Chapter 3

PROBLEM STATEMENT: SECURITY FLAWS AND DESIGN ISSUES IN CLOUD INFRASTRUCTURE

In this section, we present the threat models where an insider can manipulate user VM's in the node controller of cloud platform. Here, we assumed that cloud service provider is malicious and cloud consumer is not having any security constraints to access their cloud assets. The model described in two locations in cloud infrastructure.

3.1 SCOPE OF INSIDER ATTACKS

3.1.1 Attacks in the local Machine

We assume that the legacy OS with malicious insider at cloud platform. Such that, it clearly states that kernel and user mode is not modified. Such that, an insider can compromise VM's, those are running on controller of target cluster. For example, an insider can modify target VM kernel and they can launch VM with malicious intention without any permissions from VM owner (user). It leads to sensitive data breach of cloud storage of other VM and target VM. With reference of this attack pattern, VM's those are running on NC are in great threat from insiders.

3.1.2 Attacks in the cloud administrator

The cloud service provider contains ring 0 privileges to access any content of cloud users and physical resources hosted at cloud data center [72]. To launch insider attack on resources, insider obtains a memory dump of target VM. Initially malicious insider has no

idea about credentials stored in dump of VM kernel image. To obtain a password from kernel, an attacker or insider simply devises a method on obtained kernel of VM. The kernel image filtered using *strings* command, it thoroughly checks dump and returns available strings with name of password. Once insider obtains credentials from kernel of VM, the following are expected issues:

- A cloud service provider can access guest OS contents by using their privileges. With effect of this cloud client might lose their data confidentiality and integrity. As said earlier, cloud service provider can save, restore, reboot, and shutdown any guest operating system.
- In [73,74] demonstrated various attack scenarios and those pose great threats in cloud computing virtual environment.
- A malicious insider or malicious cloud service provider can change or breach data upon agreed with competitors of the client company. Attackers (insiders) inside the company have great risk to information resources because they are sophisticated about internal structure.
- Malicious insider cannot access the hypervisor but they can access secondary storage and network I/O. With this maliciousness cloud service provider, can perform any task without any permission from owner of Domain or Virtual Machine.

3.2 SECURITY DESIGN FLAW IN CURRENT VIRTUAL MACHINE MONITORS

This chapter provides a proof of the research problem that we addressed in this research work. We studied and analyzed the insider attacks in cloud infrastructure. This

chapter provides a complete detail of addressed problem in cloud environment and various design flaws in virtual machine monitors that leverages and violates the integrity and confidentiality rules of client virtual machines.

The addressed research problem is a hypervisor or virtual machine monitor without least privileges. The failure to defend the privileged user or cloud administrator (malicious) access to the sensitive data which is not accessible that holds cryptographic keys resident in memory space of virtual machine monitor [74].

The proof of concept presented in this chapter consists of attacks performed in the virtual machine monitor from three major providers of Virtual software i.e., Amazon Web Services (AWS), Microsoft azure, etc. We chose to demonstrate the problem with the most commercial solutions such as VMware ESXi and major open source hypervisor Xen and Linux KVM [83]. This section demonstrates problem in multiple hypervisor vendors that argues in favor of a design issues instead of an implementation of fault tolerance in hypervisor or virtual machine monitor.

This chapter organized as follows: We introduce the attack scenario of our research problem. After, we provide brief description bout the attack we use used against the virtual machine and platforms. The virtual machine introspection is used for detail implementation purpose in two sections. The following two subsections provide information about two dedicated software components and attacking nature that applied on virtual machine and virtual machines. The below section provides the insider threat model of the proposed system. It shows the all the possible flaws which is present in the present hypervisors or virtual machine monitor.

3.3 THREAT MODEL

The threat model considers the insiders are malicious with their privileged authentication techniques, those are considered in this section. We have certain assumptions regarding the insider threat implementation.

- Assume 1: Insider can modify the hypervisor or virtual machine monitor
- Assume 2: An insider can rebuild, compile, and execute an arbitrary software within the cloud infrastructure environment.
- Assume 3: Hardware components are unmodified and not used in attack.

