Network Security in Cloud Computing

Virtual Networks and Cloud-based Security

Abin Sam
Computer Science Department
George Mason University
Fairfax, USA
asam@gmu.edu

Rishi Mehta
Computer Science Department
George Mason University
Fairfax, USA
rmehta7@gmu.edu

Zain AM Ibrahim
Computer Science Department
George Mason University
Fairfax, USA
zzahid3@gmu.edu

***Abstract*—Cloud computing is the next generation of networking computing since it can convey both software and hardware as an on-interest benefit over the Web. It gives dynamic asset pools, virtualization, and high accessibility. Security is one of the significant issues which decreases the development of Cloud Computing. Confusions with information security and information insurance keep on affecting the market. In this paper, we concentrate on the virtual system vulnerabilities and security of virtual systems in virtualized environment. We propose an answer for Hybrid Cloud security, concentrating on a Virtual Intrusion Detection System (V-IDS), DVIDS, firewalls and securing VMs with VLANS. We will talk about security concerns for VLANS, approaches to implement segmentation and virtual switch protection. We also describe the VENOM (Virtualized Environment Disregarded Operations) vulnerability. This paper also exhibits a novel virtual network framework focused control the intercommunication among virtual machines deployed on physical machines with higher security. We conclude with the best practices to secure most virtualized networks and cloud environment and how to keep away from system vulnerabilities.**

Keywords—V-IDS; DVIDS; VENOM; VLANS; Cloud

I. INTRODUCTION

Recently we have witnessed a mass expansion in cloud providers and several businesses hosting their complete infrastructure with these data centers. Each customer of the cloud provider uses a small slice of the entire computing power. This computing power is infinitely scalable according to the customer demands. The problem arises when improperly configured virtual components pose a security threat and an attacker gains access to this infinitely scalable computing power. Depending on the intent of the attacker there can be various levels of damage to the data center. The primary means of attack is through the network that interconnects all the devices. Fiber optic and Cat-6 or Cat-6a cables offer massive transmission speeds which when coupled with infinitely scalable computing power can cause enormous damage to an infrastructure. With the network as a primary means of access to the computing power, it makes logical sense to have the first line of defense at the network. Today any virtual component can be replicated within a few seconds to be available to use. If the first instance of the virtual component was misconfigured then when replicated this would only compound the issue and open up several avenues of attack. This paper discusses the ways and means in which we can prevent such misconfigurations and how we can use different types of Intrusion Detection Systems (IDS) and firewalls to identify and prevent unauthorized access to the infrastructure. We also discuss how a virtual machine can be used to take control of the physical machine using the VENOM vulnerability. In order to counteract these flaws and secure all avenues of attack, we propose a network model.

II. NETWORK SECURITY IN VIRTUAL MACHINES

A. *Virtual Network Vulnerabilities*

Network Virtualization is defined as combining available network hardware and software resources into a single software-based administrative entity. Network virtualization improves the network speed, reliability, flexibility, scalability and security [1]. In cases where the load on the network infrastructure has unplanned spikes and large surges in traffic, network virtualization is most effective. Depending on how a network is subdivided or combined there are two different types of network virtualization. They are: Internal virtualization and External virtualization. Internal virtualization is when a single system with software containers is made to emulate a physical network with software [1]. External virtualization is when a single LAN (local area network) is combined or subdivided into virtual networks to improve a large networks efficiency and reduce complexity in network management by automating actions. This means that systems on the same local network can either be separated into VLANS (virtual local area networks) or systems on separate LAN's can be combined into a single VLAN that spans segments of a larger network [1].

