Ensuring Higher Accuracy of Intrusion Detection in Virtual Environment using Countermeasure Selection

Prajakta Godambe , Rohini Pise , Swachchhanda Patil , Shital Ghate

Department of Information Technology
Pimpri Chinchwad College of Engineering Pune , Maharashtra , India.

***Abstract*— Nowadays Cloud Computing is one of the most emerging Technology and many user works on cloud environment; but the Security in the Cloud environment is the key challenge to such system. Most of the time attackers attack on a machine in a network and try to compromise it as Zombie and Originate DDoS-Distributed Dos Attack from it. There is a need to Detect such Compromised Machine involve in suspicious activities named as Zombies and preventing them from spreading DDoS attacks in the network. Detection of such Zombies in Cloud system particularly in Infrastructure as a Service (IaaS) Cloud is very difficult job.**

***Keywords*— Zombie,DDoS,cloud computing, Zombie Detection.**

I. Introduction

In the today's Internet word major challenge is presence of malicious system which is compromised as a zombie. Most of the time attackers are controlling such machines to spread various attacks such as DDoS, malwares, spam messages, identity theft [1]. These types of attacks mainly focus on particular machine on network and they will create lots of unwanted traffic to destroy the load capacity of system or particular machine. There is need to detect such compromised machines and then to block those machines. So we are trying to implement such system to detect and prevent the network from different attacks like DDoS or Zombies using NIDS.

Network Intrusion Detection System (NIDS) is a system that traces all the malicious activities running on network by examining the network traffic. A recent CSA (Cloud Survey Alliance) survey shows that from all security threats the misuse and reprehensible use of cloud computing is inspected as the main security issue. Previously data centres were monitored by administrators only. All the controlling and managing activities were performed by those administrators. There were no such systems to monitor such malicious activities. Nowadays cloud users can install vulnerable softwares on their virtual machines that lead to escape clauses in cloud environment. It may lead to loopholes in cloud and may cause attackers to target machines in network and compromise them as zombies. Such zombie machines can be further used to spread distributed denial of service attacks in whole network.

In a cloud system where the infrastructure is shared by millions of users, so the wrong and vicious use of these resources gives chance to attackers to utilize vulnerabilities of the cloud and use its resource to spread attacks in more efficient ways. These attacks are very harmful and dangerous in cloud environment and cloud users generally share resources in huge amount, e.g. sharing same ports which are connected to each other, data files. The similar setup for VMs in the cloud, e.g., virtualization techniques, VM OS, installed vulnerable software, networking, etc., attracts attackers to compromise multiple VMs.

In this article, we propose Intrusion Detection and Prevention system With Higher Degree of Accuracy Using Alert Corelation Algorithm in Virtual Networking to establish a defense-in-depth intrusion detection framework. For better attack detection, our system incorporates attack graph analytical procedures into the intrusion detection processes. We must note that the design of above system does not intend to improve any of the existing intrusion detection algorithms; indeed, our system employs a reconfigurable virtual networking approach to detect and counter the attempt to compromise VMs, thus preventing zombie VMs.

II. Related work

An attack graph is capable to symbolize a sequence of exploits called atomic attacks that direct to an undesirable state. There are several automation tools to build attack graph. A proposed method based on a customized symbolic replica checking and Binary Decision Diagrams (BDDs) to build attack graph. Their model can produce all possible attack paths. On the other hand the scalability is a huge problem for this solution. In other words attackers never need to back down. Intrusion Detection System (IDS) and firewall are broadly used to check and notice distrustful events in the network. Conversely the false alarms and the large volume of raw alerts from IDS are two major troubles for any IDS implementations. In order to recognize the source or target of the interruption in the network especially to perceive multi-step attack the alert correction is a must-have tool. The primary goal of alert correlation is to afford system support for a global and compacted view of network attacks by analysing raw alerts.

III. Existing System

Cloud users works on their VMs, they may install and use susceptible software. Due to these types of vulnerable softwares, there are chances of getting ambiguities in cloud environment. The security of cloud may also be compromised. There is a need of an effective system that detects such types of vulnerability. But it is a big issue in cloud. Cloud offers shared infrastructure between numbers of users. It may benefit assaulters to make nefarious use of shared environment. They may make use of ambiguities in cloud and can deploy attacks in virtual environment in cloud. As the resources are shared by cloud users, these types of attacks are more difficult to identify and detect, e.g., being connected through the same switch, sharing with the same data storage and file systems between multiple users. The configuration of cloud VMs is similar, e.g., virtualization techniques, VM OS, installed vulnerable software, networking. Such configuration can direct assaulters to compromise multiple VMs and make misuse.

