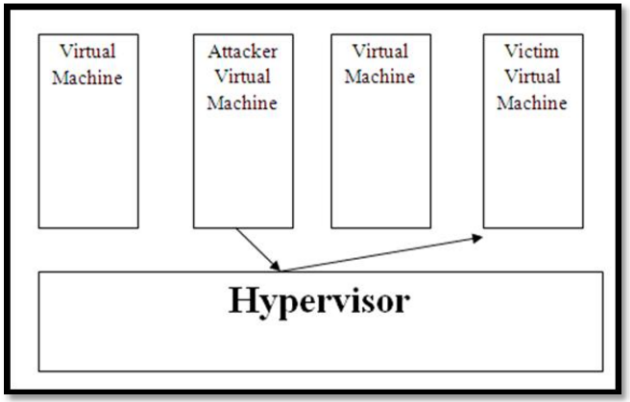
Supporting Security Sensitive Tenants in a Bare-Metal Cloud

Amin Mosayyebzadeh, A. Mohan, S. Tikale,
M. Abdi, N. Schear, C. Munson, T. Hudson,
L. Rudolph, G. Cooperman, P. Desnoyers, O. Krieger



Problems with Existing Cloud Offerings

1. A Virtualization-based shared hardware offering is prone to side-chanel, covert-channel, hyperjacking, etc.



Problems with Existing Cloud Offerings

OpenStack software controls large pools of compute, storage, and networking resources throughout a datacenter, managed through a dashboard or via the OpenStack API. OpenStack works with popular

kubernetes

openstack.

As Kubernetes Nears 2 Million Lines of Code, Commit Velocity and ...

https://globenewswire.com/.../As-Kubernetes-Nears-2-Million-Lines-of-Code-Commit-...

Dec 11, 2018 - Most common emails (**size** ~ log of #) ... The number of API endpoints exported in the **Kubernetes codebase** is stabilizing at 16,000 which ...

2. Cloud orchestration softwares have huge trusted computing base (TCB) and a massive attack surface

Problems with Existing Cloud Offerings

3. Limited visibility and control over implementation and operation; tenants needs to trust non-maliciousness and competence of the provider

Problems with Existing Cloud Offerings

4. Adheres to one-size-fits-all security solutions for operational efficiency

Problems with Existing Cloud Offerings

Bare-Metal clouds overcome the problems faced by virtualized offerings but are prone to firmware-based attacks and still possess other public cloud problems (2, 3 and 4)

Bare metal cloud servers are vulnerable to attack: Eclypsium

By David Heath

Research by security firm Eclypsium shows that vaca… wiped by hosting providers and may be used as an…

Data Centre ▸ Cloud

After IBM SoftLayer fails to scrub bare-metal box firmware of any lurking spies, alarm raised over cloud server security