In a physical setup, the security configuration and maintenance would be limited to each of the separate physical components that make up the infrastructure. However, in the case of a virtual setup where resources can be easily replicated several times over within a very short period of time, it is possible that a single machine that has been misconfigured will be replicated several times over with the same faulty configuration, thus opening multiple avenues of attack to the outside world. It is then necessary to have a solid default configuration and other risk management processes. The

management and configuration of virtual network security can then be carried out more easily as compared to physical infrastructure. In addition to this, failure to separate duties and deploy least privilege controls can result in a virtualized network being compromised by privileged insiders. An investigation by Verizon Business reported that out of all the breaches studied 22 percent of them were caused by insiders with high privilege access [2]. Compliance regulations such as PCI DSS (Payment Card Industry Security Standard) and FISMA (Federal Information Security Management Act) that require separation of duties and least privilege access to protect sensitive records must be enforced strictly to ensure there are no breaches. Lastly, failure to coordinate policy between virtual machines and virtual networks is a major complication. Isolation and security zones created using physical network devices can also be used for their virtual counterparts, however, the trouble arises when virtual machines or networks are moved around. Access control protections with the use of TCP Wrappers, pluggable authentication modules and iptables must be implemented to provide protection for the host operating system.

B. *Virtual Intrusion Detection System and Distributed Virtual Intrusion Detection System in Cloud*

The distributed nature of Cloud Computing makes it the most vulnerable target for intruders to attack. As a part to enhance the efforts to establish safety, Intrusion Detection systems can be used for such environment. This includes a systematic examination of logs, configurations, and network traffic. Intrusion detection system plays an important role in the security and perseverance of active defense system against intruder hostile attacks for any business and IT organization. But IDS execution in distributed computing requires a productive, versatile and virtualization-based methodology. An IDS ceaselessly gathers and breaks down information from a registering framework, planning to recognize meddlesome activities. As for the starting point of dissected information, there are two primary methodologies for interruption identification: Network-based IDS (NIDS) – in view of watching the network traffic flowing through the systems to monitor and Host-based IDS (HIDS) – in light of watching local activity on a host, network connections, system calls, logs etc. Traditional IDSs are not suitable for cloud environment as network-based IDSs cannot detect encrypted node communication. On the other hand host-based IDSs are not able to find the hidden attack trail. The primary shortcoming of host-based interruption location is its relative delicacy: in order to gather system activity data, a HIDS agent should be installed in the machine to monitor the activity. This agent can be deactivated or altered by a successful intruder, in order to mask his/her presence, turning the detection system useless. Techniques used for analyzing the intrusion detection of collected data can be classified as signature detection and anomaly detection.

As appeared, host-based IDS are vulnerable to local attacks, on the grounds that attackers can incapacitate or alter them. [3]Therefore, such a system cannot be viewed as reliable. The utilization of virtual machines gives an answer for this issue. The proposition introduced here permits assembling more solid host-based interruption recognition frameworks. The proposition's primary idea is to encapsulate the system to monitor inside a virtual machine, which is checked from outside. The intrusion detection and response are executed outside the virtual machine, i.e. out of reach of attackers. The proposal considers a type II virtual machine screen, so the detection and reaction system can be implemented as host system processes. Figure 1[3] outlines the fundamental components of the proposed design.

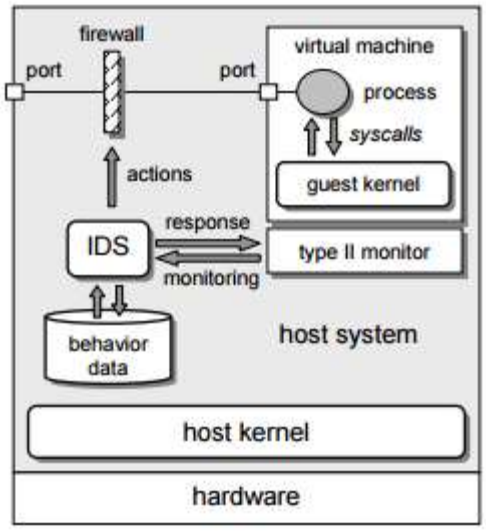

Fig: 1 Architectural proposal[3]

