ARED Distributed PAAS/IAAS Architecture Technical document - Extension

In the following sections, we will define ARED's software architecture strategy, which will outline the key components, technologies, and interactions that will be used to build the distributed infrastructure and platform as a service. This architecture strategy will provide a comprehensive roadmap for designing, developing, and deploying the software, and will ensure that the system meets the requirements outlined in the ARED Distributed PAAS/IAAS Architecture Technical document.

Approach:

Akraino – LF EDGE: Building an Open Source Framework for the Edge.

Detailed explanation of technical achievement for ARED on our distributed infrastructure.

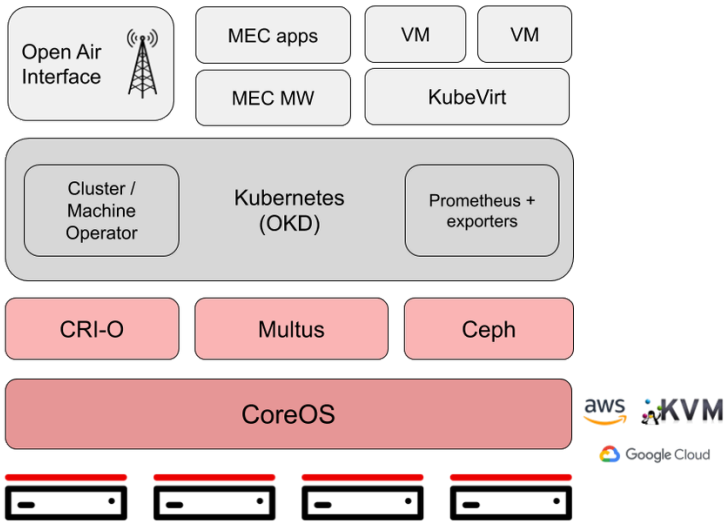
Blueprint based on Akraino project. - framework for edge compute Paas/Iaas.

Reasons for choosing Provider Access Edge (PAE) Blueprint

- ARED Business Use Cases for distributed Edge deployments for 5G, mobile edge applications, vRAN
- Technical document ressources
- Release 4 of PAE
- Provider Access Edge (PAE) Blueprint
- Also review https://www.huawei.com/en/open-source/projects
- Project – EdgeGallery

Software Platform Architecture

KNI PAE blueprint can be run on multiple environments (libvirt, AWS and baremetal) based on architecture document. The approach applicable for ARED would be the baremetal/virtual baremetal (KVM)



OS: Yocto project

KVM:

libvirt:

Kubernetes

Ceph:

Kubevirt:

Storage:

Support for Ceph storage in KNI PAE was added in KNI PAE Release 2 Notes - Akraino - Akraino Confluence

Add ceph filesystem and storage class (I76038641) · Gerrit Code Review (akraino.org)

The baremetal cluster storage yaml files can be configured to change the storage class. for example**,** by default the project supports Ceph storage as the default storage class. A storage class or persistant volume can be configured in the corresponding yaml file to support appropriate storage media in the cluster.

…/03_ceph_storage_class.yaml · Gerrit Code Review (akraino.org)

…/03_ceph_storage_filesystem.yaml · Gerrit Code Review (akraino.org)

KNI PAE Installation Guide - Akraino - Akraino Confluence

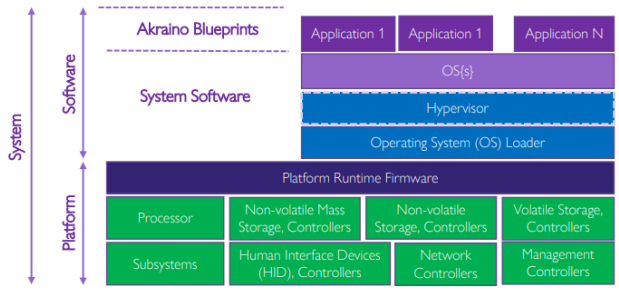
KNI PAE is based on Intel Xeon.

Other blueprints like, Integrated Edge cloud is based on ARM.

Security:

Security of Akraino execution environment (or System) consists of both Platform and Software security:

- Containerized environment security (Akraino Blueprints)
- System software security (OS loader, Hypervisor and OS{s})
- Firmware Security
- Platform Devices Security



Akraino Platform Security Objectives

- Maintain the integrity of the platform layer and provide a safe execution environment for Akraino software stack.
- Define secure boot environments based on the platform Root-of-Trust.
- Secure attesting the platform's state of integrity. ▪ Protection of key assets in the platform:
  - Platform critical data (platform Id, encryption keys, configuration data, etc.).
  - Mutable firmware components.

- Secure platform firmware update.
- Protection platform runtime environment and data.
- Provide a secure interface for the Akraino software stack to the platform firmware runtime services and devices (TPM, TEE, etc.).

