USER SECURITY GUARANTEES IN PUBLIC CLOUDS

**K.Ramkumar, Assistant Professor, Department of Computer Science,

Bharathiyar College of Engineering and Technology, Karaikal

M.Uma maheswary**,II year M.Tech Computer Science,

Bharathiyar College of Engineering and Technology, Karaikal

ABSTRACT

The infrastructure cloud service (IAAS) model offers improved resource flexibility and availability, where tenants – insulated from the minutiae of hardware maintenance – rent computing resources to deploy and operate complex systems. Large-scale services running on IAAS platforms demonstrate the viability of this model; nevertheless, many organizations operating on sensitive data avoid migrating operations to IAAS platforms due to security concerns. In this paper, we describe a framework for data and operation security in IAAS, consisting of protocols for a trusted launch of virtual machines and domain-based storage protection. We continue with an extensive theoretical analysis with proofs about protocol resistance against attacks in the defined threat model. The protocols allow trust to be established by remotely attesting host platform configuration prior to launching guest virtual machines and ensure confidentiality of data in remote storage, with encryption keys maintained outside of the IAAS domain. Presented experimental results demonstrate the validity and efficiency of the proposed protocols. The framework prototype was implemented on a test bed operating a public electronic health record system, showing that the proposed protocols can be integrated into existing cloud environments.

Keywords: *Security; Cloud Computing; Storage Protection; Trusted Computing: Remote storage: Virtual Machine*

INTRODUCTION

Cloud computing has progressed from a bold vision to massive deployments in various application domains. However, the complexity of technology underlying cloud computing introduces novel security risks and challenges. Threats and mitigation techniques for the IaaS model have been under intensive scrutiny in recent years while the industry has invested in enhanced security solutions and issued best practice recommendations . From an end-user point of view the security of cloud infrastructure implies unquestionable trust in the cloud provider, in some cases corroborated by reports of external auditors. Propose a set of protocols for trusted launch of virtual machines (VM) in IaaS, which provide tenants with a proof that the requested VM instances were launched on a host with an expected software stack.

While support data encryption at rest is offered by several cloud providers and can be configured by tenants in their VM instances, functionality and migration capabilities of such solutions are severely restricted. In most cases cloud providers maintain and manage the keys necessary for encryption and decryption of data at rest. This

further convolutes the already complex data migration procedure between different cloud providers, disadvantaging tenants through a new variation of vendor lock-in.

In this paper we present DBSP (domain-based storage protection), a virtual disk encryption mechanism where encryption of data is done directly on the compute host, while the key material necessary for re-generating encryption keys is stored in the volume metadata. This approach allows easy migration of encrypted data volumes and withdraws the control of the cloud provider over disk encryption keys. In addition, DBSP significantly reduces the risk of exposing encryption keys and keeps a low maintenance overhead for the tenant – in the same time providing additional control over the choice of the compute host based on its software stack. We focus on the Infrastructure-as-a-Service model – in a simplified form, it exposes to its tenants a coherent platform supported by compute hosts which operate VM guests that communicate through a virtual network.

I Extend previous work applying Trusted Computing to strengthen IaaS security, allowing tenants to place hard security requirements on the infrastructure and maintain exclusive control of the security critical assets. Ipropose a security framework consisting of three buiding blocks:

- Protocols for trusted launch of VM instances in IaaS;
- Key management and encryption enforcement functions for VMs, providing transparent encryption of persistent data storage in the cloud;
- Key management and security policy enforcement by a Trusted Third Party (TTP);

LITERATURE REVIEW

1. "Seeding Clouds With Trust Anchors"

Finding that customers with security-critical data processing needs are beginning to push back strongly against using cloud computing. In cloud computing, a vendor runs their computations upon cloud provided VM systems. Customers are worried that such host systems may not be able to protect themselves from attack, ensure isolation of customer processing, or load customer processing correctly. To provide assurance of data processing protection in clouds to customers, we advocate methods to improve cloud transparency using hardware-based attestation mechanisms. We find that the centralized management of cloud data centers is ideal for attestation frameworks, enabling the development of a practical approach for customers to trust in the cloud platform. Specifically, we propose a cloud verifier service that generates integrity proofs for customers to verify the integrity and access control enforcement abilities of the cloud platform that protect the integrity of customer's application VMs in IaaS clouds. While a cloud-wide verifier service could present a significant system bottleneck, we demonstrate that aggregating proofs enables significant overhead reductions. As a result, transparency of data security protection can be verified at cloud-scale.

