12-2019

Evaluating Security Aspects for Building a Secure Virtual Machine

sudip kandel
11kandelsudip@gmail.com

Evaluating Security Aspects for Building a Secure Virtual Machine

by

Sudip Kandel

A Starred Paper

Submitted to the Graduate Faculty of

St. Cloud State University

in Partial Fulfillment of the Requirements

for the Degree of

Master of Science

in Information Assurance

December, 2019

Starred Paper Committee:
Susantha Herath,Chairperson
Mark B. Schmidt
Abdullah Abu Hussein

Abstract

One of the essential characteristics of cloud computing which revolutionized the IT business is sharing of computing resources. Despite all the benefits, security is a major concern in cloud virtualization environment. Among those security issues is securely managing the Virtual Machine (VM) images that contain operating systems, configured platforms, and data. Confidentiality, availability and integrity of such images pose major concerns as it determines the overall security of the virtual machines. This paper identified and discussed the attributes that define the degree of security in VM images. It will address this problem by explaining the different methods and frameworks developed in the past to address implementing secure VM images. Finally, this paper analyses the security issues and attributes and proposes a framework that will include an approach that helps to develop secure VM images. This work aims to enhance the security of cloud environments.

Acknowledgement

I would like to thank my advisor and committee chair, Dr Susantha Herath. Thank you for your advice, constructive feedback and encouragement. You would always a source of inspiration and always motivates me at every stage. I would also like to thank Dr Abu Hussein Abdullah for helping me pick up the research topics and giving me ideas and helped me to strengthen the quality of the paper. Thank you, Dr Mark Smith, for the valuable time and feedback for reviewing the paper. Thanks to my parents for encouraging me and always being a role model. Finally, I would love to thank the library staff for assisting me to develop quality work.

Table of Contents

Chapter Page

List of Figures

Figure Page

List of Tables

Chapter I: Introduction

Introduction

Cloud computing adaptation by IT business and industries over the recent years has increased tremendously. Cloud computing is a model which provides rapid elasticity, on-demand self-service and with virtualization, it allows to share computing resources among different Virtual Machines(Hogan, Liu, Sokol, & Tong, 2011). Despite the huge advantages of cloud in reduced cost, efficiency and scalability, there is several security concerns that impede the adaptation of cloud environment. John Chambers, CEO of Cisco at the RSA conference in San Francisco referred to cloud computing and stated that “It is a security nightmare and it can't be handled in traditional ways (Duffy, 2009). More to traditional security issues like Denial-of-service, specific cloud security issues like multitenancy problems, authentication and access control level issues and virtualization level issues like VM migration and VM isolation always provide loopholes to compromise the integrity and security of the cloud.

This paper discusses VM images and identifies their exploits and then presents the essential characteristics a secure VM image should have. VM images are the specific templates that are stored in cloud repositories which are used to create the Virtual Machines(Gabor, 2016). Integrity and security of the Virtual Machine Images (VMIs) should be ensured as it defines the overall security of the cloud environment. Different approaches and frameworks have been developed over time to secure the VM images in the cloud, but they lack in certain aspects which trying to improve others. Mirage Image Management System covers the access control, unwanted information

removal, tracking modification and scanning of malware but it has several drawbacks (Wei, Zhang, Ammons, Bala, & Ning, 2009). It introduces huge performance overhead and scanning of the image does not guarantee that the image is malware free. Similarly, EVDIC approach proposed by Kazim, Masood and Shibli, can maintain privacy, integrity and access control but it lacks unwanted data removal and software updates.

Problem Statement

Many Businesses and Organizations are reluctant to adapt to cloud environment and migrate their VM to the cloud. This is mainly attributed to the lack of security. One of the major security issues is VM images protection and deterrence which is slowing down the adaptation of cloud.

The efficiency and cost-effectiveness of the cloud are solely based on the sharing of the VMI or retrieving the images from the repositories that are created by cloud providers or third parties. For instance, publishers of VM images may unintentionally release some privileged information like username and password files while making the image public. Moreover, when the retriever runs such images it is possible to develop and propagate Trojan horses lowering the overall security of the virtual machines in the cloud (Wei et al., 2009). Similarly, security patching of the dormant images from the latest security vulnerabilities is complicated and if the dormant images with some vulnerabilities are operated they may easily infect other images in the compute cloud.

A survey carried three years ago, over AMIs revealed that 98% of the Windows AMIs (249 out of 253) and 58% of the Linux AMIs (2005 out of 3432) audited had critical

vulnerabilities. Out of those 5303 images audited 612 contained at least one shell history file which contained around 160, 000 lines of command history, and identified 74 identification credentials (Marinescu, 2013).

Nature and Significance of the Problem

With the growth of cloud computing, eventually, all the data and applications of the consumers are being centralized into the cloud. The cloud image publisher can upload images in the cloud repository. Publisher's images sometimes, unknowingly or with malicious intent may disregard security holes in such images. The service providers or administrators hosting such images face issues of compliance and SLA agreement with the user. On the other hand, users downloading such public machine images or community implemented images open the back door to host malicious content and may also allow a hacker to access their system.

Objective of Study

The objective of the study is to investigate and identify the security exploits in VM images and suggest possible security controls for Virtual mobile infrastructure (VMI). First, existing VM security efforts including important frameworks and approach will be analyzed to see the security controls covered. Afterwards, the study will provide a framework of security attributes to improve the security of the VMI.

Research Goals and Questions

Following research questions were put forward before starting the research to help complete the research:

1. Highlight the state of the Art in VM image security.

   a. What are the current efforts in creating Secure VM Images (i.e. Techniques/ methods)?
   b. What are the attributes that can define a secure VM image?
2. Propose steps and procedures to develop a secure VM providing the users in the form of recommendations.

Definition of Terms

Cloud computing: Internet-based computing that provides the required resources and services on demand.

On-demand-self-service: Feature of cloud computing that allows scalable cloud computing capabilities without interacting with the service providers.

Virtual Machine Image (VMI): A template that contains Operating System, data structures and applications required to initiate a Virtual Machine.

Amazon Web Services (AWS): Includes several services Amazon has provided in the cloud for computing, storage, networking, database and applications.

Amazon Machine Image (AMI): Amazon web service-based template that contains information to launch an instance.

VM migration: Moving virtual machines from one physical environment to another.

Cloud Service Provider (CSP): Companies that provide cloud services to organizations and individuals.

Community Implemented Images: Public Images made available for the specific community. For example, community AMI in AWS.

Infrastructure as a Service (IaaS): The cloud service providers provide the networking, storage and servers but the customers manage all the layers above including the Operating System.

Platform as a Service (PaaS): This service provides all the infrastructures needed for developers and architects where they can start developing and deploying applications.

Software as a Service(SaaS): It provides all the services even the software applications are providing. It is targeted directly to end users.

International Organization for Standardization (ISO): Organization responsible for various international standards.

Patching host: Host which can mount and access an offline image like their own file system which is used to perform the file replacement actions required while patching on a running VM(Zhou et al., 2010).

Simple emulation-based patching: This approach of patching a running VM by file replacement actions from a patching host(Zhou et al., 2010).

Emulated environment: Environment set up by the patching host to perform patching (Zhou et al., 2010).

Identity and Access Management(IAM): It is a service that allows an organization to manage users and groups, and uses permission to allow or deny access to the resources. (“Identity and Access Management (IAM) - Amazon Web Services (AWS),” n.d.)

Metadata: It is the data that provides information about other data.

Role Based Access Control(RBAC): It is a method of a restricting system or network access based on the role of an individual within the organization.

Distributed Denial-of-Service Attack(DDOS): This is a cyber-attack in which multiple systems overload the resources of a system to make it unavailable to its users.

Amazon Simple Storage Service (Amazon S3): It is the storage service provided by Amazon cloud that offers scalability, availability, security and performance.

Amazon Elastic Compute Cloud (Amazon EC2): It is the web service provided by Amazon that provides computing capacity where the users can launch virtual servers.

Application Program Interface(API): API is a set of routines, tools and protocols that are used for building software that specifies how the software interacts with the users.

Hypervisor: It is a software or hardware that allows running of virtual machines. The machine which allows running of VM on top of it is called host machines and the VM are called guest machines.

Hyper Text Transfer Protocol(HTTP): It is an application protocol that allows the transmission of data from the web.

File Transfer Protocol (File Transfer Protocol): It is a Network protocol that is used to transfer files between a client and a server.

Secure Socket Shell(SSH): It is a cryptographic network protocol that is used to access the unsecured network in a secure way.

Nmap: It is a security scanner that traces packets to find hosts and services on a computer network.

Domain Name System(DNS): It is the naming system for the computer or services over a network. Domain names are the alphanumeric representation of IP addresses. Domain Name System translates between an IP address and domain names.

Summary

This chapter provided a glimpse of cloud computing components mainly the virtual machine images and benefits of the cloud. With the increased use of resources and services, there come security and integrity challenges and this section narrows it to VM images security. The next chapter discusses in detail about the VMI and different frameworks and methods adopted to maintain their security.

Chapter II: Background and Review of Literature

Introduction

This chapter explains in depth about the virtual machine images and will identify the attributes to have a secure VM image. Moreover, approaches to maintain secure VMI are discussed in detail and their methodology involved to deal with the threats and vulnerabilities. The chapter also analyzes the issues related to VM migration, sharing and isolation since they attribute to make the VM images vulnerable.

Background Related to the Problem

Virtualization is the key concept of the cloud as it has helped to improve the performance speed and lower down the cost of cloud computing but along with it comes several virtualization level issues. As stated by Hussein in his paper “Eliminating the virtualization layer will avoid the security hazards but this will exclude the vital features like VM mobility”. The components of virtualization are hypervisors, virtual machines and virtual machine images.

Whole concept of cloud computing is achieved through virtualization. It allows us to utilize the shared pool of resources to create the Virtual Machines to carry out different tasks simultaneously. A server without virtualization allows only one system to run at a time in the server which is very inefficient in terms of cost and infrastructure. But with virtualization, we can run many instances of Virtual machines on a single server. A simulated environment is created which is called the virtual machines and the instances of virtual machines are called virtual machine images. Based on the architecture of the different layers, virtualization can be hardware, network, desktop,

software, memory and storage. Most commonly used virtualization is hardware virtualization which is divided into full, para and emulated virtualization. Virtual machines in full virtualization are unknown of the underlying hardware which is different from paravirtualization.

In full virtualization, different operating systems called the guest operating systems are run on top of the host (hypervisor) which is a connection point between the VMs and the underlying physical hardware. Some hypervisor runs on the top of another Operating system and is called the host operating system (Scarfone, Souppaya, & Hoffman, 2011). Full virtualization provides the necessary hardware's interfaces and hence no modification must be done for the operating systems and its applications. In paravirtualization, the hypervisor provides the necessary interfaces for the guest OS instead of the normal hardware interfaces to be compatible with the underlying (Scarfone et al., 2011).

Virtualization architectures can be bare metal and host OS based. In bare metal hypervisors run without the host operating system. Tools availability will be reduced using such architectures, but it will reduce the potential surface attacks since there is no host OS reducing the amount of code being executed on the system. Host OS provides many utilities for controlling the virtualization Services like file-sharing, web browsers and email clients can be utilized by the VMs (Brenton, 2011).

Another form of virtualization is desktop virtualization which provides support for an application running in that specific OS and allows changes to be made in OS. Organization prefer application virtualization over desktop virtualization (Scarfone et al.,

2011). For the organization to have security over the virtualization technologies, NIST provides the following recommendations:

Component Security: Every component of the virtualization solutions should be properly secured following standard security practices, updating the software's and applying appropriate techniques to detect and prevent attacks (Scarfone et al., 2011).

Administration privileges: Utilize different techniques to manage hypervisors and limit the administrator access the virtualization system.

Hypervisor security: it is the underlying architecture of the whole virtualization solution. Disable the unused virtual hardware, disable services like file-sharing when not needed, monitor the security of each guest OS maintaining the availability of the hardware and the services to the guest OS.

Careful planning: Careful Planning of security before installing and deployment will help organizations to avoid expensive and difficult security hurdles.

