Supporting Security Sensitive Tenants in a Bare-Metal Cloud

Background

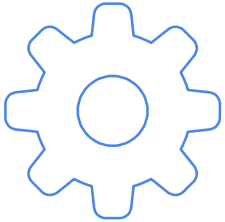
- ~70% of businesses utilize cloud
- 60% of F500 companies experienced a compromised cloud account last year
- Biggest cyber threat of 2020

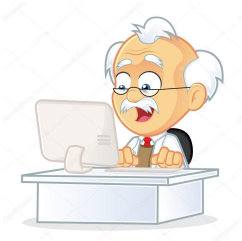
Tenants

Security Sensitive

Goldman Sachs

Security Insensitive





Security Sensitive Tenants

- Prepared to pay
- Own security arrangements
- Minimize trust in provider

Security problems with existing clouds

- Virtualized clouds
- Huge trusted computing base(TCB)
- One-size-fits-all
- Limited visibility and control

What is a bare-metal node?

- No virtualization
- Single tenant
- Tenant optimizes the server
- Avoid noisy-neighbor effect
- Efficient billing model

Bare Metal Clouds: Security Limitations

- Large parts of codebase in TCB.
- One-size fits all approach to security
- Trust the provider
- Can’t verify the firmware installed

Now introducing:

Bolted

Key goals of Bolted

- To minimize trust in provider
- Tenants with security expertise implement functionality themselves
- To enable tenants to make their own cost/performance/security tradeoffs

Security Assumptions

- Provider gives physical security
- Servers equipped with Trusted Platform Module

Components

- Isolation service
- Secure Firmware
- Provisioning Service
- Attestation Service

Isolation Service - Hardware Isolation Layer

- Allocates nodes, creates networks
- Controls provider’s switches
- Provides VLAN based isolation
- Must be deployed by provider
- Invoked by tenant

Provisioning Service- Bare Metal Imaging

- Responsible for provisioning servers
- Tenants can run their own
- Allows for diskless provisioning
- Only fetches part of the image that it uses

Attestation Service - Keylime

- Provides attestation for software
- Security sensitive tenants run continuous attestation
- Handles network and disk key distribution

LUKS and IPSec

Linux Unified Key Setup

- Disk encryption for Linux
- Keylime supported auto-configuration
- Low overhead

IPSec

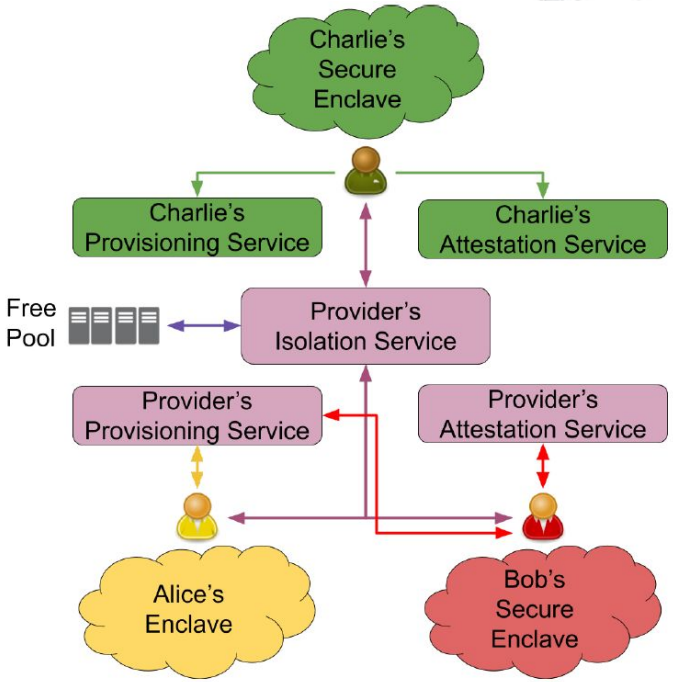
- Used for network encryption
- Higher overhead

