ISSN 2348 – 8034
Impact Factor- 3.155

GLOBAL JOURNAL OF ENGINEERING SCIENCE AND RESEARCHES

HOST BASE NETWORK INTRUSION DETECTION SYSTEM IN VIRTUAL MACHINE

Rupali Adhau[*1] and Prof. S.Pratap Singh[2]
[*1,2]Computer Department IOKCOE,Pune,India.

ABSTRACT

Now a Day's cloud computing is more popular because use of cloud increasing daily. Most of the inventor is research on cloud computing. Cloud computing is very important in data sharing application and this environment resources as an OS virtual machines. The virtual machines resides on cloud are more vulnerable to denial of services attack. The virtual machines is connect to more achiness then it more dangerous as well as very harm to all cloud network. To detect the denial of service attack is more challenging task in cloud infrastructure. At that time cloud user can install vulnerable software on the virtual machines. In our proposed system we have introduced flow network program NICE. It is detect and mitigate the attacks on virtual machines. Using the attack graph based model to build the NICE network program. We have proposed approach is to mitigate the attacks in cloud environment by selecting different countermeasure which is depending upon the percentage of vulnerability of virtual machines. Now a day's in many organization used the IDS for detect the attack in the network. Our proposed system is concrete on distributed denial of service attack in the cloud.

Keywords: *cloud computing, scenario attack graph, correlation, network analyzer, intrusion, zombies.*

I. INTRODUCTION

Now days to develop many applications is most widely using the internet. Cloud computing is at the top of security thread. The many research have shown that cloud computing is vulnerable to attacks. In cloud the number of resources are available that used the millions of user every day. The cloud is provide many services such as infrastructure, platform. We can see that number of user are moving on cloud but in this face the main issue is the security problem. The attacker to deploy the vulnerable attacks on shared virtual machines using cloud resources. The server admin is control the data management and data stored in central server. This cloud resources are shared by millions of the user and also user can install any software to shared virtual machines and this leads to the violation of the cloud security. In cloud the main challenging task is to identify attack and find solution on attack. For deploy the attack the many attacker can use the large shared infrastructure. The cloud user are sharing many resources such as Hub, Switches, operating systems, virtual machines. Using this all resources attackers is compromise their virtual machines. The main concern in the cloud computing is the business continuity and service availability from service outage. The cloud computing systems and its architecture is explain by author [1]. Transfer the cloud resources to economical mode but that economical mode is nothing but the denial of services. This system identifies that the request is generated by normal user or it get generated by bot itself.

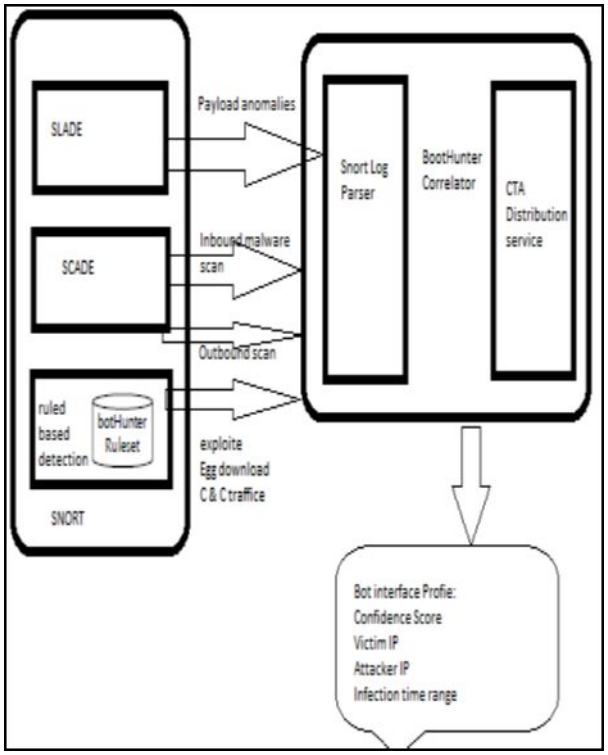
An attack graph which correlation of intrusion detection system it is include in NICE. This system to incorporate the intrusion detection process. NICE include two main phases one is install light weight mirroring which based on intrusion detection agent. At the virtual machines to scans the traffic and provide the information to the attack analyzer. According to the severity of the attack NICE decide whether to put the virtual machine in inspection or not. NICE improve the current intrusion detection or prevention method by introducing programmable or reconfigurable intrusion detection system by using software switching system [5]. In the scenario attack graph (SAG) is stored all information of VM. Using that stored information in SAG the NICE was decide the appropriate action the VM. When NICE is in suspicious mode then he not needed to block the traffic of the virtual machine. In the suspicious stage NICE incorporates the software switching technique for the virtual machine