Of all these components, security of the VMs is fundamental as it determines the security of the layers above (PaaS and SaaS). Adopting a secure VM image is a way to avoid the malicious images but other security issues such as sharing of the VM images, VM image backup and rollback and VM isolation should be considered as well. The two scenarios of providing security are securing the host environment and securing the VM images itself from such malicious host platforms(Pandey & Srivastava, 2014). Some of the vulnerabilities and security issues with the Virtual Machine Images are discussed below:

a. Multi-tenancy: Multi-tenancy which is a beautiful aspect of cloud helps to allocate the resources as per tenants. Since different VM instances lie on the same server or physical environment it possesses the problem of data leakage and data mixing between different instances which is termed as mixed-mode deployment VMs with mission-critical workload are placed on the same physical server along with less critical VM. The less critical VM implements fewer security controls in comparison to critical VMs. Also, if there is no separation of duties which allows unauthorized and inadvertent configuration of the VMs ("Best Practices for Mitigating Risks in Virtualized Environments," 2015). The attacker can deploy a VM on the same server and run brute-force attack against the targeted machine to obtain the system and network information.

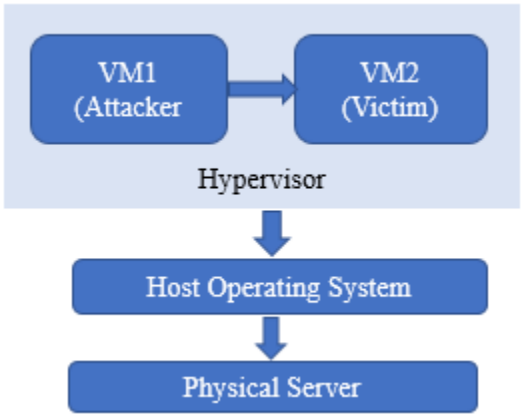

Figure 1: Multi-tenancy Security Issue

This perceived idea that VM running on the same host are isolated is technically right, but they still share a no of resources such as network bandwidth, memory and CPU. A compromised shared portion of the resources with other VMs on the network can be the entry point for attack(Glenn, Sterbentz, & Wright, 2016).

b. VM images from repository: Different cloud service providers have their own public image repositories where the retriever or consumer can use the image created and configured by the third parties. For example, Amazon Web Services has their own marketplace where there are thousands of Amazon Machine Images and community AMIs in EC2 Service. Those images may have been configured and created in such a way that they provide back-door access from an attacker (Panday & Srivastava, 2014). The provider faces the problem of hosting such malicious images which may spread over other images and repositories. Inadvertently publisher may give some critical information like login credentials which can easily be recovered and misused by the retriever.
c. Dormant VM images: Images can be running or inactive. The time when the dormant images become active again is enough for a new vulnerability to exist. Such inactive images are very difficult to patch against the new vulnerabilities. In addition, guidelines on how to activate the dormant VM images are not available. These images may also lack up-to-date access control policies and they may not be monitored for security as they are inactive which will provide loopholes for the security breach (Cloud Security Alliance, 2015).
d. Sharing of VM images: Users can create, upload and download the images in the cloud which means sharing of images is common among different users. The user with a motive to intrude may upload a malicious image and another who is concerned about privacy may happen to download the same malicious images.

In such case, the cloud administrator will also face issues of compliance and integrity of the virtual images (Modi, Patel, Borisaniya, Patel, & Rajarajan, 2013).

e. VM escaping and VM sprawl: The hypervisor which is the monitor of the virtual machine. Sometimes when an intruder runs code in the VM, it may happen that the OS running in it may escapes and interacts with the Virtual Machine Monitor (VMM) or hypervisor. When this happens all other VMs running on the same physical environment can be compromised. This uncontrolled VM is termed as VM escaping.

    VM sprawl happens when the dormant images occupy most of the resources and the active or live images are left with insufficient resources (Hussein, Alenezi, Wills, & Walters, 2016).

Literature Review of the Problem

Virtualization in cloud computing along with its advantages brings several security challenges as discussed above. Traditional security solutions against eavesdropping, DDoS and illegal invasion alone are not enough to deal with the security of the virtualization and its key components. As discussed earlier, the paper is focused on securing the virtual machine and its images. Different approaches and methods have been proposed to improve the upgrade the security level of the VM.

Cloud providers serve the purpose of centralization as well as protection of the customers VM images. In cloud supports the efficient sharing of the hardware resources but the users have no control over the hardware level. Normally, a user uploads the code and data of their workload (For example guest OS in a VM in the virtualized

solution) to cloud provider and run over the shared execution environment. Even though the VM and the workload are isolated, securing guest OS running in the hypervisor is not possible without the prior knowledge of the information and functionality of the guest OS. This along with other assumptions are made in existing cloud virtualization security techniques which raises security concern about the integrity and security of the VM and its operation (Christodorescu, sailer, Schales, Sgandurra, & Zamboni, 2009).

One important assumption made is Information about guest OS, source code and malware are known and given as blacklists. But VM can be configured with multiple guest OSes such as multi-boot VMs, and new malware can be developed constantly. Second, the guest system is clean when monitoring starts, and monitoring is done continuously throughout its lifecycle. But the points of failure here are that VM may have been compromised beforehand. More to that VM can be created, cloned and snapshots can be made and can be migrated.

This paper based on the above facts proposes an architecture without having no information about the guest OS and the source code. On top of it, they develop a technique to find out the type and version of the OS. A rootkit-detection and recovery service are also proposed to detect harmful changes in the guest kernel and restore to the valid state.

The integrity of a system is a very important task. You place the integrity monitoring system but how do you check the integrity of the integrity monitoring system itself. So, solve this problem a trusted boot process is utilized to verify such monitoring

system starting from power on. As discussed above how can you build VM introspection-based monitoring system to monitor the VM from outside. It has several challenges. There is a semantic gap between the level of detail observed by the monitoring system and the level of detail necessary to decide that the VM is secure. Also, VM lifecycle is complex. VM image can be cloned modified and migrated. This question the efficiency of the trusted boot process and continuous monitoring. To develop a secure-introspection technique following assumption are made:

1. Allow an attacker to control completely the guest VMs (Christodorescu, sailer, Schales, Sgandurra, & Zamboni, 2009).
2. Information about the software inside VMs is not provided to the cloud Service Provider
3. The hypervisor is trusted and cannot be breached.
4. VMs under the control of the cloud provider cannot be breached.

Those secure VMs under the control of the cloud provider are used as a host for the discovery and integrity solutions as shown in the figure below.

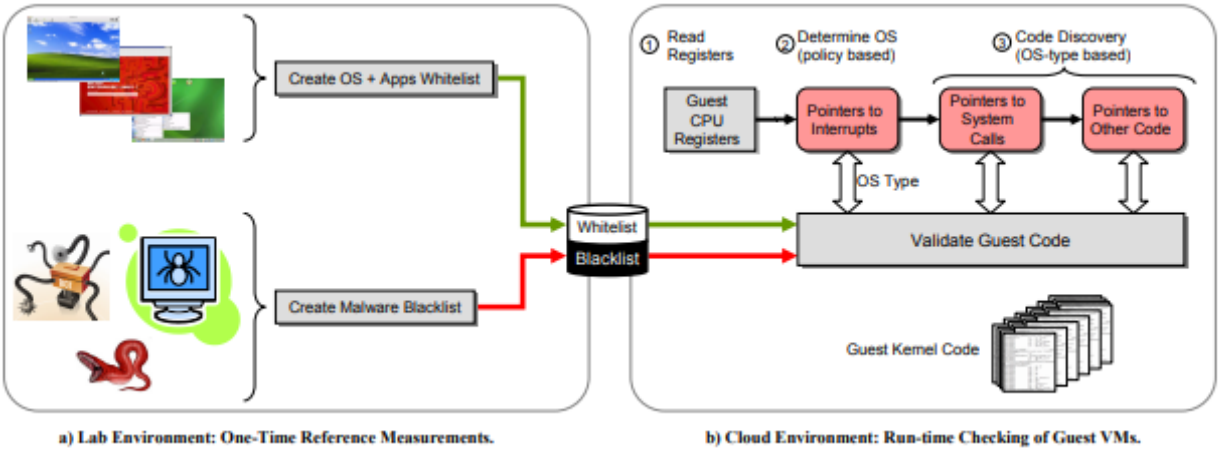

Figure 2: Overview of the Integrity Discovery System using Secure Introspection (Christodorescu, sailer, Schales, Sgandurra, & Zamboni, 2009)

Figure 2(b) shows how we identify and verify the integrity of the guest. We first read the registers in virtual CPU to find out the location. Using the hash values of in-memory code blocks and while a list of the known operating system we analyze the content obtained and determine the guest OS along with the other associated operating system structures such as a list of processes and loaded kernel modules. Using the whitelist for the guest OS we analyze all the data structures and check for modifications and validity of those modifications. Following the execution of code, we can verify the integrity of the VM during the live execution. This technique allows monitoring the VMs at any point in its lifecycle because the discovery of the OS structures depends only on the hardware state. (Christodorescu, sailer, Schales, Sgandurra, & Zamboni, 2009)

Using the while list we can monitor for the infection and already infected system. We do not need to know the guest OS and the source code since we already analyzed

from the whitelist. Assuming, hypervisors and the VM cannot be breached, this will allow to dynamically determine the integrity of all the components in the VMs.

Evaluation of the guest-OS identification was done by setting up a test network using Honeyd in a Linux VM. Honeyd was run in the generic windows machine with some services like HTTP, FTP, SSH and POP and a Cisco router with Telnet enabled as shown in the figure. Nmap was run on both the VM and OS identification code on Secure VM. “Nmap identified the honeyd “personalities” as Windows and as different network devices” (Christodorescu, sailer, Schales, Sgandurra, & Zamboni, 2009). And the code identified the VM as Linux.

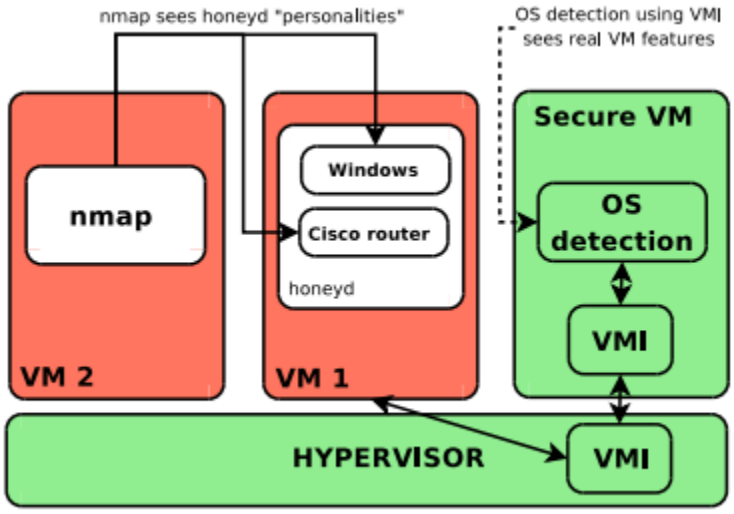

Figure 3: Experimental Setup for guest-OS Identification (Christodorescu, sailer, Schales, Sgandurra, & Zamboni, 2009).

For rootkit detection, a windows system infected with Trojan W32/Haxdoor.AU was monitored. The anti-rootkit system in placed showed changes in the six-system series and the event was detected. This also ensures that the routines on which

firewalls and antivirus rely are active and unaltered. It also detects and alerts if a rootkit penetrates these routines and fixes it. This research clearly presents the solution for where customers that run a variety of guest OS and those which need monitoring.

Wei et al. proposed an image management system called Mirage that provides access control, track the provenance of images, filtering of the images and scanning against the malware and repairing them (Wei, Zhang , Ammons, Bala, & Ning, 2009). The access control framework controls the unauthorized access of the VM images. This way the images shared by the publisher in the repository are protected for the unauthorized access. Provision tracking mechanism tracks the origin of the image and changes that are made to the images. If any operations and alteration in the images are done, the person doing so is held accountable thus reducing the possibility of the intentional malicious attempt. Sometimes the publisher may include unwanted information such as password file, login credentials and malware in the images. Image filtering mechanism will reduce the chance of publishing such unwanted information and retriever's risk of downloading such malicious content while downloading such images. The scanning feature will continuously scan to identify the bugs and it will fix such vulnerabilities. However, providence tracking mechanism introduced huge overheads in space and time (Wei, Zhang , Ammons, Bala, & Ning, 2009). Also, there is no guarantee that the scanning method will completely detect and fix the vulnerabilities present.

Hussein et al. proposed a framework gathering and the security controls from organizations such as Cloud Security Alliance (CSA) and National Institute of Standard

and Technology (Hussein, Alenezi, Wills, & Walters, 2016). He combined security control from the literature and from industry standards to develop a framework for secure sharing of VM images in the cloud. This proposed framework only provides the concept while to develop a framework with only the security measures that should be taken into consideration while sharing the VM images. The limitations of this could not be identified as it is neither tested nor implemented.

