ISSN:0975-9646

Ensuring Security in Resource Centers of Cloud Environment by Using Trusted Pools

K.P.R KrishnaChaitanya, Dr. S. Srinivasu, K. Naresh Kumar

Department of CSE, Anurag Engineering College, Kodad, Andhra Pradesh, India.

Abstract **- Now a day's cloud computing is one of the emerging Technologies. This created a new era in the IT Industry With its Service Oriented Architecture [1] .As the services of Organization Increases manganese of these services also becoming one of the problems. Management of resources will create some of security issues which will question the quality of service. Intrusions are one of the main security problems in cloud resource centers. Cloud resource administrators are facing a lot of problems in tracing out the anomalous and instructors. This paper explains how to overcome this problem. Here we are given some levels of security for cloud centers, when the level increasing the security levels also increases. To ensure the correctness of users' data in the cloud, we propose an effective and flexible distributed scheme with two salient features, opposing to its predecessors. By utilizing the two main issues that we address in context of such response policies are that of policy matching, and policy administration. The identification of misbehaving server(s). Extensive security and performance analysis shows that the proposed scheme is highly efficient and resilient against Intrusions, malicious data modification attack. Implementation of Trusted pools also one of the postulated solutions for the cloud environment .**

Keywords **- Cloud Computing [2], Security, Policy Administrations, Trusted Pool. Rivest, Shamir and Adleman (RSA), Public Key Infrastructure (PKI).**

I. INTRODUCTION

Cloud services and virtualization are driving significant shifts in IT spending and deployments. Cloud services give companies the flexibility to purchase infrastructure, applications, and services, from third-party providers – with the goal of freeing up internal resources and recognizing cost savings. Virtualization allows maximum utilization of hardware and software, increasing cost savings, as well.

Moving data into the cloud offers great convenience to users since they don't have to care about the complexities of direct hardware management. The pioneer of Cloud Computing vendors, Amazon Simple Storage Service (S3) and Amazon Elastic Compute Cloud (EC2) are both well known examples.

While these internet-based online services do provide huge amounts of storage space and customizable computing resources, this computing platform shift, however, is eliminating the responsibility of local machines for data maintenance at the same time. As a result, users are at the mercy of their cloud service providers for the availability and integrity of their data. But the most of the cloud users and providers are mainly facing challenges regarding security in cloud.

Firstly, traditional cryptographic primitives [3] for the purpose of data security protection cannot be directly adopted due to the users' loss control of data under Cloud Computing [2]. Therefore, verification of correct data storage in the cloud must be conducted without explicit knowledge of the whole data. Considering various kinds of data for each user stored in the cloud and the demand of long term continuous assurance of their data safety, the problem of verifying correctness of data storage in the cloud becomes even more challenging. Secondly, Cloud Computing is not just a third party data warehouse [4]. The data stored in the cloud may be frequently updated by the users, including insertion, deletion, modification, appending, reordering, etc. To ensure storage correctness under dynamic data update is hence of paramount importance. However, this dynamic feature also makes traditional integrity insurance techniques futile and entails new solutions. Last but not the least, the deployment of Cloud Computing is powered by data centers running in a simultaneous, cooperated and distributed manner. Individual user's data is redundantly stored in multiple physical locations to further reduce the data integrity threats. Therefore, distributed protocols for storage correctness assurance will be of most importance in achieving a robust and secure cloud data storage system in the real world. However, such important area remains to be fully explored in the literature.

Intrusion detection (ID) [5] is a crucial technique that has to be part of any comprehensive security solution for high-assurance database security. Our approach to an ID mechanism consists of two main elements, known good user [6] and trusted pool (TP). Known good user is a user who is authorized to access the resources of a cloud data center based on the hypervisor [7].

TP is a combination of resources which will provide the resources for its known good users.

II. KNOWN GOOD USER

While the public IT cloud has a silver lining for many adopters, it isn't without drawbacks, especially in regards to data protection. Once data has gone into a public cloud, data security and governance control is transferred in whole or part to the cloud provider. Yet cloud providers are not assuming responsibility, e.g. Amazon's web services contract states "we strive to keep your content secure, but cannot guarantee that we will be successful at doing so, given the nature of the internet". When handing over the data, the enterprise forfeits all control of the security of the data, unless they protect the data beforehand.