The organization of this paper is as follows. Section II describe the existing work done for the network intrusion detection in the cloud environment. In Section III we have described our proposed system, the proposed security measurement, mitigation, and countermeasures. Section IV explains NICE in terms of network performance and security. Finally, Section V concludes this paper.

ISSN 2348 – 8034
Impact Factor- 3.155

II. LITERATURE SURVEY

In this section we will see the literature of the highly related to NICE in existing methods. The detection of the spam in the network is explain in [6] author. It is work SPOT based on the sequentially scanning of outgoing messages while deploying statistical method sequential probability ratio test method. Whether host machine is compromised or not determines this method. Botsniffer [7] define the malware in the system according to the several stages which follows the correlation in the alarm triggered by the inbound traffic. In this system to maintain the attack graph which shows that serious of the exploits. The attack graph is construct number of automation tools. To construct the attack graph therefor [9] proposed a modified symbolic model checking system and binary diagram. In this method to generate the graph for all attack but in this main issues is scalability. P. Ammann [11] introduced the monotonicity assumptions, which states that the precondition of a given exploit and it is never invalidated by the successful application of another exploit. The logic programming approach and issues of the data log language to model and to analyze the network attack detection system is introduce the [12]. Attack graph can be generated by accumulating true facts of the network. In the network present number of facts which is polynomial therefore the attack graph generation process is terminate. We have introduces attack correlation graph to improve monitor and attack graph system. In the network intrusion detection system is mostly used the IDS system and firewall but the main problem with this is raw generation of false alarm. The main task in the attack correlation system main is to detect the raw alert. Recently many different author have proposed the many attack correlation graphs. In [14] author proposed in memory graph called queue graph (QH) to detect the attack on each matching of the exploit. But it is difficult to detect the correlated alert in the attack graph for analysis of similar attack. For the attack graph mapping [15] proposed is modified attack correlation method. He have proposed a new function for mapping multiple function to the map. He have proposed depended attack graph to group the related alert with multiple correlation criteria. Each edge of the DG represent the subset of the alert which might be part of attack scenario. After considering attack graph to apply the countermeasure. Several method to select the optimal correlation such as attack path and attack correlation. An attack countermeasure tree (ACT) is proposed the paper [16]. Both attack countermeasure and countermeasure are same. To minimize the countermeasure they depicts some branch and bound technique. Each optimized problem can be solved by using minimized probability in the tree. [17] Proposed (BAG) baysian attack graph which address dynamic security risk management. And applied genetic algorithm to mitigate the countermeasure selection. Another method to detect the network attack is bothunter. The following figure shows the working of bothunter



ISSN 2348 – 8034
Impact Factor- 3.155

Figure 1. Working of Bothunter

Our proposed solution is based on the network control approach SDN where networking function can be programmed through software switching methods. We have taken advantages of open flow network and software switching method to select and minimize the zombie attacks on the virtual machines.

III. PROPOSED SYSTEM

In proposed system we have utilize the scenario attack graph to model the thread and vulnerability detection in the virtual network. To minimize the vulnerability in the virtual machines using NICE which is based on the reconfigurable network

A. Threat Model

In our proposed method we have consider attacker may be located inside or outside of the network. The main aim of proposed NICE is find out vulnerable virtual machine and compromise that machine as a zombie. We have introduces new software model which can resilient the zombie attack. We are to deploy the nice agent using cloud infrastructure. The proposed system predict the attacks on the virtual machines and mitigate the attack independently on the operating system. We have assumption that user can install any of the operating system he wants.

