Trusted Docker Containers and Trusted VMs in OpenStack

Raghu Yeluri
Abhishek Gupta
Outline

- **Context: Docker Security – Top Customer Asks**
- **Intel’s Focus: Trusted Docker Containers**
- **Who Verifies Trust?**
  - Reference Architecture with OpenStack
- **Demo**
- **Availability**
- **Call to Action**
Docker Overview in a Slide..

Lightweight, open source engine for creating, deploying containers

Provides work flow for running, building and containerizing apps.
Separates apps from where they run.; Enables Micro-services; scale by composition.

Underlying building blocks: Linux kernel’s namespaces (isolation) + cgroups (resource control) + ..

Components of Docker

Docker Engine – Runtime for running, building Docker containers.
Docker Repositories(Hub) - SaaS service for sharing/managing images
Docker Images (layers)
Images hold Apps. Shareable snapshot of software. Container is a running instance of image.

Orchestration: OpenStack, Docker Swarm, Kubernetes, Mesos, Fleet, Project Atomic, Lattice...
Docker Security – 5 key Customer Asks

1. How do you know that the Docker Host Integrity is there?
   o Do you trust the Docker daemon?
   o Do you trust the Docker host has booted with Integrity?

2. How do you verify Docker Container Integrity
   o Who wrote the Docker image? Do you trust the image? Did the right Image get launched?

3. Runtime Protection of Docker Engine & Enhanced Isolation
   o How can Intel help with runtime Integrity?


5. OpenStack as a single Control Plane for Trusted VMs and Trusted Docker Containers..

Intel’s Focus: Enable Hardware-based Integrity Assurance for Docker Containers – Trusted Docker Containers
Trusted Docker Containers – 3 focus areas

- Launch Integrity of Docker Host
- Runtime Integrity of Docker Host
- Integrity of Docker Images
**Trusted VMs - Summary**

- Launch VMs on Servers that have demonstrated Boot Integrity – Platform Trust
  - Measured Launch of Boot Process/Components with Intel TXT.
  - Trust Chain: HW->FW->BIOS->OS/VMM
  - What is measured at launch:
    - Current: F/W, Core BIOS, OS/VMM Kernel, Initrd
    - Ext measurements: An7OS/FS modules

- Schedulers/Orchestrators Policy Manager use Trust to launch/create/Migrate VMs.

- Extend Chain of Trust to VMs.
  - Measure & Attest VM Images prior to Launch.
  - Encrypt VM Images and decrypt based on Platform Trust (Tenant-Controls the Keys)

- Boundary Control of VMs– Control where your Trusted VMs are launching and migrating.

*Will enable the same model and use-cases for Trusted Docker Containers*
Ensure Docker Containers are launched on Trusted Docker Hosts

- Boot-time integrity of the Docker Host
  - Measured Launch of Boot Process & components with Intel TXT.
- Docker daemon and associated component added to TCB and Measured.
- Chain of Trust: H/w->FW->BIOS->OS->Docker Engine
- Remote attestation using an Attestation Authority*
**Trusted Docker Containers - 2**

Ensure that Docker Images are not tampered prior to Launch -

- Launch time integrity of Docker Images
- Chain of Trust: H/w->FW->BIOS->OS->Docker Engine -> Docker container layers (apache, Ubuntu14.04, ubuntu14,…, base)
- Docker daemon modification: prior to container launch, measure and verify Docker image (and parent layer graph recursively)

Boundary Control/Geo-Tagging applies equally to Docker Containers as well - Compliance Needs.

- Orchestrator determines location/boundary for launching Docker Images.

Exploring: Docker Image encryption & Trust-based Retrieval of Keys – Sensitive Container Images (VNFs, PCI-DSS/HIPPA Containers.. etc)
What is measured for Trusted Docker Containers

Docker Daemon
- container management engine (e.g. Docker engine)
- Measurement Agents

Initrd++ (includes a measurement agent)

Bootloader, Tboot and OS Kernel

Bios

ACM signed by manufacturer

Intel® TXT + TPM

Chain of Trust extended to application launch

Containerized application layers (e.g. Docker image layers)
What is measured – the details

Measurement Phase 1 (H/W + BIOS)
- uCode evals BIOS ACM
- BIOS ACM (evals BIOS init code)
- BIOS
- BIOS Option ROMs

Measurement Phase II (TBOOT, OS, Docker Engine...)
- Boot loader
- uCode (evals SINIT ACM)
- SINIT ACM (measures OS Kernel, initrd)
- Tboot-xm (agent in initrd) measures DockerEngine, other components

Source: Intel
Scheduler/Cluster Manager/Policy Manager

Who Verifies the Docker Host Trust?

Examples
- OpenStack
- Docker Swarm
- Kubernetes
- Mesos
- Fleet

Principles Of Operation
- Cluster Manager determines best hosts in the cluster, based on utilization, type, location compliance, etc.
- (for this host list) Cluster Manager verifies Host Integrity with the Attestation Authority.
- Attestation Authority responds with Attestation Reports for the Hosts
- Cluster Manager picks best Server that has the Integrity and instantiates Containers.
Trusted Docker Containers & VMs with OpenStack

1. Horizon/API Server: Initiate Launch of Image (with Hypervisor_Type Property)
2. Nova Scheduler: ImageProp Filter excludes Hosts that don’t meet Image Hypervisor Type.
3. Nova Scheduler: Runs Trust/Location Filter to identify Trusted Host (for VM or Docker Container)
4. Attestation Authority: Challenges Host to Attest. Provides Signed Attestation Report to Scheduler to use. – Identifies Trusted Host for VMs or Docker Containers.
OpenStack changes

1. Add hypervisor_type property to images
   Value=qemu for VM images
   Value=docker for docker images
2. Activate ImageProperties filter
   filters out hosts that don’t match Value from Image Hypervisor Type
3. Activate Trust filter in openstack scheduler and trust properties in images
4. Configure Nova-compute to use docker driver.
   DEFAULT] compute_driver =
   novadocker.virt.docker.DockerDriver
   Steps at: https://wiki.openstack.org/wiki/Docker

Docker Specific changes

For Docker Image Integrity:

- Modified Docker daemon to intercept container launch request and call measurement agent before launch
- Manifest/trust-policy created and associated with each Docker layer

Infrastructure related changes

- TXT/TPM hardware;
- TXT/TPM activation on the clusters
- Attestation Server is setup
Demo
Summary & Call to Action

- Intel’s focus: Enable Hardware-based Integrity assurance for Docker Containers – Trusted Docker Containers
  - Enabling the same model as we have done for VMs.
  - Intel TXT and Attestation Software becomes the foundation for asserting Docker Host Integrity.
  - Intel iKGT (Kernel Guard Technology) can help in runtime integrity protection of the Linux Kernel.

- OpenStack can launch VMs and Containers with the extensions that are already mainstream (Trusted Compute Pools)

- Get engaged, get started with Trusted VMs and OpenStack. Extensions to OpenStack for Trusted Docker containers, will be available in Q3 timeframe.

- iKGT is available now on 01.org. Download it and try it out.
Q & A