As stated by Zhang et al. security threats in virtual images are classified as protection of the execution environment and protection of the virtual image itself. That concept of running the VM in a protected environment is implemented in the Trusted Cloud Computing Platform (TCCP) proposed by Santos et al (Santos, Gummadi, & Rodrigues). This includes two types of components. Trusted Virtual Machine Monitor (TVMM**)** which runs on the node and hosts customer's VMs and provides a closed box execution environment to protect against the malicious privileged users. The next component is Trusted Coordinator which consists nodes where user VM's are hosted. It has its own Trusted Key (TK) which identifies the VM and ensures that it is running on the trusted environment. However, this prototype results in overloading of the nodes with every transaction of the Trusted Coordinator and Trusted Key.

Zhou et al. proposed a patching mechanism called Nuwa, for dormant images which was the problem of Mirage approach Nuwa uses simple emulation-based patching to patch offline images. Before patching it performs a safety analysis of the patch to ensure the script is safe. If the script is unsafe then it rewrites the patching script, if possible, and then only performs the patching. It uses the Mirage Image library

feature to perform scalable multiple patching. But, it cannot patch the suspended VM image offline, which includes a snapshot of the system memory state along with the file system (Wu, Zhang, Wang, & Bala, 2010) .

Aslam et al. proposed a protocol to launch VM securely using trusted computing technology on public cloud platforms (Aslam, Gehrmann, Rasmusson, & Bjorkman, 2012). To verify that the CSP provided platform is trustworthy, before launching, the image is pre-packed and encrypted by the symmetric key. Researchers also designed the framework to implement that protocol to show bonding of VMs on a certain platform to fulfil the security requirements. Besides that, flexibility in load management and upgrading platforms is limited by that bonding.

To verify the integrity of the VM instances and the host Schiffman et al. proposed a verification service called Cloud Verifier (CV) (Schiffman, Moyer, & Vijayakumar, 2010). The integrity of the CV is first verified by the cloud consumers and CV sends attestation requests to the Compute Hosts (CH), where the VM instances are launched via images. Then VM images integrity is checked by the CH and again by CV. After that verification, then only the cloud user will decide whether to launch the image. The limitation of this approach is “is the verification service transparent and scalable? “.

For the security of the Virtual Machine Image against channel attack and malicious executing environment, Zhang et al. proposed a protocol using encryption and hashing techniques (Panday & Srivastava, 2014). It is based on symmetric key’s component distribution with integrity and self-protection approach. First, the agent, which holds encrypted information of the images and client’s request, checks the

integrity and authenticity of the host based on resource configuration information and then decrypts the image and launch the instance. The main problem with this approach is that it is not yet implemented in the cloud framework and its performance is not known.

Literature Review of the Methodology

In case of Mirage, an image management system discussed above, to address the security issues, the security risks of image repositories are identified from Publisher, Retriever and the administrator's side and the system is proposed as shown below.

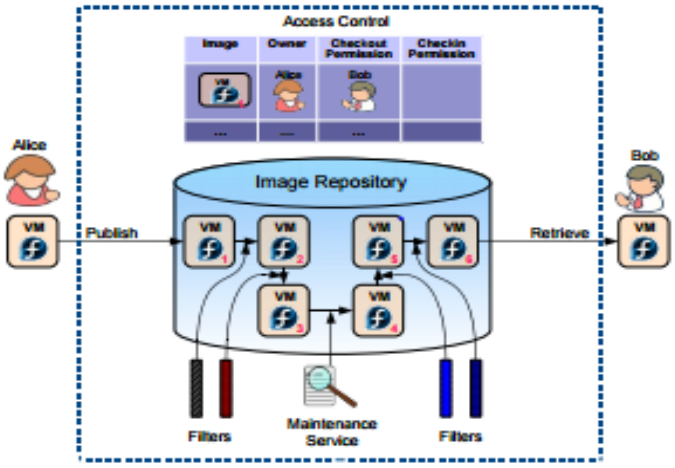

Figure 4: Mirage Image Management System (Wei et al., 2009)

Its architecture consists of four major components; access control, filters, provenance tracking and maintenance. Access control provides two permissions: check in while downloading and running the image and check out while uploading and storing the image. All these changes and deviation in the image is audited by providence

mechanism which first stores the original image when it is first uploaded. Filters are designed to be repository specific and user specific. Repository specific ensures the security best practices for the image which user-specific filters are applied right before publishing the image to avoid publisher's risk or before the retriever's downloads them to ensure it does not have malicious content. Here to avoid the storage and maintenance of similar images it is content-based, so only one content is stored if they are same even if they are from different images. It was found that performing virus scanning time in the mirage system grows slowly with the number of the image in comparison to traditional approaches.

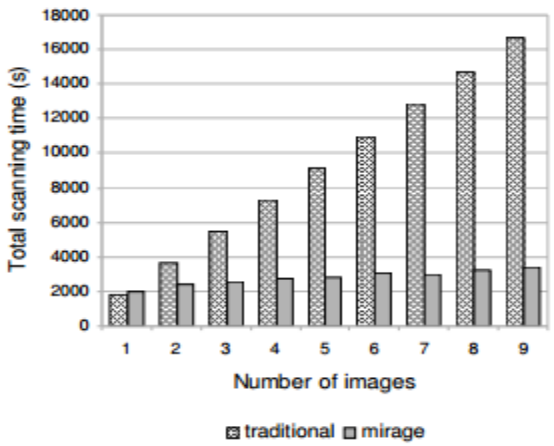

Figure 5: ClamAV (Virus Scanner) Scanning Time (Wei, Zhang , Ammons, Bala, & Ning, 2009)

Hussien et al. proposed framework based on security controls specified by CSA, ISO, NIST and so on. Such security controls were collected from academic literature as well as from industry standards and analyzed to remove the duplicates and proposed a framework with all the necessary security features for securing the shared VM image.

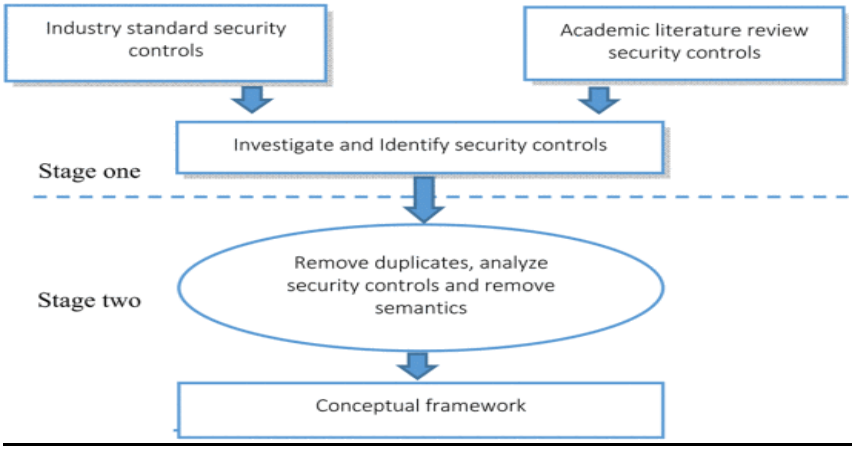

Figure 6: Framework Development Process (Hussein, Alenezi, Wills, & Walters, 2016)

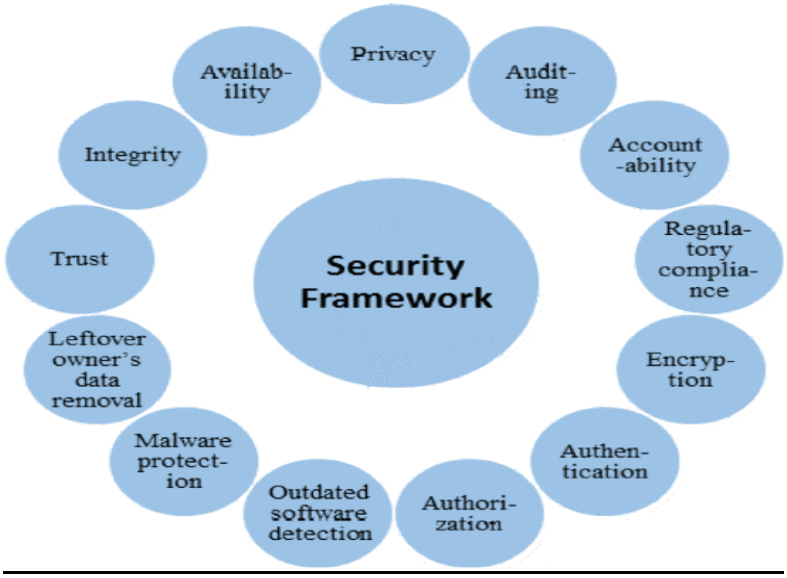

Figure 7: Framework for Securing VMI (Hussein, Alenezi, Wills, & Walters, 2016)

In the Nuwa Approach, simple emulation-based patching is used to patch dormant images in an emulated environment. Patch script analysis is done based on three concepts: impact, dependence and command classification(Wu, Zhang, Wang, & Bala, 2010). Patch Scripts is a set of command lines and arguments. Those command

lines to be executed offline are divided as safe, unsafe and unnecessary. A system is again divided into memory part if it online or running or a file system part is it is available offline depending on the availability. Those commands to be executed impacts the file system if they depend on the file system and will impact the volatile components like running processes, network and devices if they depend on the memory. While rewriting the unsafe scripts, they follow five steps:

1) Script Specialization: This ensures that the command lines should align with the arguments and the state of image filesystem.

2) Dead assignment elimination: It removes unused variables from the script.

3) Removal of unnecessary command elimination and command replacement. This technique ensures that the patching host values should match to that of the script.

4) Unnecessary Control-structure Elimination technique removes unused joint commands like *if* and *else* statements and *case* statements.

5) Final safety analysis proves that the Rewritten script is safe to be patched.

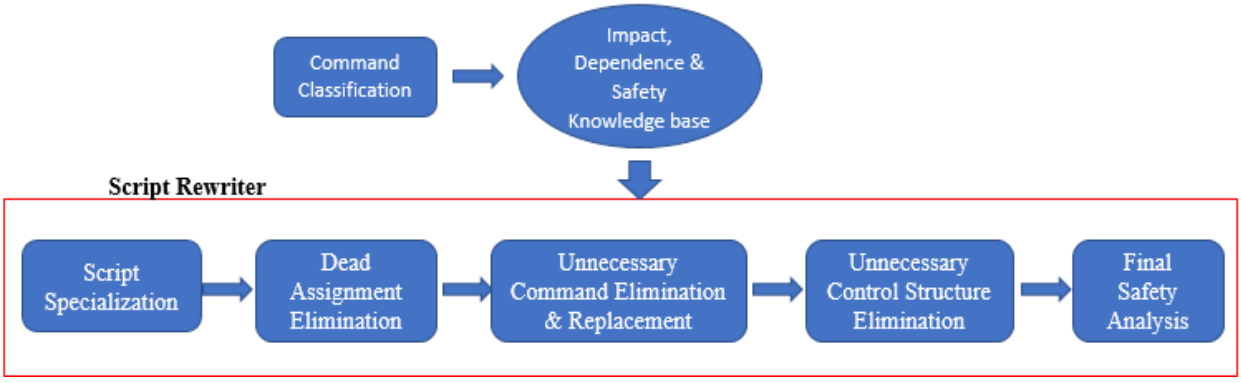

Figure 8: Flow of Script Analysis and Rewriting (Wu, Zhang, Wang, & Bala, 2010)

When moving to huge cloud environments with millions of images like Amazon EC2 then scalable batch patching seems to be very effective. Nuwa approach used Mirage Image Management System concept of storing only one of the similar files, so that batching saves time.

As seen in figure 7, in phase 1 the Mirage image extracts the patch files and imports into the Mirage. In phase 2 Virtual Mount mounts the VM images and executes the pre-installation scripts and then file replacement though Mirage Content takes place followed by the post-installation of the scripts and final check of the modified images. If rewriting cannot produce the safe script, then Nuwa resorts to online patching (Wu, Zhang, Wang, & Bala, 2010).

