Aarhat Multidisciplinary International Education Research Journal (AMIERJ)
A Peer Reviewed Multidisciplinary Journal UGC Approved Journal No 48178, 48818
Impact Factor 5.18 ISSN 2278-5655

SECURITY ANALYSIS OF CRYPTOGRAPHIC TECHNIQUES FOR AUTHENTICATING VM TEMPLATE IN THE CLOUD

Mrs. Kavita Shinde
Assistant Professor
MIT ACSC, Alandi (D),
Pune - 412105

Mrs. Bareen Shaikh
Assistant Professor
MIT ACSC, Alandi (D),
Pune - 412105

Mrs. Sangeeta Borde
Assistant Professor
MIT ACSC, Alandi (D),
Pune - 412105

Abstract: *There are number of cloud providers which lease VM Template but most of the time the cloud consumers are concerned about the fact that the VM Template in cloud which he/she is checking is authentic or not. To solve the issue raised by many consumers about authentication of VM template in the cloud, we have analyzed different cryptographic techniques with their working, advantages and disadvantages over other techniques in this paper. There are number of variations of each and every technique which is discussed in detail in our paper.*

Keywords: VM Template Cloud, Cryptographic Techniques

Introduction:

Cloud computing consists mostly of three different service types: IaaS (Infrastructure-as a Service), PaaS, (Platform-as a Service), and SaaS (Software-as a Service), and four "cloud deployment modes" (Public, Private, Community, and Hybrid) that define the ways that cloud services are delivered. Each of the three cloud computing types (IaaS, PaaS, SaaS) have separate features and structures where different functions are needed to construct and maintain the required security levels against the various types of security threats.[2]

Encryption and access control are the two primary means for ensuring data confidentiality in any IT environment. In situations where encryption is used as a data confidentiality assurance measure, the management of cryptographic keys is a critical and challenging security management function, especially in large enterprise data centers, due to sheer volume and data distribution (in different physical and logical storage media), and the consequent number of cryptographic keys.[4]

Security issues in the cloud:

Important Security Issues in the Cloud:

1. Integrity: [5]

Integrity makes sure that data held in a system is a proper representation of the data intended and that it has not been modified by an authorized person. When any application is running on a server, backup routine is configured so that it is safe in the event of a data-loss incident. Normally, the data will backup to any portable media on a regular basis which will then be stored in an off-site location.

2. Availability: [5]

Availability ensures that data processing resources are not made unavailable by malicious action. It is the simple idea that when a user tries to access something, it is available to be accessed. This is vital for mission critical systems. Availability for these systems is critical that companies have business continuity plans (BCPs) in order for their systems to have redundancy

3. Confidentiality: [5]

Confidentiality ensures that data is not disclosed to unauthorized persons. Confidentiality loss occurs when data can be viewed or read by any individuals who are unauthorized to access it. Loss of confidentiality can occur physically or electronically. Physical confidential loss takes place through social engineering. Electronic confidentiality loss takes place when the clients and servers are not encrypting their communications Importance of Security in Cloud Computing:

VM Template Authentication Methods:

A.VM Template Authentication Using Digital Signature

As Figure 2 shows, VM templates can be signed by using Cloud Provider's private key. The process of authentication is as follows [3]:

1. The Cloud Provider signs VM with its own private key.
2. The Cloud Provider sends the signed VM to the Verification Engine.
3. The Cloud Provider sends the public key to the Cloud Consumer.
4. The Cloud Consumer sends the public key to the Verification Engine.
5. The Verification Engine verifies the authentication of VM template.

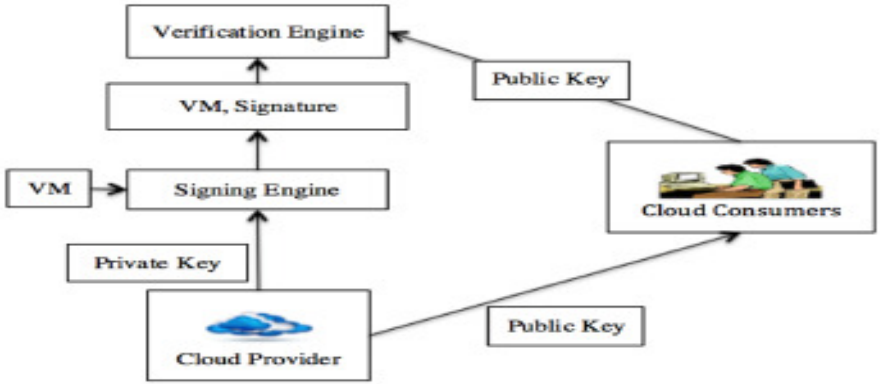

Figure 1. VM template authentication using digital signatures

Advantages: [1]