Most of the time security attacks in any environment will happen because of intruders. Even in the case of public cloud is also same problem we are facing. To resolve this problem this paper presenting the concept of known good user.

The key idea in Known good user is that a policy object is jointly administered by at least k Database Administrator (DBAs), that is, any modification made to a policy object will be invalid unless it has been authorized by at least k DBAs. We present design details of Known good user which is based on a cryptographic threshold signature scheme, and show how Known good user prevents malicious modifications to policy objects from authorized users.

The need to authenticate the identities of users, computers and even other organizations, has led to the development of the public key infrastructure (PKI). A public key infrastructure (PKI) can be defined as a set of technologies which control the distribution and utilization of unique identifiers, called public and private keys, through the utilization of digital certificates.

The set of technologies that constitute the PKI is a collection of components, standards and operational policies. The PKI process is based on the use of public and private keys to provide confidentiality and integrity of an organization's data as it is transmitted over the network. When users partake in the PKI, messages are encoded using encryption, and digital signatures are created which authenticate their identities. The recipient of the message would then decrypt the encoded message. For a PKI implementation to operate, each computer in the communication process must have a public key and private key. The public key and private key pair is used to encrypt and decrypt data, to ensure data confidentiality. When two parties use a PKI, each one obtains a public key and a private key, with the private kept only being known by the owner of that particular key. The public key on the other hand is available to the public. Before delving into the components and operations of the PKI, let's first look at what a properly designed and implemented PKI achieves:

CONFIDENTIALITY: A PKI implementation ensures confidentiality of data transmitted over the network between two parties. Data is protected through the encryption of messages, so even in cases where the data is intercepted; the data would not be able to be interpreted. Strong encryption algorithms ensure the privacy of data. Only the parties which possess the keys would be able to decode the message.

AUTHENTICATION**:** The PKI also provides the means by which the sender of the data messages and the recipient of the data messages can authenticate the identity of each other. Digital certificates which contain encrypted hashes are used in authentication, and to provide integrity.

INTEGRITY**:** Integrity of data is assured when data has been transmitted over the network, and have not been fiddled with, or modified in any manner. With PKI, any modification made to the original data, can be identified

NON-REPUDIATION**:** In PKI, non-repudiation basically means that the sender of data cannot at a later stage deny sending the message. Digital signatures are used to associate senders to messages. The digital signature ensures that the senders of messages always sign their messages. This basically means that a particular person cannot, at a later stage, deny sending the message.

A.POLICY OBJECT ADMINISTRATION POLICY:

In general a policy is set of principle or rule to guide decisions and achieve rational outcomes of resources which are shared commonly by all users within the organization. "The term is not normally used to denote what is actually done". This is normally referred to as either procedure or protocol. Policies are generally adopted by the Board of or senior governance body within an organization whereas procedures or protocols would be developed and adopted by senior executive officers. Policies can assist in both subjective and objective decision making by the administrator of that organization.

Policy Creation

Identify all the assets that we are trying to protect.

Identify all the vulnerabilities and threats and the likeliness of the threats happening in cloud network accounts .

Decide which measures which will protect the assets in a cost-effective manner.

Communicate findings and results to the appropriate parties.

Monitoring and review the process continuously for improvement.

The main issue in the administration of response policies is how to protect a policy from malicious modifications made by Admin that has legitimate access rights to the policy object.

Before the response policies can be used, some security parameters are registered with the DBMS as part of a one-time registration phase. The details of the registration phase are as follows: The parameter l is set equal to the number of Adman's registered with the Datacenters. Such requirement allows any Admin to generate a valid signature share on a policy object, thereby making our approach very flexible. Stoup's scheme also requires a trusted dealer to generate the security parameters. This is because it relies on a special property of the RSA modulus, namely, that it must be the product of two safe primes. We assume the Datacenters to be the trusted component that generates the security parameters. For all values of k, such that $2 <= k <= l-1$.

