Scenario Attack Graph (SAG): A Network Intrusion Detection and Countermeasure Mechanism

Suhelahmed Reshmi[1]and Anand S.Pashupatimath[2]

[1]M.Tech (Computer Science and Engineering), SDMCET, Dharwad

suhelreshmi@gmail.com

[2]Assistant Professor in CSE Dept., SDMCET, Dharwad

anand.pashupatimath@gmail.com

ABSTRACT

Now days as the usage of computers and networks is rising in each area of day-to-day life, it becomes very crucial to provide security to computers and also to networks in order to prevent the misuse or loss of private data. In the network security, this methodology is called Intrusion Detection System (IDS) is often utilized as a part of domain of network security and subsequently this is considered as the key exploration issues to the researchers. There are many methods and tools for IDSs suggested previously by numerous researchers, however each techniques fails at some extend while the attackers changes their attacking approach on personal computers and networks. The attacked VM (Virtual Machine) in the network is called as compromised VMs which are then used by attacker to compromise other VMs in the network. Consequently one need to have effective method for the detection of compromised VMs in network and is associated in performing the activities like spamming. To avoid these VMs from concession, we propose a multiphase distributed solution called as Scenario Attack Graph (SAG): A network intrusion detection and countermeasure mechanism. This method provides promising solution for system administrators to monitor and predict the growth of intrusions and thus to take relevant countermeasures in a timely manner.

Keywords- Network security, cloud computing, intrusion detection, attack graph, Compromised VMs

1. INTRODUCTION

The cloud computing has become the standard global data infrastructure that depends on much of today's social, cultural and commercial activities. Accompanying the increasingly crucial role the cloud computing plays in our society, an increasing number of attacks and threats aimed at this critical data infrastructure. For an organization that has a sensitive data, much more critical threats arise from sophisticated attackers, who (i) usually work in concert, (ii) can influence the resources of internet-wide uncertain 'zombie' machines that they have incorporated, and (iii) can make use of unknown exploits.

As a primary step in launching an attack, attackers examine and scan machines on the cloud, in order to discover network hosts and systems that are vulnerable. They exploit the detected vulnerabilities to compromise a system by altering the data and system configurations, establishing malicious codes, or stealing identities. Apart from attacking the compromised systems, intruders and attackers use them as stepping-stones to launch further attacks. Accordingly, attacks are increasingly multistep, spread out over hours, days or even for weeks. Recent years have seen an increased conception of sophisticated tools, thus enabling an attacker to create and launch attacks against detected vulnerabilities in a short amount of time and with little effort. Because the "discover vulnerability - create attack - launch attack" sequence has been significantly reduced, a greater number of attacks appear new to present defensive mechanisms, making it much difficult to protect against.

Specifically, the dramatic increase in legitimate activity both network and system has made it easier for assailants to hide their malicious activity. Additionally, the insecure cloud, with its large number of high speed interconnected machines, has become an effective resource that can be used by assailants to launch large-scale distributed attacks. Furthermore, vulnerabilities and flaws in protocol design and implementation, misconfigured systems, complex software code, and the failure of system operations, regularly leaves a large number of machines open to being incorporated by malicious attackers who infect them with specialized malware. Current attacks are extremely distributed. Typically, attackers compromises unattended hosts and use them as the command-and-control centers, which are then further used to compromise other machines, called zombies. In turn, these zombies are used to attack the physical target system. Therefore, watching for one or few outside attackers does not help to discover sophisticated attacks. When these attacks are launched, the command-and-control will transmit a particular signal to each zombie, which makes it function in a specific malicious manner. Under normal circumstances, zombies function like normal machines, and thus are not suspected. The capability of attackers to launch sophisticated, extensive attacks stems from their ability to yield the capability of hundreds to thousands of zombies within short notice. We propose a multiphase distributed solution called as Scenario Attack Graph (SAG): A network intrusion detection and countermeasure mechanism to detect and avert the attempts to compromise VMs, thus countering zombie VMs.

