ISSN: 2321-7782 (Online)

Volume 2, Issue 11, November 2014

International Journal of Advance Research in Computer Science and Management Studies

Research Article / Survey Paper / Case Study

Available online at: www.ijarcsms.com

Hybrid Intrusion Detection System for Virtual Networks

A.Anitha[1]
Computer Science and Engineering
Aurora's Technological and Research Institute
Hyderabad, India

Ms. Naga Aswani Puduru[2]
Computer Science and Engineering
Aurora's Technological and Research Institute
Hyderabad, India

***Abstract:** Cloud Computing is a new computing model that makes the IT world to equip with ready to use services that help in outsourcing data, computing and so on. However, security is an important concern. Adversaries are able to launch DDoS (Distributed Denial of Service) attacks. These attacks compromise some of the Virtual Machines (VMs) first and then perform DDoS activities. The compromised VMs are known as zombies. With respect to cloud computing networks detecting zombie exploration attacks is difficult. When users of cloud install vulnerable applications, this will be exploited by hackers. Chung et al. proposed a network intrusion detection system which is based on graph-based analytical models and countermeasures. This improves attack detection and also reduces consequences of attacks. This Intrusion Detection System has limitations. Very important one is that it only focuses on network IDS approach to detect zombies while ignoring the host based IDS solution. To overcome this problem, in this paper we extend this NIDS further to incorporate host based IDS approach in order to create a hybrid IDS which exploits the benefits of both the IDS. Through extensive simulations, we test our Hybrid Intrusion Detection System (HIDS) which improves detection accuracy.*

***Keywords:** Network security, cloud computing, virtualization, intrusion detection, host-based IDS*

I. INTRODUCTION

Cloud computing has become a reality and it is being used by individuals and organizations in one way or other. It is a new computing model which enables individuals and organizations to have access to plethora of computing resources that are otherwise not easy to get. There is no capital investment required. Cloud provides its services in pay per use fashion. The business models after cloud computing technology can help organizations to get customized services. The cloud computing is based on virtualization which provides affordable cloud services. On top of virtualized networks, cloud computing is built. In this context, it is very important to have security measures like intrusion detection systems in place.

The success of cloud computing depends on the paradigm that makes the cloud to work on the top of virtualized networks. This is because the virtualization technology makes the cloud services cheaper as it helps in reducing physical machines and thereby costs incurred. As VM usage does not incur monetary burdens on cloud computing the usage of Virtual Machines is prevalent. This has paved way for the research on VM Ware and other related technologies. A machine within another machine is the virtual machine which is essentially a software component which mimics like a real machine. This is the trick that lets the industry to have services for lesser costs. However, VMs are vulnerable to attacks such as DDoS that causes problems to cloud service providers and cloud users. There should be mechanisms that can prevent these attacks. Many researchers provided solutions in the form of intrusion detection systems. The intrusion detection systems are either host based or network based. Host based system takes care of protection of one machine while the network based one protects the whole network as a whole. In order to prevent such attacks and secure the whole cloud network, recently, Chung *et al*. [1] proposed an architecture in virtualization and cloud that performed intrusion detection in distributed environment.

This paper focuses on host-based IDS that can provide fool proof security that is in virtual networks environment in the context of cloud computing. The proposed IDS provide detection accuracy and are robust with respect to delay performance.

The detection agent concept is employed where an agent is deployed into each VM for monitoring intrusions. This will be done effectively as every node is protected from malicious attacks. The system can also provide scalability besides improving detection accuracy and delay performance. The remainder of this paper is structured as follows. Section II reviews literature on intrusion detection systems. Section III provides proposed system. Section IV presents experimental results while section V concludes the paper.

