IOSR Journal of Engineering (IOSRJEN) **www.iosrjen.org**
ISSN (e): 2250-3021, ISSN (p): 2278-8719
PP 47-52

Detection of Attacks using BDD approach in Virtual Environment of cloud by Honeypot

Poonam Pandire[1], Prof.Vishwajit Gaikwad[2]
[1](Department of Computer Engineering, Mumbai University)
[2](Department of Computer Engineering, Mumbai University)

Abstract: *As time progressed, the use of cloud to store data or perform different actions on data increased. Cloud services started emerging and it became more convenient for users to use cloud for storage of their data. But threat on cloud also increased considerably. After discovery of vulnerabilities in software, attacker exploited the data in the cloud. Attacks like DDOS attacks exploited the system and resulted into unavailability of services. Problems like multistep exploitation, low frequency vulnerability scanning have increased tremendously. Nice Agent which does detection or monitors the system and checks every packet. It deeply inspects the suspicious packets and detects an alertform.BDD approach is used for packet filtering. Parallel programming is used for optimisation. To prevent such attacks honeypot is used. Honeypot is a bogus IP which consist some unwanted data. When attack takes place the attacker is redirected to honeypot. It helps to prevent the system from further exploitataion.Further using Binary search diagram (BDD) approach helps in packet filtering and packet capturing. When attack takes place, the attacker is redirected to honeypot. But every time redirecting the attacker to honeypot results in time, resources, bandwidth wastage. When repeatedly such attack takes place from same IP then countermeasure such as port block etc. can be selected accordingly. Thus time is also saved and resources, bandwidth etc. is saved. This approach takes less time in preventing the system from further exploitation. Using parallel programming helps in reduction of time and helps in achieving accuracy.So no false alarm is generated.*

Keywords: *Attack Detection, BDD, Cloud Computing, Networking, Security, Parallel Programming*

I. Introduction

Facilities like allocation of space, web hosting, etc. experience hardware failure that consequence in the loss of data that are given by the Cloud. To overcome this disadvantage, users have begun using cloud computing on a large scale. A survey on Cloud Security Alliance(CSA) proves that cloud computing is considered as the most insecure from several security problems where attackers find vulnerabilities present in the cloud and exploit the resources . In earlier days systems were controlled by system administrators and so detection of vulnerabilities was possible and was prevented in a centralized manner but in cloud data centers patching known as security holes exist.. Moreover, users of cloud can install vulnerable software using cloud, creating chances of threat in cloud security. The biggest challenge is to design and develop IDS which will help to identify the attacks and minimize the impact of the attacks. The service of cloud of providing infrastructure as a service is shared across approximately a million tons of users which results in attracting attackers as they exploit the cloud's vulnerabilities and use their resources to depict various such attacks. Many more such attacks, damage the data as storage on cloud is open source and can be easily accessed by many people. Honeypot is used to prevent such situation in collaboration with the NICE (Network Intrusion Detection and Countermeasures Selection) model to achieve defense with in depth intrusion detection. The NICE model makes analysis of attack graph into the detection processes for better and accurate results. To improvise any of the existing intrusion detection algorithms, the design of NICE is not developed with this intention but, for detection and to take countermeasures to compromise VMs, NICE gives a reconfigurable networking of virtual machines approach. In two main parts NICE is divided [12]. To analyses and capture cloud traffic on cloud server, a mirroring-based light weighted agent (of NICE-A) is deployed within a cloud server. NICE-A continuously scans and monitors the vulnerabilities or anomalies in the virtual system to develop a specific Scenario Attack Graph (SAGs), NICE will come to a conclusion whether or not to put a VM in networking inspection state at the end depending on the critical condition of identified vulnerability. [13] Deep packet inspection is applied once a VM goes in the danger zone that is in suspicious mode. To make the attack behavior prominent, virtual network reconfigurations can be implemented to the inspecting VM. NICE model helps to improve the network intrusion detection significantly with the help of a programmable virtual networking approach. This intrusion detection not only detects the alert but is also reconfigurable The NICE model takes help of switching techniques and develops a mirroring based traffic capturing framework in order to minimize the disturbance. According to their vulnerability state, the existing system of the Detection model NICE helps the cloud to publish inspection models for suspicious VM's.According to SAG, this model suggest for countermeasures by taking consideration of attack type. The attack information gets stored in SAG (Scenario Attack Graph). This model focusses on important factors like detection of attacks in virtual system .It takes care of the user by not interrupting applications of user and not interrupting services provided by the Software Switching is used to improvise the detection of attack. It enables to further investigate the attacks detected. Network approaches helps in detection of attacks and probability of resistance. NICE updates and create entry of every attack in Alert correlation graph.ACG creates graph of every attack that takes place. It correlates with the previous attack. Thus in ACG every type of attack with attack and attacker details are stored which helps in further identification of same attacks.