Don't just grin and bare it: Check your provider wipes mobo before redeployment

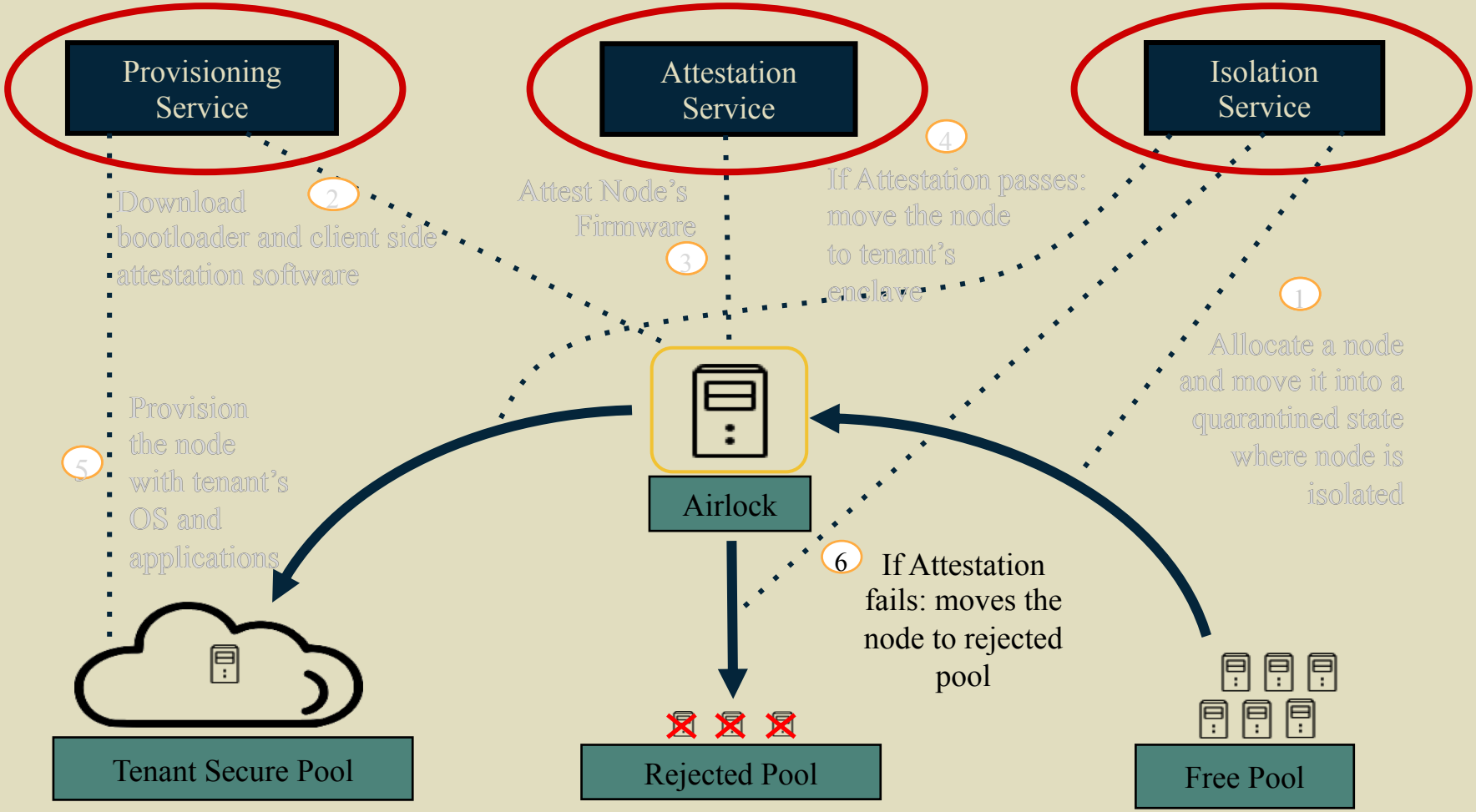
By Shaun Nichols in San Francisco 26 Feb 2019 at 08:47

26 SHARE

Is is Possible to Architect a Cloud that...

- Is appropriate for even the most security-sensitive tenants?
- Doesn’t require the tenants to fully trust the provider?
- Doesn’t impact tenants with less stringent security requirements or who are willing to trust the provider for their security?

Bolted: An Architecture for Secure Bare-Metal Cloud Service


Provisioning Service
Attestation Service
Isolation Service
1
Allocate a node and move it into a quarantined state where node is isolated
2
Download bootloader and client side attestation software
3
Attest Node's Firmware
4
If Attestation passes: move the node to tenant's enclave
5
Provision the node with tenant's OS and applications
6
If Attestation fails: moves the node to rejected pool
Airlock
Tenant Secure Pool
Rejected Pool
Free Pool

Bolted Implementation

Answering different security needs of different tenants

Minimizing the trust in the provider

Network Encryption

IPsec

- To protect against provider
- Securely bootstrapped through Keylime

LUKS
Linux Unified Key Setup

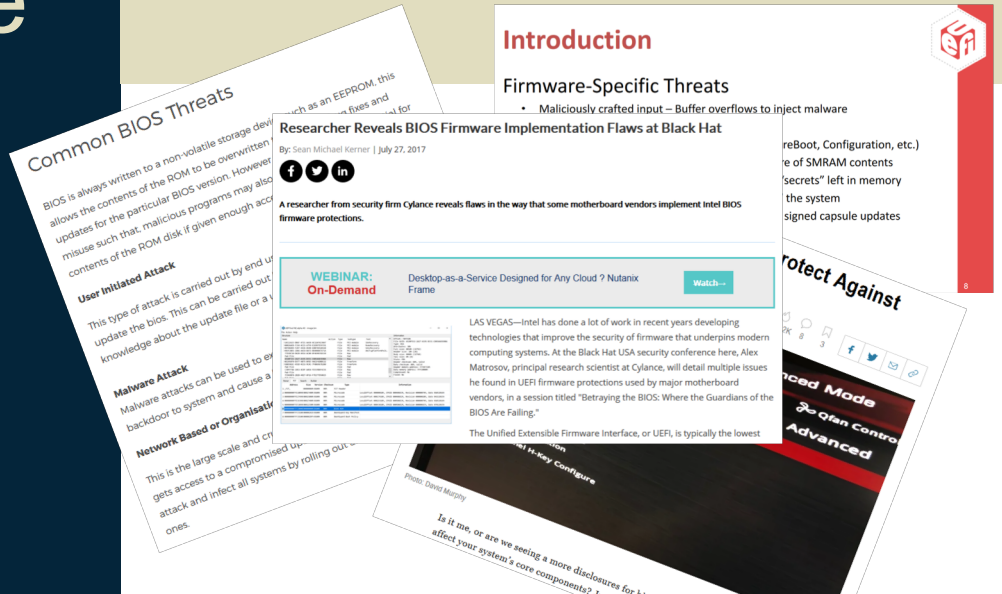
Disk Encryption

- Limits the access to tenants' remotely stored data including the provider
- Encrypted data on local disk with ephemeral keys stored only in memory
- Securely bootstrapped through Keylime

What about the firmware?