Work Cloud processing gives application and capacity administrations on remote servers. The customers don't need to stress over its support and programming or equipment upgrades. Cloud model takes a shot at the “concept of virtualization” of assets, where a hypervisor server in cloud server farm has various customers on one physical machine. Sending HIDS in hypervisor or host machine would permit the executive to screen the hypervisor and virtual machines on that hypervisor. In any case, with the quick stream of a high volume of information as in cloud model, there would be issues of execution like over-burdening of VM (Virtual Machine) facilitating IDS and dropping off information bundles. Additionally, if the host is traded off by an offending attack then the HIDS utilized on that host would be neutralized. In such a situation, a system based IDS would be more suitable for sending in the cloud like system. NIDS would be put outside the VM servers of network focuses, for example, switch, switch or gateway for system movement checking to have a worldwide perspective of the system. To handle an extensive number of information bundles stream in such a domain, a multi-threaded IDS approach can be used. The multi-threaded IDS would have the capacity to process vast measure of information and could decrease the bundle misfortune. After an effective preparing, the proposed IDS would pass the observed cautions to an outsider checking administration, who might specifically educate the cloud client about their system under attack. The outsider observing administration would likewise give master guidance to cloud administration supplier for misdoings and interruption escape clauses in the system. Figure 2 demonstrates the proposed IDS model. [4]

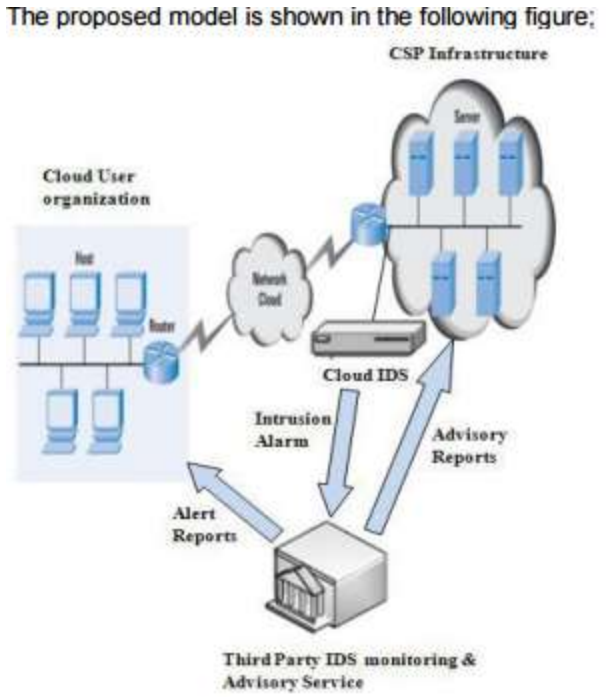

Fig: 2 Proposed Cloud IDS Model [14]

The cloud user access its information on remote servers at service provider's site over the cloud system. The user request and activities are checked and logged through a multi-threaded NIDS. The logs are promptly communicated to cloud user.

C. *Virtual Firewalls vs Physical Firewalls*

A physical firewall is a network security system that monitors and controls incoming or outgoing network traffic based on predetermined security rules. Physical firewalls can be either in the form of a software appliance that runs on general purpose hardware or hardware based firewall computer appliances [5]. In contrast, a virtual firewall is a network service or appliance that runs entirely within a virtualized environment and provides the same service as a traditional firewall [5]. While virtual firewalls may be cost-effective, easy to implement, simple and more secure than their physical counterparts they are limited by factors such as management difficulty, support and additional security risks that creep in due to virtualization [6]. As such virtual appliances have the same advantages as physical appliances, however, they suffer from problems unique to virtual appliances and in addition to this the same disadvantages as physical appliances. However, there are some specific situations where a virtual firewall is absolutely necessary since a physical firewall cannot be implemented. Situations such as virtual machines attached to the same virtual network. In this case, the network traffic of each virtual machine is open to all. Any machine on the network can monitor, read or capture the traffic between machines on this virtual network. It is possible that such a setup if connected to the outside world is susceptible to attacks like IP spoofing, MAC spoofing, packet injection, port scanning, DNS spoofing and Man in the middle attack. Once an attacker gains access to a system he can possibly convert it into a botnet and further use several of these machines to launch Distributed Denial of Service Attacks, Ping flood, SYN flood attacks, Smurf attacks [7]. A virtual firewall would be preferable in the case of intra-host inspection instead of the physical firewall. A physical firewall that protects virtualized servers must be configured to defend against hypervisor flaws. Even though the hypervisor vendor implements software hardening it is still crucial to control access to the hypervisor itself and inspect the virtualization platform for vulnerabilities. This inspection and access control cannot be achieved by a firewall running on the hypervisor itself, it must be done by a physical firewall in front of the hypervisor. Thus, in an enterprise level infrastructure that has a very large number of physical and virtual devices, it is necessary to identify the situation and accordingly use a combination of physical and virtual firewalls to complement each other.