2. "Domain based storage protection with secure access control for the cloud"

Cloud computing has evolved from a promising concept to one of the fastest growing segments of the IT industry. However, many businesses and individuals continue to view cloud computing as a technology that risks exposing their data to unauthorized users. We introduce a data confidentiality and integrity protection mechanism for Infrastructure-as a-Service (IAAS) clouds, which relies on trusted computing principles to provide transparent storage isolation between IAAS clients. We also address the absence of reliable data sharing mechanisms, by providing an XML-based language framework which enables clients of IAAS clouds to securely share data and clearly define access rights granted to peers. The proposed improvements have been prototyped as a code extension for a popular cloud platform.

3. "Secure and efficient access to outsourced data"

Providing secure and efficient access to large scale outsourced data is an important component of cloud computing. In this paper, we propose a mechanism to solve this problem in owner-write-users-read applications. We propose to encrypt every data block with a different key so that flexible cryptography-based access control can be achieved. Through the adoption of key derivation methods, the owner needs to maintain only a few secrets. Analysis shows that the key derivation procedure using hash functions will introduce very limited computation overhead. We propose to use over-encryption and/or lazy revocation to prevent revoked users from getting access to updated data blocks. We design mechanisms to handle both updates to outsourced data and changes in user access rights. We investigate the overhead and safety of the proposed approach, and study mechanisms to improve data access efficiency.

4. "Security aspects of e-health systems migration to the cloud"

As adoption of e-health solutions advances, new computing paradigms - such as cloud computing - bring the potential to improve efficiency in managing medical health records and help reduce costs. However, these opportunities introduce new security risks which can not be ignored. Based on our experience with deploying part of the Swedish electronic health records management system in an infrastructure cloud, we make an overview of major requirements that must be considered when migrating e-health systems to the cloud. Furthermore, we describe in-depth a new attack vector inherent to cloud deployments and present a novel data confidentiality and integrity protection mechanism for infrastructure clouds. This contribution aims to encourage exchange of best practices and lessons learned in migrating public e-health systems to the cloud.

PROPOSED SYSTEM

In this proposed system a "Trusted Cloud Compute Platform" (TCCP) to ensure VMs are running on a trusted hardware and software stack on a remote and initially untrusted host. To enable this, a trusted coordinator stores the list of attested hosts that run a "trusted virtual machine monitor" which can securely run the client's VM.

Trusted hosts maintain in memory an individual trusted key use for identification each time a client launches a VM.

The paper presents a good initial set of ideas for trusted VM launch and migration, in particular the use of a trusted coordinator. A limitation of this solution is that the trusted coordinator maintains information about all hosts deployed on the IAAS platform, making it a valuable target to an adversary who attempts to expose the public IAAS provider to privacy attacks .host, beyond the initial launch arguments

A decentralized approach to integrity attestation is adopted to address the limited transparency of IAAS platforms and scalability limits imposed by third party integrity attestation mechanisms. The authors describe a trusted architecture where tenants verify the integrity of IAAS hosts through a trusted cloud verifier proxy placed in the cloud provider domain. Tenants evaluate the cloud verifier integrity, which in turn attests the hosts. Once the VM image has been verified by the host and countersigned by the cloud verifier, the tenant can allow the launch.

A trusted VM launch (TL) protocol which allows tenants – referred to as domain managers – to launch VM instances exclusively on hosts with an attested platform configuration and reliably verify this.

Domain-based storage protection protocol to allow domain managers store encrypted data volumes partitioned according to administrative domains**.**

List of attacks applicable to IAAS environments and use them to develop protocols with desired security properties, perform their security analysis and prove their resistance against the attacks.

The implementation of the proposed protocols on an open-source cloud platform and present extensive experimental results that demonstrate their practicality and efficiency**.**

IMPLEMENTATION AND RESULTS

Test bed Architecture

We describe the infrastructure of the prototype and the architecture of a distributed EHR system installed and configured over multiple VM instances running on the test bed.