II. Related Work

In this section literature is review on prior works. The IDS explored in [2] works well in identifying nodes that are compromised by adversaries. Same kind of research is made in [3] and treated the compromised nodes as zombies. The characterization of zombies is done in [4] for identifying zombies in communication networks. Attack graph construction and the notion of atomic attacks are explored in [4]. These attacks can cause the VMs to have inconsistent states. MulVAL is the attack graph tool used in [5] as part of detection system. Logic programming approach was used by MulVAL while the assumption of onotonicity was proposed in [6] with a pre-condition regarding invalidation with respect to each exploit.

When attack graph is considered in [7], the tracing of alerts is done by employing a new phenomenon that is related to exploits. In-memory based data structure is used and the solution is known as queue graph for better security. IDS and firewalls came into existence and served IT industry well. The intrusion detection systems provide plethora of alerts. Such alert correlation is also performed by many researchers in the context of intrusion detection system s as part of multi-part solution. Alert correlation is very useful as explored in [8] in order to identify attacks accurately. In [9], the attack graph concept is modified and the modified attack-graph was employed for writing a better correlation algorithm.

Later in [10] Bayesian Attack Graph (BAG) for intrusion detection while countermeasure tree (ACT) was constructed for the same purpose in [11]. There is a major difference between them which is nothing but the fact that both attacks and countermeasures are in the same structure with respect to ACT. Another difference is that BAG makes use of genetic algorithms while the ACT does not use them. There are many existing intrusion detection systems such as OpenFlow protocol [12], SDN [13], OVS [14] and [15]. Recently Chung et al. proposed NICE in virtualized network environment which is meant for intrusion detection and provides scalable solution with high success rate. In this paper we implement new IDS which are influenced by NICE.

III. Proposed Simulation Model

The proposed host-based IDS in this paper are modeled for high scalability and better performance. This work was influenced by the work in NICE [1]. The concept of detection agent is used that will take care of monitoring each VM for intrusion detection. There is local attack analyzed for every VM that can be used by detection agent. The cloud servers are in place and each cloud server can host multiple virtual machines. There is network controller to control and coordinate the mechanism. When any user interacts with VM, the detection agent starts monitoring that and the status of VM is altered based on the monitoring process. The network controller coordinates the process and when intrusion is confirmed, the VM is marked as “Blocked VM”. As this kind of logic is available for all nodes in the network, this will be easier to have intrusion detection with ease.

