Red Hat Summit

Connect

Confidential Computing

with Red Hat OpenShift: concepts, use cases and roadmap

Dieter De Moitié

Specialist Solution Architect, OpenShift

What is confidential compute

Protecting data

Existing Encryption

Data At Rest

Data Encryption and System Attestation

- Protect Data on Disk using LUKS and Network Bound Disk Encryption (NBDE)
- Attest system state on boot, ensuring that system packages have not been compromised with IMA and Keylime
- Attest container workloads with Key Brokering Service (KBS)
- Ensure data remains unreachable in case of unauthorized access to the storage

Data In Transit (EU) / in Motion

Encrypt Traffic Leveraging Strong, Future Proof Cryptography

- Protect data moving through the environment using strong, smart and validated cryptography
- Centralized management and establishment of secured connections native part of workflow for Red Hat Platform
- Automation provided allows consistent deployment and security configuration

New

Data In Use

Confidential Computing Use Trusted Execution Environment Capabilities

- Protect data and workloads in use with Trusted Execution Environments as Confidential Computing is enabled on the major hardware platforms for AMD, Intel and NVIDIA
- Leverage these Trusted Execution Environments on prem or in the major cloud platforms
- Increases the security level of organizations that manage sensitive and regulated data

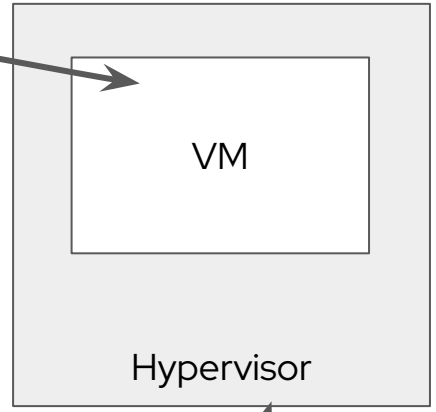
https://confidentialcomputing.io/wp-content/uploads/sites/10/2023/03/CCC_outreach_whitepaper_updated_November_2022.pdf

Why ?

- Regulation and move to the cloud
  - Example: Digital Operational Resilitience Act
    - *Financial entities shall … maintaining high standards of security, confidentiality and integrity of data,* ***whether at rest, in use or in transit.***
- Volatility in the markets and its effect.
  - Moving applications from one environment to the next.
  - Digital Sovereignty, Zero Trust and Confidential Computing

A test drive in memory forensics

```
my_super_secret_password=redhat123!
export my_super_secret_password
env
/bin/sleep 9999 # start child process
```



virtctl memory-dump get memory-dump --create-claim …

A test drive in memory forensics

$ **vol -f** memory.dump **linux.pslist**

0x8d47c1a6c780 1 1 0 systemd 0 0 0 0 2025-09-09 07:17:53.068616 UTC Disabled
0x8d47c1a68000 2 2 0 kthreadd 0 0 0 0 2025-09-09 07:17:53.068616 UTC Disabled
0x8d47c1a6a3c0 3 3 2 pool_workqueue_ 0 0 0 0 2025-09-09 07:17:53.069616 UTC Disabled
0x8d47c1a923c0 4 4 2 kworker/R-rcu_g 0 0 0 0 2025-09-09 07:17:53.069616 UTC Disabled
0x8d47c1a94780 5 5 2 kworker/R-sync_ 0 0 0 0 2025-09-09 07:17:53.069616 UTC Disabled
…
0x8d47c9e44780 1518 1518 1517 bash 1000 1000 1000 1000 2025-09-09 07:18:22.687829 UTC Disabled
…
0x8d47cbb7c780 1773 1773 1518 sleep 1000 1000 1000 1000 2025-09-09 08:13:11.465208 UTC Disabled

A test drive in memory forensics

$ **vol** -f memory.dump **linux.bash.Bash**

...

1518 bash 2025-09-09 07:20:41.000000 UTC export my_super_secret_password
1518 bash 2025-09-09 07:20:41.000000 UTC my_super_secret_password=redhat123!
1518 bash 2025-09-09 07:20:41.000000 UTC env
1518 bash 2025-09-09 08:13:12.000000 UTC /bin/sleep 9999

A test drive in memory forensics

$ **vol** -f memory.dump **linux.envars --pid 1773**

….