III. A NOVEL VIRTUAL NETWORK MODEL

As indicated by the examination of vulnerabilities existed in the virtual system above, we influence the qualities of "route" and "bridge" models and consolidate them to propose a novel virtual system model to make the communication among VM's more secure[3]. This model is made out of three layers: routing layer, firewall, and shared system, as shown in figure 3

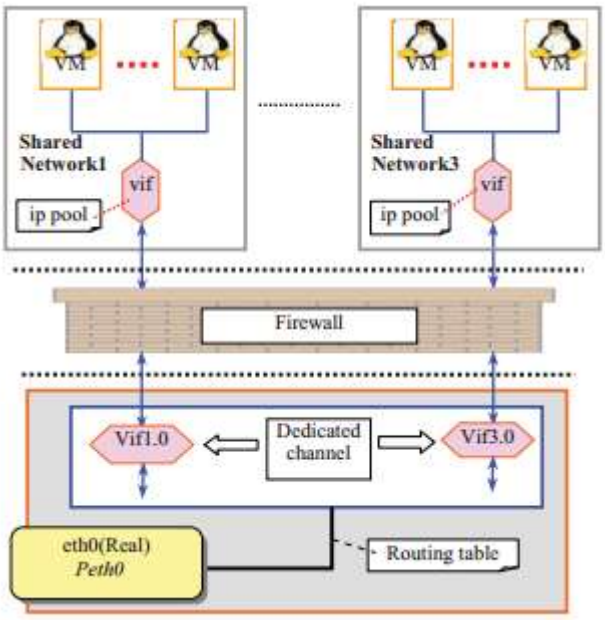

Fig: 3 A Virtual Network Model[3]

- Routing layer

The routing layer has the same capacities as the traditional route mode. The main responsibility of routing layer is to associate the physical network and make a logically devoted channel for communication between the virtual network and the physical network. In this layer, a set of unique static IDs are indicated by the administrator, ahead of time and are stored in a configuration file, which can be doled out to a common system as a unique tag. This unique tag is used to monitor the packets sent from each mutual system.

- Firewall:

The principal idea behind this layer is to prevent attacks launched from virtual shared networks by identifying the network ID specified in the configuration file. Two main security strategies are defined in this layer .Each virtual interface in the routing layer that connects with a virtual shared network cannot communicate with any other virtual shared network and any packet that try to modify the routing table will be dropped. These two security policies are implemented by using iptables packet filtering system.

- Shared network layer

In this layer, we stick to the assumption that the VMs working for the same organization or associations ought to be assigned in the same shared network. To further improve the security of this model, we indicate an arrangement of subnets (i.e., 128.128.10.0, 10.232.193.0, 10.232.194.0) ahead of time and each mutual system will be bound with a one of a kind subnet by the administrator when another shared network instance is made.

IV. VENOM VULNERABILITY

A. Introduction to VENOM

VENOM stands for Virtualized Environment Neglected Operations Manipulation. It is a security vulnerability in the virtual floppy drive controller (FDC). VENOM is used by various computer virtualization platforms. VENOM was made public on May 13, 2015. The vulnerability is in QEMU, which is an open source machine emulator and virtualizer. QEMU is utilized by various modern virtualization platforms.