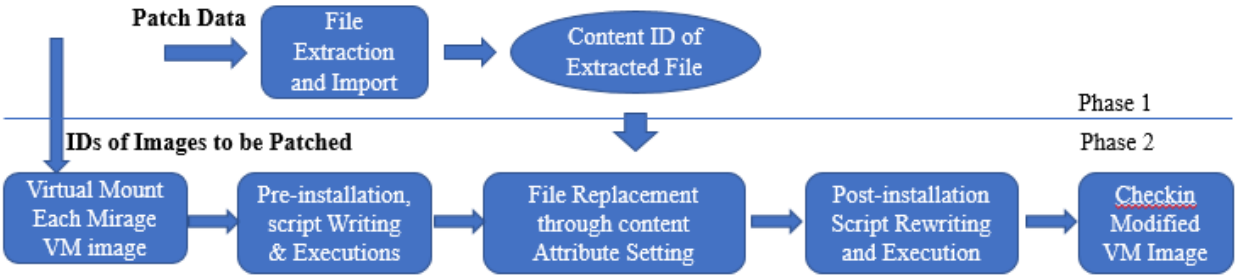

Figure 9: Batch Patching of Image via Mirage (Wu, Zhang, Wang, & Bala, 2010)

Security protocol as proposed by Anjali et al. maintains image integrity using encryption to prevent side-channel attacks and hashing techniques to prevent false resource configuration. This protocol uses cryptographic techniques to generate two key components: K1 generated by random key generator by dispatcher and K2 is created using hash operation on the resource configuration file. The architecture of the protocol consists of the scheduler which receives all the request for launching instance which is

further processed by the dispatcher as shown in figure Those instances are hosted on computer servers.

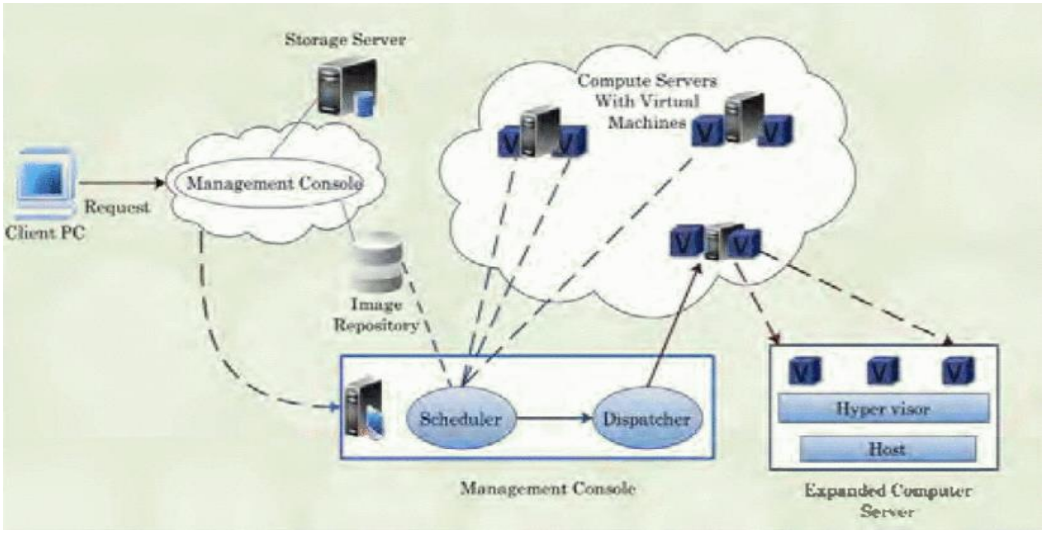

Figure 10: Architecture of the Protocol ((Panday & Srivastava, 2014)

This protocol has the following phases:

Scheduling Phase: User sends the request to launch the instance to the cloud management console. The necessary image is selected along with the required resource information to the scheduler. The resource information is hashed, and all the information is sent to the dispatcher.

Dispatching Phase: Using that information from the scheduler the dispatcher creates an encapsulated agent which is sent to the server. The encapsulated agent created on the user side consists of the encrypted virtual machine image, encrypted required resource information, encrypted key K1 given by CA (Controller Agent) when it reacts with the hypervisor, hash of image and controller agent CA and its signature (Panday & Srivastava, 2014).

Hypervisor Interaction Phase: The agent created is now executed on the server side. The hypervisor verifies the controller agent, decrypts Key K1 and generates Key K2 by hashing and send to controller agent. The controller agent then combines both keys to generate K=K1 XOR K2.

Launching Instance Phase: The key is used to decrypt the image by CA and the decrypted image is used to launch an instance.

The hypervisor sends the information about the resource used by the image to the other servers and scheduler to update the resource availability table. This way, publishing of the false resource configuration by the servers is prevented. Then, using encryption and key generation we create agents which will prevent the image from the direct interaction of the hypervisor maintaining the confidentiality of the image.

This paper proposed by Jamkhedkar et.al presents security on demand in cloud computing. The consumer requirement for security and performance of Virtual Machine varies. This framework provides a range of security options based on the threat model, that any cloud consumer can choose from. This framework also proposes the way servers are used to provide the desired security protections with a different model. To provide the service accurate mapping of security options and enforcement capabilities of the server should is required. It also provides lifetime virtual machine security by using VM live migration. When the security of the virtual machine is disturbed then it is moved into a new server. For testing the average time and downtime for such VM migration a prototype is implemented in an OpenStack cloud system.

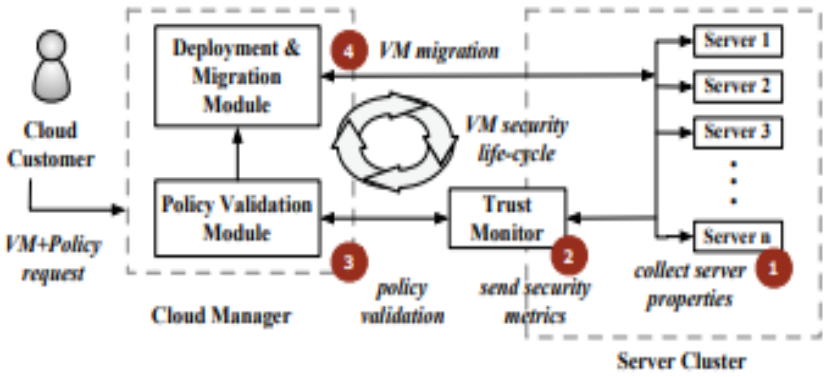

Figure 11: Security on Demand Framework (Jamkhedkar et al., 2013).

On-Demand Security (ODS) architecture above assumes that the cloud provider has different types of servers for different threat model. The VM is deployed on servers with the security options as demanded by the customer. Then VM enters security life-cycle in which the following steps occur:

- Trust monitor collects the security properties of each server.
- Trust monitor translates those security properties into security capabilities.
- The property validation module validates customer request of security policies against these capabilities.
- Then the response mechanism is triggered when the security requirements are changed or VM is not trustworthy. The response mechanism normally involves migrating VM to another server with enough security features.

This framework provides seven different security requirement types as listed below for the customer to choose from. Some of the security requirement types are:

- Basic Security: VM is protected from another VM.
- Hypervisor Protection: Protection against untrusted hypervisor.

- HW Protection: Protection against HW attacks and so on.

A customer can combine from the different sets of security features to develop a Security Request Matrix. That security request matrix is used by the providers to allocate the appropriate servers that can fulfil mentioned security requirements. The role of Trust monitor is to collect, monitor and maintain the security capabilities of the server. Trust monitor resides inside the server. The collected information about the server is then passed to the policy validation module which collects the input from trust monitor and security policy specification from customer to deploy the VM in the appropriate server. There are 2 modes in which the policy validation module runs into. One is the above-explained deployment module and the other is the relocation module which is activated for VM migration to a new server when the initial server no longer serves the purpose. VM migration helps to avoid attacks from malicious VM that were located when the VM was previously deployed.

The implementation of the framework is done on OpenStack Essex platform as shown in the figure.

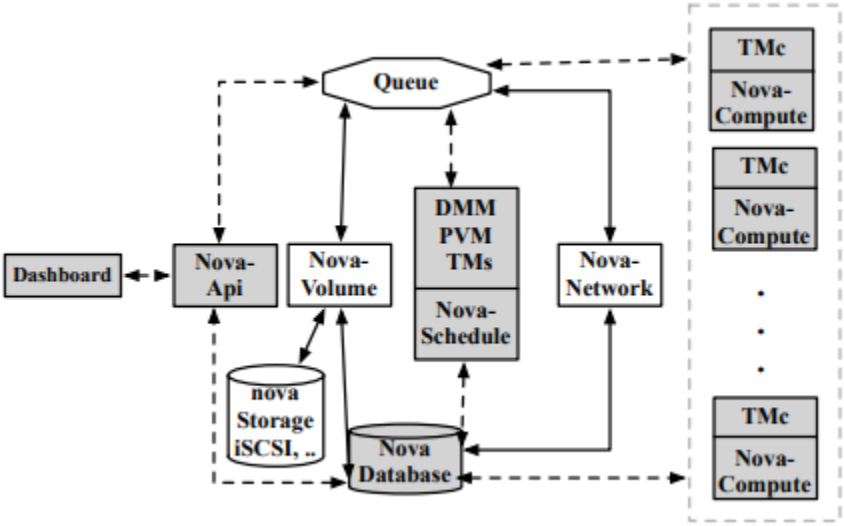

Figure 12: Implementation of Security on Demand on OpenStack Nova Components (Jamkhedkar et al., 2013)

The Nova modules and data paths were modified to add the features of Security on Demand framework. The key performance indicator is whether the VM migration can be done within a feasible time. VM was migrated to 7 different servers as shown in the table below to test the performance.

Table 1: Migration Performance with Different Servers. (Jamkhedkar et al., 2013)

Benchmark	Total Time(s)	Downtime(ms)	Data Sent(MB)
Mail Server	8.9	300	800.39
App Server	7.5	4050	728.21
File Server	7.2	250	697.87
Web Server	7.2	250	645.33
DB Server	7.4	450	717.52
Stream Server	8.0	650	716.21
Idle Server	2.6	200	212.34

The total time is the time between the moment when the migrate command is issued and the VM resumes at the destination. The downtime is the measure of VM

liveliness or, in other words, when VMs are paused. The data column represents the amount of data sent during migration.

The above results suggest how effective VM live migration is with the use of this framework and how it can be used not only to avoid VM-level attacks but also Server-level and cloud-level (Jamkhedkar et al., 2013).

Summary

The above chapter covers the systemic overview of past researches and methodology developed to maintain the security of the VM images. The security of VM images does not meanwhile it is running only but it should cover security in its dormant state and while it is migrating as well. Moreover, image repositories and hosting platform in which we launch the instance should also be trustworthy. The following chapter will explain the methodology to develop a framework considering the software and hardware requirements.

Chapter III: Methodology

Introduction

This chapter will discuss how the information will be collected to address the research questions and how the research will be carried out. No data collection is done instead we will analyze the cases and facts in the form of the case study that we retrieve from different research paper and journals. No conclusion is made from the facts obtained, it will only be used for analysis without breaching the confidentiality of the researchers and authors. This includes an in-depth study of incidents which will expose those hidden problems to identify the real cause of such incidents. This research paper will try to provide the guidelines to address those issues with the preventive recommendations.

This research paper will specifically discuss the different types of VM images based on different characteristics. VM images can be categorized based on activity: active and dormant images, virtualization and cloud computing services (Amazon Machine Images, Google CE and Azure VM images), based on Operating System: Windows and Linux Operating System and those images can be on public, private or hybrid cloud and based on the sizes available with different cloud services. The reason for choosing VM images with different characteristics and types depend on the need of clients. For example, Amazon EC2 has a high temporary storage limit of Up to 48 TB compared to Microsoft Azure's 4 TB. Amazon VM and Microsoft Azure support a wide range of database in comparison to Google CE. VMs in the public cloud is managed and maintained by the provider, whereas in the private cloud the company or the client

is responsible for the management and maintenance of the VMs. They are protected with the firewall and are more secure than private cloud.

Image types will be reviewed, analyzed in depth and all the related security attributes with those images will be discussed in detail. Attributes like authentication, encryption, access control and privileges, malware protection, connection channels and so on.

Process and Tools

A qualitative approach was followed in the process of addressing the research questions. This method of study will be used to gather information from different industry and academic standards and analyze the obtained resource and information. This will help to come up with guidelines and to develop more secure VM images with some recommendations. This method to acquire resources from different standards and past study is adopted because it clearly considers the fact that those information and requirements were collected, implemented, and analyzed. The literature review in this research paper is based on different articles, journals, research papers, and different trade journals published by industry experts. Source of information also includes the resources available from St Cloud State University library website and all those published articles and thesis. Two-thirds of the research contains the journal articles and research papers and the rest of it includes information from blogs and trusted websites.

The methodology used to investigate or address the research problem is suitable for evaluating since there are no security aspects related to virtual machine and the

infrastructure it is based on. This specific research methodology is more helpful to study the above-mentioned research questions and hypothesis. The analysis of all the theories, investigation and publication helped to come up with the possible solutions for a secure virtual machine. I have utilized the research papers that provided me a good analysis of the security problems and their approach to addressing those problems.