B.RSA PUBLIC-PRIVATE KEYS

The primary advantage of public-key cryptography is increased security and convenience: private keys never need to be transmitted or revealed to anyone. In a secret-key system, by contrast, the secret keys must be transmitted (either manually or through a communication channel) since the same key is used for encryption and decryption. A serious concern is that there may be a chance that an enemy can discover the secret key during transmission.

Another major advantage of public-key systems is that they can provide digital signatures that cannot be repudiated. Authentication via secret-key systems requires the sharing of some secret and sometimes requires trust of a third party as well. As a result, a sender can repudiate a previously authenticated message by claiming the shared secret was somehow compromised by one of the parties sharing the secret. For example, the Kerberos secret-key authentication system. involves a central database that keeps copies of the secret keys of all users; an attack on the database would allow widespread forgery. Public-key authentication, on the

other hand, prevents this type of repudiation; each user has sole responsibility for protecting his or her private key. This property of public-key authentication is often called non-repudiation.

The algorithm chooses p, q as two large prime numbers such that

p=2p`+1 and q=2q`+1

Where p 'and q' are themselves large primes. Let n =p*q be the RSA modulus.

Let m=p'*q'

The admin also chooses e as the RSA public exponent such that e>l. Thus, the RSA public key is PK= (n,e) . The server also computes the private key d € Z such that de = 1 mod m.

B.1. Secret Key Shares

The next step is to create the secret key shares for each of the l DBAs. For this purpose, the Datacenters sets a0 = d and randomly assigns ai from{0,….,m-1} for 1<=i<=k-1. The numbers {a0.....ak-1} define the unique polynomial p(x) of degree k-1,

$$f(x) = \sum_{i=0}^{i=k-1} (a_i x)$$

For l<=i<=l

Si= p(i)mod m.

III. TRUSTED POOL OF RESOURCES

The known good user will have the access privatizations on resources which are provided for him. If he accessing the different resources of single organization it is enough to certify him as a known good user at once. At the same way user also feel secure while accessing his/her Different resources from that same organization. For that trusted pool of resources is one of the methods.Providing services by integrating homogeneous trusted cloud computing resource centers is known as trusted pool of resources.

Trusted Computing pool (TCP) [8, 9, 10] is a set of hardware and software technologies to enable the construction of trusted platforms [11]. It maintains some hardware chip knows as Trusted Computing policy model (TCM) that is now bundled with commodity hardware. The TCM contains an endorsement private key (EK) that uniquely identifies the TCM (thus, the physical host), and some cryptographic functions that cannot be modified. The respective manufacturers sign the corresponding public key to guarantee the correctness of the chip and validity of the key

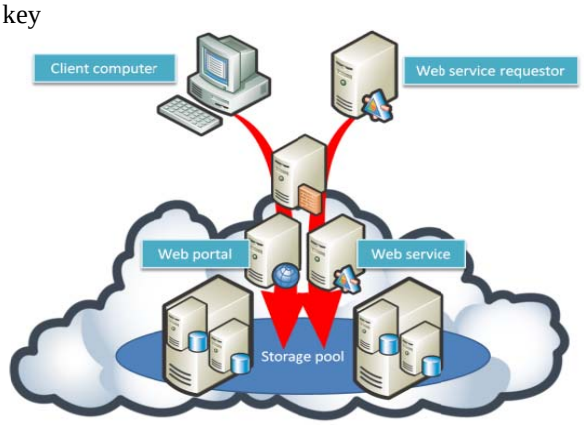

Figure 1

IV. CONCEPT OF TPR

Trusted Pool of Resources (TPR) enhances today's IaaS backend to enable closed box semantics without substantially changing the architecture. The trusted computing base of the TPR includes two components: a trusted virtual machine monitor (TVMM), and a trusted coordinator (TC). Each node of the backend runs a TVMM that hosts customers' VMs, and prevents privileged users from inspecting or modifying them. The TVMM protects its own integrity over time, and complies with the TPR protocols Nodes embed a certified TCM chip and must go through a secure boot process to install the TVMM.

