

# User Security Guarantees in Public Infrastructure Clouds

A.Mahendra<sup>1</sup>, M. Shalima Sulthana<sup>2</sup>

<sup>1,2</sup>Dept of CSE, Guest Faculty in IIIT RGUKT RKVALLEY, Vempalli

**Abstract-** In this paper, we describe a framework for data and operation security in IaaS, consisting of protocols for trusted launch of virtual machines and domain-based storage protection. 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. 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.

**Index Terms-** Mobile ad hoc networks, query processing, routing, traffic, data replacement attack, node grouping.

## I. 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. A core enabling technology of IaaS is system virtualization [8], which enables hardware multiplexing and redefinition of supported hardware architectures into software abstractions. This redefinition is performed by the hypervisor, a software component that abstracts the hardware resources of the platform and presents a virtualized software platform where guest virtual machine (VM) instances can be deployed. In addition, the hypervisor also manages the I/O communication between VM instances and external components, including storage devices allocated to the VM instance. This is one of the vulnerable areas of IaaS environments since, as demonstrated in [6], improper allocation of block storage can lead to a

breach of data confidentiality. There is a clear need for usable and cost-effective cloud platform security mechanisms suitable for organizations that rely on cloud infrastructure. One such mechanism is platform integrity verification for compute hosts that support the virtualized cloud infrastructure several large cloud vendors have signaled practical implementations of this mechanism, primarily to protect the cloud infrastructure from insider threats and advanced persistent threats. We see two major improvement vectors regarding these implementations. First, details of such proprietary solutions are not disclosed and can thus not be implemented and improved by other cloud platforms. Second, to the best of our knowledge, none of the solutions provide cloud tenants a proof regarding the integrity of compute hosts supporting their slice of the cloud infrastructure. In this project, 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. The relevant security mechanism is encryption of virtual disk volumes, implemented and enforced at compute host level. 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. Tenants can choose to encrypt data on the operating system (OS) level within their VM environments and manage the encryption keys themselves. Cloud computing offers an important technique that is platform integrity verification that support the virtualized cloud infrastructure for hosts. Many of the cloud vendors have assembled and the judicious implementations of this mechanism. Cloud storage provides us with convenient, huge, and scalable storage at low cost, but data privacy is a major problem that prevents users from storing files on the cloud trust worthy. One way to improve privacy from data owner point of view is to encrypt the files before storing them onto the cloud and decrypt the files after downloading them. To safeguard the cloud infrastructure from corporate executive threats and advanced persistent threats, we tend to see a vast improvement vectors pertaining these implementations. Secondly, to the most effective of our information, none of the solutions provides cloud tenants a symbol concerning the integrity of figure hosts supporting their way of looking forward towards the cloud infrastructure.

## II LITERATURE SURVEY

The infrastructure cloud (IaaS) service model offers tenants with an improved assets flexibility and availability, where they are encased from the trivial details of hardware maintenance, rent computing resources to be utilized and operate complex systems. Many organizations work on delicate data to avoid relocation and replication of operations to IaaS platforms due to defense concerns. In this paper we use Order-preserving encryption (OBP) to achieve efficiency and security of data stored in a cloud, we also use another techniques like auditing protocols and third party assistance for the key management updates into a cloud by which the accessing becomes easier and the security is guaranteed and the violation of the data decreases. The industry has invested for strict security and they suggest best practices [5]. The main aim of this project is to through light on IaaS. It is in its simplified form, and exposes to its users that it is coherent platform as it supports the hosts of clouds who operates VM guests can communicate by a virtual network by providing the basic requirements

that are identified when an deployment of Distributed Electronic Health Record (EHR) system for an IaaS computing platform. In these years for IaaS the threats and migration has been under the intensive security [1][2][3]. At first, details of such principal solutions are not closed totally and may therefore not be enforced and enhanced by alternative cloud platforms [3].

Nuno Santos Krishna P. Gummadi Rodrigo Rodrigues.[6] given a mechanism reliably detects whether or not the host is running a platform implementation that the remote party trusts. These platforms will effectively secure a VM running in a single host. Antonis Michalas, Nicolae Paladi and Christian Gehrman.[7] aimed for a paperless medical system where patients and doctors are able to book appointments via the Internet, create electronic prescriptions and store their medical history in a central database, easily accessible from anyone with appropriate access rights. Patrick McDaniel, Kevin Butler Radu Sion, Erez Zadok, Kui Ren and Marianne Winslett.[8] There are longstanding concerns beginning in large-scale systems. A recent report ready for the chairman and ranking member of the senate Committee on independent agency and environmental Affairs [8] highlighted beginning united of 3key future technologies for securing our national crucial infrastructure.

