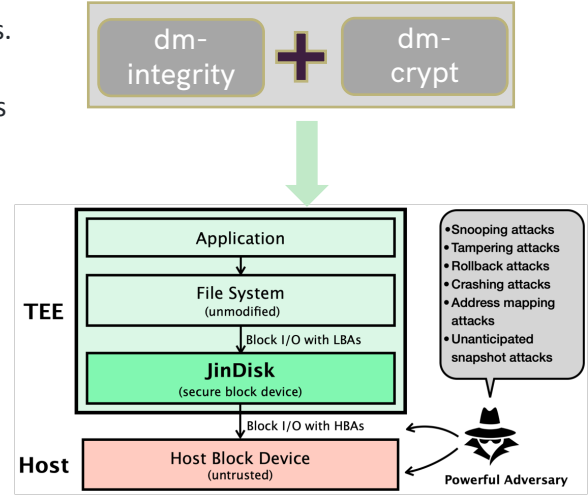


Sysbench-tpcc: --threads=64 --tables=10 --scale=100 --db-driver=mysql

Ephemeral and Persistent Storage

-- New DM Target for CoCo

- **Confidentiality** guarantees that the user data submitted by any write is not leaked and thus prevents tampering attacks.
- **Integrity** promises that the user data returned from any read are genuinely generated by the user and thus prevents snooping attacks.
- **Freshness** ensures that the user data returned from any read are up-to-date and thus prevents rollback attacks.
- **Consistency** ensures that all the security guarantees are held despite any accidental crashes or crashing attacks.
- **Atomicity** promises that all writes before a flush are persisted in an all-or-nothing manner.
- **Anonymity** avoids LBA leakage in the sense that the adversary cannot learn LBAs from the on-disk data structures directly or infer LBAs from HBAs.



https://github.com/jinzhao-dev/jinzhao-disk

Image Management for CoCo
-- Requirements

- Confidentiality
- Integrity at runtime
- High performance

+

- Quick container startup
- Download image inside guest
- Low cost
  - Low memory
  - Low network
  - Low disk

We need all of these!!!

- Encryption with authentication
- New image format & spec

+

- Cache image data on host
- Data Deduplication
- Data Compression
- Data Lazy Loading

Image Management for CoCo
-- Containerd Image Management

- Lack of confidentiality, integrity and downloading inside guest

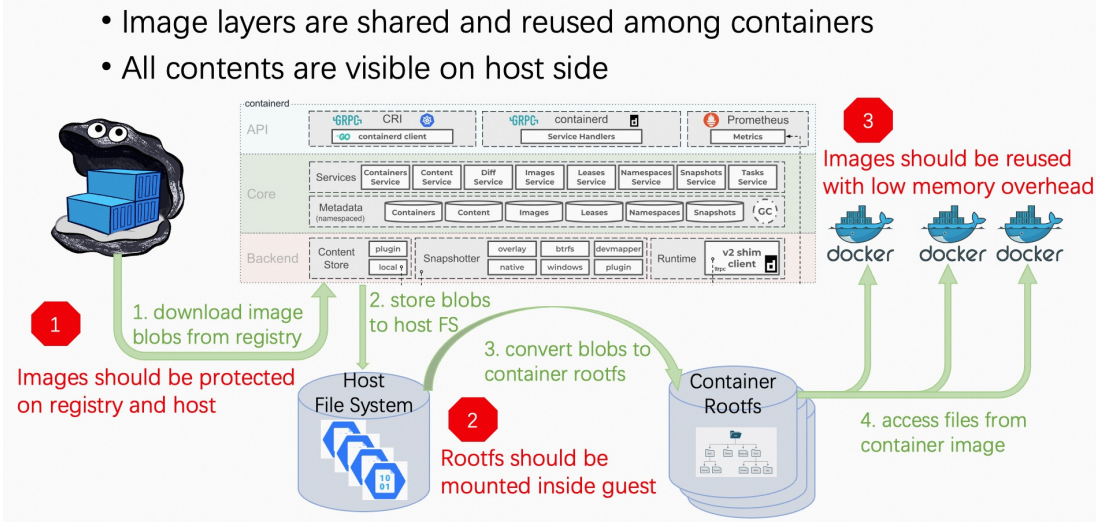
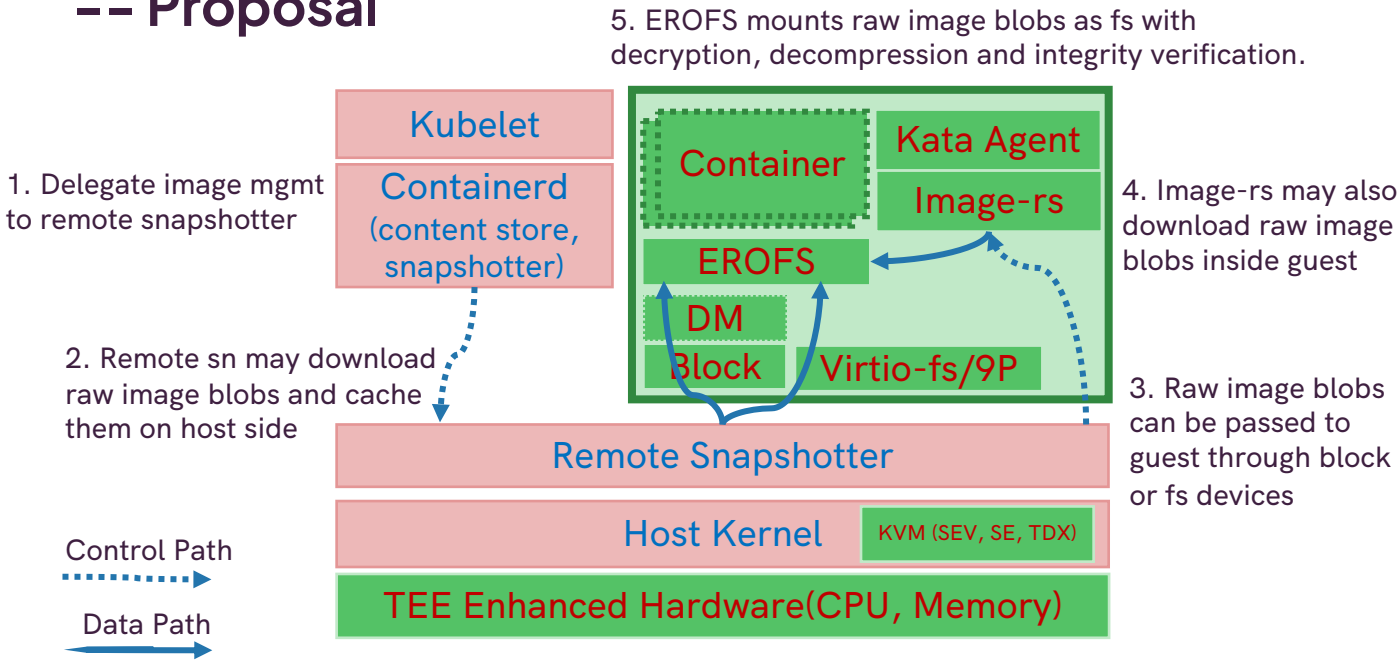


