SECURE CLOUD USING INTRUSION DETECTION AND COUNTERMEASURES SELECTION

S.Deepak[1], P. Kiruthika[2], H.Anandakumar[3], B.Anuradha[4]

PG Scholar[1], Assistant Professor[2], Assistant Professor[3], Associate Professor[4]
Department of Information Technology, SNS College of Engineering, Coimbatore.

Abstract: **In Cloud virtual machine is considered as the security threat. Virtual machines are vulnerable to Denial of Service attacks. Denial of Service attacks involves action such as low frequency vulnerability scanning, exploitation and compromising vulnerable virtual machines. This paper proposes a framework to detect and mitigate attacks within the cloud environment. To prevent vulnerable virtual machines in the cloud, a multi-phase vulnerable detection and countermeasure selection, which built an attack graph, based analytical models and reconfigurable virtual network-based countermeasures. Host-based IDS solutions are incorporated in Network Intrusion Detection and Countermeasure Selection in virtual Network Systems. The encrypted data Vulnerable can be monitor using intrusion detection agent. This framework is an Open Flow network, monitor and control over distributed virtual switches in order to improve attack detection and consequences. Port analyser is used to monitor the traffic and detect the vulnerable. The detection accuracy is exaggerated by incorporating the host based intrusion detection system. The Security evaluations demonstrate the efficiency and effectiveness of the proposed solution.**

Keywords: Network security, Intrusion detection, Compromised machine, Cloud computing, Countermeasure selection

I. INTRODUCTION

A Recent Cloud Security survey shows all security problems and abuse use of cloud computing is considered as the prime security threat in which attackers exploit vulnerabilities in cloud and utilize cloud resources to deploy attacks. Intrusion Detection System is a system that attempt to find unauthorized access to network by analyzing traffic on the network for signs of malicious activity. In ancient vulnerabilities detected and patched by the administrator in a centralized manner, patching known security hole in cloud. The challenge is to determine an efficient vulnerability attack detection and response system for accurately analyzing attacks and minimizing the impact of security breach to cloud users. Cloud Computing makes infrastructure and services available on-need basis. Cloud is a set of a network enabled scalable and services that might be accessed in a pervasive approach. SPOT is designed based on a statistical method called Sequential Probability Ratio Test. It is a strong statistical procedure which is able to accustomed check between two hypotheses. It is mainly used to analyze and detect the attacks; attacks are more effective within the cloud environment as a result of cloud users sometime share computing resources being connected through identical switch, sharing the identical information storage and file systems. This paper proposes the NICE to determine a defense-in-depth intrusion detection framework for attack detection. VM deploy a light-weight mirroring-based intrusion detection agent on every cloud server to capture and analyze traffic in network. ID's agent sporadically scans the virtual system vulnerabilities inside a cloud server to determine attack graph. It is a tool to illustrate all possible multistage, multi host attack method to grasp threats and then to make appropriate countermeasure. Once a VM enters inspection state, deep packet inspection is applied. The attacker's primary goal is to take advantage of vulnerable VMs and compromise them as zombies. The switch port analyzer is employed to monitor the network traffic and analyze the threshold of the packet. The switching techniques construct a dynamic IDS system. It sporadically scans the virtual system vulnerabilities inside a cloud to establish attack graph. A virtual network based attack detection and solutions to improve the resiliency to zombie explorations. The Host based intrusion detection is employed to improve the accuracy of detection. A correlation algorithm based attach graph is capable of detecting multiple attack scenarios. It employs a re-configurable virtual networking approach to find a countermeasure that attempts to compromise VMs. Deep Packet Inspection is applied to virtual network is deployed to the inspecting VM to make the potential attack behaviors. Host based detection and countermeasure constructs a mirroring-based traffic framework to attenuate the interference on users.