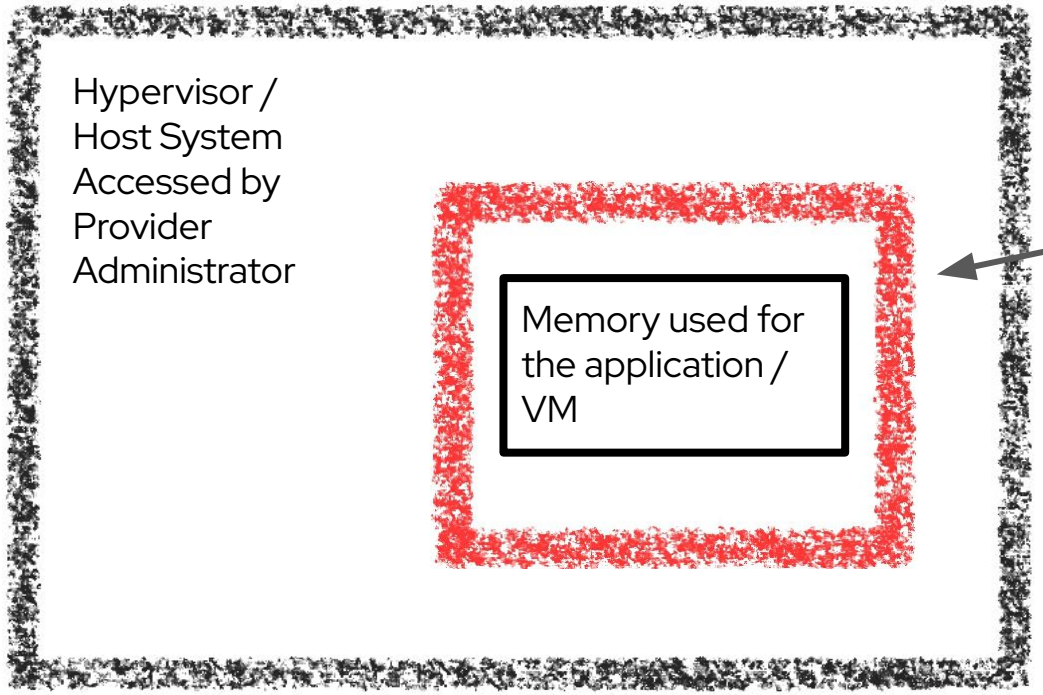
1773 1518 sleep SHELL /bin/bash
1773 1518 sleep HISTCONTROL ignoredups
1773 1518 sleep HISTSIZE 1000
1773 1518 sleep HOSTNAME memory-dump
1773 1518 sleep PWD /home/cloud-user
1773 1518 sleep LOGNAME cloud-user
1773 1518 sleep XDG_SESSION_TYPE tty
1773 1518 sleep my_super_secret_password redhat123!

What is confidential computing

Confidential Computing (CC) is the protection of **data in use** by performing the computation in a **hardware-based, attested Trusted Execution Environment**, according to the Confidential Computing Consortium's definition.

The three primary attributes of a Trusted Execution Environment are **data integrity, data confidentiality, and code integrity**.

How?



Hidden from the Operating System and Kernel inside a **Trusted Execution Environment**. Access only through special API.

Hardware implementations

- AMD SEV SNP: Confidentiality on a Core based VM and on the Memory
- Intel SGX (core based) and TDX (VM)
- IBM z HyperProtect + Secure Execution
- ARM CCA
- AWS Nitro
- RISC-V (in progress)