What about the firmware?

- BIOS, UEFI, … are huge
  - Vulnerable to attacks; potentially enabling tenants to modify FW
  - No way for tenant to inspect FW

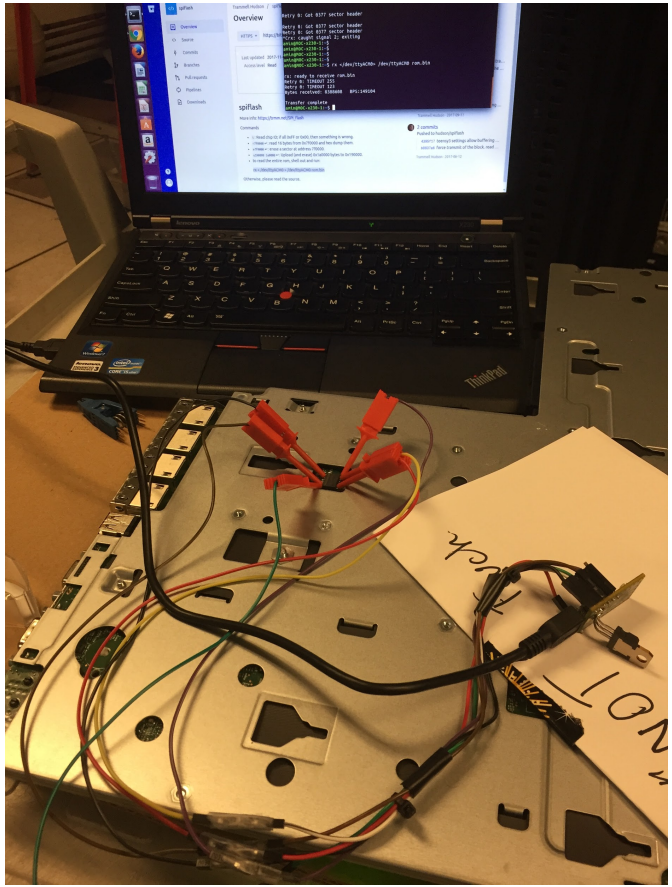


What about the firmware?

- LinuxBoot: A stripped down linux firmware
  - Open source
  - Deterministically built

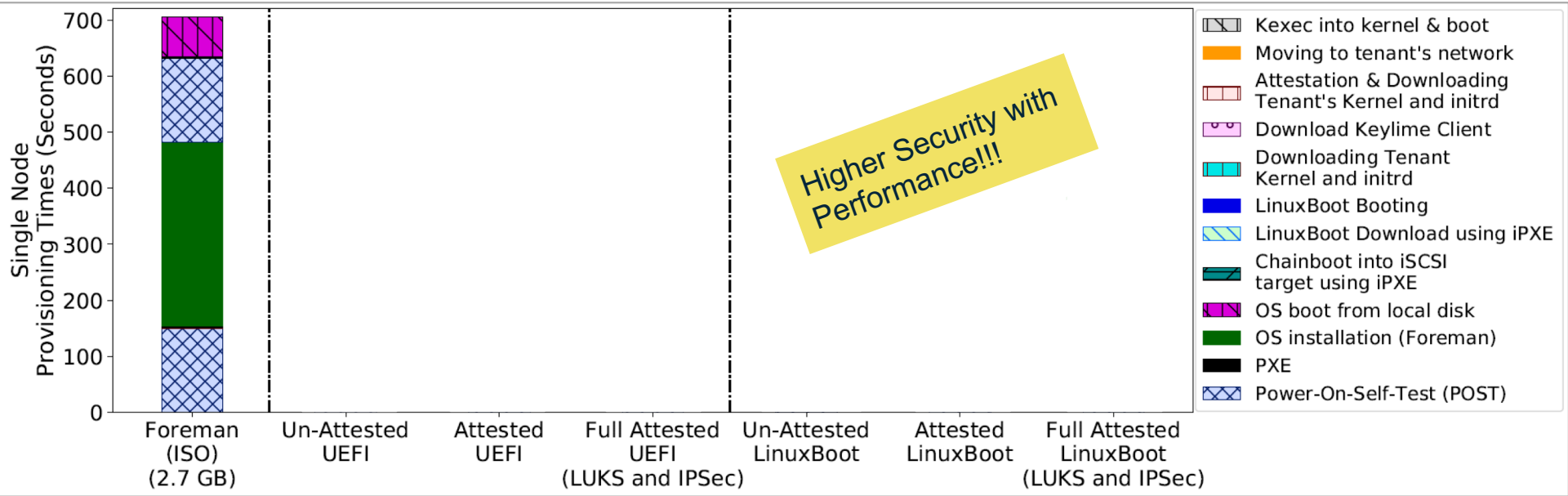
What about the firmware?