II. RELATED WORK

An attack graph represent a series of exploits refer as atomic attacks that associate undesirable state. An attacker gains access to your credentials they'll listen in on your activities and transactions, manipulate information come falsified knowledge and send your clients to illegitimate sites [1]. The Bot Sniffer can detect real world botnets with high accuracy and low false rate. Exploits uniform abstraction temporal behavior characteristics of compromised machines to detect zombies by grouping flows to server connections This makes the detection of botnet C&C a difficult drawback. It proposing associate approach that uses network based anomaly detection to spot botnet C&C channels. [2]. Bot Hunter detects compromised machines based on Perimeter monitoring strategy which focused on recognizing the infection and coordination that occurs during a malware infection, mostly on the reality that a thorough malware infection method has a range of well-defined stages that permit correlating the intrusion alarms triggered by inbound traffic with ensuing outgoing communication patterns. Attack graph is capable of detecting multiple attacks for rhetorical analysis [3]. MUL-VAL that adopts a logic programming approach and uses Data log language to model and analyze network. Firewall and Intrusion Detection System are widely used to measure and find suspicious events within the network. It is associate end-to-end framework and reasoning system [4]. Cloud Trace Back is used to find the source of denial of service attacks and introduce the utilization of a back propagation neutral network known as Cloud defender, which was trained to discover and filter attack traffic [5]. SPOT is designed based on a powerful statistical tool called Sequential Probability Ratio Test, which bounded false positive and negative error rates. It is used to monitoring outgoing messages on a network. Vulnerable machines are one of the security threats on the internet they are usually used to launch numerous security attacks, spamming and spreading malware, DDOS and determine theft. Given that spamming provides a key economic incentive for attacks to recruit the massive variety of compromised system [6]. Attack graph method come up with generate attack trees. The structure will take advantage of the penetration achieved by previous exploits in its chain also the final exploit within the chain achieves the attacker goal [7]. An Alert correlation was proposed to analyze alerts and to decrease false positives and negative, knowledge about the target system or environment is usually necessary for efficient alert correlation. An attack graph based correlation algorithm to form explicit correlations only by matching alerts to specific exploitation nodes within the attack graph with multiple functions and devised a dependencies graph to cluster connected alerts with multiple regression, identify source and target of the intrusion within the network to detect multistep attack [8] Open v Switch's forwarding path is designed to be enable to "offloading" packet processing to hardware chipsets, whether housed in a classic hardware switch chassis or in an end-host NIC. This allows for the Open v Switch control path to be able to both control a pure software implementation or a hardware switch [9].

III. EXISISTING SYSTEM

Network intrusion detection and countermeasure selection systems establish a defense-in-depth intrusion framework. This incorporates attack graph analytical procedures into the intrusion detection processes for better detection.

A. NICE System Overview

The NICE framework is distributed light weighted intrusion detection agent on network controller, VM profiling server, and an attack analyzer.

1. NICE-A

Intrusion detection agent is deployed on every cloud server to find and analyze cloud traffic. It scans the vulnerable virtual machine within a cloud. When an anomalous traffic is detected intrusion detection alerts are passed by intrusion agent to analyzer packet and attacker.

2. VM Profiling

Virtual machines in the cloud can get information about their state using the VM profiling. The states are services running, open ports, etc. VM profile contains information about the connectivity with other virtual machine. The information about the services running on a VM is to find the authenticity of alerts pertaining to that VM. Attacker can use switch port analysis program to examine the network to analyze the open ports on virtual machine. Data logs about open ports on a VM and the history of ports plays a major role in determining vulnerable virtual machine. The VM profiles information is maintained in a database and contain information about alert, traffic and vulnerability.

3. Attack Analyzer

The significant functions of attack analyzer are to generate a graph and update, alert correlation and countermeasure selection. The process of generating and utilizing the Attack Graph consists of several phases' information analyzing, graph construction, and exploit path analysis. With this data attack graph can be generated using Scenario attach graph. The alerts are sending from intrusion detection agent, alert analyzer matches the alert in the alert correlation graph. If the alert are already identifies in the graph, the attack analyzer performs countermeasure selection procedure.

4. Network Controller

Network controller is used to collect network information and provides input to the attack analyzer to model an attack graphs. Based on optimal return of investment and risk probability are calculated, the countermeasures are selected by attack analyzer and executed by network controller.

IV. PROPOSED SYSTEM

In the proposed system Host-based Intrusion Detection system analyzes the traffic and specify computer on which the intrusion detection software is installed. Host based Intrusion Detection system is integrated with NICE to improve attack detection. A host-based system monitors as well as analyzes a systems internals along with the network packets. The result are logged into a secure database and compared with the knowledge base to detect any malicious activity. A host-based system analyzes logs and consists of information regarding the status of your system.

A. System Overview

The components in Host intrusion detection framework are distributed and intrusion detection agent on each cloud server, a network controller and an attack analyzer.