B. Attack Graph model

Each node explains precondition or consequence in attack graph. Attack graph is a tool to detect the all possible multithread multi-host attacks. As the attack graph provide all detailed information about the exploited vulnerabilities. In this we can get whole picture of the security threads of the system. For the countermeasure selection helps the attack graph which is take the appropriate decision according to the current network security and can be mitigate the attack. According to the attack graph we can take appropriate decision about the vulnerable virtual machine.

Definition 1. Scenario Attack Graph:
Scenario attack graph is a tuple
S= (V, E)
Where,
V= union of set of vertices Nc, Nd, Nr.,
Nc is exploit node,
Nd is result of the exploit,
Nr is initial step of the attack.
E=E is union of Epre and Epost this are the directed edges.
Algorithm 1 (Alert Correlation) :
Require: alert ac, SAG, ACG
if (ac is a new alert) then
create node ac

n1 ← vc∈map(ac)

for all n2 ∈ parent(n1) do

create edge (n2.alert, ac)
for all si having a do
if a is last element In si then
append ac to the si

ISSN 2348 – 8034
Impact Factor- 3.155

else
create path Si+1 = {subset(Si, a), ac}
end if
end for
add ac to n1.alert

end for
end if
return S

C. VM profiling

NICE is consist of all detailed information about all virtual machines, incoming traffic toward the virtual machine in VM profiling model. In the virtual machine we can make the analysis of the vulnerability. According to the vulnerabilities in the virtual machines we have three states in the virtual machines.

1. Stable: The VM will be in stable state if and only if there is not present any vulnerability on the virtual machine.
2. Vulnerable: It is the state of vulnerable machine which may have one or more vulnerability on it but not get exploited.
3. Exploited: It is the state of virtual machine which at least one vulnerabilities is get exploited and machine is get compromised.
4. Zombie: VM totally under control of Zombie.

IV. NICE SYSTEM DESIGN

NICE system is consist of VM profiling, network analyzer, network controller, NICE agent Reconfigurable network. This figure 2 shows the nice system in cloud cluster. We have installed NICE at the host machine in proposed system. The figure 2 shows the system architecture of the proposed NICE model.

In this paper we have proposed a novel network detection and countermeasure selection procedure. Figure 1. Shows the NICE architecture

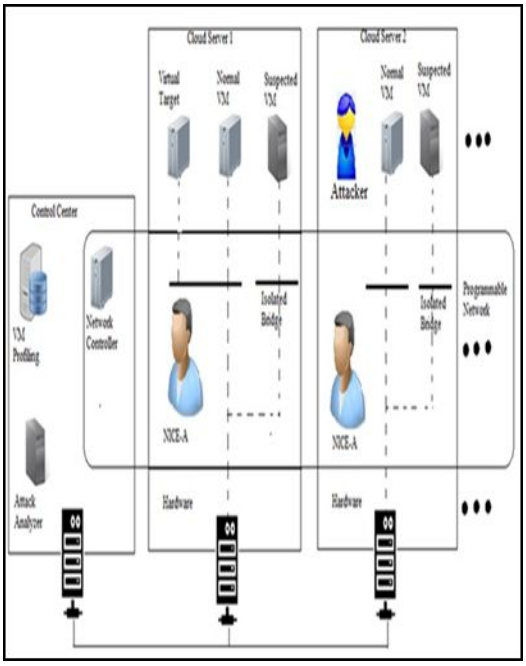


Figure. 2 NICE System architecture.