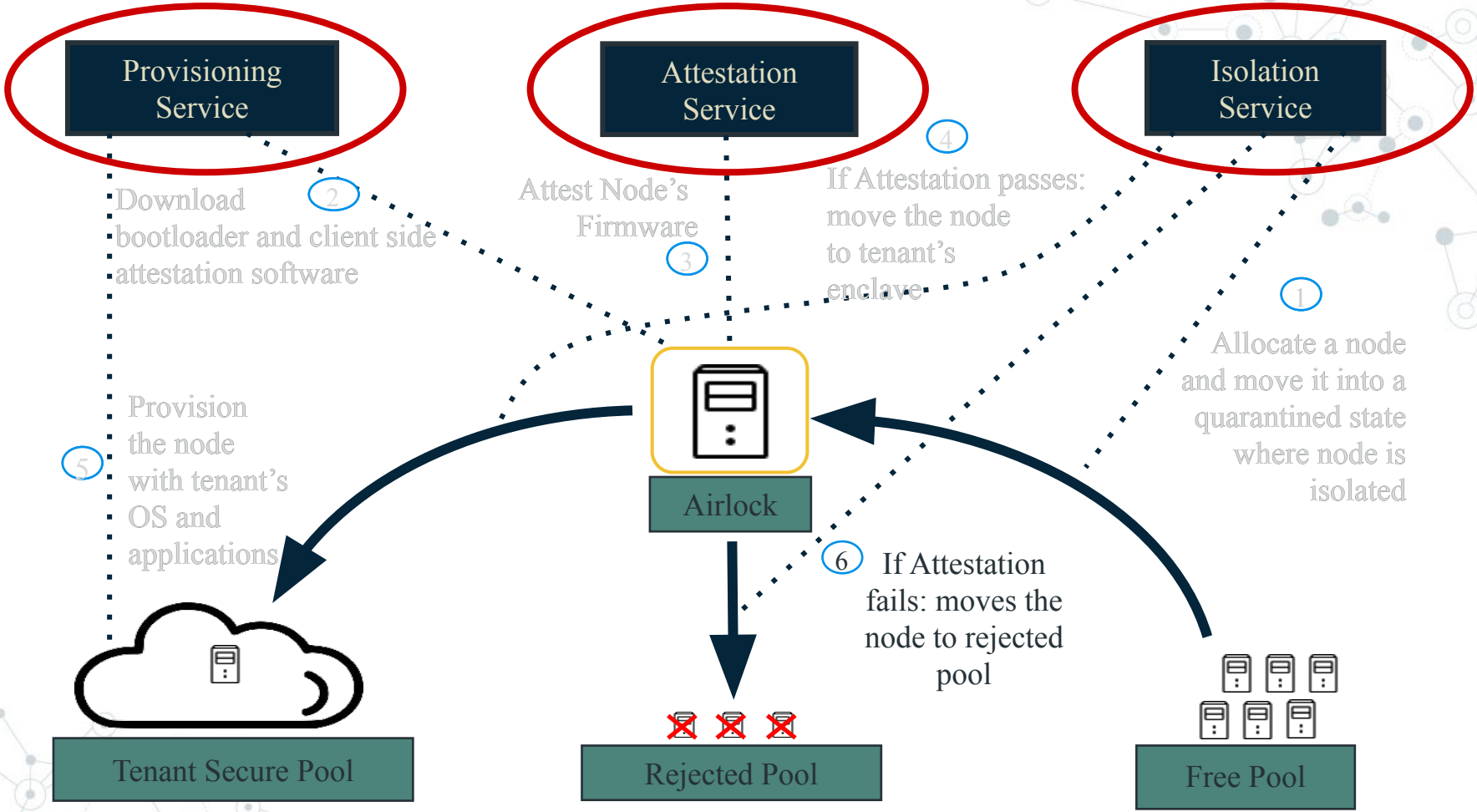
Secure Firmware - Linuxboot

- Open source
- Deterministically built
- Ensures memory scrub
- Allow attestation agent to execute
- Minimal build of Linux
- Faster to POST

Use cases

- Alice (HPC): Maximizes performance and minimizes cost; does not care about security
- Bob (Developer): Don't trust other tenants but is willing to trust the provider
- Charlie (NSA): Not only does not trust other tenants but wants to minimize his trust in the provider