II. Literature Survey

In this section, a study of related work is shown. This will give an idea of the work done and its advantages with disadvantages of the respective method. In the paper[1] SPOT was used to detect spam zombies and were monitored by system which is based on the tool of Sequential Probability radio test. False positive and false negative bounding was implemented by SPOT technique. In the paper [2] BOTHUNTER is an application which is two way between internal assets and external entities and is designed to scan the communication data flows. BotHunter recognizes crawlers or spiders which work for same engine.BotHunter makes use of three focused network packet sensors which is malware, specialized in different phases of malware infection, including exploit usage, inbound scanning ,outbound bot coordination dialogs and outbound attack propagation. Bot's activity cannot be detected by BOTHUTER if it sees the traffic's proxy. Only after seeing sniffer being placed after proxy ,Bot activity can be detected but will generate a report only when the proxy's address is same as the victim IP address. In the paper [3], the technique called as BotSniffer technique is used. BotSniffer is a process which allows the BOTMASTER to allow actions of bots in the botnets. In this process, the BOTMASTER has control over the Botnets action in BotSniffer. To check network traffic,BotSniffer which is a technique used that shows many correlation and algorithms with similar analysis, also comes to know the crowd of the hosts which shows correlation in their day to day activities as Bots of the same botnet. Many rounds of response crowds is required by BotSniffer.The accuracy of the algorithm may lag because of few response behaviour. It is time taking process. In this paper [4] To group the types of attacks and generate the attack graph, the tool is designed. Symbolic checking algorithm automatically is used. The attacks whose impact can cause heavy loss and cost required to remove its impact is analysed. To create graph automatically it fails sometimes for unknown bugs. In the paper [7] the tool, to generate the scenario attack graphs MulVal is designed. It performs multistage vulnerability,multihost, analysis on a network. It cannot detect attacks in some situations which are not stored in the data logs and due to this reason it fails to create the (SAG)scenario attack graphs. Due to creation of graph methods prevention of systems from many attacks is possible. With the help of honeypot, the attacks that take place is prevented. This bogus IP prevents the system from further exploitation of the system. With honeypot, attacker's information can also be traped.Honeypot records the known as well as unknown attacks.

III. Proposed System

As mentioned above, it is explained about the models providing protection against attacks and thus helping in network security. NICE is used in combination with honeypot to avoid exploitation of the system. NICE agent resides on physical cloud server and it contains: a controller which controls network, a Virtual Machine that profiles server, analyser which does work of detection of attack and bogus IP as application of honeypot. The NICE-A agent monitors and reports signature or unknown attack to the Attack Analyser. Attack Analyser creates and maintains attack graph and update the same in SAG and redirect it to Honeypot, thus protecting the system.BDD approach is used to reduce packets and perform packet filtering. Parallel programming is used to avoid more consumption of time to detect attack. These attacks have some impact on the system and so to avoid its maximum impact on the system countermeasures are selected based on impact and severity.VM profiler keeps checking if any port is open, services running etc.,..The Network Controller collects information of network present open flow network.

1.Parallel Programming :

Parallel Programming is used to speed up the process. Instructions are given separately to execute and are given parallel. These instructions do not have context switching and so no confusion occurs and speed increases of execution.