Image Management for CoCo
-- Proposal



Nydus Image Service for CoCo
--Implement above proposal

Introduction

- Nydus Remote Snapshotter
- Nydus Image Service
  - Data lazy loading
  - Runtime data dedup
  - Flexible integration
- Nydus Image Format
  - Build time data dedup
  - OCIv1 compatible mode

Contribution Welcomed!

- Data encryption
- Convergent encryption for dedup
- Compression + encryption
- Integration with image-rs

Thanks!

What is Nydus Image Service?

A readonly filesystem for containers (runC/Kata/Kata CC), AI models and software packages by:

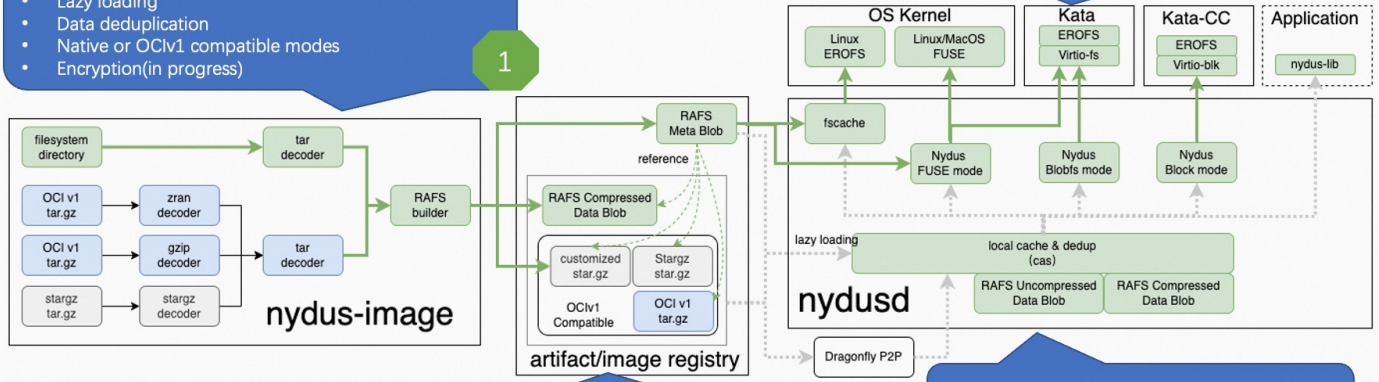
- Linux/MacOS FUSE
- Virtio-fs
- EROFS with page sharing
- User space library (in progress)

2

An image format w/ advanced features:

- Lazy loading
- Data deduplication
- Native or OCIv1 compatible modes
- Encryption(in progress)

1



An image service integrated with ecosystem:

- OCI distribution compatible
- Integrated with buildkit/containerd/cri-o/nerdctl/Kata/harbor/dragonfly

4

A node level storage subsystem with P2P, cache and data deduplication

3