3.3.1 Threat implementation

The idea for the insider attacks we took from the cold boot attacks [31]. The cold boot attack illustrates that cryptographic keys are stored in the random-access memory that contains the virtual machine contents up to some amount of time period. This creates a chance of taking virtual machine contents that an attacker can exploit.

To preserve the contents of extracted random access memory, the cold boot simple cooling techniques are applied and those are connected to some forensic analysis. The cryptographic key and sensitive information can be extracted using simple cooling technique of cold boot attack with disk encryption mechanisms. The major hurdle in the cold boot attacks is to have the direct communication or physical access to the system. In the cloud environment, the cloud employees can launch these attacks since they have physical access to the cloud resources.

The virtual machine monitor manages the virtual machines by allowing them to store the sensitive information in the cloud resources. The Figure 3.1 illustrates the

virtualization environment, where multiple virtual machines can run on top of the physical computer. The illustrated figure represents the array of byte sized with the size of M-1. The virtual machine monitor is a virtual machine player, which holds all the rights over the physical random access memory.

An empirical method proposed in this chapter shows that it is possible to launch the attack without physical access to the cloud components. The malicious insider can simple launch an attack taking exploits of the virtual machine monitor, in our case Xen is the hypervisor or virtual machine monitor. This proved that no need to have the physical access to the cloud components, simply they perform the attacks on virtual machine monitor. As we discussed earlier the cloud virtual machine are launched and monitored by the virtual machine monitor. In fact, the random access memory can be shared among the virtual machine, such that attacker directly extracts the memory contents which are assigned to the consumer virtual machine [30].

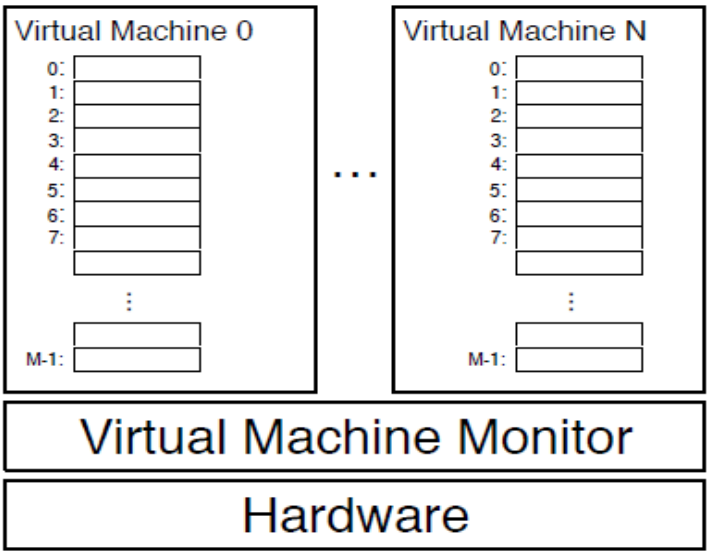

Fig. 3.1 Virtual Machine's Memory Space.

To implement this attack scenario, we used Linux operating system along with Xen hypervisor because both are open source software's and it is possible to change and access the internal contents of the physical resources, which are hosted in the cloud server side.

The empirical attack concepts directly bypass the concept of principle of least privilege can undermine the security concepts of the virtual machine physical host. The attacker can launch the attack on virtual machine monitor by exploring the vulnerabilities present in the cloud virtualization environment and these attack scenarios are still issues in the open source software components which are majorly used in cloud computing environment.

3.3.2 RSA Key Structure in Memory

The RSA private key structure reveals the everything which is loaded in the memory components. If the attacker obtains the private key of the cloud consumer virtual machine, then the attacker can simply uses the RSA private key to take a memory snap of the virtual machine. The RSA key structure is represented in PKCS#12 [32]. The representation of the private key in OSI – Abstract Syntax Notation One (ASN.1), which is also represented with four, parts ITU-TX 680 [33,68]. This structure specifies the RSA key in the random access memory.