2. Model showing Threat :[12]

This is the first model in proposed system that is explained. By assumption, the attacker be internal or outsider of the virtual network system. Exploitation of the vulnerabilities in a networking system is every attacker's goal and gain Intel of the data existing in the system. The proposed threat model focuses on the detection of attack and the reconfiguration of the solutions which will improve the resistance of the zombie machines. The use of IDS on a host based system is included in this model. In order to avoid addressing, the encrypted traffic for detecting attacks. Infrastructure-as-a-Service (IaaS) is launched in proposed cloud networking system and is with an assumption that the one who is the service provider of the cloud network has begun. Also assumption is done that the users of cloud have installed any Operating System .Protection of a cloud network manually is impossible. We assume that we have secured the hypervisor and freed it from any further vulnerability. The chances of any unregistered host attacking the system will be directed to a bogus IP. Due to the presence of Honeypots in the network system it secures the network. These protect the cloud system to an appropriate extent from known as well as unknown threats.

3. Graph Model of Attack [12]

A special tool attack graph is used to explain all the possible multiple stage, multiple host attacks which are important to understand these threats and accordingly decide appropriate measures. In this model, each node displays the existing conditions in this model or their resulting preventing measures. The actions are not necessary to display the attacks as normal interactions can be used as a preventive measure. Probable threats and known vulnerabilities within the system are traced. With the help of MulVAL logic notation, the result of actions is represented that is defined as Scenario Attack Graph (SAG) and is shown by X. Ou et al. If found attack is new type of attack, then attack analyser will refer Alert Correlation graph algorithm and it will update ACG and Scenario Attack graph. If the attack resembles the signature of previous attacks and if the attack is known then attack analyser will execute the Scenario attack graph algorithm

4.BDD Approach and Hashing:

In the proposed system,BDD approach and hashing is used for to reduce repeated packets.BDD approach is used for optimization.

5.Tracking of unauthorized user by honeypot :

Honeypot tracker is nothing but an application with bogus IP and data. Whenever an attacker tries for some illegal access of the system or tries to attack. the tracker traces the activity. This tracker keeps track of the path of the attack. The tracker stores the information about attackers Source IP, Source Port, Destination IP and Destination Port. When the attacker attacks, attacker feels that attacker has gained access to the data but in reality it is redirected to some bogus IP. Furthermore countermeasures are applied depending on the scenario by selecting from countermeasure pool. This increases the security of the cloud. The attacker in this case has no knowledge of his activities getting trapped. When a certain threshold value is crossed referring to a predefined threshold, then countermeasures like blocking the port etc. can be done after redirecting to honeypot. Countermeasure like port blocking etc. increases the security of the cloud as the attacker is unable to proceed further due to block.

6. Virtual Machine Profiler model [12]

The profiler contains the information of any open port on the network and it also contains history of open ports. The attacker with some technique can exploit any open port on a virtual machine by applying various ways like running port scanning program. Therefore it very much essential to keep an eye on open port which can be only done by admin.

IV. Methodology

There are two algorithms that are required to be used here depending on a known or an unknown attack. These includes :

Algorithm 1: BDD approach for packet filtering and scenario updating in SAG (Scenario attack Graph)

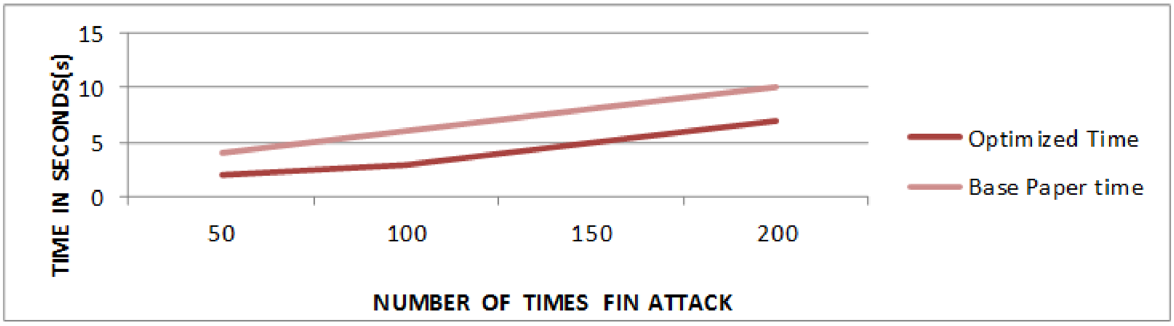
```
Require: AG, SAG, alert generated when attack is detected
1: BDD approach and hashing is used to reduce repeated packets
2 : When new alert is generated then it is recorded in the alert graph
3: create node which is new attack in Attack Graph (AG) and trap information of the attacker
4: Increment SAG (scenario attack graph) according to new attack.
5: Create new path containing information about attacker
```