The research did not conduct any lab or practical experiments that was outside the scope of the starred paper. It always provides room for future research to cover all aspects in a broader term and to carry out lab experiment and implement as a thesis.

Result Analysis

Although the use of the image is very effective from the cloud providers and subscriber point of view as well but the risk that it brings is quite high. There may be specific problems based on the different cloud environment and the types of images, but the following are the problems that were common in most of the cloud images.

Image Security risks

As discussed in NIST Special publication 800- 190, an image with vulnerabilities, defect configuration, embedded malware, embedded secrets, untrusted software and insecure registries, is at risk.

Image vulnerabilities

Images which are not running or are not active are called dormant images. These dormant images, as discussed before, lack security updates and software and applications running in them may be outdated. When running instances from those

images, there is always a risk that they have developed vulnerabilities which will be inherited in the instances and the images created from those instances.

CSP or the image may build an image around a software or an application which has a vulnerability, which opens the door for the attacker. Such vulnerabilities not just inject the malicious files but may attempt to elevate the privileges and may abuse the host itself to attack other machines in the host. Software running on windows images are more vulnerable than in Linux image.

Most of these vulnerabilities are inherited from the third-party images because the users do not properly analyze the security aspects while using such images. The vulnerabilities can be less vulnerable and may also include critical vulnerabilities like remote code execution(Balduzzi, Zaddach, Balzarotti, Kirda, & Loureiro, 2012). If such vulnerabilities exist, then the attacker used a network probing mechanism to gain identity information about the target instance such as IP address. Then, the attacker sets up his machine in the same host if possible and tried to brute force to get into the machine. After the successful attempt, the malicious instances are used to run side channel attack into the targeted VM. To sum up the above concept, the problem is multi-tenancy.

Image Configuration

Poor configurations in user and privileges always provides the backdoor for the images to be compromised even though the image is free from vulnerabilities. For example: If Images configured as root is made public (Souppaya, Morello, & Scarfone, 2017).

Embedded malware and secrets

Images are the files that are used to create VM. These can include malicious files intentionally or unintentionally within them. There is a chance of such risk if the image is provided by the third parties whose provenance is unknown. (National Institute of Standards and Technology, 2017). These images with the malicious files may be used to compromise the host environment or other images within the same physical environment. It is found that the Windows environment is more prone to malware like viruses, spyware and trojans then Linux Operating systems. These malwares can perform keylogging in your operating filesystem, stealing the data and files from the machine. They are also capable of decrypting the hashes of password files and recover the passwords from the browser history (Balduzzi, Zaddach, Balzarotti, Kirda, & Sergio, 2012).

Untrusted Images

Images possess the high risk of malware, data leakage and other vulnerabilities when untrusted and outdated software's are run in the images. Portability and reusability of the images increase the chance that the users may run the images from the external sources that is not validated and trustworthy. For example: when troubleshooting a problem with the application configuration in the images, there may be updated and upgraded applications which do not need for troubleshooting, in the images provided by the third party. This compels the user to move for those images and which possess several risks. (National Institute of Standards and Technology, 2017) Portability of images saves times and is efficient but also VM images are packages in a

run-time format with hard disk images and configurations that is suitable for a hypervisor. Even though it is supported with any cloud platform, but it may not be 100% compatible which lefts the room for vulnerabilities.

Backdoor and leftover credentials

Most of the Linux machine images used SSH keys for remote service. The public part of the SSH key is handed to the CSP when a user rents an image. For example: when AMI is rented, a user's public key is stored in *authorized keys* in the home directory by Amazon. "If a user is malicious and if a user does not remove her public key before making that image public, then the user can log in into any running instances of the AMI" (Balduzzi, Zaddach, Balzarotti, Kirda, & Sergio, 2012). Even if the attacker does not have an idea about the live machine he can run brute force attack on all the instances created from the images to find out the IP address of the running instance to try to use the credentials they got. Gaining access to the IP address they can send traffic to flood the instance and carry out Denial of service attack as well.

More to that, SSH server provides password-based authentication in addition to SSH keys. Likewise, as discussed above, if the password information is not removed while pushing the images in the cloud then the image can easily be compromised. More to that, password provides a large attack vector than SSH authentication because for SSH authentication only the user with the private key, normally the developer or creator of the image, can access the image. The password authentication mechanism for signing in for the resources in the cloud is not enforced. So, anyone can use a

password cracking mechanism to extract the password from the hashes in the images to crack them (Balduzzi, Zaddach, Balzarotti, Kirda, & Sergio, 2012)

Even if the login credentials are left over inadvertently, third parties or the malicious user can utilize such weakness to create a big security problem over the whole cloud environment.

Unsolicited connections

The cloud instances are an open connection to other external applications or applications within the cloud environment. It is very difficult to find out that the connection is legitimate, or the connection is malicious. In a Linux machine, the syslog service is used to log different types of events to a syslog server. It can be the logging information of the incoming traffic, connected hardware or the login and logout information to the machine. All these log files are stored in the var/log directory. These connections if analyzed provides a lot of information about where the connection is going. For example, the connection can be for some software updates or it may be a connection to an attacker's server which will be used to gather information about the images such as IP address of the machine. That IP address can later be used to carry out the attack on the instances via backdoor.

Privacy Risks

In the public cloud, virtual machine images are shared among the different organizations or users. The creator and the publisher who publishes the images may include the SSH private keys or forgot removing access keys that are used to start the instance from the image. More to that, the attacker can exploit the shell history, login

attempts and browser history to de-anonymize the user and the image creator (Balduzzi, Zaddach, Balzarotti, Kirda, & Sergio, 2012). The user who is using the VM images for running instance may provide inadvertently increase the risk of adding vulnerability to the images or provide a backdoor to the attackers. Images of those instances, when shared in the cloud, inherit all the vulnerabilities and security loopholes along with it (Livraga & Zhu, 2017).

1. Private keys: Cloud consumers use SSH keys to identify if the image they are using is authentic and to authenticate the API calls sent by the user to the VM Management interface of the cloud providers, API keys are used. By default, images use a certain filesystem to store those keys. For example, Amazon uses id_rsa for the SSH keys and *.pem as the extension for the AWS API keys. These API keys are not password protected. So, once the attacker gets hold of the API keys, he/she can launch the instances from the image. More to that, although, protection of SSH keys with the passphrase is the best practice, but it is not followed more often. It is common that the users use the same keys to access the other images or other virtual machines from the same image. If an attacker gets into one image, then there are chances that the other images owned by the user can also be compromised. Form the log files, the attacker also can use the failed login attempts to figure out the passwords or at least try to guess it. (Balduzzi, Zaddach, Balzarotti, Kirda, & Sergio, 2012)

2. Logfiles and shell history: The log information obtained from authentication of different applications are stored in log files. For example: When accessing the web application, HTTP GET requests which contain login information and passwords are stored in log files.

Also, many common shell history files are left on the image during their creation. Balduzzi et al., found that of 612 Amazon machine images tested they found that around 12% have at least one shell history files. Of all those files 74 different authentication credentials were found in the command line in those log files. These credentials can easily allow the attacker to compromise the components and services hosted in the images in the images (Balduzzi, Zaddach, Balzarotti, Kirda, & Sergio, 2012).

Remote credentials like DNS management password will allow the attacker to modify and access the Domain Name Server (DNS) and change the DNS configuration and redirect all the traffic from the host to his machine (Balduzzi, Zaddach, Balzarotti, Kirda, & Sergio, 2012).

3. Web environment and browsers: Web Browsers and its history is also another concern for the images. Browser history files and information can be de-anonymize to get the information about the user and the image creator. (Balduzzi, Zaddach, Balzarotti, Kirda, & Sergio, 2012). Many browsers will have add-ons that are not updated, which will lead vulnerabilities to persist. Use of social media on browsers will invite the risk on the browsers and its

platform. This will invite backdoor trojan, keystroke logger or other malware while the user uses browsers for accessing the images (Jansen, 2011).

4. Recovery of Deleted files: So, what does it mean when we say delete the files from the image before publishing it? Deleting a file does not destroy the whole file, rather it will prevent from accessing the file. To completely delete the file, the file should be overwritten. Using file shredders, the user can overwrite the files, but it should be done many times to destroy completely all the traces of the file. Still, not all file shredders will deal with the slack spaces. So, by utilizing forensic tools one can recover the traces of original file even after wiping or shredding. As discussed in the paper by Balduzzi et al, where the paper focuses on the Amazon cloud images are created by a process called bundling. A user follows the following procedure of the bundling process to create an image 1) create an image from the mounted filesystem, 2) compress and encrypt the image and 3) split it into manageable parts so that it can be uploaded for storage (Balduzzi, Zaddach, Balzarotti, Kirda, & Sergio, 2012). It has two levels of bundling: filesystem and block level. Amazon uses for methodologies of bundling to create the images and as shown in the table out of four bundling method, three of them are prone to undeleting attack.

Table 2: Different Methods of Bundling in Amazon Cloud (Balduzzi et al., 2012)

Bundle Method	Level	Vulnerable
Ec2-bundle-vol	File System	No
Ec2-bundle-image	Block level	Yes
From AMI Snapshots	Block	Yes
From VMWare	Block	Yes

So, it suggests that there is a big security concern regarding the recovery of the deleted files. It shows that one can easily use commands like *extundelete* and *windundelete* to recover the contents from the deleted files.

Image registries

The images, once created, are pushed to the registries for storage. Then managing the registries is equally as important as managing the security of the images.

1. Configuration of Access control in registries: In the cloud, registries use buckets to store the images. For example, Google has a cloud storage bucket to store the container images and Amazon EC2 Container Registry.

   Improper configuration of the permission and roles in the registries will surely provide the backdoor to compromise the images. As for example, an image/storage object viewer should only have the read-only permission.

2. Access control governance: To ensure only the valid users can access the cloud resources, CSP defines access control governing policies that include risk management, governance and compliance.
3. Insufficient authentication and authorization restrictions: Authentication methods used to push, or pull is also a concern. It is always essential to use secure and short-lived access for the images such as access token. Registries contain images that are very sensitive and are used to run proprietary applications and to access sensitive data. Insufficient authentication and authorization in the registries will lead to intellectual property loss and expose to the technical details of an application that will help the attacker to compromise the images. Moreover, registries store many the image and if a registry is compromised it will lead to the compromise of the other registries and the host environment (Souppaya, Morello, & Scarfone, 2017).

   Different authentication techniques such as password authentication, Trusted Platform Module(TPM) based authentication and implicit authentication. Weak password practice such as regular use of the same password increases security risk. Password authentication is susceptible to dictionary attacks since the password functions are stored in the server. In TPM, a hardware-based security uses crypto-processer to store cryptographic keys. It is susceptible to a cold boot attack. An intruder once escapes the disk encryption and reveal master password with social engineering.

Another authentication based on user's behavior such as location, application usage and motion, is implicit authentication. Data that are collected for profiling will disclose confidential information (Tan Fong, Ang, M. L Mat, Kiah, & Shu Yun, 2017).

4. Insecure network connections to registries: Images contain sensitive information like organization's proprietary software and secrets. Poor or insecure network connections to the registries for the access of the image and its content always possess the risk of data loss and unauthorized alteration of the images. Attackers can intercept the network to carry out man-in-the-middle attack and steal the developer or administrator credentials and may inject malware contents in the image (Souppaya et al., 2017).
5. Dormant Image in registries: Images that are not used for running the instances falls under stale or dormant images. These images cannot or are not updated with the latest security patches and have outdated version of software and applications. Such images, when pulled out of the repository, increases the chances of a vulnerable version of the virtual machine (Souppaya et al., 2017). In addition to that, tool and techniques used to secure the normal files should not be used in securing the dormant images because it is not just a file, but it contains the OS and the system requirements to run a virtual machine.

Solutions and Recommendations

Image vulnerabilities

Organizations or users can use image-specific vulnerability tools and process to detect the vulnerabilities present in applications and components of the image. Tools used should not be specified to a certain design of the image because tools and software used for vulnerability checking give a false sense of security if there is a certain change in the architecture in images. Countermeasures for such vulnerabilities include:

1. Detecting the vulnerabilities early in the image development process and implementing solutions to prevent such images from being deployed in the cloud environment (Souppaya et al., 2017).
2. Vulnerability monitoring and detecting mechanism should not be limited to the base layer where the OS lies but it should cover from base layer to all the software and the application that is used in the images.
3. Organizations who use the image and the Cloud Service Provider hosting the images should only allow the image which meets the organization's vulnerability and configuration policies. The CSP and the subscriber can use the Common Vulnerability Scoring System (CVSS) to determine if the image is vulnerable or not.
4. To detect the outdated software and old images that needs patching and updates, organizations can use different approaches. The customers when initiating the virtual machine can use Update Checker which checks for the

outdated software and applications regardless of the machine is running or is dormant. This approach is only suitable for the Linux machines. The update checker has the centralized database which contains all the information about the software packages, their version that is in the images. If any new version or new software is installed in the machine, then the information is updated in the central repository. The update checker takes information from the different databases like package DB, metadata DB and repository Cache and matches the software packages to detect if there are any outdated software. Those results are passed on to the user and it also gives information about the which software should be prioritized.