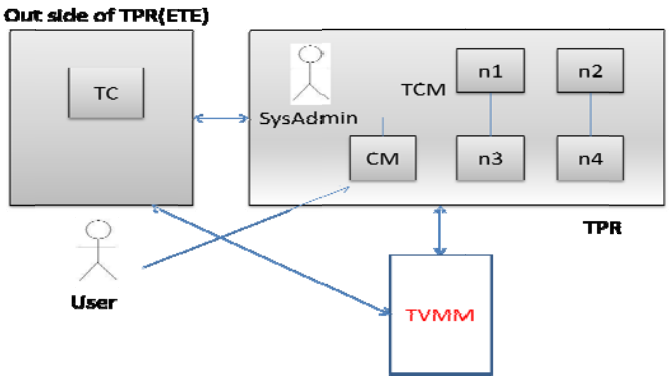

Figure 2

Due to space limitations we will not go into detail about the design of the TVMM, and we refer the reader to for an architecture that can be leveraged to build a TVMM that enforces local closed box protection against a malicious system admin. The TC manages the set of nodes that can run a customer's VM securely. We call these nodes trusted nodes. To be trusted, a node must be located within the security perimeter, and run the TVMM. To meet these conditions, the TC maintains a record of the nodes located in the security perimeter, and attests to the node's platform to verify that the node is running a trusted TVMM implementation. The TC can cope with the occurrence of events such as adding or removing nodes from a cluster, or shutting down nodes temporarily for maintenance or upgrades. A user can verify whether the IaaS service secures its computation by attesting to the TC. To secure the VMs, each TVMM running at each node cooperates with the TC in order to 1) confine the execution of a VM to a trusted node, and to 2) protect the VM state against inspection or modification when it is in transit on the network. The critical moments that require such protections are the operations to launch, and migrate VMs. We assume an external trusted entity (ETE) that hosts the TC, and securely updates the information provided to the TC about the set of nodes deployed within the IaaS perimeter, and the set of trusted configurations. Most importantly, system admin that manage the IaaS have no privileges inside the ETE, and therefore cannot tamper with the TC. We envision that the ETE should be maintained by a third party with little or no incentive to collude with the IaaS provider e.g., by independent companies analogous to today's certificate authorities like VeriSign.

V. CONCLUSION

This paper, concerns about the confidentiality and integrity of their data and computation are a major deterrent for enterprises looking to embrace cloud computing. We present the design of a Trusted Computing pool (TCP) that enables IaaS services such as Amazon EC2 to provide a closed box execution environment. TCCP guarantees confidential execution of guest VMs, and allows users to attest to the IaaS provider and determine if the service is secure before they launch their VMs. We plan to implement a fully functional prototype based on our design and evaluate its performance in the near future.

Security is not a destination its journey.

REFERENCES

[1]Dummies 2nd IBM Limited Edition Mini eBook

[2]The NIST Definition of Cloud Computing :http:// csrc.nist.gov/publications/nistpubs/800-145/SP800-145.pdf

[3]Cryptographic primitive : http://en.wikipedia.org/wiki/ Cryptographic_primitive.

[4]Oracle9i Data Warehousing Guide: http://docs.oracle. com/cd/B10501_01/server.920/a96520/concept.htm.

5]NIST SP 800-94, Guide to Intrusion Detection and Prevention: Guide to Intrusion Detection and Prevention Systems (NIST).

[6]Intel Known Good User(TxT): http:// www.intel.com /content/www/us/en/architecture-and-technology/trusted-execution-technology/malware-reduction-general-technology.html.

[7]VMH: http://en.wikipedia.org/wiki/Hypervisor.

[8]TCG: https://www.trustedcomputinggroup.org.

[9]S. Berger, R. C´aceres, K. A. Goldman, R. Perez, R. Sailer, and L. van Doorn. vTPM: virtualizing the trusted platform module.In Proc. of USENIX-SS'06, Berkeley, CA, USA, 2006.

[10]D. G. Murray, G. Milos, and S. Hand. Improving Xen security through disaggregation. In Proc. of VEE'08, pages 151–160, New York, NY, USA, 2008.

[11] trusted platforms http://en. wikipedia. org/wiki/ Trusted_Platform_Module