Confidential computing organizations

Confidential Computing Consortium (CCC) - 2019

https://confidentialcomputing.io

AMD arm Google HUAWEI intel

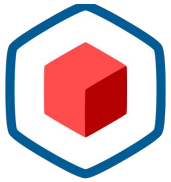
Meta Microsoft NVIDIA Red Hat Shielded

TikTok

OpenInfra Foundation

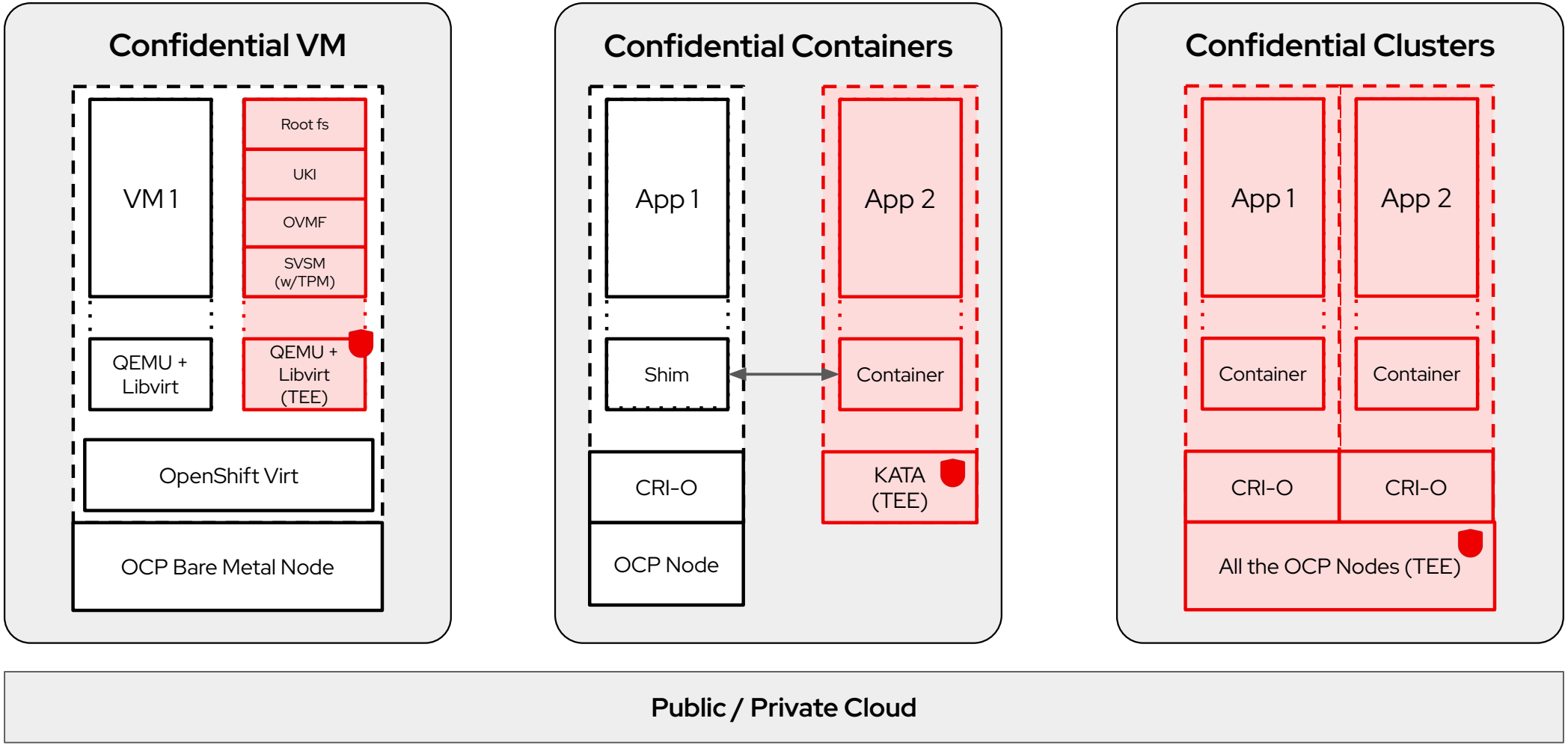
Kata Containers

Cloud Native Computing Foundation (CNCF)



CONFIDENTIAL CONTAINERS

3 projects with OpenShift



Attestation

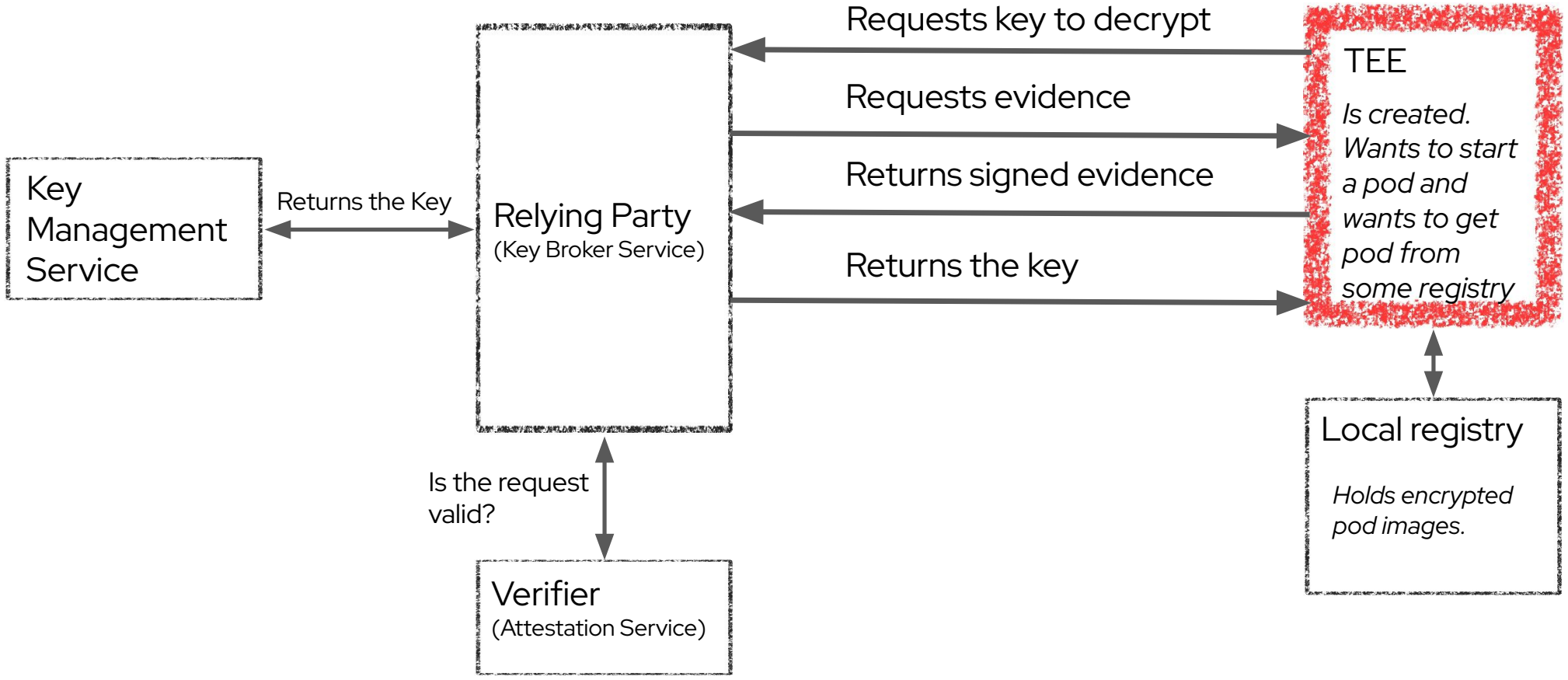
What is attestation ?

Attestation is a **cryptographic process** that allows a trusted execution environment (TEE) to **prove to a remote party** that it **is running on genuine hardware and with the expected software and configuration**, ensuring the integrity and confidentiality of the data and workload.