We must note that the configuration of intrusion detection framework does not mean to enhance any of the current interruption discovery calculations; in-deed, it utilizes a reconfigurable virtual systems administration methodology to discover and counter the endeavours to bargain VMs, in this manner anticipating zombie VMs. Decent consolidates a delicate product exchanging answer for isolate and review suspicious VMs for further examination and assurance. Through programmable system approaches, intrusion detection framework can enhance the assault location likelihood and enhance the flexibility to VM abuse assault without intruding on existing ordinary cloud services.

Disadvantage of Existing System:

1. There is no such technique or model is available for discovering attacks in virtual networks.
2. The attack detection mechanism does not provide a high accurate solution.

IV. Proposed System

In this article, we propose Intrusion detection framework selection in virtual network systems using Countermeasure. This framework comprises of attack graph analytical processes into network intrusion detection processes. This framework does not reconfigure or redevelop any available detection algorithms. The framework applies a new virtual networking measure to discover and counter the endeavors to misuse VMs. This also helps to prevent the distributed DoS attacks in virtual environment.

We propose Resolving security issues in virtual system frameworks to build a protection inside and out interruption discovery schema. For better threat identification, the framework joins threat chart explanatory techniques into the proposed interruption discovery forms. We propose multistage disseminated helplessness recognition, estimation, and countermeasure determination component which is based on threat chart based diagnostic models and reconfigurable virtual system based countermeasures. The proposed skeleton powers Open Flow system programming APIs to manufacture a screen and control plane over disseminated programmable virtual switches to essentially enhance threat location and moderate threat outcomes. The framework and security assessments exhibit the proficiency and adequacy of the proposed result.

Advantage of Proposed System:

1. We prepare a framework for virtual environment that catches and monitors mistrustful traffic not disrupting the user's normal work and services.
2. The attack identification and reporting capability is increased.
3. The configuration of cloud server is optimized to reduce resource expenditure.
4. The framework uses minimum computation as compared to other network.

V. System Architecture

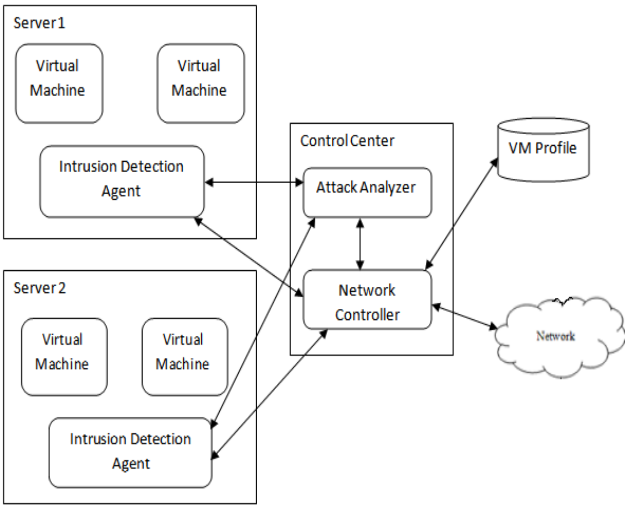


Fig. 1 Architecture of Intrusion Detection Framework.

The proposed framework is illustrated in figure 1.It shows the framework within two cloud server clusters. The framework comprises of different components that are distributed in nature. There is an Intrusion Detection agent installed in each cloud server, a controller that monitors the network, an attack analyzer, a VM profiling database. The other components excluding Intrusion Detection agent are placed in a centralized control center. This control center is again connected to switches on each cloud server. Intrusion Detection agent is a software agent available in each cloud server connected to the control center through a predefined secure medium. This medium or channel is different from data packets of VLAN.

The network controller determines and applies the required countermeasures according to information received from attack analyzer. Whenever the mistrustful traffic is identified, the received alert is passed to attack analyzer in control center. Attack analyzer measures the severity of received alert based on the previously set attack graph. From attack graph it decides the appropriate countermeasure solutions to perform and then start it using the network controller.

VI. Algorithm

Algorithm 1 Countermeasure Selection

Step 1*: Wait for client request for accessing applications and then accept it.*

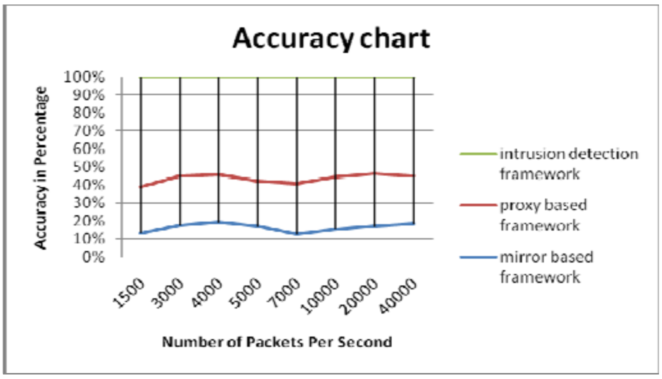
Step 2*: Accept the connection. Checks the blacklist for client's IP address, if it exists then do not allow client to access the application and go to step 1.*