A. Network Analyzer

ISSN 2348 – 8034
Impact Factor- 3.155

The following figure shows the working of NA. This is main function is to collect the all information from the NICE agent, and maintain virtual machine profiling, the scenario attack graph and attack correlation graph. According to the vulnerability of the virtual machine network analyzer decide or select the countermeasure and forward this message to the network controller. The network analyzer decide sending alert is new or old, if it is new alert then it make new entries in virtual machines and if alert is old then it update the attack correlation graph and scenario attack graph using network analyzer. The network analyzer collect the information from different parts of the system and maintain two graph and it will operate the network. The following figure shows the working of network analyzer which present in network side. To detect the denial of service attack on the virtual machine by using attack correlation. After selecting appropriate countermeasure it will forward the message to the network controller.

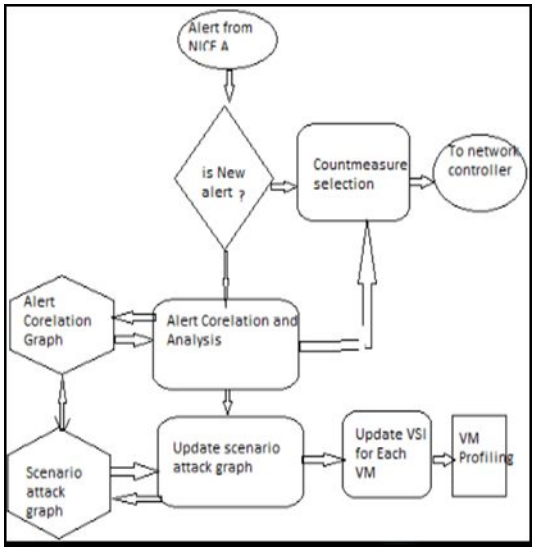


Figure 3. Working of Network Analyzer

B. Network Controller

The network controller is main part of our proposed system. It is acts as assistant for network analyzer. The network controller perform the countermeasure selected through network analyzer. The reconfigure the network and manage the processes running on the virtual machine this is main function of network controller.

C. Advantages of NICE

1) NICE is very effective at detecting attacks without generating an overwhelming number of false alarms.
2) It can detect unusual behavior and thus have the ability to detect symptoms of attacks without specific knowledge of details.
3) Can produce information that can in turn be used to define signatures for misuse detectors.
4) Can detect attacks that cannot be seen by existing IDS
5) Can operate in an environment in which network traffic is encrypted
6) Unaffected by switched networks
7) Can help detect Trojan horse or other attacks that involve software integrity breaches.

V. COMPARISON BETWEEN EXISTING AND PROPOSED SYSTEM

Existing System	Proposed System
Less effective over vulnerability	More effective over vulnerability

ISSN 2348 – 8034
Impact Factor- 3.155

It may have difficult processing all packets in a large or busy traffic network	Can operate in an environment in which network traffic is encrypted
Modern switch-based networks make NIDS more difficult	Unaffected by switched networks
It cannot analyse encrypted information.	It can analyse encrypted information.