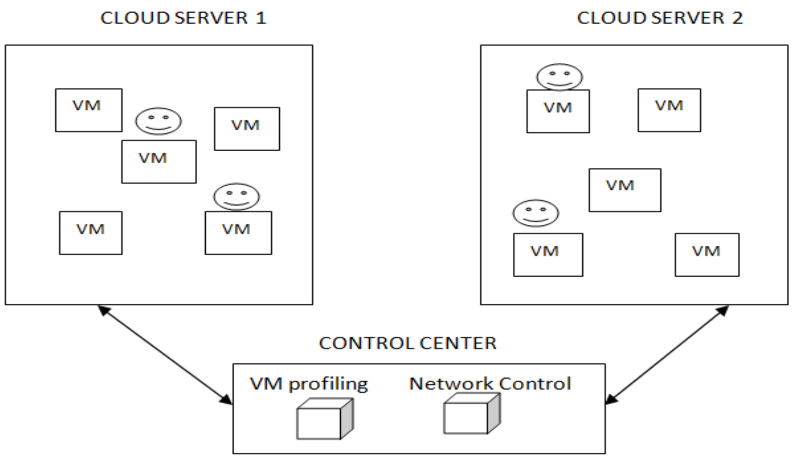


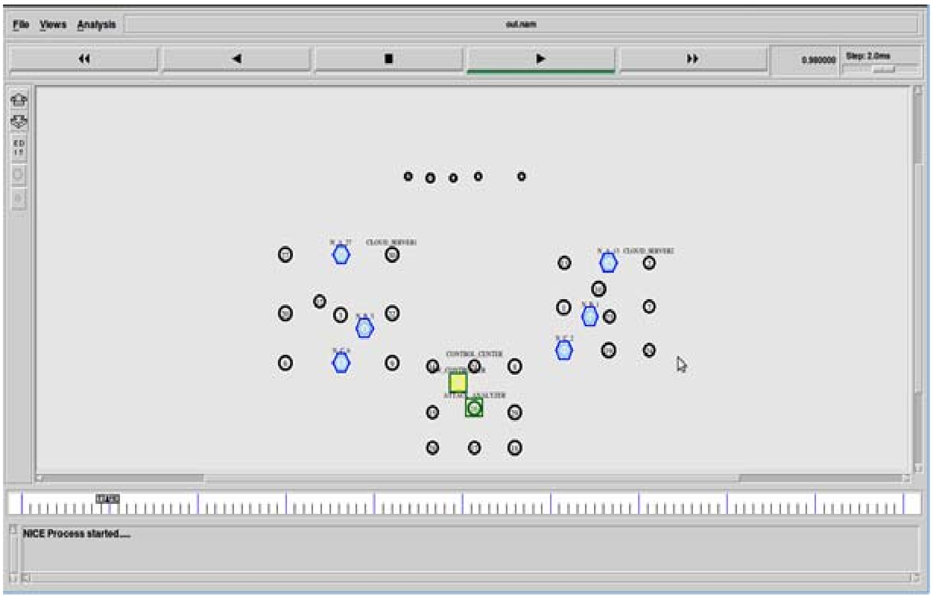
Figure 1 – Broad architectural overview of the proposed system

where ☺ indicates NICE-Agent

As can be seen in Figure 1, it is evident that the architecture of the proposed system has multiple cloud servers and each server has multiple VMs. In each VM, detection capabilities are there with the help of detection agent and also attack analyzer. This way the proposed IDS take care of full security of the communications network in virtualized network environment.

IV. Experimental Results

We built a simulation model as described in the previous section. The environment used is Ubuntu 12.4 Linux based OS. NS2 version 2.3.4 is used along with Tool Command Language. The experiments are done in terms of detection of intrusion, intrusion accuracy and also the performance in terms of delay, success rate, QoS, and CPU utilization.

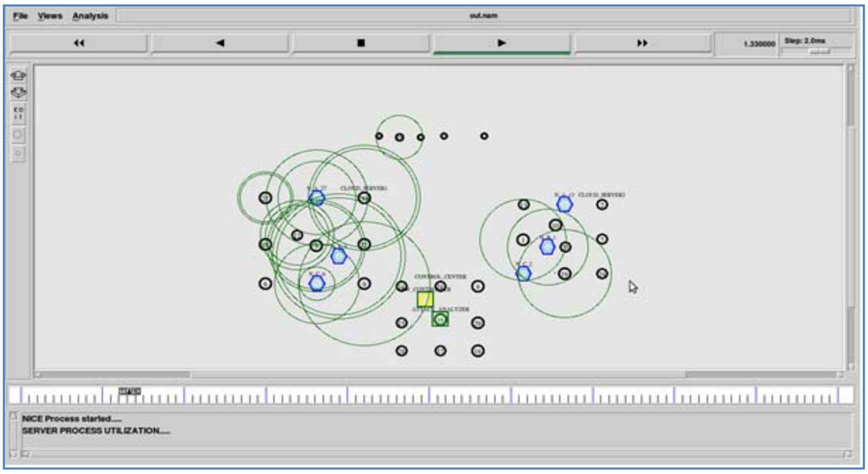


NICE Process started.....

Figure 2 – The network virtualization is in place and the detection process starts

As can be seen in Figure 2, it is evident that there are two cloud servers. Normal nodes are presented in black color that acts as the nodes for background processing. There are some Detection Agents represented in blue color. These are actually VMs containing detection agents. Network controller makes decisions when malicious attacks are observed. For instance when a malicious attack is made, a VM gets compromised. Network controller might mark the VM as suspected or blocked based on the severity of maliciousness.

