



Verifying Cloud Integrity: Making the cloud do the dirty work



Joshua Schiffman, Thomas Moyer, Hayawardh Vijayakumar,
Trent Jaeger and Patrick McDaniel

Introduction

Customers are increasingly concerned that cloud computing platforms are unable to protect their security-critical data. Promises of “hardened code,” armed guards, and SLAs have done little to allay fears in light of growing threats to virtualized platforms. While recent work has enabled remote parties to verify system integrity, it is not clear how a practical trusted cloud can be designed to allow its customers to verify it.

Problem: Trusting Public Clouds

Clouds offer multi-tenant systems to host numerous VMs on the same physical system thereby maximizing utilization of hardware. The threat of misconfigured VMMs that allow rogue VMs to compromise the VMM and other hosted VMs makes the security of these platforms uncertain.

While encryption approaches that protect data stored on the cloud are becoming popular, they do not address the issue of ensuring the integrity of cloud computations and the data produced by those processes.

Challenges

Integrity Measurement: Trusted computing hardware enables remote parties to inspect the system configurations, but requires *a priori* knowledge of high integrity configurations and a PKI for managing key identities.

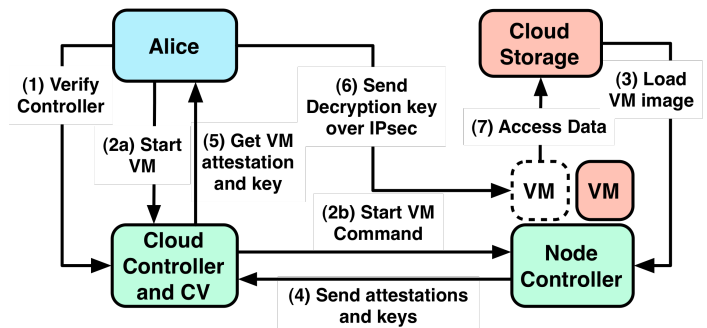
Integrity Criteria: Customer may have different integrity requirements. A solution must provide proofs that are comparable to various criteria.

Complexity: Clouds contain many multi-layered systems which affect each other’s integrity. A verification system must generate proofs that represent all dependents systems efficiently.

Solution: Cloud Verifier

We propose a more *transparent* cloud design that enables customers to build trust in the specific pieces of the cloud they depend upon. To that end, we introduce the **Cloud Verifier (CV)**, a simple and verifiable component that continually verifies the integrity of VMMs in the cloud based on a public **integrity criteria**. Customers are able to build transitive trust in the core service of the cloud by first verifying the CV and then its criteria against their own.

Once verified, the CV sends an IPsec key bound to the VMM’s integrity state that hosts the customer’s VM. The customer may poll the CV to check if the key is still valid and thus the VMM’s integrity. This key is used to establish a trusted connection to the VM so that the owner may provide keys to access encrypted data stored on the cloud.



Preliminary Evaluation

We built a proof of concept cloud using the Eucalyptus open source cloud platform to evaluate our design. We leverage previous work in installing verifiable VMMs to build the verification framework and designed the CV as a service in the cloud’s web frontend that users access. Our CV is able to handle 7, 000 requests per second and introduced only a small one second delay for verification before a VM is launched.