Developing a Technique to Detect Intrusion in the Network and Evaluate Security Performance Analysis

Pinki[1] Avni Khatkar[2]
[1]Student [2]Assistant Professor
[1,2]Department of Electronics & Communication Engineering
[1,2]MRIEM, Rohtak

Abstract— *Intrusion Detection and Prevention Systems (IDPS) are used: to identify possible attacks, collecting information about them and the trying to stop their occurrence and at last reporting them to the system administrator. These systems are used by some organizations to detect the weaknesses in their security policies, documenting existing attacks and threats and preventing an individual from violating security policies. Because of their advantages these systems became an important part of the security infrastructure in nearly every organization. In a Cloud computing environment, attackers can determine the vulnerabilities in the cloud systems and compromise the virtual machines to set out large scale Distributed Denial-of-Service (DDOS) attack. To avert these virtual machines from concession, we propose a multi-phase solution NICE (Network Intrusion Detection and Countermeasure selection in Virtual Network Systems).*

Keywords— Intrusion Detection System, Attack Analyzer, Network Controlled, VM Profiling

I. INTRODUCTION

NICE (Network Intrusion detection and Countermeasure selection in virtual network systems) to establish a defense-in-depth intrusion detection framework. For better attack detection, NICE incorporates attack graph analytical procedures into the intrusion detection processes. We must note that the design of NICE does not intend to improve any of the existing intrusion detection algorithms; indeed, NICE employs a reconfigurable virtual networking approach to detect and counter the attempts to compromise VMs, thus preventing zombie VMs. In computer security, a Network Intrusion Detection System (NIDS) is an intrusion detection system that attempts to discover unauthorized access to a computer network by analyzing traffic on the network for signs of malicious activity. A recent CSA (Cloud Survey Alliance) survey reports that among all security issues exploitation and despicable use of cloud computing is considered as the main security threat. In traditional data centers, where system administrators have full control over the host machines, vulnerabilities can be detected and patched by the system administrator in a centralized manner. However, patching known security holes in cloud data centers, where cloud users usually have the privilege to control software installed on their managed VMs, may not work effectively and can violate the Service Level Agreement (SLA).

Furthermore, cloud users can install vulnerable software on their VMs, which essentially contributes to loopholes in cloud security. The challenge is to establish an effective vulnerability/attack detection and response system for accurately identifying attacks and minimizing the impact of security breach to cloud users. In a cloud system where the infrastructure is shared by potentially millions of users, abuse and nefarious use of the shared infrastructure benefits attackers to exploit vulnerabilities of the cloud and use its resource to deploy attacks in more efficient ways. Such attacks are more effective in the cloud environment since cloud users usually share computing resources, e.g., being connected through the same switch, sharing with the same data storage and file systems, even with potential attacker.

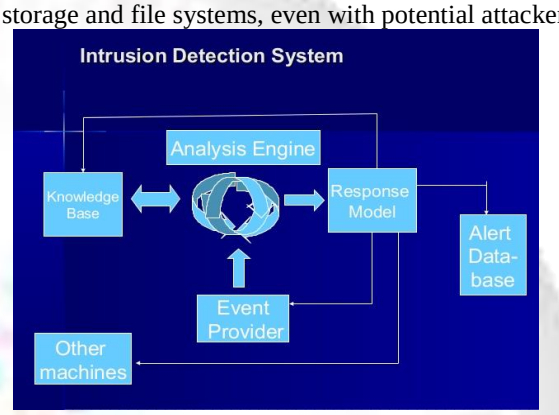

Fig. 1: IDS

II. OBJECTIVE

The main aim of this project is to prevent the vulnerable virtual machines from being compromised in the cloud server using multi-phase distributed vulnerability detection, measurement, and countermeasure selection mechanism called NICE. There are few objectives of this research. By studying the previous techniques and by knowing the loopholes in the previous developed techniques we can develop a new technique to enhance the security of the cloud computing.

III. PROPOSED SYSTEM

In this article, we propose NICE (Network Intrusion detection and Countermeasure selection in virtual network systems) to establish a defense-in-depth intrusion detection framework. For better attack detection, NICE incorporates attack graph analytical procedures into the intrusion detection processes. We must note that the design of NICE does not intend to improve any of the existing intrusion detection algorithms; indeed, NICE employs a reconfigurable virtual networking approach to detect and counter the attempts to compromise VMs, thus preventing zombie VMs.

A. Advantages of Proposed System:

The contributions of NICE are presented as follows: We devise NICE, a new multi-phase distributed network intrusion detection and prevention framework in a virtual networking environment that captures and inspects suspicious cloud traffic without interrupting users' applications and cloud services.