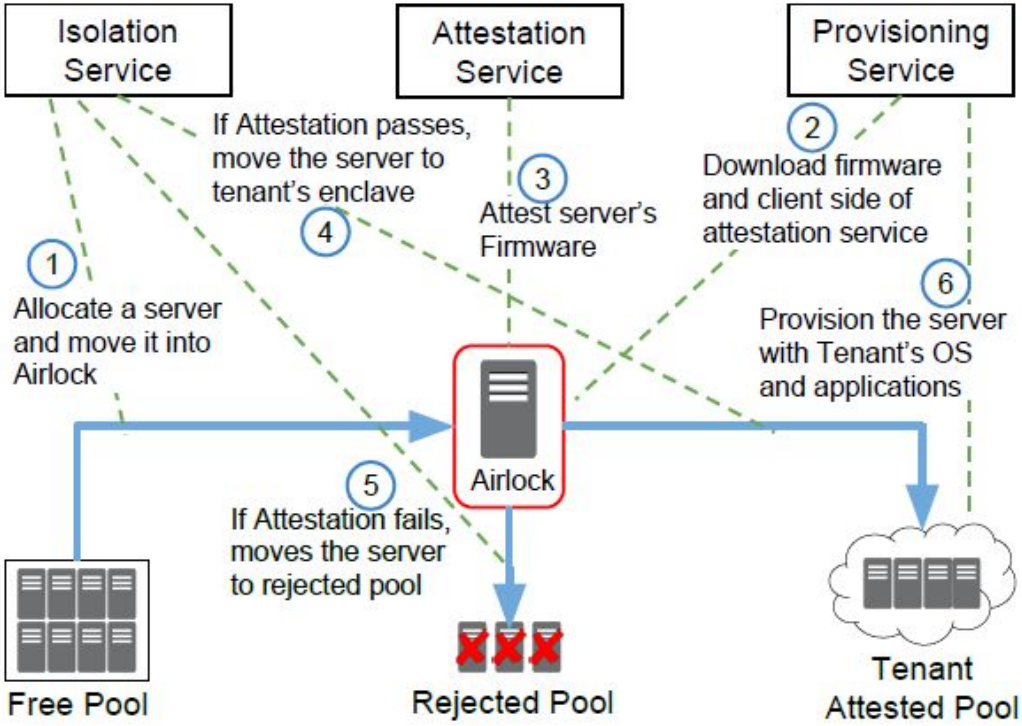



Provisioning Service
Attestation Service
Isolation Service
2
Download bootloader and client side attestation software
Attest Node's Firmware
3
4
If Attestation passes: move the node to tenant's enclave
1
Allocate a node and move it into a quarantined state where node is isolated
5
Provision the node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

Bolted's Architecture

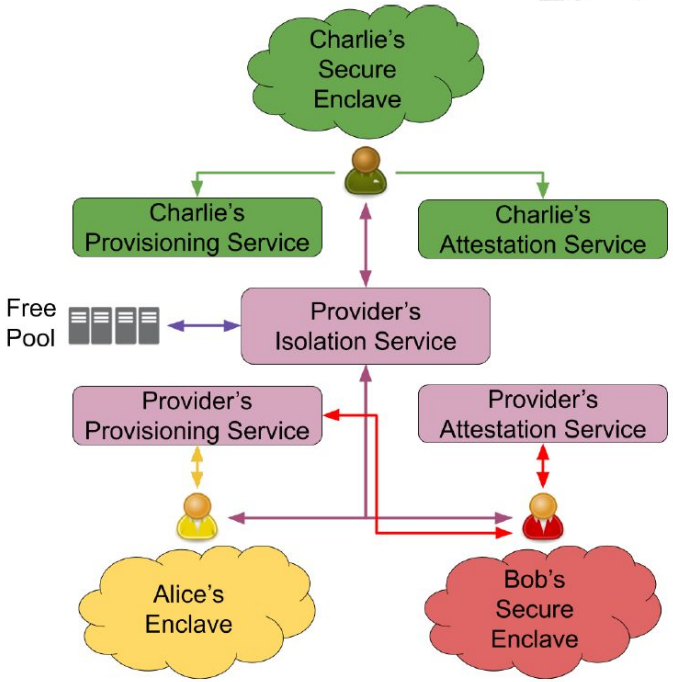
Key Components:

- Isolation Service
- Provisioning Service
- Attestation Service
- Secure Firmware



Use cases

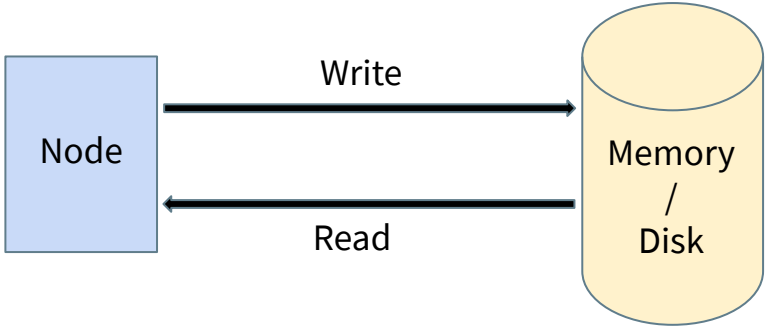
- Alice (HPC): Maximizes performance and minimizes cost; does not care about security
- Bob (Developer): Don't trust other tenants but is willing to trust the provider
- Charlie (NSA): Not only does not trust other tenants but wants to minimize his trust in the provider