Infrastructure Description:

The test bed resides on four Dell Power Edge R320 hosts connected on a Cisco Catalyst 2960 switch with 801.2q support. The prototype IAAS includes one "controller" running essential platform services (scheduler, PKI components, SDN control plane, VM image storage, etc.) and three compute hosts running the VM guests. Transactions on Cloud Computing reflects three larger domains of the application-level deployment (front-end, back-end and database components) in three virtual LAN (VLAN) networks. The compute hosts use libvirt6 for virtualization functionality. We modified libvirt 1.0.2 and used the "libvirthooks" infrastructure to implement the SC for the TL and DBSP protocols. SC unlocks the volumes on compute hosts and interacts with the TPM and TTP .It uses a generic server architecture where the SC daemon handles each request in a separate process. An inter process communication (IPC) protocol defines the types of messages processed by the SC. The IPC protocol uses synchronous calls with several types of requests for the respective SC operations; the response contains the exit code and response data.

Trusted Third Party

Application Description:

This system contains one client VM, two front-end VMs, two back-end VMs, a database VM and an auxiliary external database VM. Six of the VM instances operate on Microsoft Windows Server 2012 R2, with one VM running the client application operates on Windows 7. Load balancing functionality provided by the underlying IaaS allots the load among front-end and back-end VM pairs. The hosts of the cluster are compatible with the TL protocol, which allows an infrastructure administrator to perform a trusted

Performance evaluation

0 20 40 60 80 100

VM Launch number

10000

12000

14000

16000

18000

20000

22000

Duration, ms

Trusted VM launch

Vanilla VM launch

Overhead induced by the TL protocol during VM instantiations. Trusted launch: Figure 6 shows the duration of a VM launch over 100 successful instantiations: the TL protocol extends the duration of the VM instantiation (which does not include the OS boot time) on average by 28%. However, in our experiments we have used a minimalistic VM image (13.2 MB), based on CirrOS 7, while launching larger VM images takes significantly more time and proportionally reduces the overhead induced by TL. DBSP Processing time: Table 1 shows a breakdown of the time required to process a storage unlock request, an average of 10 executions. Processing a volume unlock request on the prototype returns in _2.714 seconds; however, this operation is performed only when attaching the volume to a VM instance and does not affect the subsequent I/O operations on the volume. A closer view highlights the share of the contributing components in the overall overhead composition. Table 1 clearly shows that the TPM unseal operation lasts on average _2.7 seconds, or 99.516% of the execution time. According to Section 4.2, in this prototype we use TPMs v1.2, since a TPM v2.0 is not available on commodity platforms at the time of writing. Given that the vast majority of the execution time is spent in the TPM unseal operation, implementing the protocol with a TPM v2.0 may yield improved results. DBSP Encryption Overhead: Next, we examine the processing overhead introduced by the DBSP protocol.

Transactions on Cloud Computing

13

presents the results of a disk performance benchmark obtained using IOmeter8. To measure the effect of background disk encryption with DBSP, we attached two virtual disks to a deployed server VM described in 6.1.2. The storage volumes were physically located on a different host and communicating over iSCSI. We ran a benchmark with two parallel workers on the plaintext and DBSP-encrypted volumes over 12 hours. Next, we disabled in the host BIOS the AES-NI acceleration, created and attached a new volume to the VM and reran the benchmark. This has produced three performance data result sets: plaintext, DBSP encryption and DBSP encryption with AES-NI acceleration.

It is visible that the measurements '4 KiB aligned (DBSP) with AES-NI' and '1 MiB (DBSP) with AES-NI' are roughly on par with the plaintext baseline: '4 KB aligned' and '1 MB'. The performance overhead induced by background encryption is at 1.18% for read IO and 0.95% for write IO. We can expect that this performance penalty will be further reduced as the hardware support for encryption is improved. Disk encryption without hardware acceleration ('4 KB aligned (DBSP)' and '1 MB (DBSP)') is significantly slower, as expected, with a performance penalty of respectively 49.22% and 42.19% (total IO). It is important to reemphasize that the runtime performance penalty is determined exclusively by the performance of the disk encryption subsystem. DBSP only affects the time required to unlock the volume when it is attached to the VM instance,

0
20
40
60
80
100
120
140
160
180
4 KB aligned 1 MB 4 KB aligned
(DBSP)
1 MB (DBSP) 4 KB aligned
(DBSP) w. AES-NI
1 MB (DBSP) w.
AES-NI
Ioops
Read Ioops
Write Ioops

Benchmarks results on identical drives: plaintext, with DBSP, with DBSP and AES-NI acceleration.