1. Intrusion detection Agent

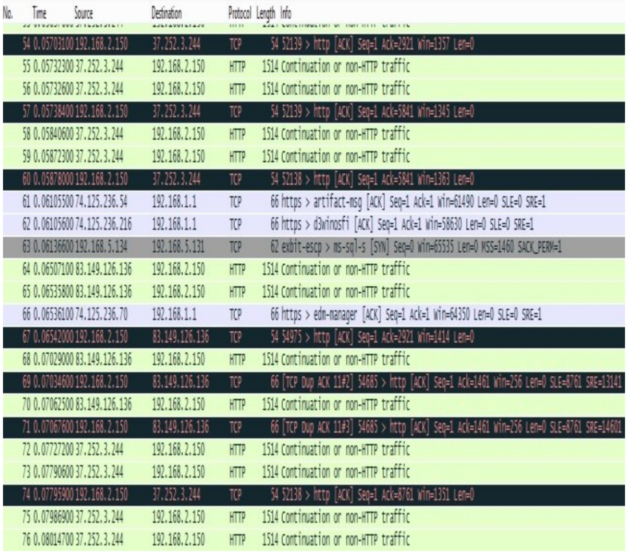
Host intrusion detection agent is implemented in each cloud server and analyzes the packets transferred within the network. Virtual switches built on one or multiple virtual machines connected to the control center through a dedicated and secure channel. Intrusion agent will monitor the whole cloud for the request and the response and might realize the attacker within the network.

2. Analyze Packets

The Agent can analyze the packets transferred in the network. It contains the packet size, Number of packets and also the source ip and destination ip also present in it, Duration of the packets are also present in the captured packets using these details the attacker is mitigated.

$$\text{Packet threshold size} = \frac{\text{total packet in all transfer}}{\text{Number of transfers}}$$

$$\text{Duration threshold value} = \frac{\text{sum of all duration in all transfers}}{\text{Number of transfers}}$$



```
No.  Time       Source          Destination     Protocol Length Info
54 0.05703100 192.168.2.150   37.252.3.244     TCP      54 52139 > http [ACK] Seq=1 Ack=2921 Win=1357 Len=0
55 0.05732300 37.252.3.244    192.168.2.150    HTTP   1514 Continuation or non-HTTP traffic
56 0.05732600 37.252.3.244    192.168.2.150    HTTP   1514 Continuation or non-HTTP traffic
57 0.05738400 192.168.2.150   37.252.3.244     TCP      54 52139 > http [ACK] Seq=1 Ack=5841 Win=1345 Len=0
58 0.05840600 37.252.3.244    192.168.2.150    HTTP   1514 Continuation or non-HTTP traffic
59 0.05872300 37.252.3.244    192.168.2.150    HTTP   1514 Continuation or non-HTTP traffic
60 0.05878000 192.168.2.150   37.252.3.244     TCP      54 52138 > http [ACK] Seq=1 Ack=5841 Win=1363 Len=0
61 0.06105500 74.125.236.54   192.168.1.1      TCP      66 https > artifact-msg [ACK] Seq=1 Ack=1 Win=61490 Len=0 SLE=0 SRE=1
62 0.06105600 74.125.236.216  192.168.1.1      TCP      66 https > d3winosfi [ACK] Seq=1 Ack=1 Win=58630 Len=0 SLE=0 SRE=1
63 0.06136600 192.168.5.134   192.168.5.131    TCP      62 exbit-escp > ms-sql-s [SYN] Seq=0 Win=65535 Len=0 MSS=1460 SACK_PERM=1
64 0.06507100 83.149.126.136  192.168.2.150    HTTP   1514 Continuation or non-HTTP traffic
65 0.06535800 83.149.126.136  192.168.2.150    HTTP   1514 Continuation or non-HTTP traffic
66 0.06536100 74.125.236.70   192.168.1.1      TCP      66 https > edm-manager [ACK] Seq=1 Ack=1 Win=64350 Len=0 SLE=0 SRE=1
67 0.06542000 192.168.2.150   83.149.126.136   TCP      54 54975 > http [ACK] Seq=1 Ack=2921 Win=1414 Len=0
68 0.07029000 83.149.126.136  192.168.2.150    HTTP   1514 Continuation or non-HTTP traffic
69 0.07034600 192.168.2.150   83.149.126.136   TCP      66 [TCP Dup ACK 11#2] 54685 > http [ACK] Seq=1 Ack=1461 Win=256 Len=0 SLE=8761 SRE=13141
70 0.07062500 83.149.126.136  192.168.2.150    HTTP   1514 Continuation or non-HTTP traffic
71 0.07067600 192.168.2.150   83.149.126.136   TCP      66 [TCP Dup ACK 11#3] 54685 > http [ACK] Seq=1 Ack=1461 Win=256 Len=0 SLE=8761 SRE=14601
72 0.07727200 37.252.3.244    192.168.2.150    HTTP   1514 Continuation or non-HTTP traffic
73 0.07790600 37.252.3.244    192.168.2.150    HTTP   1514 Continuation or non-HTTP traffic
74 0.07795900 192.168.2.150   37.252.3.244     TCP      54 52138 > http [ACK] Seq=1 Ack=8761 Win=1351 Len=0
75 0.07986900 37.252.3.244    192.168.2.150    HTTP   1514 Continuation or non-HTTP traffic
76 0.08014700 37.252.3.244    192.168.2.150    HTTP   1514 Continuation or non-HTTP traffic
```