VI. RESULTS AND DISCUSSION

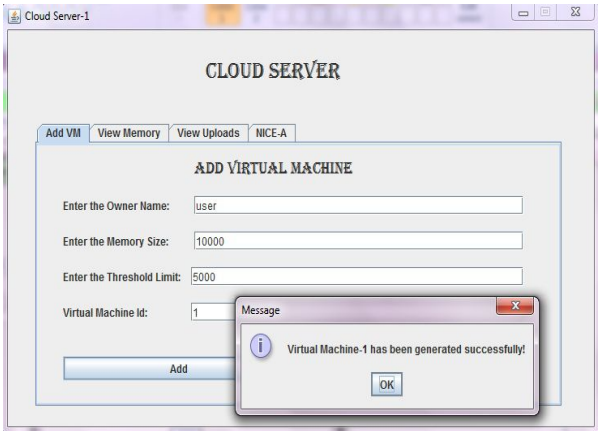


Figure 4 Add VM

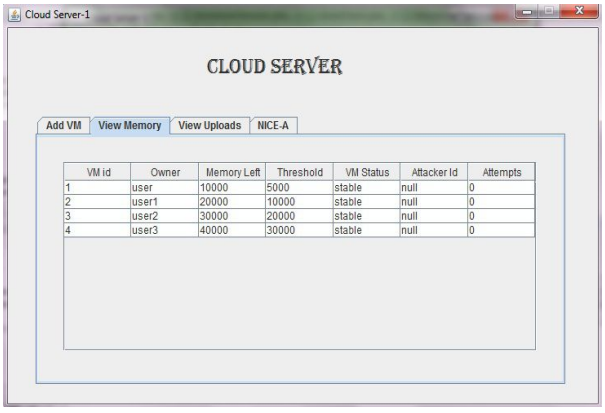


Figure 5 View Memory

ISSN 2348 – 8034
Impact Factor- 3.155

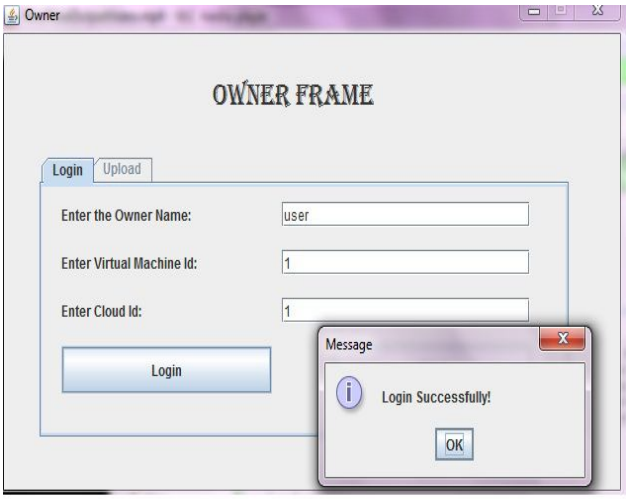


Figure 6 Owner Login

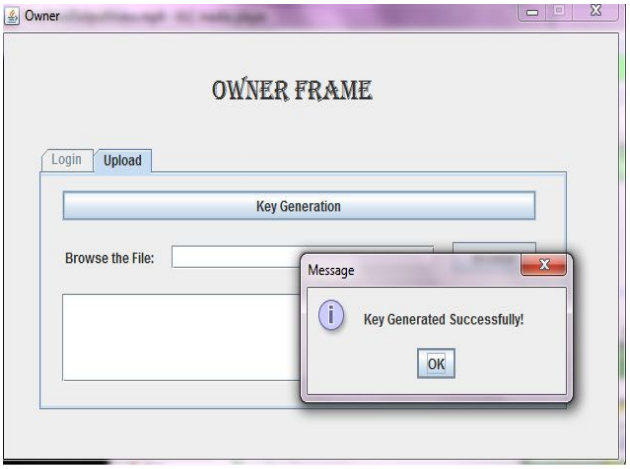


Figure 7 Key Generation

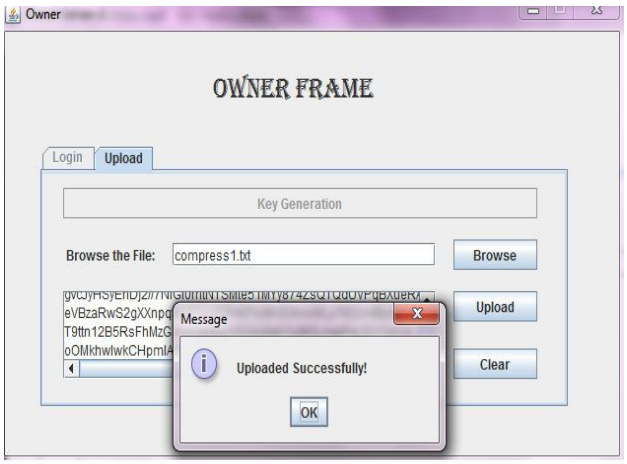


Figure 8 File Upload

ISSN 2348 – 8034
Impact Factor- 3.155

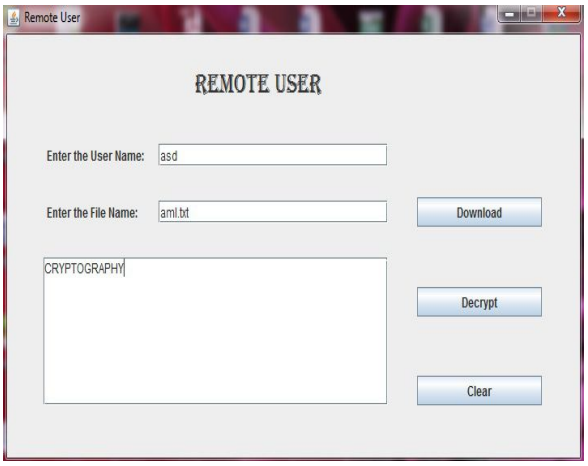


Figure 8 File Download

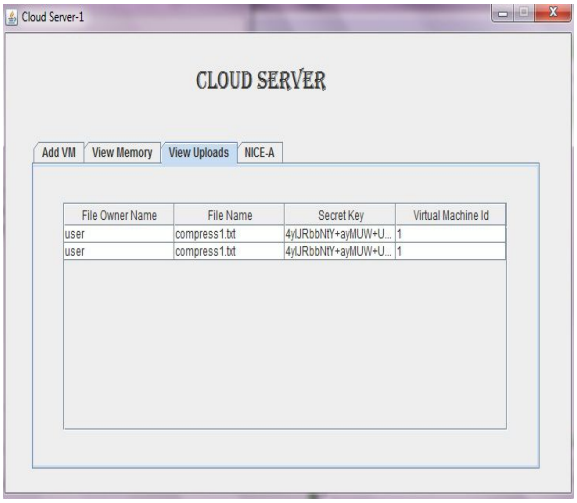