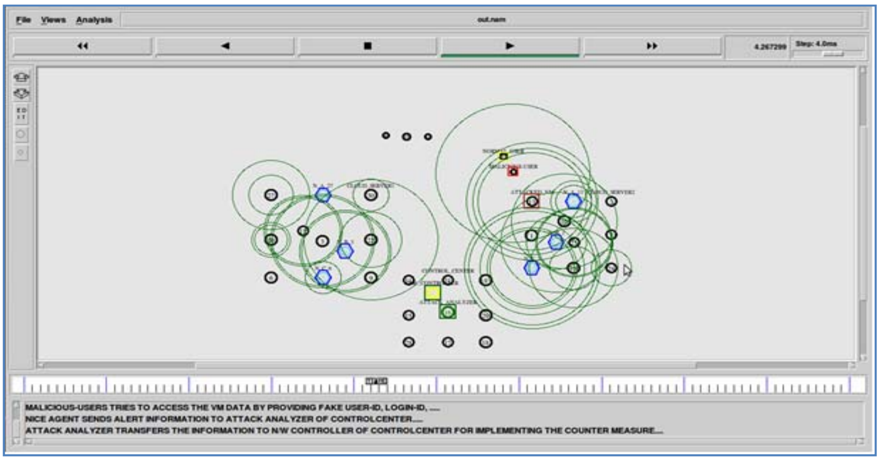
As shown in Figure 3, it is evident that the network is being used and the nodes are performing their respective duties. Most importantly every server starts working with VM deployment and other activities in cloud. Each VM is associated



NICE Process started....
SERVER PROCESS UTILIZATION....

Figure 3 – Host-based intrusion detection is working

with detection Agent who is responsible for detecting malicious behavior. The simulation also reflects the common communication rounds.

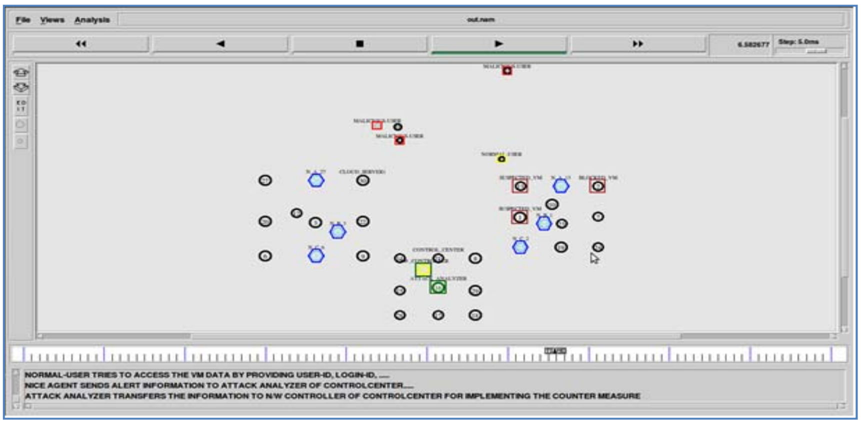


MALICIOUS-USERS TRIES TO ACCESS THE VM DATA BY PROVIDING FAKE USER-ID, LOGIN-ID, ...
NICE AGENT SENDS ALERT INFORMATION TO ATTACK ANALYZER OF CONTROLCENTER....
ATTACK ANALYZER TRANSFERS THE INFORMATION TO N/W CONTROLLER OF CONTROLCENTER FOR IMPLEMENTING THE COUNTER MEASURE....