Another approach that can be used as discussed before is Nuwa, patching of old an offline image. It implies the patch script to the software that are outdated. It enables scalable patching to many images in a batch. To make sure that the patch script is also risk-free, Nuwa performs a safety analysis and rewriting of the script.

Image configuration

Security recommendations to avoid image configuration defects include:

1. Following up standards and best practices to configure the images. For example, an organization can always adhere to NIST special publications for cloud computing.
2. Enforce compliance requirements, images which are not in compliance can be rejected.

3. Use base layers that are stable, updated and those with less attack surface areas. For example, Select the base layers from minimalistic technologies like Alpine Linux and Windows Nano Server (National Institute of Standards and Technology, 2017).
4. The network configuration and the remote administration tools for the images and VMs should not be enabled from the machines. This is because they possess a greater threat to network-based attacks. Instead, remote access and the network configuration should be run through the runtime APIs which may be accessed via orchestration tools or by creating remote shell sessions on the host on which the instances are running (National Institute of Standards and Technology, 2017).

In addition to the above-mentioned security configurations, there are other security measures that can be employed during the configuration, deployment and the networking of the security images, especially in marketplace. Limit the attack surface by minimizing the footprints with only necessary Server roles, features, services and networking ports. It is highly recommended to remove the bass/shell history, SSH keys, log files and unnecessary certificates. Do not use the LVM (Logical Volume Management).

The image should include SSH server by default and should not contain any custom configuration such as resolv.conf. It is recommended to create a single partition on the OS disk while deployment (Barclayn, n.d.).

Embedded malware and secrets

To minimize the risk through malware and secret keys and certificates following security measures can be implemented.

1. Monitoring process includes all know malware signatures and behavioral detections.
2. Secrets should be outside of images and provided only during runtime. Integrate secret management system in the hypervisor to store the secrets. These tools provide APIs to the hypervisor to retrieve secrets. For the images, they do not have to manage those secrets and if any attack is done, no any secrets can be leaked.Provide the encryption on a need to know basis to the images and the VMs and always encrypt the image with approved cryptographic algorithms (National Institute of Standards and Technology, 2017).
3. Configurations settings of Images obtained from third parties should be carefully analyzed and follow best practices to avoid any risks.

Untrusted Image

To mitigate the risk of using untrusted images or the components in it, the following approaches are recommended:

1. To deal with the security issue coming from the portability and reusability of the images from untrusted sources implement a centrally-controlled mechanism to determine their validity before they run in the cloud environment.

2. Implement cryptographic signatures to identify and validate the images have not been tampered.
3. Run and store the images only from the approved list of developers when it comes to third-party images.
4. Continuously monitor and maintain the repositories for outdated images or image that requires updates in their software and configuration. Even if the login credentials are left over inadvertently, third parties or the malicious user can utilize such weakness to create a big security problem over the whole cloud environment.

Backdoor and Leftover Credentials

Tools to detect all the credential exposed within the image should be implemented by the Cloud Service Providers. For example, AM exposed tool can be used for scanning AMIs for common credential leakage. This will look for the presence of SSH authorized-keys or SSH identity keys that will allow the potential backdoor and access to other hosts. Such tools can also be used to detect the private keys and certificate files present in the images (Feinstein & Jarmoc, n.d.).

1. Password Complexity if not enforced in the cloud should be strictly enforced. To add more security to that, multifactor authentication will certainly reduce the risk of image compromise**.**
2. Frequent rotation for SSH keys and certificates should be. A key rotation will generate the new cryptographic key and will back up the old key to decrypt the information it has encrypted.

3. Organizations should be clear and identify the normal users and the superusers while giving access to the images.
4. Protocols for data portability needs to be enforced.

Private keys

By analyzing the security concerns, the following measures can be utilized to remove the security concerns of the image associated with private keys.

1. Users should always use IAM services and do not use the same APIs keys for all the resources to minimize the risk of APIs keys being exposed.
2. Always look for the SSH configuration. Protection of SSH keys with the passphrase is the best practice but it is not followed more often. A customer when using the public images should check for the authorized-keys in every home directory and delete all the public keys present.
3. Use CSP provided inbound firewalls to restrict the IP range to block login via SSH.

Log and shell history files

Following are the countermeasures that can be used to protect the virtual machine images from leakage of information through log and history files.

1. Provide tutorials and educational videos to the customers to help secure use of shared public images.
2. Publisher of the images should always scan the log files and delete them before publishing into the cloud repository.

Web browsers and history

Users and organization can employ hardening of the browser environments. Organization can encrypt network exchanges against keystroke logging (Jansen, 2011). Users can have their built-in browsers in their own virtual environment and should be used only for the specific service. For example, Banks can implement this strategy to access very secure transaction or a bank service. This will help to minimize browser attacks and session hijacking.(Dunn, n.d.)

Protection against file recovery

Files once deleted are not deleted completely from the filesystem instead they are made unavailable. So, administrator and users should always look for ways to prevent the forensic analysis of the deleted image information. Following countermeasures are helpful in preventing deleted files recovery.

1. In the case of S3-backed or filesystem backed images, API keys are used for bundling authorization. Use extended bundling command in such a way that it raises an alert if the API keys are included in the bundling. For example, as proposed by Bugiel et al., the bundling tool below issues a warning if API keys are included in bundling (Bugiel, Nürnberger, Pöppelmann, Sadeghi, & Schneider, 2011).

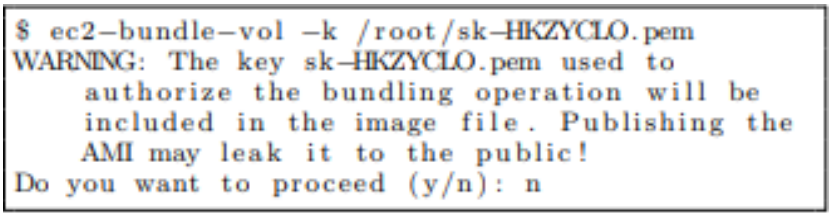

Figure 13: Improved Bundling Tool (Bugiel et al., 2011)

2. EBS-backed images or volume level images can be easily generated from the instances and can be made public. They need a different approach then filesystem backed images. CSP can provide a certain mechanism to inform the users of the potential risk of private data being included in the AMIs and provide tools to prevent forensic analysis of the data (Bugiel et al., 2011).
3. File shredders which the capability not just to delete the files but to overwrite those files with some random data.

Image Registries

Countermeasures for authentication, authorization and access control configuration

Following security recommendations are useful in minimizing the security concerns related to authentication and access control:

1. All access to the registries should be authenticated and classified according to the roles of the user. Access writes to the image should be provided on a need-to-know basis. For example, Developer of the image should be only allowed to push an image to a certain repository instead of granting them access to all the repositories (Souppaya et al., 2017).
2. Logging and auditing of the write access to the registries and careful analysis of that login information. Integrate the automatic vulnerability scanning and compliance assessment before they are being pushed into the repository and are being deployed. This will reduce the chance of using the vulnerable and non-compliant images (Souppaya et al., 2017).

3. Weakness in password authentication can be reduced with randomization of the password. Password authentication protocol must be challenge-response type so that the server will not have the password function to avoid dictionary attack on the server. Asymmetric Password-Authenticated Key Exchange(APAKE) is the solution where the client only knows the password (Tan Fong, Ang et al., 2017).

A platform proposed by R. Chow et al. supports the authentication decision based on an authentication score calculated on the user's behavior. This authentication method provides a balance between transparent user access and trust in the security of the method (Chow et al., 2010).

Access Control Governance, risk management and compliance

User certification should be provided to verify the roles and privileges to the resources. Based on those roles, privileges and behavior risk score can be calculated which are certified by risk score-based certification. This will eventually provide effective governance and administration. Good organization policies and implementation would avoid phishing attacks and loss of control which in the end would help maintain compliance (Indu, Anand, & Bhaskar, 2018).

Countermeasures against insecure network connections

1. Users or Organization should configure the development tools and services to only connect to registries over encrypted channels. Pushing and pulling of images from the repository or registry should be done

through an encrypted channel. Organizations should utilize Hypertext Transfer Protocol Secure (HTTPS) to have a secure connection and the image at rest should also be encrypted. For example, Amazon utilized server-side encryption to secure the images at S3 (“AWS | Amazon Elastic Container Registry | Product Details,” n.d.).

2. Registries should be timely updated and checked for unsafe and outdated dormant images that are no longer in use. Such images should be removed from the registries.
3. Always specify the version of the images while naming the images. This will help the organization look for the old version and remove them after they were replaced with the new ones.

Dormant Image in Registries

Organizations should follow following measures against the use of unsafe and unused images:

1. Organizations should categorize the unsafe and unused images and work on downsizing them or remove them from the repository if they are no longer be used. This can be done by automating the process based on time triggers and labels associated with the image (Souppaya et al., 2017).
2. Tags provided to the images should be configured in such a way that it will provide you with the specific version so that it will be easy to understand which versions of the images are outdated (Souppaya et al., 2017).

Hardening of Virtual Image

Hardening of the image is a process that will limit or remove the use of unnecessary applications in a VM images to reduce the vulnerability that can be exploited by an attacker. The hardened images are more secure than the original images. Some of the ways to harden the images are as follows: (“ Cloud Computing Design Patterns,” n.d.)

1. Close unused or unnecessary server ports
2. Disable unnecessary internal root accounts and services
3. Disable guest access to system directories
4. Uninstall redundant software
5. Establish memory quotas

The cloud provides will implement the above-mentioned ways to harden the image and the hardened images is saved in the VM repository. Hardening of images will help against insufficient authentication and DOS attacks.

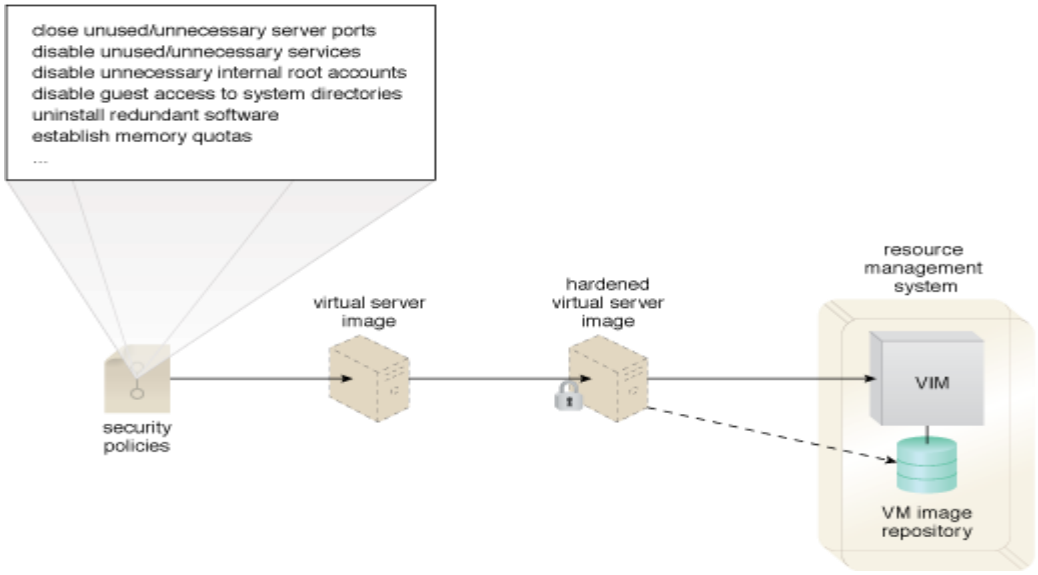

Figure 14: Hardening of the virtual image. (“Cloud Computing Design Patterns,” n.d.)

Summary

This section described the methodology involved in the research. The next chapter highlights the main content of the study and the necessary information. Different types of security concerns associated with the cloud images were discussed in relation to the different cloud service providers based on the research and information available. The chapter after analyzing all the security concerns has provided with the recommendations to minimize or to handle such risk while using the cloud images. Those recommendations are providing with a very deep research and thinking.