Figure 9 View Upload

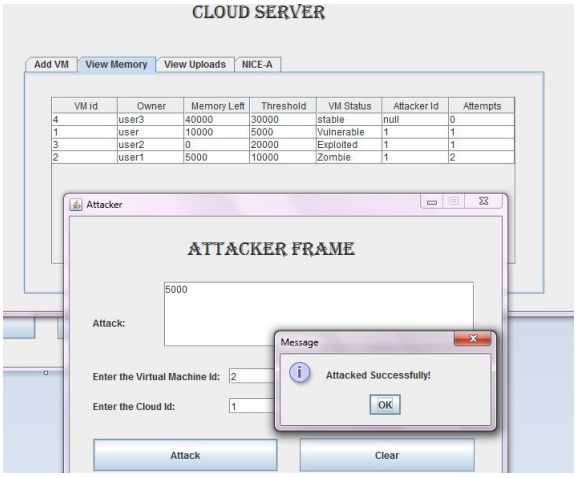


Figure 10 Attack

ISSN 2348 – 8034
Impact Factor- 3.155

Figure 11 VM Profiling

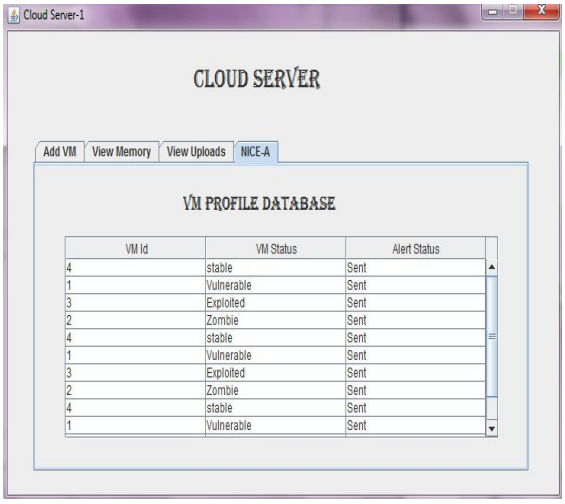


Figure 7.9 Nice-A

VII. FUTURE ENHANCMENT

NICE only investigates the network IDS approach to counter zombie explorative attacks. In order 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 project, I will investigate the scalability of the proposed NICE solution by investigating the de-centralized network control and attack analysis model based on current study.