Attestation

- Architectural decision: all content in a Trusted Execution Environment (TEE) coming from the outside which is encrypted
  - data / container images / kubernetes secrets / storage
- This key is provided through Attestation making it Security by Design
- Local attestation
  - Cloud provider is providing an attestation service
  - cloud provider and the attestation provider are in the same company
- Remote attestation
  - attestation service is not located within (or administered by) the cloud provider
  - better protection

Flow of attestation



Red Hat Build of Trustee

- Trustee = open source project to implement attestation service
- Red Hat Build of Trustee to add Enterprise support.
- Is deployed as an operator on an OpenShift System
- Comes with OpenShift Container Platform or OpenShift Platform Plus subscription

Trusted Computing Base

- A trusted computing base (TCB)
  - all hardware, firmware, and software components of a computer system that are critical to its security.
  - Bugs, vulnerabilities jeopardize the security of the entire system

- In the context of confidential containers: Trusted Execution Environment (TEE), the confidential container itself, and the minimal software running inside it.

- In the context of attestation: The attestation service needs to run in a TCB.
  - chicken and egg problem

Root of trust

- Typically on-prem datacenter
  - Presumed to be the most secure, best control over this environment
  - Also contains the key management system

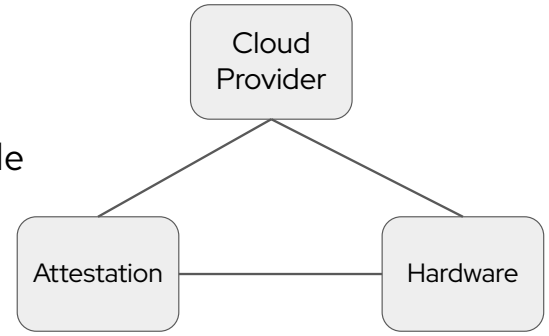
- Trusted single use platform
  - Laptop
  - Used to start initial attestation source and start the chain of trust

- Trusted third party attestation provider
  - Service provider you trust implicitly

Separation of responsibilities

- Cloud Provider
  - Can be forced to provide access to the data through regulations
- Hardware Provider
  - Intel, AMD, ...
- Attestation Provider
  - Provides attestation services and has access to Key Management Services.
- Software Provider
  - Provides the Software to create and run the Confidential Container and provide the Attestation Software
  - Red Hat OpenShift and Build of Trustee in our case
- Client
  - Always assumed secure as for the customer
- **To break through the security mechanisms you need roles to collaborate (cloud provider + other)**

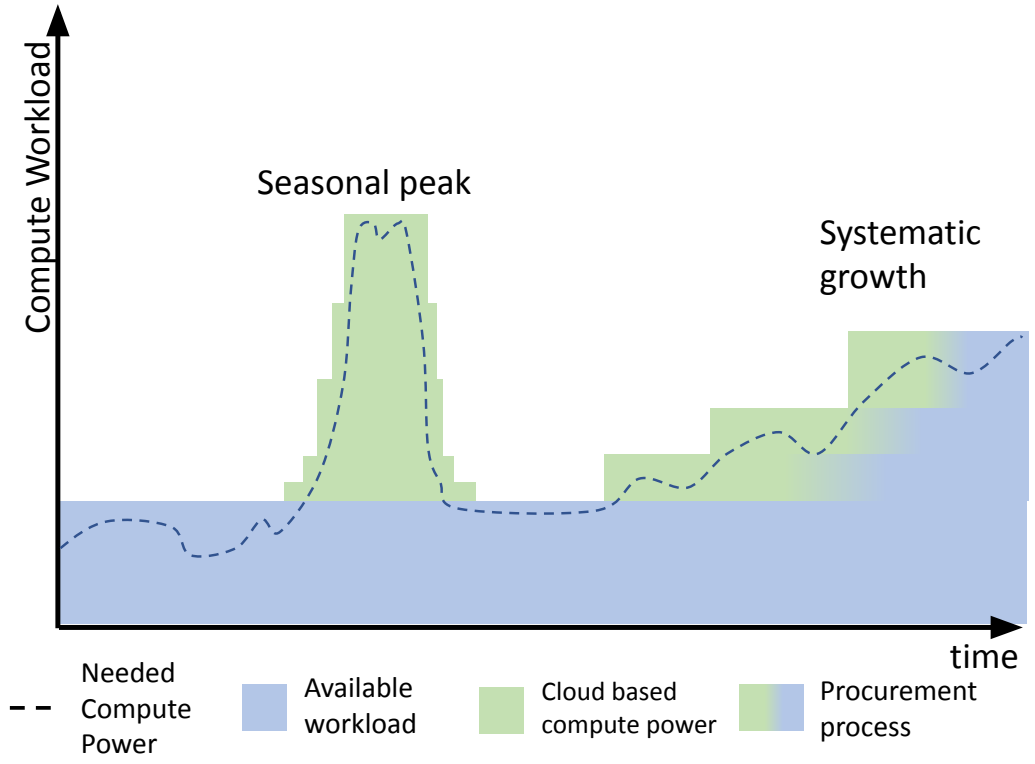
Separation of responsibilities



- Attack vector: The Hardware providers participates.
  - CPU is not started in Conf.Mode but provides evidence to convince Attestation service to provide key.
- Attack vector: The Attestation provider participates.
  - CPU is not started in Conf.Mode and the provides non confidential evidence. But Attestation service provides access to the key anyway.
- The Software if it is open source it can be externally verified not to include backdoors.
- The attestation service should be deployed in the companies TCB.
  - **=> Unique to Red Hat**