Chapter IV: Analysis of Results

Introduction

Images in the cloud are basically divided into two types. Images that are provided and maintained by the cloud service provider, open-source communities and third-party vendors are called public images. All the projects and everyone can use such images to create the instances. Customers have trust issues in the safety and security of such images.

Custom Images are available to a specific set of accounts and are kept private. A subscriber or a customer can create such images from boot disks or other images and updated the changes made in the configuration to run instances

Security is one of the biggest challenges in the cloud environment itself which implies in case of the cloud images which are used to initiate the Virtual Machine. In this section, the paper tries to focus on the concerns and risks present in the images to create a virtual machine image. How the attacks happen on the image is out of the scope of this paper. The focus of this research paper in the public images which will cover the risk and issues of the private images as well. Analysis of the image is basically done with the Linux and Windows images.

Use Cases

This paper will use the findings of experiments and security breaches of an image as a scenario for the describe the use cases. Use cases will describe in steps, the security concerns of the VM images from the time of creation to the launching the

instances and suggesting the solutions or the recommendations to solve such kinds of problem.

The scope of these use cases is to suggest the users of the image understand the security concerns present while using it and how those issues can be minimized or prevented so that they can launch a trustworthy virtual machine.

To carry out tests of the images is out of the scope of this paper because it is expensive, massive technological knowledge and time-consuming. Moreover, it is unethical to carry out the testing in the cloud since the image that is used for the experiments contains very sensitive information.

Image Privacy Risk

Intent: A cloud user wants to use a safe and secure VM image. This includes the image to be safe from vulnerabilities, threats, malware and dealing with private keys and left-over credentials.

Scenario: Balduzzi et.al, used an automated system to test the Amazon machine image. The test was conducted for five months. The system consists of the 3 components:

1. Robot: It initiates AMIs and fetches the login credentials. It was configured to look for common usernames like Root and Ubuntu in Linux machines.
2. Remote Scanner: It looks for open ports using Nmap tool.
3. Local Scanner: It used Nessus tool to run vulnerability scanning tests with administrator privileges. The test was categorized as general, network, security and privacy.

Of the total, 5,303 Amazon Machine Image were analyzed. This included Linux machine images and windows machine images from different data centers located around the world. Following results were obtained from scanning of the Amazon Machine Images by Balduzzi et.al (Balduzzi et al., 2012).

Findings 1: Software vulnerabilities: 98% of the windows and 5% of the Linux AMIs have critical software vulnerabilities such as remote code execution. Those vulnerabilities have been found to be more than two years old.

Findings 2: Utilizing ClamAv tool after analyzing the filesystems with 850,000 malware signatures 2 windows AMIs were infected with Trojan spy and Trojan agent.

Findings 3: In two AMIs*, /var/log* directory showed that the syslog daemon was configured to send messages to the remote host *whictheh* was not in control of the user.

Findings 4: Login credentials and private keys were left over intentionally or forgotten. With the result from Nmap scan and matching the SSH keys obtained from the AMIs, over 2,100 AMIs were identified.

Table 3: Left Over Credentials per AMI (21% of the scanned AMIs shows left over credentials)

	East	West	EU	Asia	Total
AMIs (%)	34.8	8.4	9.8	6.3	21.8
Password	67	10	22	2	101
SSH keys	794	53	86	32	965
Both	71	6	9	4	90

Findings 5: Of total AMIs scanned 612 AMIs (around 125 contained at least one shell history files. One those files 74 different authentication credentials were found.

Table 4: Credentials in History Files

Findings	Total	Image (local)	Remote
Amazon RDS	4	0	4
DNS	1	0	1
SQL	7	6	1
MySQL	58	45	13
Web App	3	2	1
VNC	1	1	0
Total	74	54	20

Finding 6: It was found that the data and files were recovered from the deleted files which is a concern for public images. It was also found that deleted files were able to be recovered from an official image published by Amazon itself.

Table 5: Recovered Data from Deleted Files

Type	#
Home files (/home, /root)	33,011
SSH private keys	232
Access keys and certificates	293
Password file(/etc/shadow)	106

Possible Solutions

The problem and the security challenges found in the findings can be analyzed from the consumer, administrator and publisher perspective. All the solutions presented below are analyzed from the above recommendations provided. They are then categorized based on actors involved in creating, hosting and using images.

Publisher's approach: Publisher may follow the following measures to gain the trust in the images they publish:

Build and deploy practices: The publisher before publishing the image in the repository should utilize the best practices while creating the image. Organizations or the third parties publishing the image should deploy an effective solution to plan, automate and govern the deployment of applications, tools and resources.

Clean Images: Utilize the user-specific filters to reduce the data leakage and remove sensitive information from the image.

Crypto-Shredding: Utilize shredding, wiping, sfill, scrumb or zerofree to avoid recovery of the deleted files. It involves secure deletion of all the sensitive material.

Image Configuration: As discussed earlier, configure the images and network to avoid security issues and component failures.

Administrator's Approach

The administrator should hold the control of authentication, authorization and logging and control of the images.

Software Updates and Patching: Software vulnerabilities were present there for more than two years in the image. The longer the vulnerabilities remain there, more significant will be the threat to other machines. For online images, you can use a few online patching utilities such as Microsoft Virtual Machine Servicing Tool to patch but for offline images, such a traditional model does not work. Injecting the patching in the offline images so that they will run when the image is online is another option, but it will delay the startup of the machine. So offline patching solution Nuwa is the best fit to patch the dormant images and the VM image will be up-to-date when it is initiated. Cloud providers should form a security team to reduce the vulnerability of an image to spread all over. Identifying and categorizing those vulnerable images and make such image private so the attacker will have fewer chances to attack those vulnerabilities.

Maintenance of Image: Images which are stored in the repository is the responsibility of the administrator. Those images should be scanned for malware to reduce the risk of liability for hosting malicious image.

Awareness and Support: Administrator should help the publisher to publish the image in a secure way and provide information on what to do and do not do. Support them with a tool to scan and fix those vulnerabilities before publishing those images for

the publisher who have less technical knowledge in sharing of the VM image in a secure manner.

Securing Private Keys: Using the same SSH keys for different resources in the cloud should be avoided. The cloud provider is responsible for adding the key regeneration script in the image for generating the host key. This host key is used for authentication of the SSH server.

Retriever's Approach:

The biggest risk for the retriever is that he provides the door for the attacker to enter his network with the use of malicious image, especially from the third parties. Following approach would be helpful in minimizing the security issues while acquiring images:

Updates and Configuration: Consumer of the cloud should automate the process of updating the software's and the licenses before they become illegal. Setting up the insecure configuration and using pirated software's will expose the vulnerability of the images. Before using the image check the remote configuration of the images to the attacker's machine.

Filters and scanning: Users should utilize tools like Secure Clean, privacy protection to remove information that is injected maliciously in the images to pull out the login credentials. Schedule a periodic task to scan dormant images for viruses and licenses. Utilize the filters to avoid the risk of consuming the harmful content.

Secure virtual machine Repository

Problem: Publisher who advertently publishes the malicious image in the repository, when used by the user will create an infected virtual machine. So not having a clean image will provide room to carry outside channel attack on the other VMs on the same network. The problem becomes huge when there is not enough security access control inviting malicious actions. There should be a balance between the security access control implemented otherwise strict security control may bring performance overhead. To identify the security holes, auditing and logging of information is very important.

Intent: The intent of the use case is to prevent the poisoning of the VM image during its building phase that may be introduced via the developer of the image. The access control strategies should be in place to prevent data leakage.

Solutions: As stated by Fernandez et.al, the virtual machine repository contains a reference monitor that has the filter to scan the VM image before being publishing and retrieving. User authentication is done via authentication to allow access for publishing and retrieving of images in the registry. The security auditor will do logging of all the activities of the virtual machine repository. The following figure will clearly state how this solution works. (Fernandez, Monage, & Hashizume, 2013)

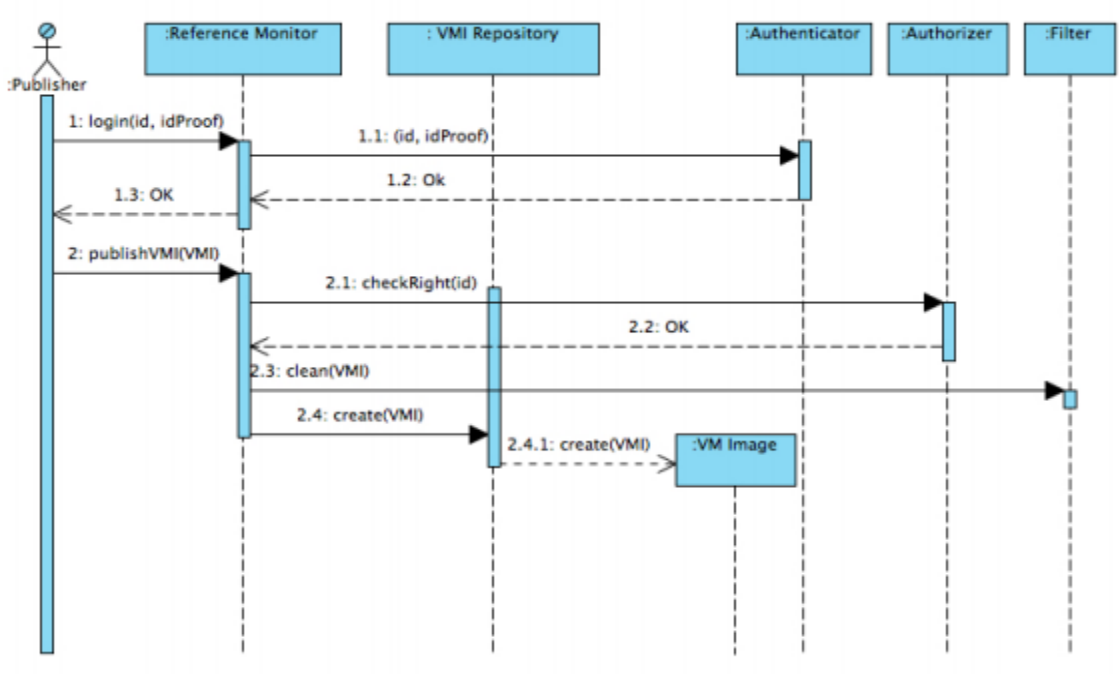

Figure 15: Working of Virtual Machine Image Repository (Fernandez, Monage, & Hashizume, 2013).

Role Based Access Control (RBAC) can be implemented to control the VMI repository. The roles can be varies depending on the type of privileges the user needs.

Above stated frameworks and solutions have provided effective solutions to the problem mentioned above. The framework proposed by Wei et. Al addresses the security concern of the VM images with a Mirage Image Management System. The approach provides filters and scanners before they are published and retrieved. The vulnerable images upon detection will be tracked and maintained. Update Checker as proposed by Schmidt et.al which checks for the outdated software and applications

regardless of the machine is running or is dormant. The NIST Security Reference will provide with all the necessary security requirements for the Vmi and its repository.

Summary

To support that those recommendations are useful, the paper has analyzed the findings from the research paper and provided solutions from all the concerned users of virtual machine images. Later, use cases that were presented to reflect the problem and suitable solutions for those problems provided by analyzing the above-mentioned recommendations.

Chapter V: Results, Conclusions and Recommendations

Introduction

This chapter will discuss in brief about the findings of the paper from the previous chapters and tries to draw the conclusion. This will also explain the methodology used to obtain the results and how those methodologies help to carry out the research. The main concept of the paper lies in the research questions. The chapter will try to explain how the above-mentioned research questions are answered in previous chapters. It will try to answer if those reasons are valid or not. Will the above reasoning need further research and study before it can be practically implemented? What are the limitations of this paper before the users can practically use it. It will also explain what improvement and additions can be done in the future to make this paper a very good resource for those working towards securing the virtual machine images.

Discussions

This paper utilizes qualitative research to gain an understanding of the security concerns of the images, the research paper related to it and help to develop ideas that will be helpful to overcome such concerns. The research paper methodology utilized white papers, technology blogs, manuals, articles and other publications to gain ideas and to answer the research questions.

Every security concerns were specific to the cloud images were described. More focus and attention were put onto be more specific and not to deviate to the other types of images or the security concerns related to hypervisor or cloud computing. Every type of security risk related to the images including configurations risks was analyzed from

different perspectives. Risk related to the images registries were better explained and information was provided on how they differ from the general filesystem. Understanding all those security concerns, solutions to them were provided in the form of recommendations. To prove the credibility of those recommendations one use case with the issues was presented in the form of findings. And for each of those issues, possible recommendations were enlisted analyzing the solutions provided above from the publisher, retriever and administrator approach.The following section summarized how the paper will answer the above-mentioned research question.