Figure 4 – Detection of malicious attacks

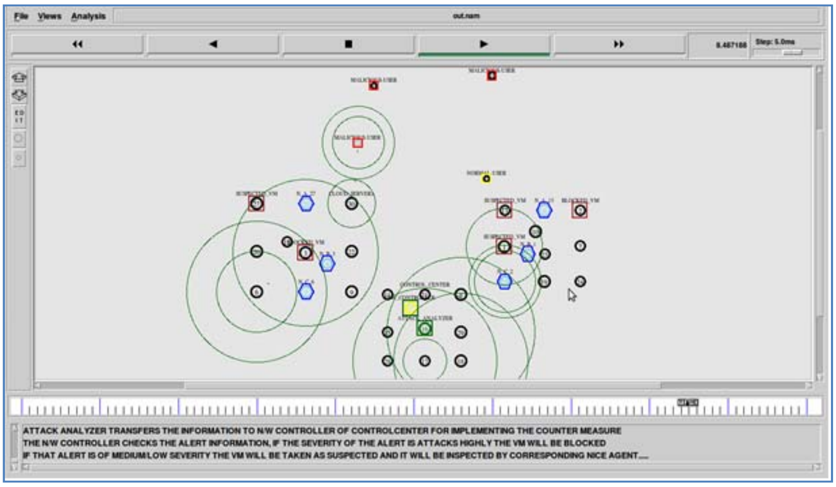
As seen in Figure 4, malicious user is trying to access a VM. Then the DA associated with VM is able to inform about the presence of malicious user. Then the attack analyzer performs intrusion detection process and the result is notified to network controller where appropriate decision is made. When normal user interacts with VM, the normal procedure takes place and network functions as expected. The Network controller starts the process. The VM attacked by malicious user severely is marked

As Blocked VM and if the attack sensitivity is less the VM is marked as a suspected VM. This process has been carried out in server 2.



NORMAL-USER TRIES TO ACCESS THE VM DATA BY PROVIDING USER-ID, LOGIN-ID, ...
NICE AGENT SENDS ALERT INFORMATION TO ATTACK ANALYZER OF CONTROLCENTER....
ATTACK ANALYZER TRANSFERS THE INFORMATION TO N/W CONTROLLER OF CONTROLCENTER FOR IMPLEMENTING THE COUNTER MEASURE

Figure 5 – Network controller marking VMs attacked as blocked/suspected



ATTACK ANALYZER TRANSFERS THE INFORMATION TO N/W CONTROLLER OF CONTROLCENTER FOR IMPLEMENTING THE COUNTER MEASURE
THE N/W CONTROLLER CHECKS THE ALERT INFORMATION, IF THE SEVERITY OF THE ALERT IS ATTACKS HIGHLY THE VM WILL BE BLOCKED
IF THAT ALERT IS OF MEDIUM/LOW SEVERITY THE VM WILL BE TAKEN AS SUSPECTED AND IT WILL BE INSPECTED BY CORRESPONDING NICE AGENT....

Figure 6 – Network controller marking VMs attacked as blocked/suspected

The Network controller starts the process. The VM attacked by malicious user severely is marked as Blocked VM and if the attack severity is less the VM is marked as a suspected VM. This process has been carried out in server 1.

As can be seen in Figure 7, it is evident that the delay statistics are visualized in the graph. The delay analysis of the proposed system is compared with existing system. The proposed host-based intrusion detection reveals improved performance in terms of delay.