2. RELATED WORK

The existing IDS methods are based upon the use of centralized processing methods, hierarchical processing methods and distributed acquisition methods. Such IDS designs later go through from few limitations mainly in the robust analysis of elements with higher loading and often leads to the failure of the single point. The main source of malicious activities, intrusion, attacks are Internet now days and its does specifically through the web applications installed in the cloud. The worms in the Internet extended over various networks through the activities like searching, attacking as well as inevitably infecting the remote machines. Thus such types of intrusions are now days become viable threats for secured information. To determine the security against such intrusions in the network is that to establish the various properties of virus in which the effect of patching is included, the impact of network traffic, even the ways how these malicious codes reside in a certain hosts and awareness of other human countermeasures are included. An attack graph is able to perform a series of exploits called as atomic attacks that leads to an undesirable state means a state where an attacker obtains the administrative access to a machine. There are numerous automation tools such as Binary Decision Diagrams (BDDs) and new symbolic model checking NuSMV which are used to construct attack graph. These models can generate all potential attack paths; however, the scalability is a major problem for this solution. Firewall and Intrusion Detection System (IDS) are extensively used to detect and monitor suspicious events in the network. However, the large volume of raw alerts and false alarm from IDS are two major problems for some IDS implementations. To identify the target or source of the intrusion in the network, particularly to detect multi-step attack, the alert correction is a must have tool. The main goal of alert correlation is to determine system support for a global and concise view of network attacks by examining raw alerts.

3. SYSTEM ARCHITECTURE

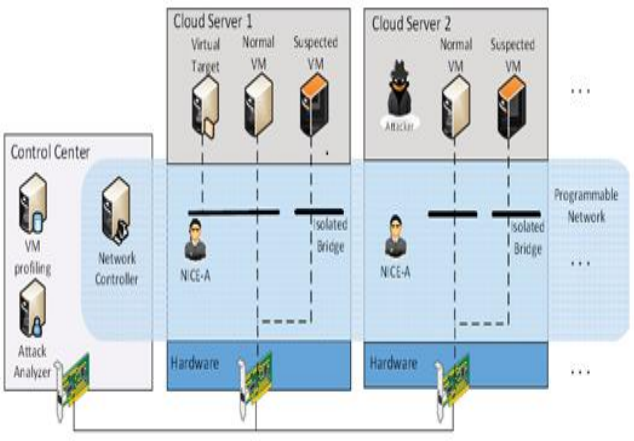

Fig-1 System architecture of SAG

The various modules used in the system are:

1. NICE-A
2. VM Profiling
3. Attack Analyzer
4. Network Controller

3.1. 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. Each bridge forms an isolated subnet in the virtual network and connects to all related VMs.

3.2. VM PROFILING:

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. VM profiles are maintained in a database and contain comprehensive information about vulnerabilities, alert and traffic.

3.3. ATTACK ANALYZER:

The major function of attack analyzer 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.

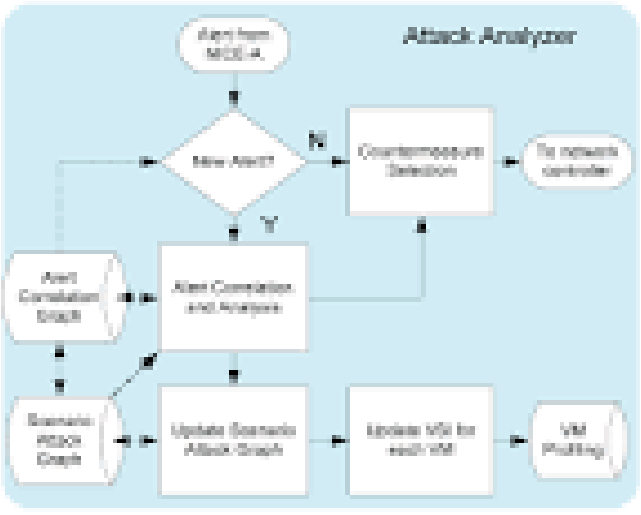

Fig-2 Attack Analyzer

The Attack Analyzer also handles alert correlation and analysis operations. This component has two major functions: (1) Constructs Alert Correlation Graph (ACG), (2) Provides threat information and appropriate countermeasures to network controller for virtual network reconfiguration.