Fig.1 Packet Analysisi

3. Attack Mitigation

The threshold id find out for all the terms like packet size, duration, number of packets. Then the attack graph is generated for the captured packets in the network. Then the risk probability is also calculated for the captured packets in the network.

$$\text{Packet average value} = \frac{\text{individual packer size}}{\text{Total packet size}}$$

$$\text{Risk probability} = 1 - \text{packet average value for each communication}$$

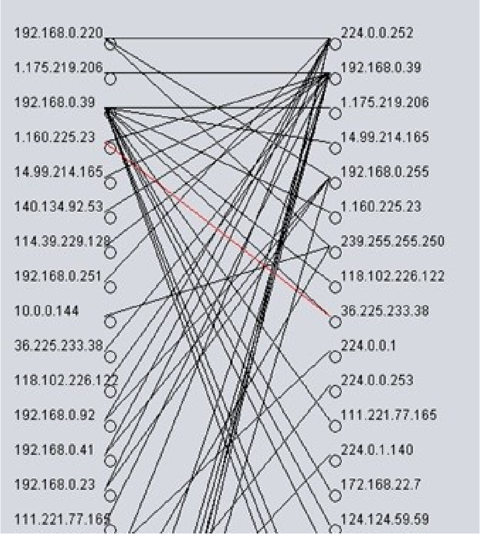


Fig.2 Attack Graph

4. Resource Allocation

The users in the network send the request to the server for resource. Then the server allocates the resource for the users who send request, based on the users request the resource is allocated. The allocated resources are utilized by the clients in the network.

5. Analyze Attacker

The users utilize the resource by transfer the data to the allocated space. The data transferred to the resource are analyzed by the server. The attacker transfer the malicious data to the cloud, the malicious data makes the server vulnerable. By analyzing the incoming packet in the resource the data vulnerable can be detected and appropriate countermeasure are taken by server. The server identifies vulnerable data and analyzes the packet source and identifies the user resource in the cloud environment and remove the entire space from cloud environment

B. Countermeasure Selection

Algorithm presents the optimal countermeasure for a given attack scenario.

Table I. Countermeasure

No.	Countermeasure	Intrusiveness	Cost
1	Traffic redirection	3	3
2	Traffic isolation	4	2
3	Deep Packet Inspection	3	3
4	Creating filtering rules	1	2
5	MAC address change	2	1
6	IP address change	2	1
7	Block port	4	1
8	Software patch	5	4
9	Quarantine	5	2
10	Network reconfiguration	0	5
11	Network topology change	0	5

V. SYSTEM DESIGN

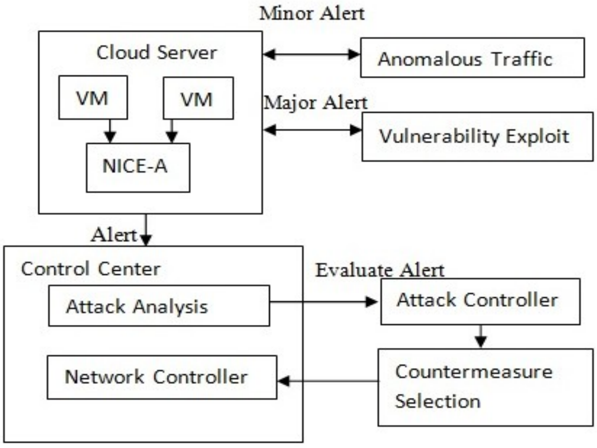


Fig.3 Virtual Network Security Evaluation

VI. CONCLUSION

In this paper, Host-based Intrusion Detection system is incorporated to improve the detection accuracy and Cover the whole spectrum of Intrusion Detection System in the cloud system. It utilizes the attack graph model to conduct attack detection and prediction The DDOS attack can be reduced in the cloud virtual system environment. It can reduce the risk of the cloud system from being exploited and abused by attackers.

VII. 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] G. Gu, J. Zhang, 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.

[3] G. Gu, P. Porras, V. Yegneswaran, M. Fong, 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.

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

[5] 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.

[6] 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.

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

[8] S. Roschke, F. Cheng, and C. Meinel, "A New Alert Correlation Algorithm Based on Attack Graph," Proc. Fourth Int'l Conf. Computational Intelligence in Security for Information Systems, pp. 58-67, 2011.

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