An ISO8824 defines the set of notations and definitions for the values and data types, where as a data value is an instance of respective data type. The set of encoding rules specifies the value of the octets that carry application semantics, referred as transfer syntax. The Figure 3.2 illustrates the RSA private key ASN.1 in consumer virtual machine dump or snapshot. The ITU-TX.690 [77] presents the set of rules includes Canonical Encoding

Rules (CER), Basic Encoding Rules (BER), and Distinguished Encoding Rules (DER) for encoding the abstract objects in binary form. The BER consists the CER and DER definitions and differs from each other in a set of boundaries.

```
RSAPrivateKey ::= SEQUENCE {
    version           Version,
    modulus           INTEGER,  -- n
    publicExponent    INTEGER,  -- e
    privateExponent   INTEGER,  -- d
    prime1            INTEGER,  -- p
    prime2            INTEGER,  -- q
    exponent1         INTEGER,  -- d mod (p-1)
    exponent2         INTEGER,  -- d mod (q-1)
    coefficient       INTEGER,  -- (inverse of q) mod p
    otherPrimeInfos   OtherPrimeInfos OPTIONAL
}
```

Fig.3.2 RSA Private Key ASN.1 type.

The encoding rules are defined in ITU-TX.690 or ISO-8825-1 [34]. The encoding should contain the four distinguished components: content octets, length octets, end-of-content octets and identifier octets. These octets must be following this order: identifier octets, content octets, length octets, and end-of-content octets, such that it is a order dependent. The main concentration in our discussion is an identifier octet, which encodes the ASN.1 tag for the different types of data value, set of tags, and is described in [35]. For instance, an integer value has a tag with a hexa decimal value of 0x02. The identifier block consists of three components such as constructed bit, two-bit classification and primitive type and it is a starting octet of any ASN.1 encoding scheme [103].

The RSA private key is defined as ASN.1 objects identifier available in [32,67], which elaborates the object identifiers for both RSA private and public keys in memory locations. Our primary objective is to extract the private key from consumer virtual machine dump, that we focused in cloud infrastructure to represent the vulnerabilities in

the ASN.1 format. The following Figure 3.3 represents the components of VMI Architecture in the binary representation as per ASN.1 type.

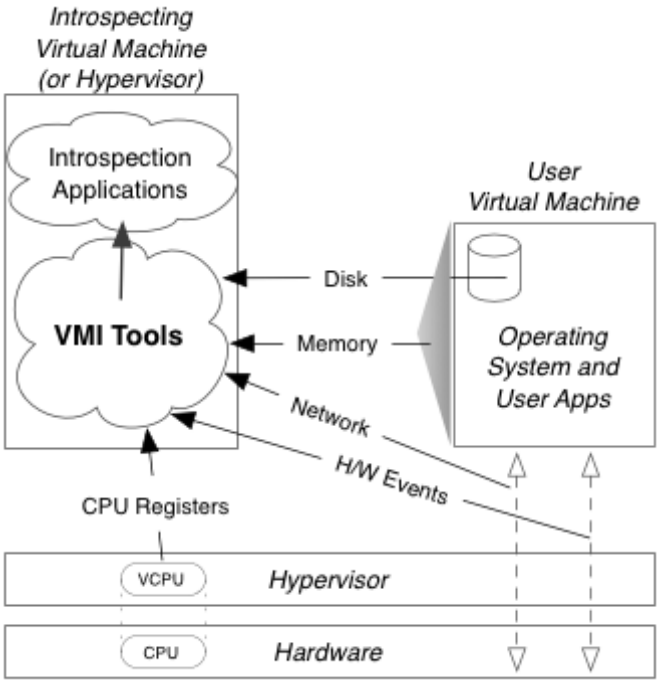

Fig.3.3 lib VMI Architecture [103].

The private key can be searched in the consumer virtual machine dump can be introduced in the cold boot attacks consists of looking to verify the features of DER encoding. As per the cold boot attack pattern no false positive but it happening with our tests to be true. The search algorithm starts to verify the class tag 0x10, but since its encoding must constructed in ASN.1 SEQUENCE type. Then sequence byte chanced to 0x30, and then constructed encoding means the constructed bit in the identifier is active.