- Bolted works with either UEFI or LinuxBoot
  - With UEFI, download LinuxBoot runtime (Heads) as execution environment for Keylime client
  - We have burned Heads into a small number of servers

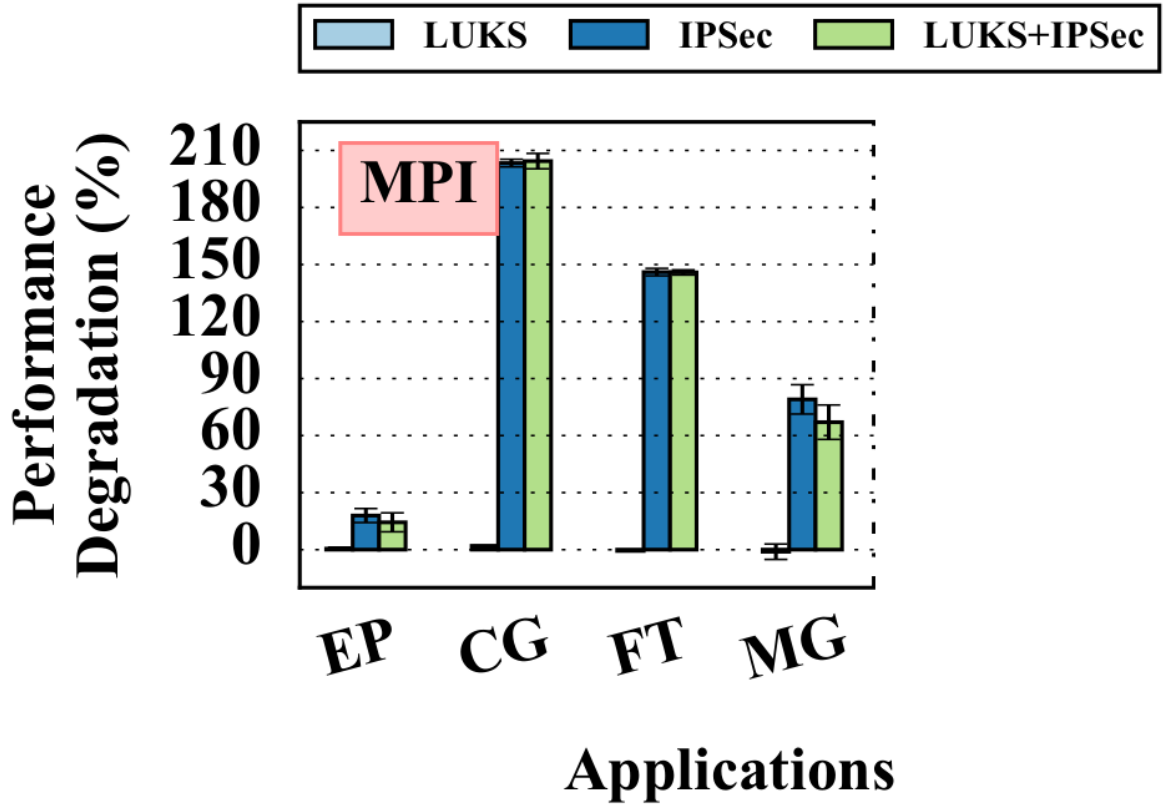


Boot Time

- Dell R630 server
  - 2 Xeon E5-2660 v3 2.6 GHz
  - 256 GB RAM



The Cost of Minimizing Trust on the Provider



Open Source Code

- HIL
  - https://github.com/cci-moc/hil
- BMI
  - https://github.com/cci-moc/ims
- Keylime
  - https://github.com/mit-ll/python-keylime
- LinuxBoot
  - https://github.com/osresearch/linuxboot

Concluding Remarks

- It is possible to measure all components needed to boot a server securely
- Small Microservices; most can be deployed by tenants and not in TCB
  - Minimizing trust in the provider
  - Provider does not need to deploy a global security policy
- Supporting even the most security sensitive tenants
- Tenants can make the cost/performance/security tradeoff