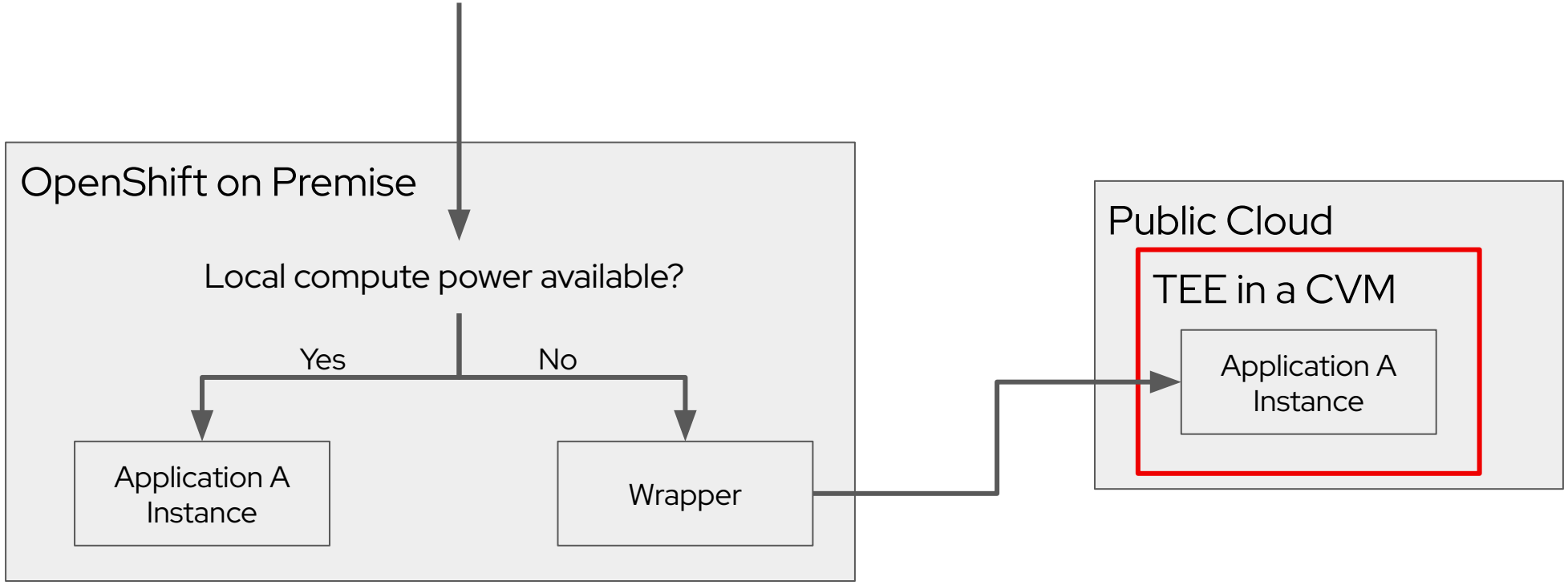
Use cases

Use case: Dynamic Secure Cloud Bursting



public cloud acting as a consumption based flexible extension of the on premise Datacenter.

Use case: Dynamic Secure Cloud Bursting

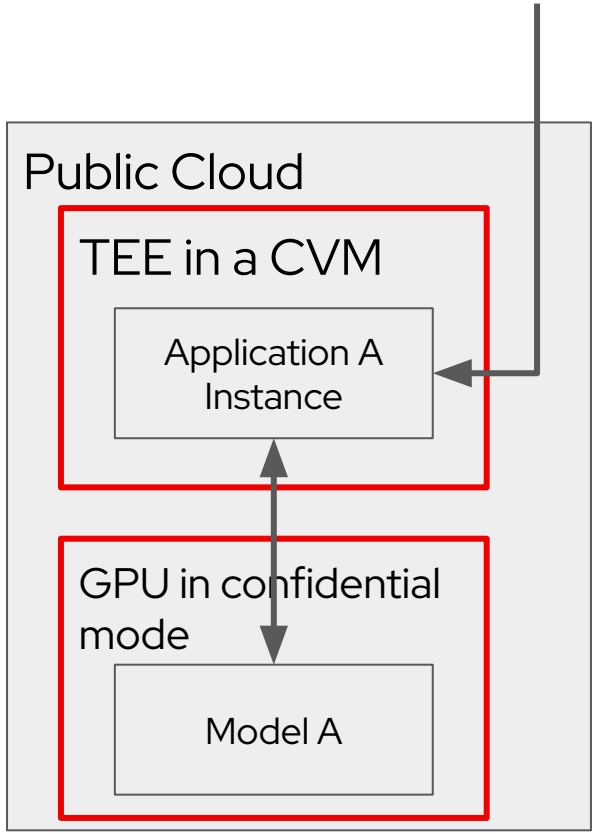


Demo-ed using KEDA (Kubernetes event-driven autoscaler)

https://www.redhat.com/en/blog/secure-cloud-bursting-leveraging-confidential-computing-peace-mind