The threat that VENOM poses can risk access to corporate intellectual property (IP), also sensitive and personally Identifiable Information (PII), which might probably affect thousands of organizations and numerous end users that rely on affected VMs for the allocation of shared computing resources.

VENOM is different because unlike most of the VM escape vulnerabilities discovered before this, it is not only exploitable in non-default configurations but a number of other default configurations. Unlike all the older vulnerabilities, VENOM is applied to various virtualization platforms and allows arbitrary code execution.

B. How VENOM works

How this vulnerability works is that the guest operating system communicates with the FDC by sending commands to the FDC. QEMU's virtual FDC uses a buffer to store these commands and the data parameters associated along with them. The FDC knows how much data is expected for each command and once all the data for a given command is received from the guest system, the FDC executes the command and resets the buffer. The moment all the FDC commands are completely processed, the buffer reset is immediately performed. This buffer reset is performed for all commands except two. An attacker can send these commands and specially crafted parameter data from the guest system to the FDC to overflow the data buffer and execute arbitrary code in the context of the host's hypervisor process. [10]

Keeping the vulnerability issues in mind, a privileged guest user is able to manipulate this vulnerability to crash the guest or even execute arbitrary code on the host with the privileges of the host's QEMU process. In a statement, Red Hat [10] said:

"This issue affects the versions of the KVM and XEN packages as shipped with Red Hat Enterprise Linux 5, the versions of the qemu-kvm packages as shipped with Red Hat Enterprise Linux 6 and 7, and the versions of qemu-kvm -rhev packages as shipped with Red Hat Enterprise Virtualization 3. Future updates for the respective releases will address this flaw."

C. Products Affected

While a lot of products have been affected by this bug, there are a few who remain unaffected. QEMU, Amazon, Ubuntu, Oracle, Cisco, UpCloud, f5, Barracuda Networks, IBM, HP, Rackspace, Linode, Citrix, Redhat were affected by the bug and released patches and advisories for the same. Products like VMware, Microsoft Hyper-V, and Bochs hypervisors remained immune to this vulnerability.

VENOM is agnostic of the host operating system (Linux, Windows, Mac OS, etc.). An attacker would need to have administrative or root privileges in the guest operating system to cause trouble by exploiting VENOM.

Here is what Oracle had to say regarding the bug fixes: "We will release a VirtualBox 4.3 maintenance release very soon. Apart from this, only a limited amount of users should be affected as the floppy device emulation is disabled for most of the standard virtual machine configurations,"

V. SECURING VIRTUAL MACHINES WITH VLAN'S

Server virtualization is now seen as a solution to problems faced with physical computers that fail to handle incremental applications and huge processing workloads. It uses special software to create virtual machines that run in sync with a physical machine, simultaneously to share resources. The physical machine here is the 'host'. This virtualized environment includes a 'virtual network' or a 'virtual LAN'. The 'Virtual LAN' or 'VLAN' creates a virtualized local area network infrastructure within the host machine for communication. A VLAN is a set of switch ports. VLANs are network segments, we can assign each VLAN an IP address scope.

The foremost principle in securing a VLAN network is security at the physical level. If an organization wants to avoid its devices to be tampered with, physical access is supposed to be strictly controlled. VLAN security includes switch security and correct VLAN configuration. Some of the most common attacks against VLAN can be avoided with proper attention to proper configuration practices. A switch in a VLAN network does a lot more than just switch packets

between ports. VLANs majorly comprise of two types of interfaces or links. We can connect together multiple switches thanks to these links. Depending on their configuration they are called Access Links or Trunk Links.