The search algorithm finds the RSA version number in the together with the DER encoding tag of next field. The RSA version key is almost zero in all the cases except the cases where multi-prime is used in its DER encoding. The final result from the search is expected octets bytes 0x02 0x01 0x00 0x02, which represent the RSA version of 0x02 is

the next type field. Decomposing the RSA version of byte to distinguish the identifier octet of an integer (0x02) that has zero value (0x00) and one byte of length (0x01). We use this search algorithm to locate private RSA keys in memory dumps of virtual machines. More details can be found in the below sections 3.4 with case studies.

3.4 CASE STUDIES

This section provides empirical methods that an insider or employee can perform with privileged malicious tasks on consumer data. Demonstrated an attack that damages the confidentiality and integrity of user data in the cloud computing environment. The first attack was easy to perform that takes an advantage of administrator access privileges to obtain the consumers virtual machines snapshot or dump in the cloud environment. We used Xen hypervisor to perform these attacks. The memory dump of the client can be obtained using the single command as root user [103].

3.4.1 Case Study 1

Extracting the clear text passwords

The first attack was shown in [75] that extract the clear text passwords from the operating system memory dump in Linux environment. We demonstrate that this attack is possible in cloud environment over the client virtual machines. To launch, this attack the malicious insider issue a command *dump-core,* it is a management command in the Xen user interface (xm or xl). The task of the dump-core command is to prepare the dump memory of targeted virtual machine. A malicious insider has to specify the virtual machines in the dump-core command. After obtain the memory dump, we use cat command to check or verify the password in the dump file and grep is used to locate the

password. These commands are listed below, the privileges or password found using theses commands were used for the login into the virtual machines and Apache RSA key that are loginpwd and apachersapwd respectively. Practically, an insider has no idea about the client password those are found in the memory dump. It is possible to automate the password search using TrueCrypt, once an insider obtained a memory dump. In both ways an insider can find the password from the dump in the clear text manner.

$ xm dump-core 2 -L sekhardomu.dump

Dumping core of domain: 2 ...

$ cat sekhardomu.dump | strings | grep loginpwdloginpwdloginpwd

$ cat sekhardomu.dump | strings | grep apachersapwd

apachersapwd

apachersapwd

apachersapwd

3.4.2 Case Study 2

Obtaining private keys using memory snapshots

The primary objective of second attack is to acquire the private key of private-public key pair in the cloud environment. This attack scenario demonstrates how a key obtained from the Apache web server. The key is used for creating or establishing a secure channel with clients. As shown in the earlier attack, the private key is stored in memory dump in the form of plain text format. Here, RSA key is a number either 1024 or 2048 bits.

To launch this attack, a malicious insider obtains a memory dump of client virtual machine, as earlier attack. Now, the insider having keys in the memory dump but memory dump size is minimum of hundreds of megabytes. In this attack, the same technique is used to obtain the private keys from the memory dump as cold boot attack. The cryptographic keys are stored in the memory dump are in recognized format i.e., most using PKCS#1 that represents the keys in ASN.1 object format. Such that, ASN.1 having known structure of RSA key in the memory dump. The *rsakeyfind* tool searches the memory dump to extract the RSA keys in known object structure. The following command shows the attack command sequence on Linux platform.

$ xm dump-core 2 -L sekhardomu.dump

Dumping core of domain: 2 ...

$ rsakeyfindsekhardomu.dump

found private key at 1b061de8

version = 00 modulus = 00 d0 66 f8 9d e2 be 4a 2b 6d be 9f de 46 db 5a ...

publicExponent = 01 00 01

privateExponent = ...

prime1 = ...

prime2 = ...

With the above mechanisms, an insider can obtain credentials from the consumer virtual machines dump. To implement these methods, we used Xen hypervisor on Ubuntu 13.04 with Linux kernel 3.0.1.

3.5 CONCLUSION

This chapter provides information about problem definition of the research work. This section devises an empirical method to prove the vulnerabilities in the existing methods. We addressed and demonstrate possible empirical attacks patterns to strengthen the proposed system. These are performed on virtualization layer of the cloud infrastructure with xen hypervisor as virtual machine monitor and implemented those attacks in the Ubuntu 12.04 LTS 32-bit operating system.