This approach has the advantage that the cloud Provider is able to create and modify multiple VM templates, and all cloud Consumers can verify the source and integrity of the VM template via digital signature verification. It also has the advantage of simplified key management. All that is required are the following: (a) the cloud Provider needs to create a single public/private signature key pair and protect the private key from unauthorized use and from unauthorized disclosure, (b) the cloud Provider needs to provide the public key in a trusted manner[4] to each cloud Consumer; and (c) the cloud Consumer needs to protect the public key from undetected, unauthorized modification.

Disadvantages: [1]

The approach has some disadvantages as well. While on the surface, the approach seems highly secure, there are several security concerns with it:

1. First of all, how does the cloud Consumer communicate securely with the verification engine to provide the public key and to obtain the verification results?

Let us assume that the cloud Consumer can establish a secure session using TLS or SSH.

2. Then the question becomes: how does the cloud Consumer trust the verification engine running in the cloud Provider. If the cloud Consumer cannot trust or authenticate the verification engine, it has no basis to trust the response from the verification engine regarding the VM template signature verification.

3. Furthermore, whatever means the cloud Consumer uses to establish trust in the verification engine, why not use the same means to trust the VM template and forego the extra step of having to first establish trust in the verification engine?

B.VM Template Authentication Using Cryptographic Hash Functions

As Figure 3 shows, using Cryptographic Hash Functions, such as SHA-256, can authenticate VM templates. This approach doesn't need any key management. The process of authentication is as follows [3]:

1. VM will be send to the Verification Engine and Hashing Engine.
2. The Hashing Engine computes the hash value of VM template.
3. The Hashing Engine sends the hash value of VM template to the Cloud Provider.
4. The Cloud Provider sends the hash value of VM template to the Cloud Consumer.
5. The Cloud Consumer sends the hash value of VM template to the Verification Engine.
6. The Verification Engine verifies the authentication of VM template by creating the hash of VM template and comparing it with the one, which was received from the Cloud Consumer.

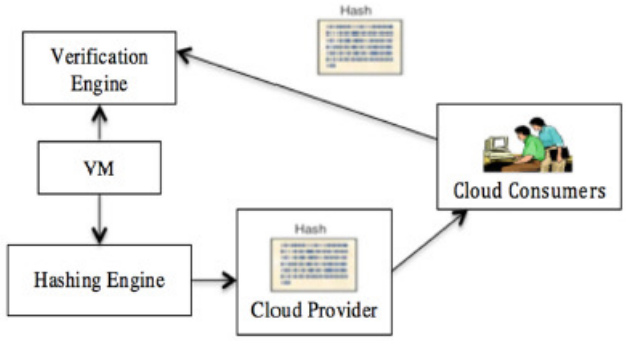

Figure 3. VM Template Authentication Using Cryptographic Hash Function

Figure 2. VM Template Authentication Using Cryptographic Hash Function

Advantages: [1]

The approach has the advantage of requiring no key management as it is required in Digital Signature. However, the hash value of the VM template needs to be provided to the consumers using means that assure its integrity and source (e.g., physically). The cloud Consumer provides this hash value for comparison during VM template authentication.

Disadvantages: [1]

1. This approach has the limitation that each time the VM template' is modified, a new hash value needs to be promulgated using a secure, out-of-band means.

2. The approach has the limitation that each VM template hash value needs to be promulgated using secure, out-of-band means. One can assume that the cloud will have multiple VM templates.

3. Just like the digital signature, this approach does not solve the problem of the cloud Consumer communicating securely with the verification engine to provide the hash value and obtaining the verification results. Let us assume that the cloud Consumer can establish a secure session using TLS or SSH.