6: Update depending on new attacks AG(Alert graph) and SAG(Scenario attack graph)
7: finally return new path of attack

Algorithm 2: Parallel Programming and Hashing to reduce optimization
Require:Graph(Counters,Attacktypes), CounterMeasureCM,Alert
1.If attack is detected by NICE agent, then trap details of the attacker.
2. Parallel Programming is used to give instructions to cores and speed up the process of trapping.
3. Attack gets redirected to Honeypot
4: Set probability of alert is equal to 1 when attacker does attack.
5:Risk_Probrability (T) or probability of risk is calculated
6: Initialize benefit received when appropriate countermeasure is applied.
7: Risk Probability is risk in attack performed and detection be calculated as every attack type which has different impact on the vulnerable system
8: Hashing is used for optimization

V. Results

1)FIN ATTACK :
When more than 8 FIN packets are sent from zero length window, IDS detects the attack as FIN attack.



Fig(a) Time required to detect FIN Attack

As shown in Fig (a), the execution time of algorithm increases as FIN packets increases. When maximum no. of FIN packets increases for example 200 then time increases to 7 seconds whereas base paper requires 10 seconds.

2) SYN ATTACK:
When more than 8 SYN packets are sent from zero length window, IDS detects the attack as SYN attack.

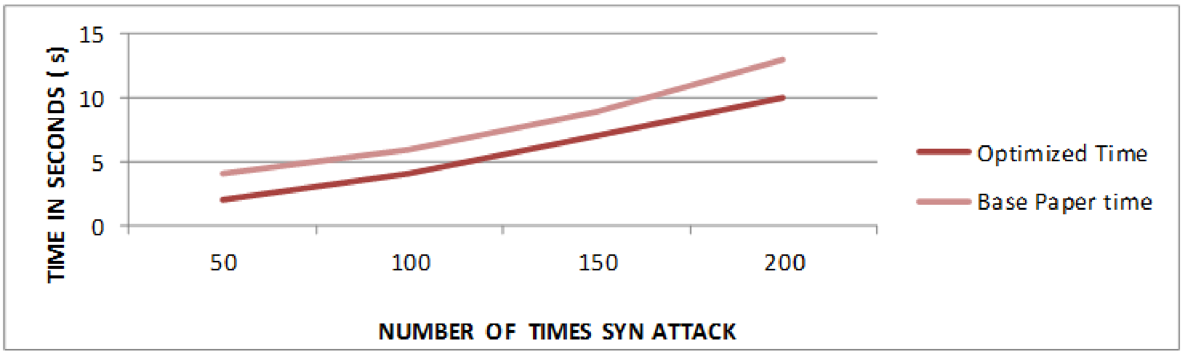
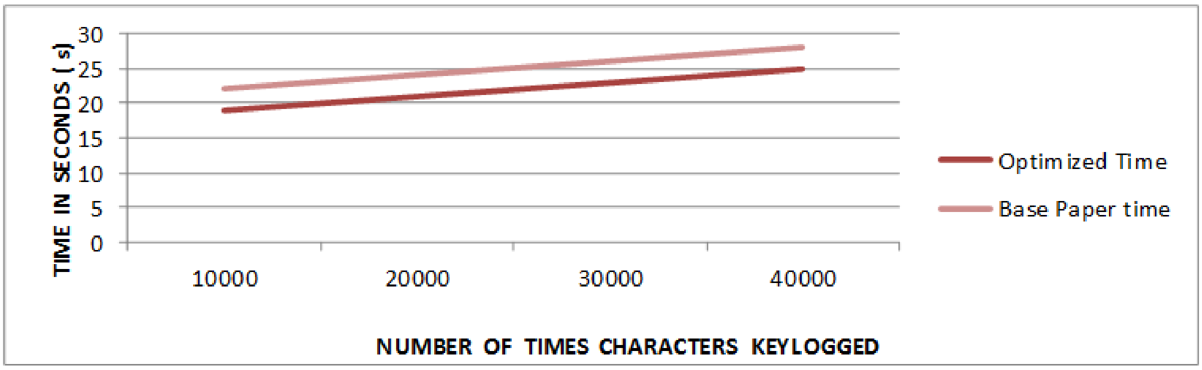