Access Links are the most common type of links. To gain access to the local network, the network hosts connect to the Access Link. These links are configured in a special way, so one may plug a computer into them and access a network. For Trunk Links to be able to carry packets, they require the ability to carry these packets from all of the available VLANs. This is due to the fact that VLANs span usually over multiple switches. A Trunk Link is a port configured to carry packets for any VLAN. Network segments can be enhanced by making them security zones. This is a very smooth way of securing your VLAN network. For a VLAN to be a security zone, it requires appropriate monitoring and filtering in a proper environment. A VLAN by itself is not a security zone. Here are a few notable best practices as mentioned by Chris Partsenidis [12]

- Removing console-port cables and introducing password-protected console or virtual terminal access with specified timeouts and restricted access policies
- Disabling high-risk protocols on any port that doesn't require them (e.g CDP, DTP, PAgP, UDLD)
- Deploying VTP domain, VTP pruning and password protections
- Controlling inter-VLAN routing through the use of IP access lists.

VLAN technology offers numerous enhancements to the network and provides paths to run multiple services in isolated environments without affecting its speed, quality, and network availability. If the necessary basic security guidelines are taken into consideration during initial implementation and then during ongoing administration, a VLAN can dramatically reduce administrative overhead. It should not be forgotten that any network is only as robust as its weakest link, and therefore, an equal amount of attention needs to be given to every layer to assure the soundness of the entire structure. [13]

VI. Conclusion

In cloud computing, odds of interruption are more with the learnedness of intruder's attacks. Diverse IDS procedures are utilized to counter pernicious attacks in custom networks. In Cloud computing, gigantic network access rate, giving up the control of information and applications to the service provider and distributed vulnerability, an effective, dependable and data straightforward IDS are required. This paper depicts a proposition to build the security of registering frameworks utilizing virtual machines. The premise of the proposition is to screen visitor from activities through an interruption recognition framework, outside of the virtual machine. The information utilized as a part of interruption identification is gotten from the virtual machine screen and examined by an IDS handle in the basic machine. The detection framework is out of reach to virtual machine forms and can't be subverted by gatecrashers. The principle goal of the venture, to impede the execution of a suspect process in the virtual machine and thus stay away from the framework trade off, was reached with the present model. Be that as it may, correlative work should be done to reduce the virtualization overhead and to enhance the execution of the present interruption detection and reaction mechanism. We are additionally concentrating on more adaptable approaches to interface with the visitor piece, permitting executing or suspending particular suspect procedures. Likewise, the communications between the IDS and the host framework firewall should be refined.

Different inquiries to be conducted include observing systems in light of other significant information, similar to the system movement created by the virtual machine, and the conduct of visitor clients on their procedures. More refined calculations for interruption discovery can be executed based on such data, diminishing the event of false results (both positive and negative).

References

[1] https://en.wikipedia.org/wiki/Network_virtualization.

[2] https://www.sans.org/reading-room/whitepapers/analyst/top-virtualization-security-mistakes-and-avoid-them-34800.

[3] http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.1.9231&rep=rep1&type=pdf

[4] Irfan Gul, M. Hussain, "Distributed cloud intrusion detection model", International Journal of Advanced Science and Technology Vol. 34, September 2011.

[5] https://en.wikipedia.org/wiki/Firewall_(computing)

[6] https://en.wikipedia.org/wiki/Virtual_firewall

[7] http://www.cioupdate.com/trends/article.php/11047_3670581_2/The-Pros-and-Cons-of-Virtual-Appliances.htm

[8] http://www.symantec.com/connect/articles/security-11-part-3-various-types-network-attacks

[9] http://venom.crowdstrike.com

[10] https://access.redhat.com/security/cve/CVE-2015-3456

[11] http://www.zdnet.com/article/venom-security-flaw-millions-of-virtual-machines-datacenters/

[12] Chris Partsenidis on behalf of fedtechmagazine.com.

[13] http://www.firewall.cx/networking-topics/vlan-networks/226-vlan-security.html

[14] http://www.ijstr.org/final-print/may2012/Intrusion-detection-system-for-cloud-computing.pdf