What are the current efforts in creating Secure VM Images (i.e. Techniques/ methods)?

The research paper first explained security controls from different organizations like Cloud Security Alliance (CSA) and National Institute of Standards and Technologies. Based on those guidelines provided frameworks and systems were developed by the researchers. Those frameworks and systems which is more specific to the image security are discussed in detail. The focus of those research paper is on preventive measures to reduce risk and issues related to the research paper. Importance was given to the systems and frameworks which are verified such as Nuwa and Mirage Image Management System. Security of the executing environment was also discussed in detail since it also affects the image security, but it was just to provide the information so that the readers will have a better understanding of the image security issues.

What are the attributes that can define a secure VM image? Identifying the Security Concerns of the VMI.

The security concerns of the image right from development, build to deployment phase are identified. The attributes identified mostly covers the security risk on public images since most of the security risk arises when the image is made public. More to that, the private image concerns will be covered within the public image security risks. Vulnerable and malicious images in the repository are left unintentionally or intentionally. When such images are used the VM created lack confidentiality and integrity in them. Instances that have a connection to the other applications within the cloud environment or external to the cloud environment highlights the network insecurity. Another aspect of image security, a poor configuration is also explained in this paper. Leftover credentials such as SSH keys, private keys and API keys are other weakness explained to provide a valid explanation for the question.

Recovery of deleted files is another big security concern that is prevalent over images provided by most CSPs.

Propose steps and procedures to develop a secure VM providing the users in the form of recommendations.

All the involved actors such as publisher, administrator of the image and retriever of the images can follow the above-provided recommendation to develop and build the images and launch the machines from safe images. The solutions explained will try to answer every possible issues and risk related to the image. From the problems and the

recommendations provided it is found out that the publisher of the images is more involved in securing the image.

Image Security Risks: Organizations and users should use specific vulnerability tools and process to detect vulnerability early from the development process. Different approaches that are explained above can be used to detect the outdated software and application. Update checker is one way to check the centralized database that contains all the information. Nuwa is another good option that can be used to patch offline images that are dormant. Standards and best practices available should be followed to configure their image following up the configuration requirements. To prevent network-based attacks network and remote access should not be enabled from the machines but should be run through APIs.

Privacy risks: Sharing of the virtual machine will increase the chances of the exploitation of history files, login audits and private keys. Creating awareness to the user seems to be the perfect way to minimize those risks. It is now clear from the above recommendations that users should not use the same API keys for all resources and should have firewalls to restrict the exploitation of SSH login.

Image registries: The above-provided recommendations made clear that the access control to the registries where the image is stored should be enforced on a need-to-know basis. Use of encrypted channels for pushing and pulling of the image is a very important factor to keeps the data and information leakage in the network. Categorize and downsizing the image is found to be the best way for the dormant images and utilizing the right way of tagging seems to be helpful for that case.

Conclusion

This research paper's focus was to clearly analyze the security problems of using virtual machine images especially the cloud images. This paper described the security concerns and what needs to be done to remove those security concerns in detail. Security concerns were mentioned in this paper analyzing all the available research paper from a different perspective. Based on those security concerns, it was easy to provide recommendations for those issues and risk. The attributes and the recommendations were discussed being very specific to the cloud machine images not diverting to the cloud environment and other virtual images. The use cases were presented in the paper to support that the recommendations will be helpful. It is found that the provided recommendations will be able to fix the issues while using the cloud images.

In conclusion, this research paper's aim was to prove a collective document that would help the consumers to guide towards choosing the right images and the procedures to use the images to initiate the machine in a secure manner. By reading this paper, a publisher and retriever should understand the hidden issues while publishing and retrieving the images. As an administrator of the cloud image repository, various security guidelines and procedures to be followed so that administrator will be less accountable.

Future work

The scope of the paper was limited to the security recommendations is a more discrete and practical manner. It is because the cost and the time frame required to verify those

recommendations and demonstrate is out of the scope of this paper. However, upon investing more time and resources more extensive study and implementation of those recommendations would be a huge step towards securing the images in the cloud environment.

Some suggestions for future work that can be done in this area is listed as follows:

- Work with the security team of the cloud providers and look for approval to conduct experiments to make sure that that the recommendations help in image security.
- Careful consideration of ethical issues. Is it ethically justifiable to conduct experiments on the cloud environment? Especially while scanning to detect the malware and secrets does it hamper the integrity and availability of the user information.
- The organization that provides the cloud service should form a security team to detect and access the risk. It is because there will be consistency over development and enforcement. Development, deployment and testing in different environments will create will not give the accurate result. Continuous integration and deployment would ensure consistency and automated deployment of security policies across creation, development and deployment, storage in registries to the running of the images.

- Portability of the image is another important factor. But at the same time research done to make the security as portable as the image. Tools and techniques can be developed that will over every platform.

References

Aslam, M., Gehrmann, C., Rasmusson, L., & Björkman, M. (2012). Securely launching virtual machines on trustworthy platforms in a public cloud.

AWS | Amazon Elastic Container Registry | Product Details. (n.d.). Retrieved May 1, 2018, from https://aws.amazon.com/ecr/details/

Balduzzi, M., Zaddach, J., Balzarotti, D., Kirda, E., & Loureiro, S. (2012). A security analysis of amazon's elastic compute cloud service. In *Proceedings of the 27th Annual ACM Symposium on Applied Computing* (pp. 1427–1434). ACM.

barclayn. (2017, October 17). Security Recommendations for Azure Marketplace Images. Retrieved November 26, 2018, from https://docs.microsoft.com/en-us/azure/security/security-recommendations-azure-marketplace-images

Best Practices for Mitigating Risks in Virtualized Environments. (2015). Retrieved from https://downloads.cloudsecurityalliance.org/whitepapers/Best_Practices_for%20_Mitigating_Risks_Virtual_Environments_April2015_4-1-15_GLM5.pdf

Brenton, C. (2011). virtualization-security, 17.

Bugiel, S., Nürnberger, S., Pöppelmann, T., Sadeghi, A.-R., & Schneider, T. (2011). AmazonIA: when elasticity snaps back. In *Proceedings of the 18th ACM conference on Computer and communications security* (pp. 389–400). ACM.

Chow, R., Jakobsson, M., Masuoka, R., Molina, J., Niu, Y., Shi, E., & Song, Z. (2010). Authentication in the clouds: a framework and its application to mobile users. In *Proceedings of the 2010 ACM workshop on cloud computing security workshop*

- *CCSW '10* (p. 1). Chicago, Illinois, USA: ACM Press. https://doi.org/10.1145/1866835.1866837

cloud Computing Design Patterns. (n.d.). Retrieved November 25, 2018, from http://cloudpatterns.org/

Duffy, J. (2009, April 23). Cisco's Chambers: cloud computing a security "nightmare." Retrieved April 30, 2018, from https://www.networkworld.com/article/2235425/cisco-subnet/cisco-s-chambers--cloud-computing-a-security--nightmare-.html

Dunn, J. E. (n.d.). Virtualised USB key beats keyloggers. Retrieved September 11, 2018, from https://www.techworld.com/news/security/virtualised-usb-key-beats-keyloggers-3213277/

Feinstein, B., & Jarmoc, J. (n.d.). "Get Off of My cloud": cloud Credential Compromise and Exposure, 31.

Gabor, K. (2016). *Developing Interoperable and Federated cloud Architecture*. IGI Global. Retrieved from https://books.google.com/books?id=dhL4CwAAQBAJ&printsec=frontcover#v=onepage&q&f=false

Glenn, C., Sterbentz, D., & Wright, A. (2016). *Cyber Threat and Vulnerability Analysis of the U.S. Electric Sector* (No. INL/EXT--16-40692, 1337873). https://doi.org/10.2172/1337873

Google Compute Engine Documentation | Compute Engine. (n.d.). Retrieved May 1, 2018, from https://cloud.google.com/compute/docs/

Hogan, M., Liu, F., Sokol, A., & Tong, J. (2011). NIST cloud computing standards roadmap. *NIST Special Publication*, *35*, 6–11.

Hussein, R. K., Alenezi, A., Wills, G. B., & Walters, R. J. (2016). A framework to secure the virtual machine image in cloud computing. In *Smart cloud (Smartcloud), IEEE International Conference on* (pp. 35–40). IEEE. Retrieved from http://ieeexplore.ieee.org/document/7796151/

Identity and Access Management (IAM) - Amazon Web Services (AWS). (n.d.). Retrieved December 2, 2018, from https://aws.amazon.com/iam/

IEEE Xplore - Under Maintenance. (n.d.). Retrieved December 2, 2018, from http://webservices.ieee.org/xplore/xplore-ie-notice.html?targetUrl=https%3a%2f%2fieeexplore.ieee.org%2fdocument%2f7796151%2fauthors

Indu, I., Anand, P. M. R., & Bhaskar, V. (2018). Identity and access management in cloud environment: Mechanisms and challenges. *Engineering Science and Technology, an International Journal*, *21*(4), 574–588. https://doi.org/10.1016/j.jestch.2018.05.010

Jamkhedkar, P., Szefer, J., Perez-Botero, D., Zhang, T., Triolo, G., & Lee, R. B. (2013). A Framework for Realizing Security on Demand in cloud Computing. In *2013 IEEE 5th International Conference on cloud Computing Technology and Science* (pp. 371–378). Bristol, United Kingdom: IEEE. https://doi.org/10.1109/cloudCom.2013.55

Jansen, W. A. (2011). cloud Hooks: Security and Privacy Issues in cloud Computing. In *2011 44th Hawaii International Conference on System Sciences* (pp. 1–10). Kauai, HI: IEEE. https://doi.org/10.1109/HICSS.2011.103

Livraga, G., & Zhu, S. (Eds.). (2017). *Data and Applications Security and Privacy XXXI: 31st Annual IFIP WG 11.3 Conference, DBSec 2017, Philadelphia, PA, USA, July 19-21, 2017, Proceedings*. Springer International Publishing. Retrieved from //www.springer.com/gp/book/9783319611754

Marinescu, D. C. (2013). *cloud Computing: Theory and Practice*. Newnes.

Modi, C., Patel, D., Borisaniya, B., Patel, A., & Rajarajan, M. (2013). A survey on security issues and solutions at different layers of cloud computing. *The Journal of Supercomputing*, *63*(2), 561–592.

Pandey, A., & Srivastava, S. (2014). An approach for virtual machine image security. In *2014 International Conference on Signal Propagation and Computer Technology (ICSPCT 2014)* (pp. 616–623). https://doi.org/10.1109/ICSPCT.2014.6884997

Santos, N., Gummadi, K. P., & Rodrigues, R. (2009). Towards Trusted cloud Computing. *Hotcloud*, *9*(9), 3.

Scarfone, K. A., Souppaya, M. P., & Hoffman, P. (2011). *Guide to security for full virtualization technologies* (No. NIST SP 800-125). Gaithersburg, MD: National Institute of Standards and Technology. https://doi.org/10.6028/NIST.SP.800-125

Schiffman, J., Moyer, T., Vijayakumar, H., Jaeger, T., & McDaniel, P. (2010). Seeding clouds with trust anchors. In *Proceedings of the 2010 ACM workshop on cloud computing security workshop* (pp. 43–46). ACM.

Souppaya, M., Morello, J., & Scarfone, K. (2017). *Application container security guide* (No. NIST SP 800-190). Gaithersburg, MD: National Institute of Standards and Technology. https://doi.org/10.6028/NIST.SP.800-190

Study Design and Methodology. (n.d.). University of North Texas. Retrieved from http://courses.unt.edu/wmoen/dissertation/ch3.pdf

Tan Fong, Ang, M. L Mat,Kiah, & Shu Yun, L. (2017). Security Issues and Future Challenges of cloud Service Authentication. *Acta Polytechnica Hungarica*, *14*(2). https://doi.org/10.12700/APH.14.2.2017.2.4

Virtualization-security. (n.d.), 17.

Wei, J., Zhang, X., Ammons, G., Bala, V., & Ning, P. (2009). Managing security of virtual machine images in a cloud environment. In *Proceedings of the 2009 ACM workshop on cloud computing security* (pp. 91–96). ACM.

Zhou, W., Ning, P., Zhang, X., Ammons, G., Wang, R., & Bala, V. (2010). Always up-to-date: scalable offline patching of VM images in a compute cloud. In *Proceedings of the 26th Annual Computer Security Applications Conference* (pp. 377–386). ACM.