Fig (b) Time required detecting SYN Attack

As shown in Fig (b), the execution time of algorithm increases as FIN packets increases. When maximum no. of FIN packets increases for example 200 then time increases to seconds 10 whereas base paper requires 13 seconds.

3) KEYLOGGER ATTACK:
Key logger is an illegal activity which runs in background. This is the application where the words characters passwords etc similarly any key pressed are recorded.

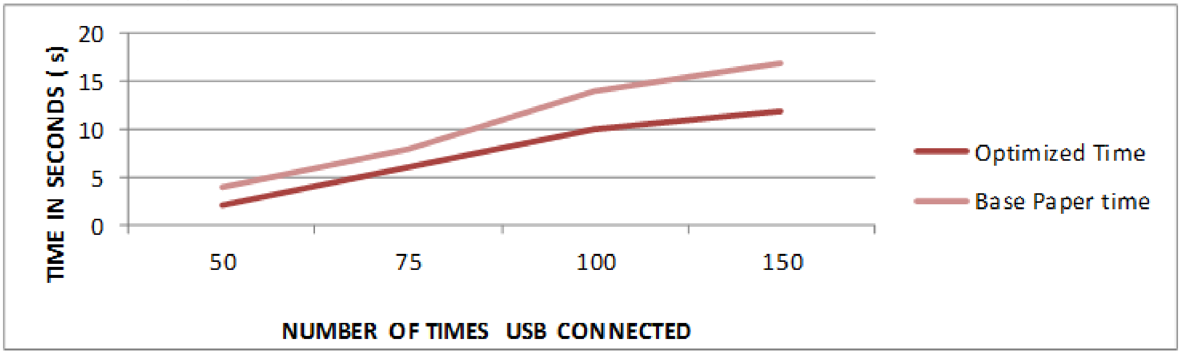


Fig(c) Time required detecting Key logger Attack

As shown in Fig.(c),the execution time of algorithm increases as number of times Keylogged words increases. When maximum no. of Key logged words increases for example 4000 then time required to execute the algorithm increases to 25 seconds whereas base paper requires 28 seconds.

4) USB ATTACK :

USB Attack is illegal act. It is a type of attack where USB is connected to server by unauthorized user



Fig(d) Time required to detect USB Connected

As shown in Fig.6.3, the execution time of algorithm increases as number of times USB Attack increases. When maximum no. of USB attack increases for example 150 then times required to execute the algorithm increases to 12 seconds whereas base paper requires 17 seconds.

5)BUFFER OVERFLOW ATTACK :

Ping packets are sent from one source IP to target IP. When threshold of 8 is crossed, it is considered as overflow attack. IDS detects it as Buffer Overflow attack and creates alertform that buffer overflow has occurred.