Akraino Platform Security Goals

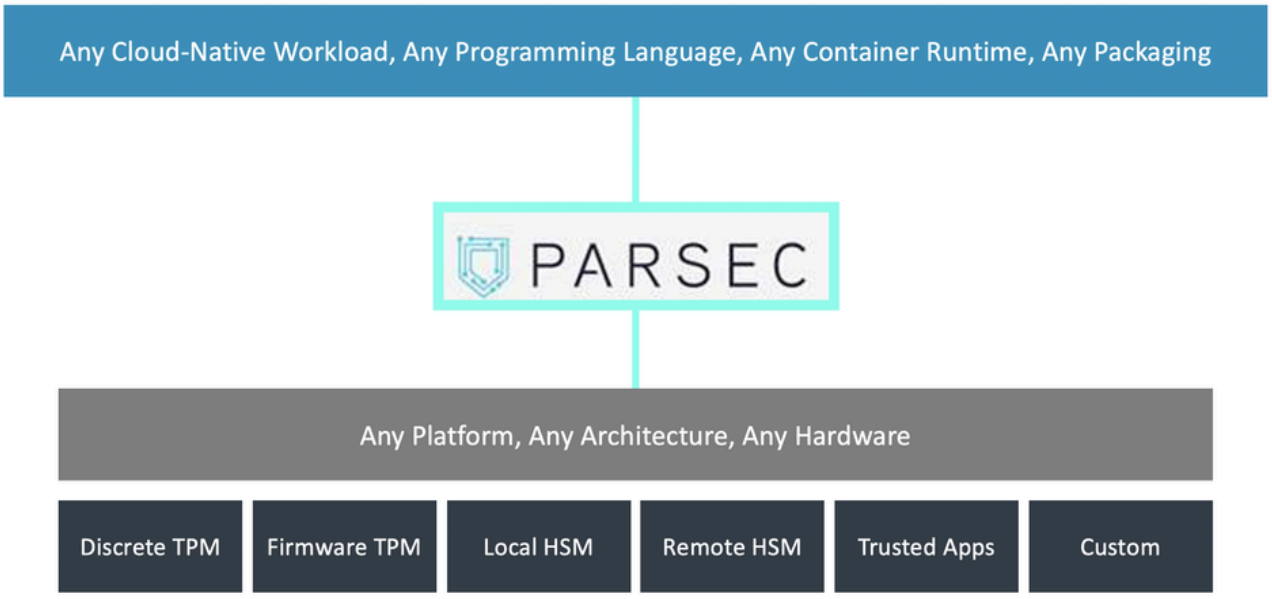
- **Unique identification.** Devices shall be uniquely identifiable.
- **Security lifecycle.** Devices shall support a security lifecycle. The device states shall be attestable and may impact access to data that is bound to the device.
- **Attestation**. Devices shall be securely attestable.
- **Software authorization.** Devices shall ensure that only authorized software is executed. Secure boot and secure loading processes are necessary to prevent unauthorized software from being executed.
- **Secure update.** Devices shall support secure update of software, or platform critical data like hardware configuration.
- **Anti-rollback**. Devices shall prevent unauthorized rollback of updates
- **Isolation.** Devices shall support isolation. Isolation of trusted services from one another and from less trusted services is essential to protect confidentiality and integrity of that service.
- **Interaction.** Devices shall support interaction over isolation boundaries. The interfaces must not be used to compromise confidentiality and integrity of the device
- **Device binding of stored data**. All devices shall support unique binding of stored sensitive data to the device.
- **Cryptographic and trusted services.** All devices shall support a minimum set of trusted services and cryptographic operations that are necessary to support other security goals

Akraino Security Platform Abstraction:

Platform Abstraction Interface should be available for the Akraino blueprints for securely accessing the platform's runtime services and secure devices.

PARSEC:

Platform AbstRaction for SECurity, an open-source initiative to provide a common API to hardware security and cryptographic services in a platform-agnostic way. This abstraction layer keeps **workloads decoupled from physical platform details, enabling cloud-native delivery flows within the data center and at the edge.**



PARSEC provides the following:

▪ **Abstraction** – a common API that is truly agnostic and based on modern cryptographic principles

▪ **Mediation** – security as a microservice, brokering access to the hardware and providing isolated key stores in a multi-tenant environment

▪ **Ergonomics** – a client library ecosystem that brings the API to the fingertips of developers in any programming language: “easy to consume, hard to get wrong”

▪ **Openness** – an open-source project inviting contributions to enhance the ecosystem both within the service and among its client libraries

GitHub - parallaxsecond/parsec: Platform AbstRaction for SECurity service

Why PARSEC?

Use Parsec when you need:

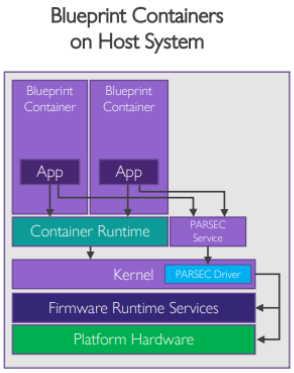
- A portable interface to your platform's Root of Trust in order to manage keys and perform cryptographic operations without knowledge of the hardware.
- A simple and portable way to access the best available security of your platform in your preferred programming language.

What PARSEC Provides

The value proposition of Parsec is that it provides the following:

- **Abstraction** – a common API that is truly agnostic and based on modern cryptographic principles
  Mediation – security as a microservice, brokering access to the hardware and providing isolated key stores in a multi-tenant environment
- **Ergonomics** – a client library ecosystem that brings the API to the fingertips of developers in any programming language: “easy to consume, hard to get wrong”
- **Openness** – an open-source project inviting contributions to enhance the ecosystem both within the service and among its client libraries

Akraino Blueprints in Native Host Environment With PARSEC



Akraino Security Development Lifecycle

Akraino Security Development Lifecycle - Akraino - Akraino Confluence