ijecs open access www.ijecs.in
International Journal Of Engineering And Computer Science ISSN:2319-7242
Volume - 3 Issue - 8 August, 2014 Page No. 7661-7664

Comparison between NICE and NICE-1 Algorithms In Virtual network systems

P. Suganya[1], G. Thailambal[2], M. Umadevi[3]

[1] M.Phil (Student), Department of Computer Applications, School of Computing Sciences, Vel's University, suganya315@gmail.com.
[2] Assistant Professor, Department of Computer Applications, School of Computing Sciences, Vel's University, aishusri2009@gmail.com
[3] M.Phil (Student), Department of Computer Applications, School of Computing Sciences, Vel's University, umakaali@yahoo.co.in

ABSTRACT

Cloud computing has gained adequate attention in the recent times due to its ease and customizability. However, cloud security has been a never ending issue faced by the users. Attackers take hold of the cloud system to gain access to third party information. Network Intrusion Detection and Countermeasure selection (NICE) is one particular technique that adopts multipath distributed vulnerability detection with the help of OpenFlow network programming API's. On the other hand, NICE-1 adopts a similar architecture but makes use of an efficient host-based IDS involving firewall that manages any number of attackers. The latter is compatible and cost effective when compared to the former.

Index terms- Network Security, Virtual Network, Intrusion detection, Cloud computing

1 INTRODUCTION

The major advantage of cloud network lies in the availability for all. Cloud users can directly install the vulnerable software on their machines thereby adding complexities to cloud security. In the usage of multiple VMS, there is a probability to ignore certain security loopholes unnoticed.

In this paper, the existing NICE algorithm is compared with the proposed NICE-1 algorithm. The design of NICE-1 differs from NICE inorder to eliminate cost overheads and manage cloud traffic to better extent. While NICE-A scans the vulnerabilities; NICE-1 fixes a strong mechanism involving firewall where the problem faced due to cloud security is limited.

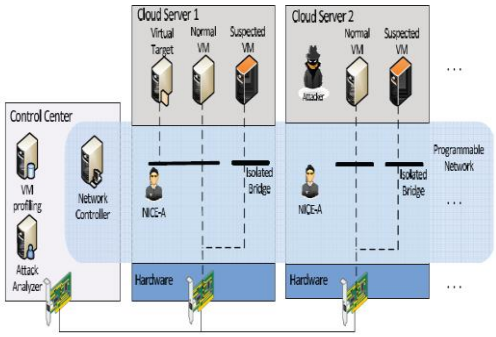
The remainder of this paper is organized as follows: Section 2 presents the NICE algorithm. NICE-1 proposed algorithm is described in Section 3 while Section 4 presents the comparison between existing and proposed structure in terms of performance and security. Section 5 discuss on conclusion of the paper with listing of future works.

2 NICE algorithm

In this section, the existing NICE framework is illustrated. The architecture involves a control center

and multiple cloud servers. The control center manages the VM profiles and controls networks [1]. At this stage, the false alarm rate might be reduced but still, the detection accuracy is a limitation as shown in Fig 1

Fig 1: NICE system architecture



The attack analyzer plays a major role in dealing and mitigating the incoming attacks. The NICE attack graph maintains following information in database:

- Cloud system information
- Vulnerability information
-

Virtual network topology and configuration information

2.1 Security Analysis

In security metrics, there has been a significant effort in development of quantitative security metrics [2][3]. There are many options in using these security metrics model for security risk that will be a better choice. To access network security risks, we need a current network configuration, these security metrics are needed to measure the risk factors. After constructing an attack graph, vulnerable information is included in this graph. For initial node, the priori probability is assigned as likelihood of threat source that becomes active and is difficult to exploit the vulnerability. In attack graph, the relations between exploits can be conjunctive according to how they relate to dependency conditions [4].

2.2Performance Evaluation

- A registered user means the one who uses a data resource/ files system and provides them credentials, effectively by proving them identity. Any person can become a registered user by providing some credentials, mainly to get the permission for accessing. After that, one can access information and privileges unavailable to non-registered users, these are referred as guests.
- The attackers can be located either on outside or inside of the virtual networking system. The attacker's primary goal is to exploit vulnerable VMs and compromise them as zombies. Our protection model focuses on virtual-network-based attack detection and reconfiguration solutions to improve the resiliency to zombie explorations.
- NICE system are performed by attack analyzer, which includes procedures such as attack graph construction and update, alert correlation, and countermeasure selection. Each node in the attack graph represents an

exploit by the attacker. Each path from an initial node to a goal node represents a successful attack. .