### *Towards trusted cloud computing*

The design of a trusted cloud computing platform (TCCP). TCCP enables Infrastructure as a Service (IaaS) providers such as Amazon EC2 to provide a closed box execution environment that guarantees confidential execution of guest virtual machines. Moreover, it allows users to attest to the IaaS provider and determine whether or not the service is secure before they launch their virtual machines [1].

### *Seeding Clouds with Trust Anchors*

In this paper, we identify three main challenges that cloud providers face when generating proofs that can placate a

User's concerns:

- 1) That cloud vendors provide a proof of data security protection of their hosts and customer processing;
- 2) That such proofs have a clear meaning to cloud customers; and

3) That such proofs can be generated effectively and efficiently in a cloud computing environment [2].

#### *Domain Based Storage Protection with Secure Access Control for the Cloud*

Despite the variety of available open source cloud management platforms (e.g. Open Stack, Eucalyptus, Open Nebula), allocation of read-write permissions for shared data between collaborating tenants still remains an open problem. In this paper we address the outlined gap. We improve and extend previous work by adding capabilities to both grant access to data to other IaaS cloud clients and assign access permissions [3].

#### *Security aspects of e-health systems migration to the cloud*

In this paper, we will present current state of the art research in this field. We focused of several shortcomings current healthcare solutions and standards, particularly for platform security, privacy aspect and requirements which is a crucial aspect for the overall security of healthcare IT systems. [5]

A decentralized approach to integrity attestation is adopted by Schiffman et al. [2] 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. Our protocol maintains the VM launch traceability and transparency without relying on a proxy verifier residing in the IaaS. Furthermore, the TL protocol does not require additional tenant interaction to launch the VM on a trusted host, beyond the initial launch arguments. Platform attestation prior to VM launch is also applied in [7], which introduces two protocols – “TPM-based certification of a Remote Resource” (TCRR) and “Verify My VM”. With TCRR a tenant can verify the integrity of a remote host and establish a trusted channel for further communication. In “Verify My VM”, the hypervisor running on an attested host uses an emulated TPM to verify on-demand the integrity of running VMs. Our approach

is in many aspects similar to the one in [7] in particular with regard to host attestation prior to VM instance launch. We overcome this limitation and generalize the solution by adding a verification token, created by the tenant and injected on the file system of the VM instance only if it is launched on an attested cloud host.

In [8], the authors described a protocol for trusted VM launch on public IaaS using trusted computing techniques.

To ensure that the requested VM instance is launched on a host with attested integrity, the tenant encrypts the VM image (along with all injected data) with a symmetric key sealed to a particular configuration of the host reflected in the values of the platform configuration registers (PCR) of the TPM placed on the host. The proposed solution is suitable in trusted VM launch scenarios for enterprise tenants as it requires that the VM image is pre-packaged and encrypted by the client prior to IaaS launch. However, similar to [7], this prevents tenants from using commodity VM images offered by the cloud provider to launch VM instances on trusted cloud hosts. Furthermore, we believe that reducing the number of steps required from the tenant can facilitate the adoption of the trusted IaaS model. We extend some of the ideas proposed in [8], address the above limitations – such as additional actions required from tenants – and also address the requirements towards the launched VM instance and required changes to cloud platforms.

### III. PROPOSED MODEL

When providers are offering security enhancements such as protection of data at rest, end-users have limited or no control over such mechanisms. There is a clear need for usable and cost-effective cloud platform security mechanisms suitable for organizations that rely on cloud infrastructure. Traditional public auditing protocols, another important task of the Third-party assistance (TPA) is to check the integrity of the client’s files stored in cloud. The TPA does not know the real secret key of the client for 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 un-trusted 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 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. Tenants can choose to encrypt data on the operating system (OS) level within their VM environments and manage the encryption keys themselves. However, this approach suffers from several drawbacks: first, the underlying compute host will still have access encryption keys whenever the VM performs cryptographic operations; second, this shifts towards the tenant the burden of maintaining the encryption software in all their VM instances and increases the attack surface; third, this requires injecting, migrating and later securely withdrawing encryption keys to each of the VM instances with access to the encrypted data, increasing the probability than an attacker eventually obtains the key. Proposed system presents experimental results to demonstrate the validity and efficiency of the proposed protocols to overcome the drawbacks of existing system. A basic structure underlying a system, concept, prototype is implemented on a transparent and replicable testing of scientific theories, computational tools, and new technologies, operating a public electronic health record system, showing that the proposed protocols can be integrated into existing cloud environments. Threats and mitigation is another technique where its Blinding technique with holomorphic property to form the encryption algorithm to encrypt the secret keys held by the TPA. It makes our protocol secure and the decryption operation efficient. Meanwhile, the TPA can complete key.