4. Then the question becomes: how does the cloud Consumer trust the verification engine running in the cloud Provider. If the cloud Consumer cannot trust or authenticate the verification engine, it has no basis to trust the response from the verification engine regarding the VM template verification.

5. Furthermore, whatever means the cloud Consumer uses to establish trust in the verification engine, why not use the same means to trust the VM template and forego the extra step of having to first establish trust in the verification engine?

C.VM Template Authentication Using Message Authentication Code (MAC)

As Figure 4 shows, using Message Authentication Code (MAC) can authenticate VM templates.

The process of authentication is as follows [3]:

1. The Cloud Provider uses a secret key and keyed hash function to create a MAC.
2. The Cloud Provider sends the MAC to the Verification Engine.

3. The Cloud Provider sends secret key to the Cloud Consumer.

4. The Cloud Consumer sends secret key to the Verification Engine.

5. The Verification Engine verifies the authentication of VM template.

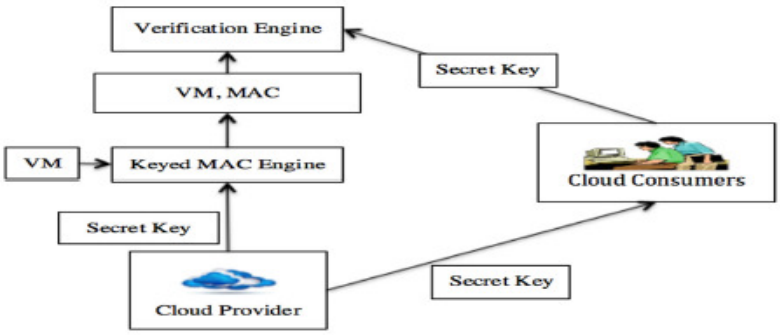

Figure 4. VM Template Authentication Using Message Authentication Code (MAC)

Figure 3. VM Template Authentication Using Message Authentication Code (MAC)

Advantages: [1]

1. The approach has the advantage of the cloud Provider being able to create and modify multiple VM templates and all cloud Consumers being able to verify the source and integrity of the VM template via MAC verification.

2. It also has the advantage of simplified key management. All that is required are the following:

(a) The cloud Provider needs to create a single secret key and protect it from unauthorized use and from unauthorized disclosure.

(b) The cloud Provider needs to provide to each cloud Consumer with the secret key in a secure manner.

(c) The cloud Consumer needs to protect the secret key from unauthorized disclosure.

Disadvantages: [1]

1. Unless the secret key is unique per Consumer, this approach is vulnerable to one Consumer modifying a VM template to compromise another Consumer. Having unique keys for each Consumer will increase a cloud Provider's key management

challenge.

2. Just like the use of a digital signature, this approach does not solve the problem of the cloud Consumer communicating securely with the verification engine to provide the secret key and to obtain the verification results. Let us assume that the cloud Consumer can establish a secure session using TLS or SSH.

3. Then the question becomes: how does the cloud Consumer trust the verification engine running in the cloud Provider. If the cloud Consumer cannot trust or authenticate the verification engine, it has no basis to trust the response from the verification engine regarding the VM template authentication.

4. Furthermore, whatever means the cloud Consumer uses to establish trust in the verification engine, why not use the same means to trust the VM template and forego the extra step of having to first establish trust in the verification engine?

Conclusion:

Cloud Computing represents one of the most significant shifts in information technology. Full adoption of Cloud Computing has been concern regarding the security and privacy of information because of security. Much work has been done regarding the security of the cloud and authenticating VM template in the cloud using different cryptographic techniques, but until now no technique is that much strong to solve the problem of the cloud Consumer communication securely with the verification engine

References:

[1] https://link.springer.com/chapter/10.1007/978-1-4614-9278-8_1 [Last retrieved on 2nd Dec 2017]

[2] https://www.cryptomathic.com/news-events/blog/key-management-challenges-to-iaas-providers-in-the-cloud [Last retrieved on 10th Dec 2017]

[3] Ramaswamy Chandramouli, Michaela Iorga, Santosh Chokhani (2014). Cryptographic Key Management Issues and Challenges in Cloud Services (pp. 1-30). Springer New York.

[4] http://nvlpubs.nist.gov/nistpubs/ir/2013/NIST.IR.7956.pdf [Last retrieved on 27th Nov 2017]