Virtual machines in the cloud can be profiled to get precise information about their state, services running, open ports, and so on. An attacker can use port-scanning program to perform an intense scan for network to look for open ports on any VM.

The network controller is a key component to support the programmable networking capability to realize the virtual network reconfiguration feature based on Open Flow protocol. In NICE, within each cloud server, there is a software switch which is used as the edge switch for VMs to handle traffic in and out from VMs handled by physical Open Flow-capable Switch (OFS).

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.

3 NICE-1 Algorithm

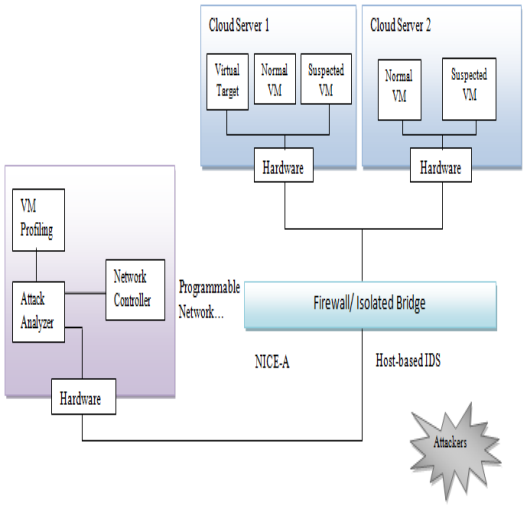
In attacker's store, a malware program is detected, and it sends the program to NICE scanning node, it gathers the file information and sends information to the Analyzer. The analyzer draws a graph, and it shows the status and it sends to the controller. In the software part decision of the switch software is taken and is sent through the switching software bridge of NICE scanning, and the process is again repeated.

NICE technique to establish a defense-in-depth intrusion detection framework. For better attack detection, NICE incorporates attack graph analytical procedures into the intrusion detection processes.

Lightweight mirroring-based network intrusion detection agent (NICE-A) acts on each cloud server to capture and analyze cloud traffic and identified vulnerability.

Once a VM enters inspection state, Deep Packet Inspection (DPI) is applied, and/or virtual network reconfigurations can be deployed to the inspecting VM to

make the potential attack behaviors prominent as shown in FIG 2.



1 system architecture

- NICE 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 [5] [6].

4 Comparison between NICE and NICE-1 Algorithm

NICE Algorithm	NICE-1 Algorithm
1.Cost is expensive	Cost is less
2.It involves multiple firewalls	It has only one firewall
3.Its architecture is so much complicated	Its architecture is very simple

4. Network controller is individually managed	Network controller is managed as a whole and it is connected to any number of VM profiles.

5 CONCLUSION

NICE-1 is easy and efficient to implement. In proposed solution, it investigates how to use the programmability of software switches-based solutions to improve the detection accuracy and defeat victim exploitation phases of collaborative attacks. In future, to improve the detection accuracy, host-based IDS solutions are needed to be incorporated and to cover the whole spectrum of IDS in the cloud system [7][8][9]. This should be investigated in the future work.

REFERENCES

1. B. Joshi, A. Vijayan, and B. Joshi, "Securing Cloud Computing Environment Against DDoS Attacks," Proc. IEEE Int'l Conf. Computer Comm. and Informatics (ICCCI '12), Jan. 2012.
2. H. Takabi, J.B. Joshi, and G. Ahn, "Security and Privacy Challenges in Cloud Computing Environments," IEEE Security and Privacy, vol. 8, no. 6, pp. 24-31, Dec. 2010.
3. K. Kwon, S. Ahn, and J. Chung, "Network Security Management Using ARP Spoofing," Proc. Int'l Conf. Computational Science and Its Applications (ICCSA '04), pp. 142-149, 2004.
4. O. Sheyner, J. Haines, S. Jha, R. Lippmann, and J.M. Wing, "Automated Generation and Analysis of Attack Graphs," Proc. IEEE Symp. Security and Privacy, pp. 273-284, 2002.
5. O. Database, "Open Source Vulnerability Database (OVSDB)," http://osvdb.org/, 2012.
6. R. Buyya, J. Broberg and A. Goscinski, Cloud Computing Principles and Paradigms. Wiley, 2011, vol. 81.
7. G. Tomsho, Guide to Networking Essentials. Course Technology Ptr, 2011.
8. P. Mell and T. Grance, The nist definition of cloud computing, National Institute of Standards and Technology, vol. 53, no. 6, 2009
9. C. Mazzariello, R. Bifulco, and R. Canonico, Integrating a network ids into an open source cloud computing environment, 2010.