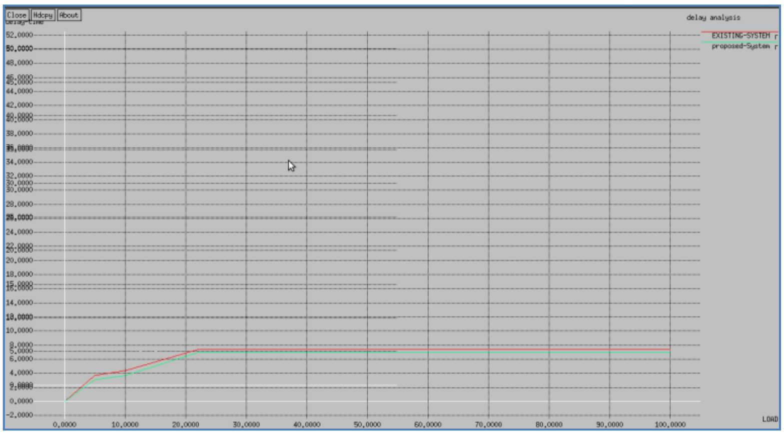


Figure 7 – Comparison of delay analysis

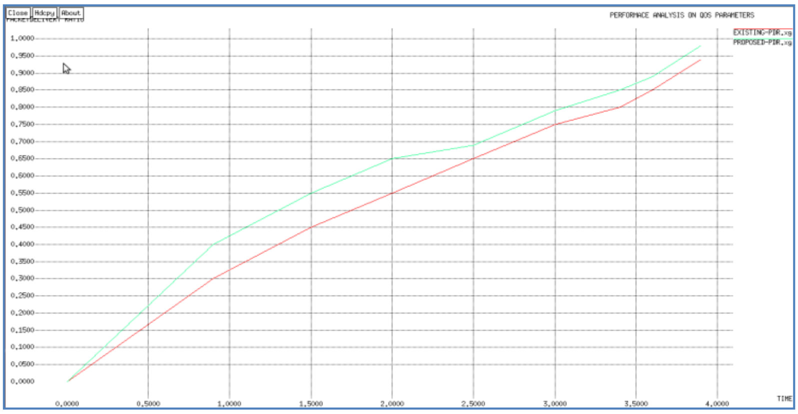


Figure 8 – Comparison of QoS

As can be seen in Figure 8, it is evident that the QoS statistics are visualized in the graph. The QoS of the proposed system is compared with existing system. The proposed host-based intrusion detection reveals improved QoS performance.

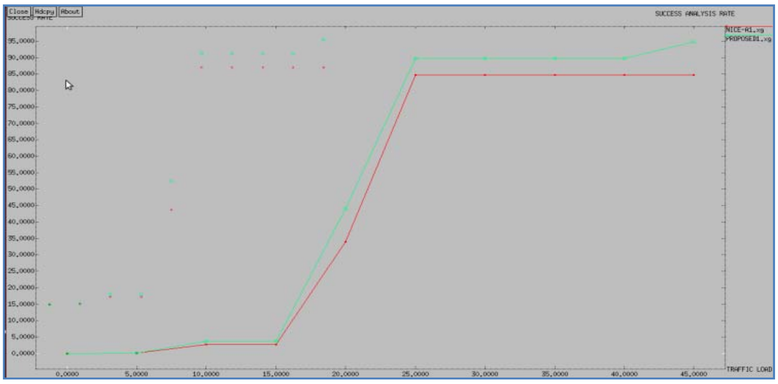


Figure 9 – Comparison of success rate

As can be seen in Figure 9, it is evident that the success rate statistics are visualized in the graph. The success rate of the proposed system is compared with existing system. The proposed host-based intrusion detection reveals improved performance in terms of success rate.

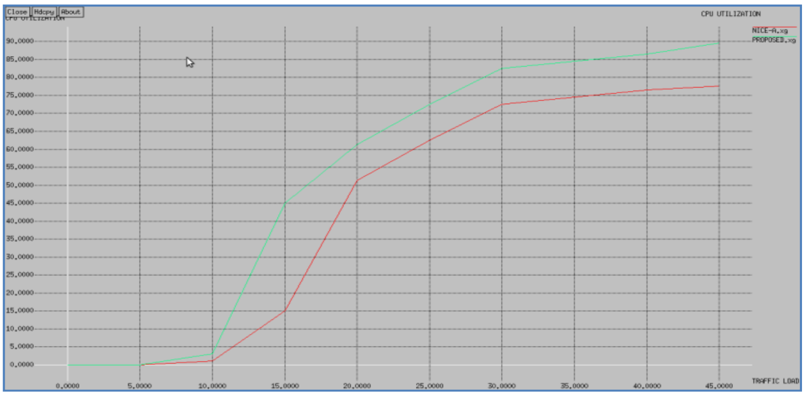


Figure 10 – Comparison of CPU utilization

As can be seen in Figure 10, it is evident that the CPU utilization statistics are visualized in the graph. The CPU utilization of the proposed system is compared with existing system. The proposed host-based intrusion detection reveals improved performance in terms of CPU utilization.