VIII. CONCLUSION

In the proposed system we have proposed a novel method to detect and mitigate the attacks in the cloud virtual environments. In NICE to find and mitigate the attacks on the virtual machine using the attack graph model. The NICE also introduces a programmable network model which helps to mitigate the attacks on the virtual machines. We have used host based approach to mitigate and to provide security to the whole cloud system. The proposed system can mitigate the attacks on the virtual machines. NICE also investigate the counter zombie attack in the network ids.

IX. ACKNOWLEDGEMENTS

ISSN 2348 – 8034
Impact Factor- 3.155

Author would like to take this opportunity to express our profound gratitude and deep regard to my (Project Guide name), for his exemplary guidance, valuable feedback and constant encouragement throughout the duration of the project. His valuable suggestions were of immense help throughout my project work. His perceptive criticism kept me working to make this project in a much better way. Working under him was an extremely knowledgeable experience for me.

REFERENCES

1. *S CoudSercurity Alliance, "Top threats to cloud computing v1.0," https://cloudsecurityalliance.org/topthreats/csathreats.v1.0.pdf, March 2010.*
2. *M. 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 Commun., vol. 53, no. 4, pp. 50–58, Apr. 2010.*
3. *B. Joshi, A. Vijayan, and B. Joshi, "Securing cloud computing environment against DDoS attacks," IEEE Int'l Conf. Computer Communication 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 & Privacy, vol. 8, no. 6, pp. 24–31, Dec. 2010.*
5. *"Open vSwitch project," http://openvswitch.org, May 2012.*
6. *Z. Duan, P. Chen, F. Sanchez, Y. Dong, M. Stephenson, and*
7. *J. Barker, "Detecting spam zombies by monitoring outgoing messages," IEEE Trans. Dependable and Secure Computing, vol. 9, no. 2, pp. 198–210, Apr. 2012..*
8. *J. Li, X. Chen, M. Li, J. Li, P. Lee, andW. Lou. Secure deduplication with efficient and reliable convergent key management. In IEEE Transactions on Parallel and Distributed Systems, 2013.*
9. *C.-K Huang, L.-F Chien, and Y.-J Oyang, "Relevant Term Suggestion in Interactive Web Search Based on Contextual Information in Query Session Logs," J. Am. Soc. for Information science and Technology, vol. 54, no. 7, pp. 638-649, 2003.*
10. *S. Bugiel, S. Nurnberger, A. Sadeghi, and T. Schneider. Twin clouds: An architecture for secure cloud computing. In Workshop on Cryptography and Security in Clouds (WCSC 2011), 2011.*
11. *W. K. Ng, Y. Wen, and H. Zhu. Private data deduplication protocols in cloud storage. In S. Ossowski and P. Lecca, editors, Proceedings of the 27th Annual ACM Symposium on Applied Computing, pages 441–446. ACM, 2012.*
12. *R. D. Pietro and A. Sorniotti. Boosting efficiency and security in proof of ownership for deduplication. In H. Y. Youm and Y. Won, editors, ACM Symposium on Information, Computer and communications Security, pages 81–82. ACM.*
13. *S. Quinlan and S. Dorward. Venti: a new approach to archival storage. In Proc. USENIX FAST, Jan 2002.*
14. *A. Rahumed, H. C. H. Chen, Y. Tang, P. P. C. Lee, and J. C. S. Lui. A secure cloud backup system with assured deletion and version control. In 3rd International Workshop on Security in Cloud Computing, 2011.*
15. *R. S. Sandhu, E. J. Coyne, H. L. Feinstein, and C. E. Youman. Role-based access control models. IEEE Computer, 29:38–47, Feb 1996.*
16. *J. Stanek, A. Sorniotti, E. Androulaki, and L. Kencl. A secure data deduplication scheme for cloud storage. In Technical Report, 2013.*
17. *C. Ng and P. Lee. Revdedup: A reverse deduplication storage system optimized for reads to latest backups. In Proc. of APSYS, Apr 2013. .*