Bolted implementation

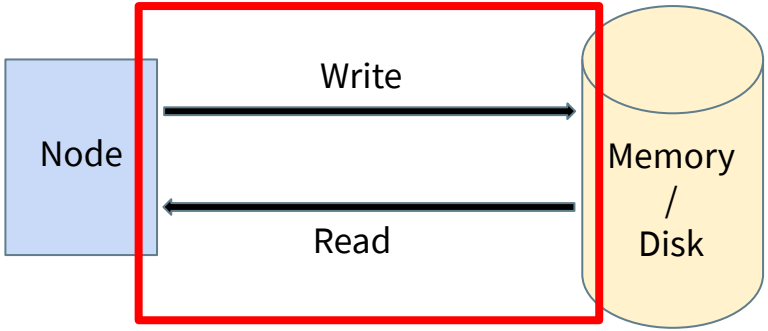
- Bare Metal Imaging (BMI) - Provisioning Service
- Hardware Isolation Layer (HIL) - Isolation Service
- Keylime - Attestation Service
- LinuxBoot - Firmware to speed up provisioning
- LUKS - Memory/Disk Encryption
- IPSec - Network Encryption

Network Encryption vs Memory Encryption



Network Encryption vs Memory Encryption

Network Encryption (IPSec)

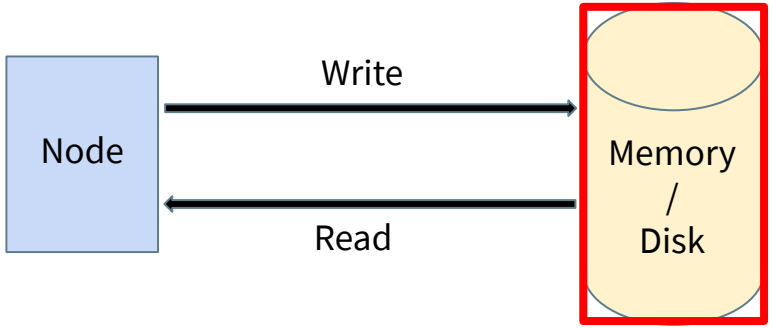


Protects against Man-in-the-middle

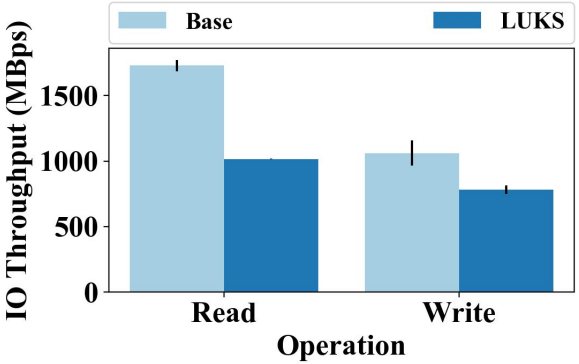
Also applies to Node-Node communication (i.e. parallel programming - MPI)

Network Encryption vs Memory Encryption

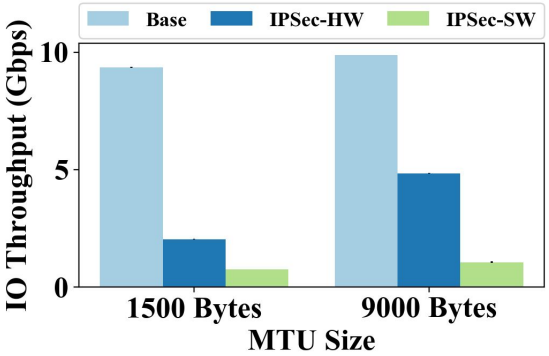
Memory Encryption (LUKS)