#### IV METHODOLOGY

We now describe two protocols that constitute the core of this papers contribution. These protocols are successively applied to deploy a cloud infrastructure providing additional user guarantees of cloud host integrity and storage security. For protocol purposes, each domain manager, secure component and trusted third party has a public/private key pair (pk/sk).

The private key is kept secret, while the public key is shared with the community. We assume that during the initialization phase, each entity obtains a certificate via a trusted certification authority. We first describe the cryptographic primitives used in the proposed protocols, followed by definitions of the main protocol components.

##### *Cryptographic Primitives*

The set of all binary strings of length  $n$  is denoted by  $\{0,1\}^n$ , and the set of all finite binary strings as  $\{0,1\}^*$ . Given a set  $U$ , we refer to the  $i$ th element as  $u_i$ . Additionally, we use the following notations for cryptographic operations throughout the paper:

##### *Protocol Components Disk encryption subsystem:*

A software or hardware component for data I/O encryption on storage devices, capable to encrypt storage units such as hard drives, software RAID volumes, partitions, files, etc. We assume a software based subsystem, such as dm-crypt, a disk encryption subsystem using the Linux kernel Crypto API. 4.3 Trusted Launch Construction We now present our construction for the TL, with four participating entities: domain manager, secure component, trusted third party and cloud provider (with the „scheduler“ as part of it). TL comprises a public-key encryption scheme, a signature scheme and a token generator.

We now analyse the TL and DBSP protocols in the presence of an adversary. We prove the security of both schemes 4 Key retrieval is currently not covered in the security analysis due to space limitations through a theoretical analysis, showing that our protocols are resistant to the attacks presented in Section

Option a can only succeeds if ADV can break the mutual authentication in the secure channel setting. Given that the selected secure channel scheme is sound and  $\tau$  is sufficiently long and selected using a sound random generation process, the ADV fails to

break the last protocol step. Hence, as long as the secure channel protocol is sound, the overall protocol construction is also sound against this attack option. Option b can only succeed if the adversary either manages to guess a value  $\tau_0 = \tau$  when launching vm or manages to either obtain  $\tau$  when DMi launches vmi l or replace the association between  $\tau$  and vmi l with an association between  $\tau$  and vm when DMi launches vmi l, by attacking any of the protocol steps preceding the final mutual authentication step. A successful attack in this case has the probability  $\tau_0 = \tau$  equals to  $1/2^n$ , where n is the length of the token value and is infeasible if n is large enough. Below, we show why the adversary also fails with respect to the last option.

## V. IMPLEMENTATION AND RESULTS

We next describe the implementation of the TL and DBSP protocols followed by experimental evaluation 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.

**Performance evaluation 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  $\approx 2.714$  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  $\approx 2.7$  seconds, or 99.516% of the execution time. According to Section

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.

## VI. CONCLUSION

In this paper, we have proposed an system architecture about providing user security guarantees

in public infrastructure clouds and single keyword search scheme to search the encrypted data files efficient and also the data security over the cloud. However, some extensions are still possible of our current work remaining. In future, We would like to propose a multi-keyword search scheme as our OPE algorithm is a simple one, another extension is to find a powerful algorithm which will not harm the efficiency. 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. This work has covered only a fraction of the IaaS attack landscape. The additional concept is also implementing attribute based file sharing system.

## REFERENCE

- [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. Gehrman, "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. Gehrman, "Security aspects of e-health systems migration to the cloud," in the 16th International Conference on Ehealth Networking, Application & Services (Healthcom'14), pp. 228–232, IEEE, Oct 2014.
- [7] B. Bertholon, S. Varrette, and P. Bouvry, "Certicloud: a novel tpm based approach to ensure cloud IaaS security," in *Cloud Computing, 2011 IEEE International Conference on*, pp. 121–130, IEEE, 2011.
- [8] M. Aslam, C. Gehrman, L. Rasmussen, and M. Björkman, "Securely launching virtual machines on trustworthy platforms in a public cloud – an enterprise's perspective.," in *CLOSER*, pp. 511–521, SciTePress, 2012.
- [9] A. Cooper and A. Martin, "Towards a secure, tamper-proof grid platform," in *Cluster Computing and the Grid, 2006. CCGRID 06. Sixth IEEE International Symposium on*, vol. 1, pp. 8–pp, IEEE, 2006.
- [10] W. Wang, Z. Li, R. Owens, and B. Bhargava, "Secure and efficient access to outsourced data," in *Proceedings of the 2009 ACM workshop on Cloud computing security*, pp. 55–66, ACM, 2009.
- [11] D. Song, E. Shi, I. Fischer, and U. Shankar, "Cloud data protection for the masses," *IEEE Computer*, vol. 45, no. 1, pp. 39–45, 2012.