Analysis on DDoS Attacks and Solutions in Virtualized Cloud Environment

Dr. Amit Kr. Chaturvedi[#1], Punit Kumar[*2], Dr. Kalpana Sharma[#3]

[#]*Assistant Prof., Govt. Engineering College, Ajmer*
Ajmer, Rajasthan, India

** Ph.D. Scholar, Bhagwant University, Ajmer, Rajasthan, India*

Abstract— *Cloud computing offers the services at SaaS, PaaS, and IaaS layers on "pay-per-use" basis. There are two major advantages of shifting on cloud computing, one is access of service online from anywhere and anytime and the second is the scalability i.e. users can demand the VMs, Storage, Servers, etc as much as required. The cloud servers will provide the resources as per the requirements of the users. The attackers are using these two qualities i.e. access of service from anywhere , anytime and scalability for attacking on the data saved on the server or the user's data. In this paper, we have presented a study on the strategies of attack using bot-nets, DDoS attacks, and solutions proposed for the prevention from such attacks. The future of the botclouds is also discussed in this paper.*

Keywords—*Botnet, Botcloud, cloudcomputing, cybercriminal, virtualization, virtual machine, IaaS, PaaS, SaaS, resource pooling.*

I. INTRODUCTION

Cloud computing is defined as a pool of virtualized computer resources. Generally, Cloud providers use virtualization technologies combined with self-service abilities for computing resources via network infrastructures, especially the Internet and multiple virtual machines are hosted on the same physical server. Based on virtualization, the cloud computing paradigm allows workloads to be deployed and scaled-out quickly through the rapid provisioning of Virtual Machines or physical machines. A cloud computing platform supports redundant, self-recovering, highly scalable programming models that allow workloads to recover from many inevitable hardware/software failures.

Virtualization is a technique which allows creating an abstract layer of system resources and hides the complexity of hardware and software working environment. Virtualization commonly implemented with hypervisor technology, which is a software or firmware elements that can virtualizes system resources. Virtualization gives the power to create logical environments as per the requirements and supports to overcome from the complexity of the hardware and network complexity. It also supports the scalability of the resources as per the need, and to create multiple resource pools for different levels of users (end users, developers, providers, etc.)

Virtualization has proved its value and accelerated the realization of cloud computing. Then, virtualization was mainly server virtualization, which in an over-simplified statement means hosting multiple server instances with the same hardware while each instance runs transparently and in isolation, as if each consumes the entire hardware and is the only instance running. A hypervisor or virtual machine monitor (VMM) is computer software, firmware, or hardware that creates and runs virtual machines. A computer on which a hypervisor runs one or more virtual machines is called a host machine, and each virtual machine is called a guest machine. The hypervisor presents the host operating systems with a virtual operating platform and manages the execution of the guest operating systems.

A hypervisor is a function which abstracts or isolates operating systems and applications from the underlying computer hardware. This abstraction allows the underlying host machine hardware to independently operate one or more virtual machines as guests, allowing multiple guest VMs to effectively share the system's physical compute resources, such as processor cycles, memory space, network bandwidth and so on. A hypervisor is sometimes also called a virtual machine monitor. A hypervisor can be of one of the following categories: (1) Native or Bare Metal Hypervisors, (2) Hosted Hypervisors, (3) Virtual Machine Monitors.

Today, VMMs have become popular again for a multitude of reasons. Server consolidation is one such reason. In many settings, people run services on different machines which run different operating systems (or even OS versions), and yet each machine is lightly utilized. In this case, virtualization enables an administrator to consolidate multiple OSes onto fewer hardware platforms, and thus lower costs and ease administration.

Another reason is testing and debugging. While developers write code on one main platform, they often want to debug and test it on the many different platforms that they deploy the software to in the field. Thus, virtualization makes it easy to do so, by enabling a developer to run many operating systems. Platform virtualization software, specifically emulators and hypervisors, are software packages that emulate the whole physical computer machine, often providing multiple virtual machines on one physical platform. Some of the Platform virtualization softwares examples are DOSBox, FreeBSD Jail, Hercules, Hyper-V, INTEGRITY etc.