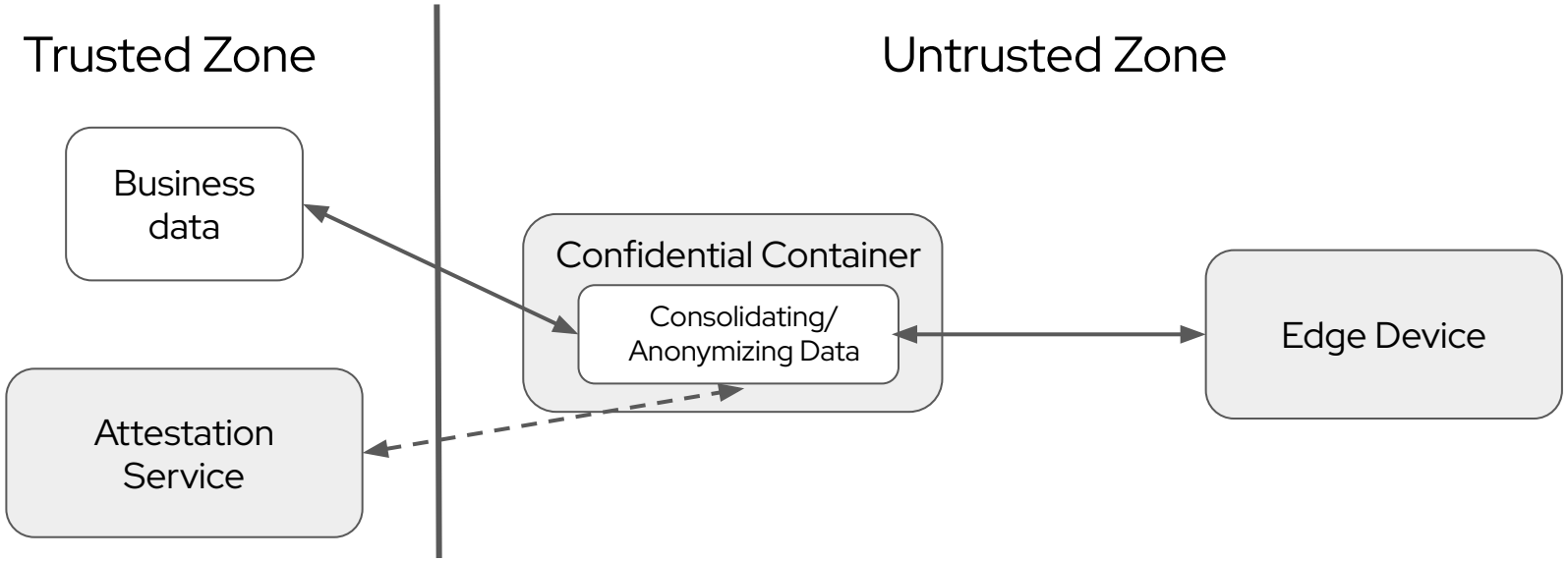
Use case: Dynamic Secure Cloud Bursting

- Integrating the execution of the AI model into a Confidential Computing context.
- The GPU is external to the CPU
  - Modern GPUs support TEE
  - Requires multipart attestation (roadmap)