Step 3: *Allow accessing the applications and is monitored by XVDS.*

Step 4: *If IP address crosses set entropy value then it is informed to Attack Analyzer.*

Step 5: *Attack analyzer starts monitoring the IP address with new entropy value for any misbehaviour.*
Step 6: *If IP address crosses entropy value at Attack Analyzer then IP address is checked first at Local database for any history of misbehaviour.*
Step 7: *If result is negative at Local database, IP address is checked at Global database for any history of misbehaviour.*
Step 8: *If result is positive in any databases, the IP address is blocked immediately from accessing any applications.*
Step 9: *For that particular IP address Scenario Attack Graph is generated then the IP address is blocked.*
Step 10: *The blocked IP address is updated in internal database and in network controller.*

VII. RESULT



VIII. CONCLUSION

The proposed system detects and mitigates attacks in cloud virtual environment. The system overcomes risk of raw alerts and false alarms. This approach concentrates only on DDoS attacks. The proposed solution can considerably reduce the risk of the cloud system from being exploited and abused by internal and external attackers. NICE only investigates the network IDS approach to counter zombie explorative attacks. In order to get better detection accuracy, host-based IDS solutions are needed to be incorporated and to cover the whole spectrum of IDS in the cloud system. In future researches can be performed for other attacks in cloud.

ACKNOWLEDGMENT

We are thankful to our Teachers, Friends and Department of Information Technology for their constant guidelines and support. We also thank the researchers as well as publishers for making their resources available. We are also thankful to college authorities for providing the required infrastructure and support.

REFERENCES

[1] Chun-Jen Chung, Pankaj Khatkar, Tianyi Xing, Jeongkeun Lee, and Dijiang Huang, "NICE: Network Intrusion Detection and Countermeasure Selection in Virtual Network Systems" IEEE transactions on dependable and secure computing, vol. 10, no. 4, july/august 2013.

[2] Armbrust, A. Fox, R. Griffith, A.D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, "A View of Cloud Computing," ACM Comm., vol. 53, no. 4, pp. 50-58, Apr. 2010.

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

[4] 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.

[5] R. Sadoddin and A. Ghorbani, "Alert Correlation Survey: Framework and Techniques," Proc. ACM Int‟l Conf. Privacy, Security and Trust: Bridge the Gap between PST Technologies and Business Services (PST ‟06), pp. 37:1-37:10, 2006.

[6] Coud Sercurity Alliance, "Top threats to cloud computing v1.0,"https://cloudsecurityalliance.org/topthreats/csathreats.v1.0.pdf , March 2010.

[7] "Open vSwitch project," http://openvswitch.org, May 2012.

[8] Z. Duan, P. Chen, F. Sanchez, Y. Dong, M. Stephenson, 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. IEEE TRANSACTIONS ON DEPEDABLE AND SECURE COMPUTING.

[9] G. Gu, P. Porras, V. Yegneswaran, M. Fong, and W. Lee, "BotHunter: detecting malware infection through IDS-driven dialog correlation," *Proc. of 16th USENIX Security Symp. (SS '07)*, pp. 12:1–12:16, Aug. 2007.

[10] G. Gu, J. Zhang, and W. Lee, "BotSniffer: detecting botnet command and control channels in network traffic," *Proc. of 15th Ann. Network and Distributed Sytem Security Symp. (NDSS '08)*, Feb. 2008.

[11] O. Sheyner, J. Haines, S. Jha, R. Lippmann, and J. M. Wing, "Automated generation and analysis of attack graphs," *Proc. IEEE Symp. on Security and Privacy*, 2002, pp. 273–284.

[12] "NuSMV: A new symbolic model checker," http://afrodite.itc.it: 1024/~numb. Aug. 2012.

[13] S. H. Ahmadinejad, S. Jalili, and M. Abadi, "A hybrid model for correlating alerts of known and unknown attack scenarios and updating attack graphs," *Computer Networks*, vol. 55, no. 9, pp. 2221–2240, Jun. 2011.

[14] S. Roschke, F. Cheng, and C. Meinel, "A new alert correlation algorithm based on attack graph," *Computational Intelligence in Security for Information Systems*, LNCS, vol. 6694, pp. 58–67. Springer, 2011.

[15] A. Roy, D. S. Kim, and K. Trivedi, "Scalable optimal countermeasure selection using implicit enumeration on attack countermeasure trees," *Proc. IEEE Int'l Conf. on Dependable Systems Networks (DSN '12)*, Jun. 2012.

[16] "Open flow." http://www.openflow.org/wp/learnmore/, 2012.