- NICE incorporates a software switching solution to quarantine and inspect suspicious VMs for further investigation and protection. Through programmable network approaches, NICE can improve the attack detection probability and improve the resiliency to VM exploitation attack without interrupting existing normal cloud services.
- NICE employs a novel attack graph approach for attack detection and prevention by correlating attack behavior and also suggests effective countermeasures.
- NICE optimizes the implementation on cloud servers to minimize resource consumption. Our study shows that NICE consumes less computational overhead compared to proxy-based network intrusion detection solutions.

IV. MODULES DESCRIPTION

A. Nice-A:

The NICE-A is a Network-based Intrusion Detection System (NIDS) agent installed in each cloud server. It scans the traffic going through the bridges that control all the traffic among VMs and in/out from the physical cloud servers. It will sniff a mirroring port on each virtual bridge in the Open vSwitch. Each bridge forms an isolated subnet in the virtual network and connects to all related VMs. The traffic generated from the VMs on the mirrored software bridge will be mirrored to a specific port on a specific bridge using SPAN, RSPAN, or ERSPAN methods. It's more efficient to scan the traffic in cloud server since all traffic in the cloud server needs go through it; however our design is independent to the installed VM. The false alarm rate could be reduced through our architecture design.

B. VM Profiling:

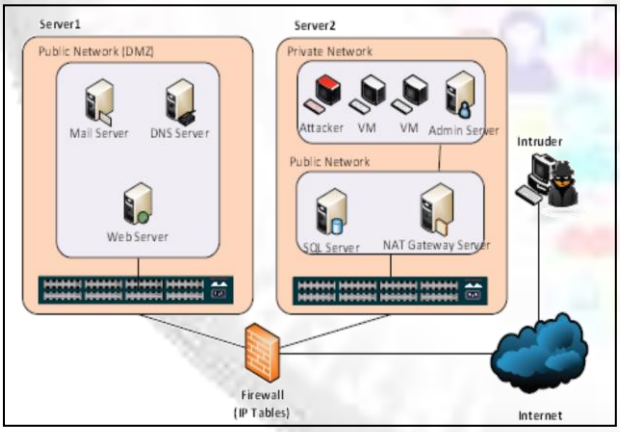

Fig. 1: Virtual network topology for security evaluation

Virtual machines in the cloud can be profiled to get precise information about their state, services running, open ports, etc. One major factor that counts towards a VM profile is its connectivity with other VMs. Also required is the knowledge of services running on a VM so as to verify the authenticity of alerts pertaining to that VM. An attacker can use port scanning program to perform an intense examination of the network to look for open ports on any VM. So information about any open ports on a VM and the history of opened ports plays a significant role in determining how vulnerable the VM is. All these factors combined will form the VM profile. VM profiles are maintained in a database and contain comprehensive information about vulnerabilities, alert and traffic.

C. Attack Analyzer:

The major functions of NICE system are performed by attack analyzer, which includes procedures such as attack graph construction and update, alert correlation and countermeasure selection. The process of constructing and utilizing the Scenario Attack Graph (*SAG*) consists of three phases: information gathering, attack graph construction, and potential exploit path analysis. With this information, attack paths can be modeled using SAG. The Attack Analyzer also handles alert correlation and analysis operations. This component has two major functions:

- Constructs Alert Correlation Graph (*ACG*),
- Provides threat information and appropriate countermeasures to network controller for virtual network reconfiguration.

NICE attack graph is constructed based on the following information: Cloud system information, Virtual network topology and configuration information, Vulnerability information.

D. Network Controller:

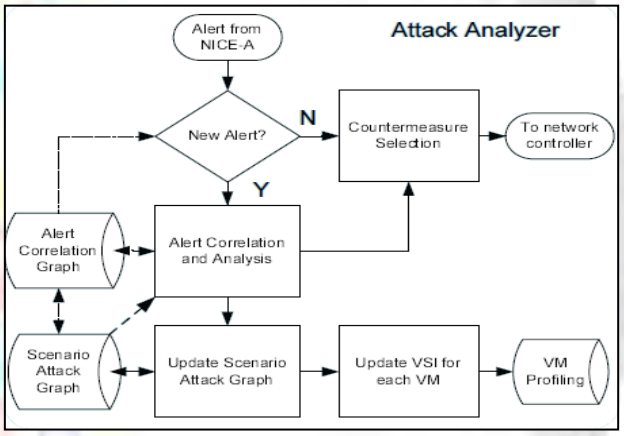

Fig. 2: Workflow of Attack Analyzer

The network controller is a key component to support the programmable networking capability to realize the virtual network reconfiguration. In NICE, we integrated the control functions for both OVS and OFS into the network controller that allows the cloud system to set security/filtering rules in an integrated and comprehensive manner. The network controller is responsible for collecting network information of current Open Flow network and provides input to the attack analyzer to construct attack graphs. In NICE, the network control also consults with the attack analyzer for the flow access control by setting up the filtering rules on the corresponding OVS and OFS. Network controller is also responsible for applying the countermeasure from attack analyzer. Based on *VM* Security Index and severity of an alert, countermeasures are selected by NICE and executed by the network controller.