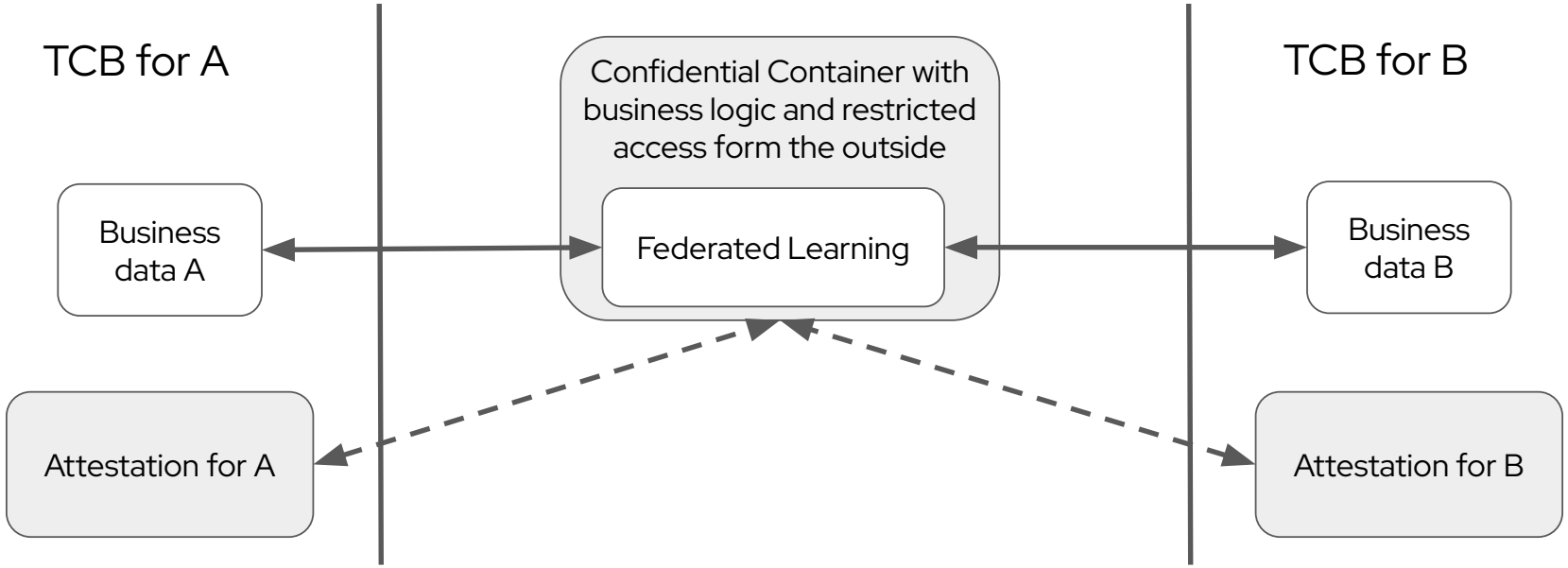
Red Hat

Use cases: edge integration (public cloud, roadmap)



Securing privacy data from external access in an edge device.

Use cases: partner interaction (public cloud, roadmap)



Interaction of different partners to exchange information for example for a federated learning scenario but aren't willing to share the business data.

Centralized component which restricts outside interaction.

With Confidential Container no participant has access to training data.

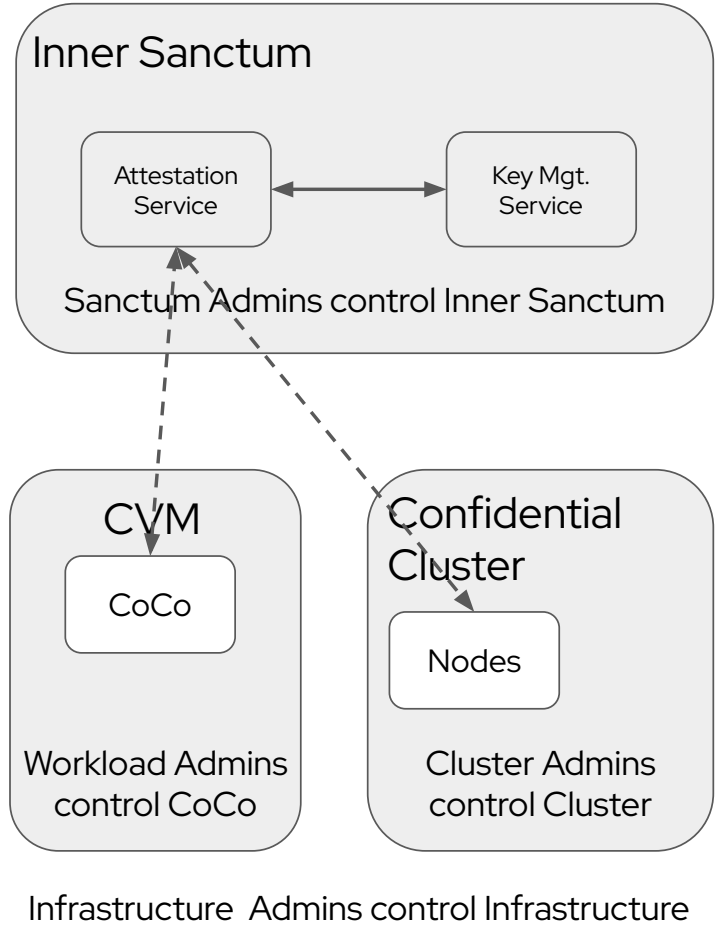
Use cases: The inner sanctum (on-premise)

Segregating access of

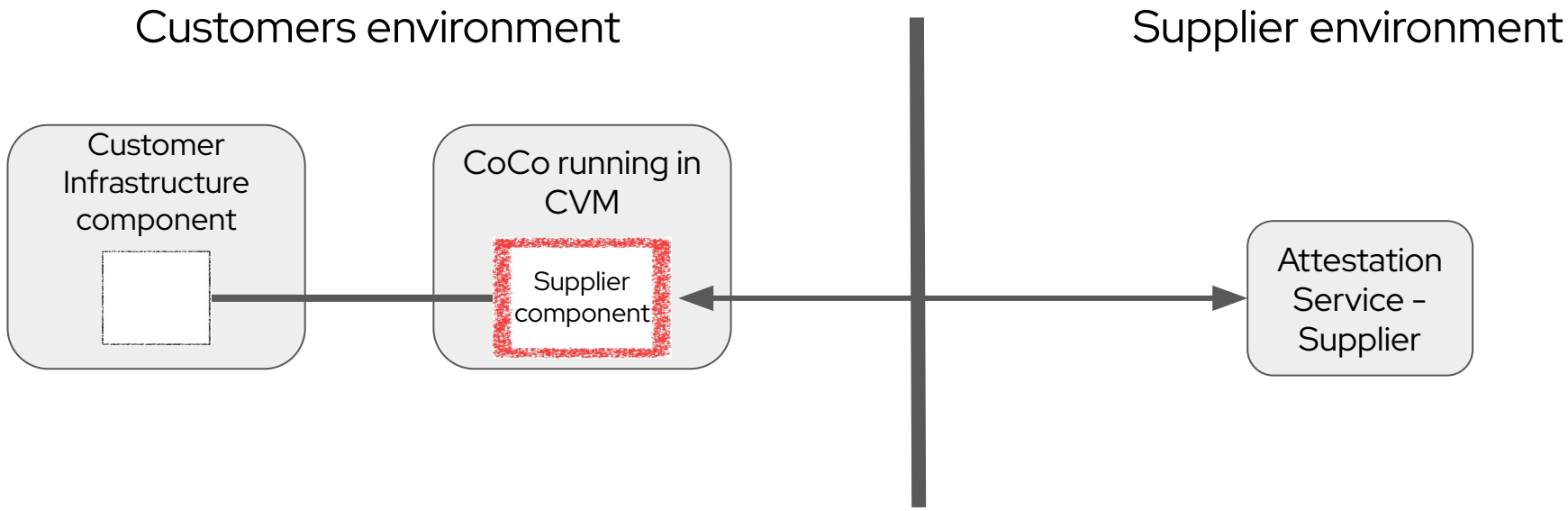
- Infrastructure Administrators
- Cluster Administrators
- Workload Administrators