Cost of encryption



Disk Encryption

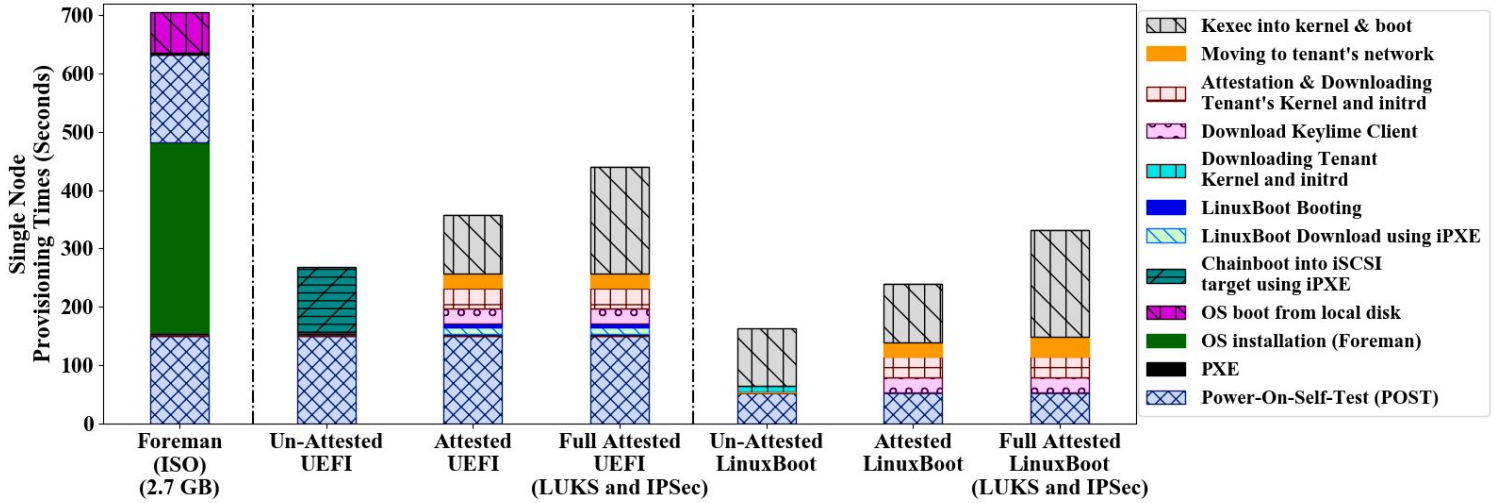


Network Encryption

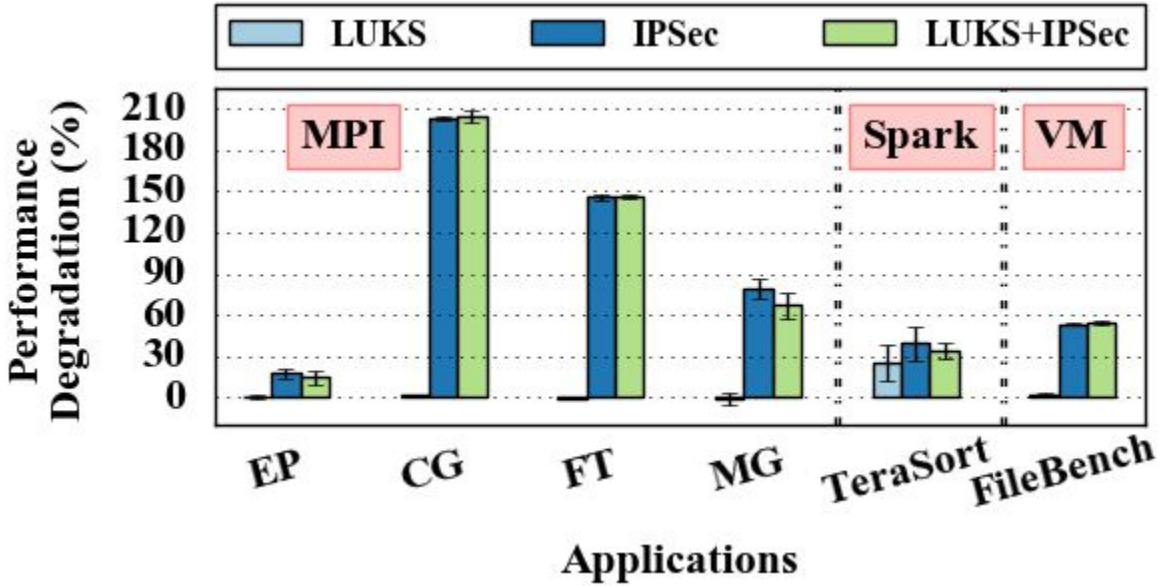
LUKS: a disk encryption specification originally intended for Linux

IPSec: an Internet Engineering Task Force (IETF) standard suite of protocols between 2 communication points across the IP network that provides network encryption

Provisioning time



Application performance



Concluding remarks

- Minimize trust tenants need to place in the provider
- Supporting even the most security sensitive tenants
- Tenants can make the performance/security tradeoff

Thank you!
Any questions?