E. Security Performance Analysis:

To evaluate the security performance, a demonstrative virtual cloud system consisting of public (public virtual servers) and private (VMs) virtual domains is established as shown in Figure 3. Cloud Servers 1 and 2 are connected to Internet through the external firewall. In the Demilitarized

Zone (DMZ) on Server 1, there is one Mail server, one DNS server and one Web server. Public network on Server 2 houses SQL server and NAT Gateway Server. Remote access to VMs in the private network is controlled through SSHD (i.e., SSH Daemon) from the NAT Gateway Server. Table 2 shows the vulnerabilities present in this network and table 3 shows the corresponding network connectivity that can be explored based on the identified vulnerabilities.

Host	Vulnerability	Node	CVE	Base Score
VM group	LICQ buffer overflow	10	CVE 2001-0439	0.75
	MS Video ActiveX Stack buffer overflow	5	CVE 2008-0015	0.93
	GNU C Library loader flaw	22	CVE-2010-3847	0.69
Admin Server	MS SMV service Stack buffer overflow	2	CVE 2008-4050	0.93
Gateway server	OpenSSL uses predictable random variable	15	CVE 2008-0166	0.78
	Heap corruption in OpenSSH	4	CVE 2003-0693	1
	Improper cookies handler in OpenSSH	9	CVE 2007-4752	0.75
Mail server	Remote code execution in SMTP	21	CVE 2004-0840	1
	Squid port scan	19	CVE 2001-1030	0.75
Web server	WebDAV vulnerability in IIS	13	CVE 2009-1535	0.76

Fig. 3: Vulnerabilities in the virtual network

V. CONCLUSION

Here a technique is proposed to detect and mitigate collaborative attacks in the cloud virtual networking environment. The proposed technique is an NIDS. A "network intrusion detection system (NIDS)" monitors traffic on a network looking for suspicious activity, which could be an attack or unauthorized activity. For this purpose a large NIDS server can be set up on a backbone network in order to monitor all traffic; or smaller systems can be set up to monitor traffic for a particular server, switch, gateway, or router. NIDS server does not replace primary security such as firewalls, encryption and other authentication methods. Existing intrusion detection systems are based on the assumption that an intruder will behave differently from the legitimate user and hence can be easily identified. It also assumes that nearly all the unauthorized actions are noticeable.

REFERENCES

[1] Denning, D. E. and Neumann, P. G. “Requirements and Model for IDES -- a Real-Time Intrusion Detection System”, Tech. report, Computer Science Lab, SRI International, 1985.
[2] K. Scarfone, P. Mell, “Guide to Intrusion Detection and Prevention Systems (IDPS)”. Computer Security Resource Center (National Institute of Standards and Technology).
[3] Chun-Jen Chung, Student Member, IEEE, Pankaj Khatkar, Student Member, IEEE, Tianyi Xing, Jeongkeun Lee, Member, IEEE, and Dijiang Huang Senior Member, IEEE-“NICE: Network Intrusion Detection and Countermeasure Selection in Virtual Network Systems”- IEEE TRANSACTIONS ON DEPEDABLE AND SECURE COMPUTING, 2013.
[4] Denning, D. E. and Neumann, P. G. “Requirements and Model for IDES -- a Real-Time Intrusion Detection System”, Tech. report, Computer Science Lab, SRI International, 1985.
[5] K. Scarfone, P. Mell, “Guide to Intrusion Detection and Prevention Systems (IDPS)”. Computer Security Resource Center (National Institute of Standards and Technology). Ptacek, Thomas H. & Newsham, Timothy N. (January 1998);"Insertion, Evasion, and Denial of Service: Eluding Network Instrusion Detection".
[6] Lunt, Teresa F., "Detecting Intruders in Computer Systems," 1993 Conference on Auditing and Computer Technology, SRI International.
[7] Sebring, Michael M., and Whitehurst, R. Alan., "Expert Systems in Intrusion Detection: A Case Study," The 11th National Computer Security Conference, October, 1988.
[8] Jackson, Kathleen, DuBois, David H., and Stallings, Cathy A., "A Phased Approach to Network Intrusion Detection,"14th National Computing Security Conference, 1991.