4. NETWORK CONTROLLER

The network controller is a key component to support the programmable networking capability to realize the virtual network reconfiguration. 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.

5. CONCLUSION AND FUTURE SCOPE

SAG utilizes the attack graph model to conduct attack detection and prediction. The proposed solution investigates how to use the programmability of software switches-based solutions to improve the detection accuracy and defeat victim exploitation phases of collaborative attacks. The proposed solution can significantly reduce the risk of the cloud system from being exploited and abused by internal and external attackers.

SAG only investigates the network IDS approach to counter zombie explorative attacks. 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. This should be investigated in the future work. Additionally, as indicated in the paper, we will investigate the scalability of the proposed SAG solution by investigating the decentralized network control and attack analysis model based on current study.

REFERENCES

[1] Chun-Jen Chung, Student Member, IEEE, Pankaj Khatkar, Student Member, IEEE, Jeongkeun Lee, Tianyi Xing, 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.

[2] M. Armbrust, A.D. Joseph, A. Fox, R. Griffith, G. Lee, R. Katz, A. Konwinski, I. Stoica, D. Patterson, A.

Rabkin, and M. Zaharia, "A View of Cloud Computing," ACM Comm., vol. 53, no. 4, pp. 50-58, Apr. 2010.

[3] B. Joshi and B. Joshi, "Securing Cloud Computing Environment Against DDoS Attacks," Proc. IEEE Int'l Conf. Computer Comm. and Informatics (ICCCI '12), Jan. 2012.

[4] J.B. Joshi, H. Takabi and, G. Ahn, "Security and Privacy Challenges in Cloud Computing Environments," IEEE Security and Privacy, vol. 8, no. 6, pp. 24-31, Dec. 2010.

[5] "Open vSwitch Project," http://openvswitch.org, May 2012.

[6] Z. Duan, Y. Dong, P. Chen, M. Stephenson, F. Sanchez, , and J. Barker, "Detecting Spam Zombies by Monitoring Outgoing Messages," IEEE Trans. Dependable and Secure Computing, vol. 9, no. 2, pp. 198-210, Apr. 2012.

[7] G. Gu, M. Fong, V. Yegneswaran, P. Porras, and W. Lee, "BotHunter: Detecting Malware Infection through IDS-driven Dialog Correlation," Proc. 16th USENIX Security Symp. (SS '07), pp. 12:1-12:16, Aug. 2007.

[8] J. Zhang, G. Gu, and W. Lee, "BotSniffer: Detecting Botnet Command and Control Channels in Network Traffic," Proc. 15th Ann. Network and Distributed Sytem Security Symp. (NDSS '08), Feb. 2008.

[9] O. Sheyner, R. Lippmann, S. Jha, J. Haines, and J.M. Wing, "Automated Generation and Analysis of Attack Graphs," Proc. IEEE Symp. Security and Privacy, pp. 273-284, 2002,

[10] "NuSMV: A New Symbolic Model Checker," http://afrodite.itc. it:1024/nusmv. Aug. 2012.

[11] P. Ammann, S. Kaushik, and D. Wijesekera, "Scalable, graph based network vulnerability analysis," Proc. 9th ACM Conf. Computer and Comm. Security (CCS '02), pp. 217-224, 2002.

[12] S. Govindavajhala, X. Ou, and A.W. Appel, "MulVAL: A Logic- Based Network Security Analyzer," Proc. 14th USENIX Security Symp., pp. 113-128, 2005.

[13] D.S. Kim, A. Roy, and K. Trivedi, "Scalable Optimal Countermeasure Selection Using Implicit Enumeration on Attack Countermeasure Trees," Proc. IEEE Int'l Conf. Dependable Systems Networks (DSN '12), June 2012.

[14] R. Dewri, N. Poolsappasit, and I. Ray, "Dynamic Security Risk Management Using Bayesian Attack Graphs," IEEE Trans. Dependable and Secure Computing, vol. 9, no. 1, pp. 61-74, Feb. 2012.