APPLICATION DOMAIN

The presented results are based on work in collaboration with a regional public healthcare authority to address some of its concerns regarding IaaS security. We have deployed the prototype described in Section 6, further extended by integrating a medication database, and evaluated it through end-user validation and performance tests. Our results demonstrate that it is both possible and practical to provide strong platform software integrity guarantees to IAAS tenants and efficiently isolate their data using established cryptographic tools. Platform integrity guarantees allow tenants to take better decisions on both workload migration to the cloud and workload placement within IAAS. This contrasts with the current, "flat" trust model, where all IAAS hosts declare the same – but unverifiable for the tenant – trust level.

CONCLUSION

From a tenant point of view, the cloud security model does not yet hold against threat models developed for the traditional model where the hosts are operated and used by the same organization. However, there is a steady progress towards strengthening the IaaS security model. In this work we presented a framework for trusted infrastructure cloud deployment, with two focus points: VM deployment on trusted compute hosts and domain-based protection of stored data. We described in detail the design, implementation and security evaluation of protocols for trusted VM launch and domain-based storage protection. The solutions are based on requirements elicited by a public healthcare authority, have been implemented in a popular open-source IaaS platform and tested on a prototype deployment of a distributed EHR system. In the security analysis, we introduced a series of attacks and proved that the protocols hold in the specified threat model. To obtain further confidence in the semantic security properties of the protocols, we have modeled and verified Finally, our performance tests have shown that the protocols introduce a insignificant performance overhead. This work has covered only a fraction of the IAAS attack landscape. Important topics for future work are strengthening the trust model in cloud network communications, data geolocation and applying searchable encryption schemes to create secure cloud storage mechanisms. Our results show that it is possible and practical to provide strong platform software integrity guarantees for tenants and efficiently isolate their data using established cryptographic tools. With reasonable engineering effort the framework can be integrated into production environments to strengthen their security properties.

REFERENCES

[1] N. Santos, K. P. Gummadi, and R. Rodrigues, "Towards trusted cloud computing," in Proceedings of the 2009 Conference on Hot Topics in Cloud Computing, HotCloud'09, (Berkeley, CA, USA), USENIX Association, 2009.

[2] J. Schiffman, T. Moyer, H. Vijayakumar, T. Jaeger, and P. McDaniel, "Seeding Clouds With Trust Anchors," in Proceedings of the 2010 ACM Workshop on Cloud Computing Security, CCSW '10, (New York, NY, USA), pp. 43–46, ACM, 2010.

[3] N. Paladi, A. Michalas, and C. Gehrmann, "Domain based storage protection with secure access control for the cloud," in Proceedings of the 2014 International Workshop on Security in Cloud Computing, ASIACCS '14, (New York, NY, USA), ACM, 2014.

[4] M. Jordon, "Cleaning up dirty disks in the cloud," Network Security, vol. 2012, no. 10, pp. 12–15, 2012.

[5] Cloud Security Alliance, "The notorious nine cloud computing top threats 2013," February 2013.

[6] A. Michalas, N. Paladi, and C. Gehrmann, "Security aspects of e-health systems migration to the cloud," in the 16th International Conference on E-health Networking, Application & Services (Healthcom'14), pp. 228–232, IEEE, Oct 2014.

[7] B. Bertholon, S. Varrette, and P. Bouvry, "Certicloud: a novel tpmbased approach to ensure cloud IaaS security," in Cloud Computing, 2011 IEEE International Conference on, pp. 121–130, IEEE, 2011. [8] M. Aslam, C. Gehrmann, L. Rasmusson, and M. Bjorkman, "Se- ¨ curely launching virtual machines on trustworthy platforms in a public cloud - an enterprise's perspective.," in CLOSER, pp. 511– 521, SciTePress, 2012.