establishing Inner Sanctum Admin Group to restrict access to Confidential Computing Services like Attestation and KMS.

Implementation of 4 eye principle to get physical access.



Use cases: IP/integrity protection (on-premise)



A supplied component is running inside a customers infrastructure to provide services.

The supplier wants to ensure that the software used in this cannot be accessed by a customer.

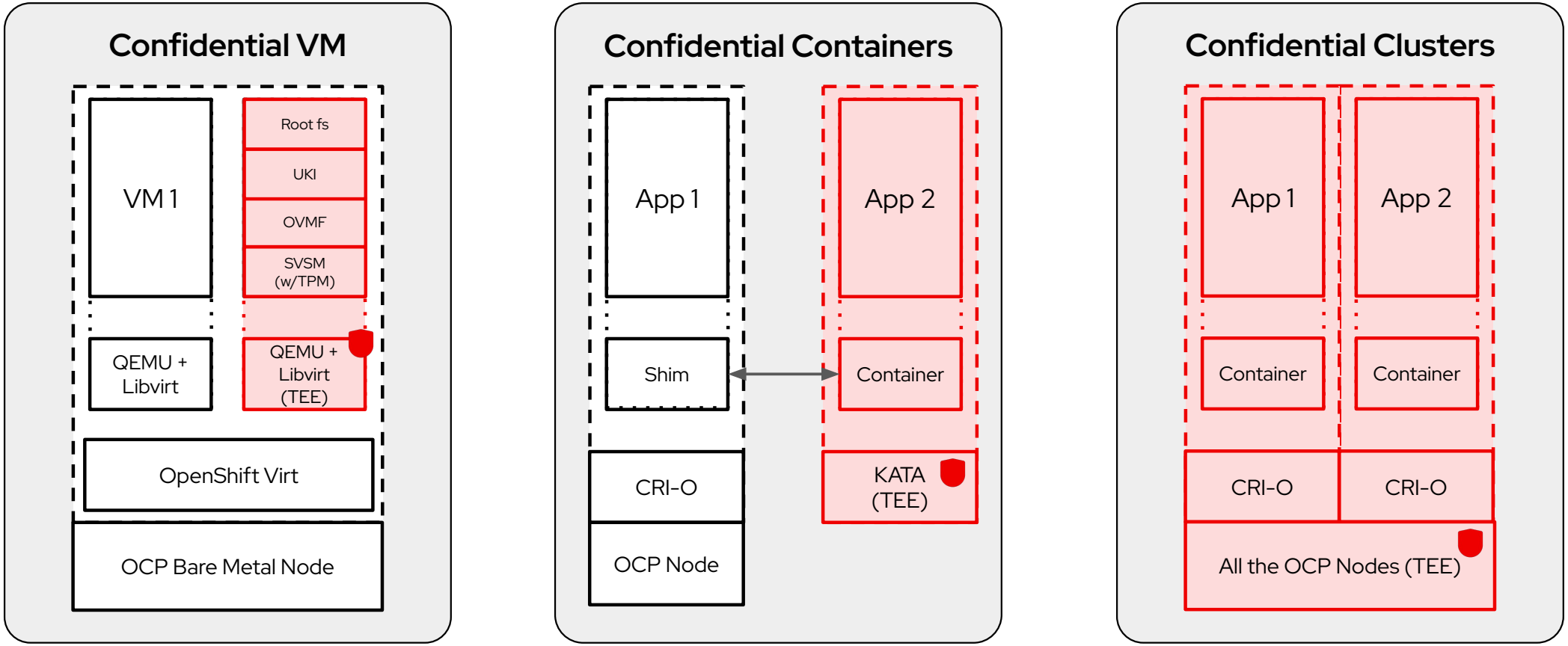
Example: robot in a factory.

Summary

Summary

- **Confidential Computing (CC)** allows to **protect data in use**
- The computation in a hardware-based **Trusted Execution Environment**
- **Attestation** allows to prove that it is **running on genuine hardware and with the expected software and configuration**
- Red Hat provides the **Red Hat Build of Trustee** for attestation
- Run the attestation service in a trusted environment
- **Remote attestation** and **open source software** makes this more secure
- 3 projects => see next slide

3 projects with OpenShift



Red Hat Summit

Connect

Thank you


linkedin.com/company/red-hat


facebook.com/redhatinc


youtube.com/user/RedHatVideos


twitter.com/RedHat