V. Conclusion And Future Work

In this paper, we studied intrusion detection systems in distributed virtual network environments. Virtualization technology is widely used at server side processing as it can leverage and extend the functionality of physical servers. This technology is the basis for the success of cloud computing. In fact it made cloud computing cheaper and affordable to general public. In the context of virtual networking environments, security problems have been reported by researchers and also the Cloud Security Alliance (CSA). The deployed VMs in cloud environment are vulnerable to DDoS attacks. It is done by adversaries by compromising a VM and performing attacks through it. NICE is the recently proposed network-based intrusion detection system meant for virtual network environment which has limited scalability. To overcome this problem, we implemented host-based IDS that improved accuracy and delay performance. Our NS2 simulations reveal that the host-based IDS can scale well besides showing comparable performance in terms of delay, QoS and success rate. However, CPU utilization is more due to the tradeoff between other performance gains and CPU utilization. Improving it is an important direction for future work.

References

1. Coud Sercurity Alliance, “Top threats to cloud computingv1.0,”https://cloudsecurityalliance.org/topthreats/csathreats.v1.0.pdf, March 2010.
2. 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.
3. Z. Duan, P. Chen, F. Sanchez, Y. Dong, M. Stephenson, andJ. Barker, “Detecting spam zombies by monitoring outgoing messages,*IEEE Trans. Dependable and Secure Computing*, vol. 9, no. 2, pp. 198–210, Apr. 2012.

4. 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.

5. X. Ou, S. Govindavajhala, and A. W. Appel, “MulVAL: a logicbased network security analyzer,” *Proc. of 14th USENIX Security Symp.*, pp. 113–128. 2005.

6. 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.

7. L. Wang, A. Liu, and S. Jajodia, “Using attack graphs for correlating, hypothesizing, and predicting intrusion alerts,” *Computer Communications*, vol. 29, no. 15, pp. 2917–2933, Sep. 2006.

8. R. Sadoddin and A. Ghorbani, “Alert correlation survey: framework and techniques,” *Proc. ACM Int’l Conf. on Privacy, Security and Trust: Bridge the Gap Between PST Technologies and Business Services (PST ’06)*, pp. 37:1–37:10. 2006.

9. S. Roschke, F. Cheng, and C. Meinel, “A new alert correlation algorithm based on attack graph,” LNCS, vol. 6694, pp. 58–67. Springer, 2011.

10. N. Poolsappasit, R. Dewri, 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.

11. A. Roy, D. S. Kim, and K. Trivedi, “Scalable optimal countermeasure selection using implicit enumeration on attack counter measure trees,” *Proc. IEEE Int’l Conf. on Dependable Systems Networks (DSN ’12)*, Jun. 2012

12. “Openflow.”http://www.openflow.org/wp/learnmore /, 2012.

13. Open Networking Fundation, “Software-defined networking: The new norm for networks,” *ONF White Paper*, Apr. 2012.

14. “Open vSwitch project,” http://openvswitch.org, May 2012.

15. N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner, “OpenFlow: enabling innovation in campus networks,” *SIGCOMM Comput. Commun. Rev.*, vol. 38, no. 2, pp. 69–74, Mar. 2008.

AUTHOR(S) PROFILE

A. Anitha,pursuing M.Tech in computer science and engineering from Aurora’s Technological and Research Institute, uppal, Hyderbad, Telangana. Her Areas of interest are Cloud Computing and Networking.

Ms. Naga Aswani Puduru working as a Assistant professor in the Department of Computer Science and Engineering in Aurora’s Technological and Research Institute with a teaching experience of 1year and worked as a Software Test Engineer .She had received her M.Tech in Computer Science (Software Engineering) from JNTU Hyderabad. Her areas of interest include Cloud Computing, Network Security and her specialization in Cloud Computing.