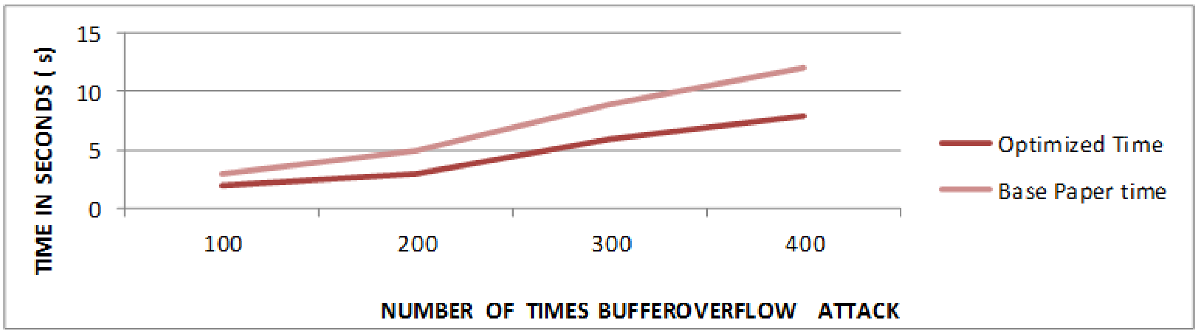


Fig (e) Time required to detect Buffer Overflow Attack

As shown in Fig.6.4,the execution time of algorithm increases as number of times repeated packets increases. When maximum no. of repeated packets increases for example 400 then time required to execute algorithm increases to 8 seconds whereas base paper requires 12 seconds.

VI. Conclusion

Using cloud, it becomes easier to send knowledge to a different person. Storing personal knowledge includes security issues. However issues like different types of attacks like DDOS attacks and anomaly primarily based attacks hampers the confidential knowledge and exploits the system by disabling the system services. Existing system contains NICE model that develops a mirroring based mostly, traffic capturing framework so as to attenuate the disturbance by taking facilitate of switch techniques. Design of the detection model NICE helps the cloud to publish examination models for suspicious VM's in keeping with their vulnerability state. This model determines acceptable actions betting on the behaviour of the VM within the SAG. The good model contributions square measure to develop a pleasant model based mostly multi-phase distributed, network intrusion and detection of attacks and framework of hindrance in an exceedingly virtual atmosphere that may examine malicious traffic while no interruption of the users applications and cloud services. NICE includes a package switch answer to discover and examine suspicious VMs for additional Investigation. Using BDD approach can filter the packets. Parallel programming is used to give instructions to the cores. This speeds up the process and execution is done with more speed. In parallel programming there is no context switching so lot of confusion of switching instructions is avoided and more accuracy is achieved. Hence there is no false alarm generation. In future host based IDS can be implemented so that whole network can be covered.

References

Journal Papers:

[1]. 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.

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

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

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

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

[6]. Ashara Banu MohamedNorbik Bashah IdrisBharanidharan Shanmugum," A Brief Introduction to Intrusion Detection System,(CCIS, volume 330)

[7]. X. Ou, S. Govindavajhala, and A. W. Appel, "MulVAL: a logic-based network security analyzer," Proc. of 14th USENIX SecuritySymp., pp. 113–128. 2005.

[8]. Cloud Sercurity Alliance,"Top threats to computingv1.0," https://cloudsecurityalliance.org/topthreats/csathreats.v1.0.pdf, March 2010.

[9]. M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. Katz, A. Kon-winski, 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.

[10]. Dr. S.Vijayarani and Ms. Maria Sylviaa.S, "INTRUSION DETECTION SYSTEM – A STUDY,International Journal of Security, Privacy and Trust Management (IJSPTM) Vol 4, No 1, February 2015

[11]. S Deepa Lakshmi , G Arunkumar , V Madhu Viswanatham,"Network Security Enhancement through Honeypot based Systems" Vol 7 No 1 Feb-Mar 2015

[12]. NICE: Network Intrusion Detection and Countermeasure Selection in Virtual Network Systems by Chun-Jen Chung,Student,IEEE,Pankaj Khatkar, Student Member,IEEE,TjanyiXing,2012

[13]. Ankit Punia, Vedang Ratan Vatsa"Current Trends and Approaches of Network Intrusion Detection System, IJCSMC, Vol. 6, Issue. 6, June 2017, pg.266 – 270

[14]. lating, hypothesizing, and predicting intrusion alerts," Computer Communications, vol. 29, no. 15, pp. 2917–2933, Sep. 2015.

[15]. 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, 2016.

Books:

[16]. Understanding DDOS attack and its effect in cloud Environment Rashmi .Deshmukh, Kailash.Devadkar

Theses:

[17]. Chun-Jen Chung ,*SDN based proactive defense based mechanism in cloud system,2015*