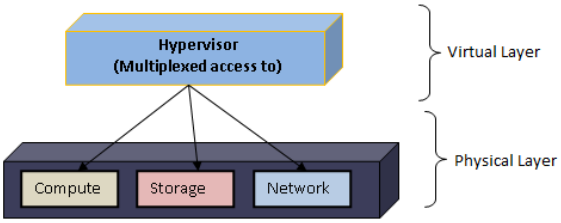

Figure 1: Role of Hypervisor in virtual computing

A hypervisor has an inherent knowledge of application's requirements. It also has a global view of architecture and is also hardware agnostic.

II. BOT, BOTNET, AND BOTCLOUD NETWORKS

A bot is a partially autonomous piece of software that can be controlled remotely. The person controlling a bot is referred to as a botmaster. A group of bots under control of a botmaster is referred to as a botnet. A botnet is created by first infecting a computer without the knowledge or consent of its owner by, for example, sending a virus as an email attachment (Ianelli and Hackworth, 2005). Once a computer is infected with bot software, it contacts the botmaster. At this moment, the bot becomes part of the botnet. The botmaster can then send orders to the bot to carry out (malicious) tasks. Botnets can comprise thousands or even millions of bots. Botnets create a serious threat to the network, business, partners and the potential customers. Botnets rival the power of today's most powerful cloud computing platforms. These dark clouds are controlled by cybercriminals, botnets borrow your network to server malicious business interests [4].

Common attacks launched by botnets include (1) launching Distributed Denial of Service (DDoS) attacks, (2) sending spam email and spreading malware, (3) stealing private information and (4) performing click fraud (Jing et al., 2009; Ianelli and Hackworth, 2005).

DDoS attacks are used to overload a target's servers so that legitimate traffic can no longer access them (Mirkovic and Reiher, 2004). This is achieved by simultaneously flooding a target domain with requests until the response time to load a webpage is longer than a legitimate user is willing to wait. Thus, the system appears to have crashed. In some cases, just the threat of a DDoS attack is often enough for criminals to extort money from businesses. Another common use of botnets is the sending of spam and malware. It is estimated that more than 97% of all email is unsolicited, bulk email (also known as spam) and the majority is generated by botnets (Anselmi et al., 2010). E-mail containing spam messages or malicious attachments can be sent from an infected machine using either a user's personal account or their Internet Service Provider's (ISP) e-mail server. If each single machine sends only 10 such messages per day, the massive size of most botnets can thus send millions of spam messages each day.

Private information is stolen from bot infected computers using keyloggers or making periodic screenshots (Ianelli and Hackworth, 2005). Keyloggers capture passwords and usernames at the moment that these are entered into a protected website, such as a personal email site. In addition, a bot can identify and copy sensitive financial data, such as credit card information.

Click fraud can be used to attack different targets for different reasons. A typical use of click fraud is to intentionally click on advertisements from pay-perclick providers, such as Google or Adsense (Kshetri, 2010). There are two general motivations for this kind of fraud: inflationary and competitive (Wilbur and Zhu, 2009). Inflationary click fraud is when attackers can earn money by clicking on advertisements they themselves are hosting. The pay-per-click providers then charge the targeted company for these clicks and pass this money on to those who actually host the advertisements, in this case the attackers.

Competitive click fraud is when attackers click the advertisements of a company with the goal of driving up that company's advertising costs. In this case, the attackers click on any advertisement of that company, regardless of where they are hosted and regardless of who receives money for the click.

Another use of click fraud is to artificially influence online polls. Online polls allow users to vote on various topics. Examples include news sites that poll how their readers feel about a certain news story or software developers that poll their users for feature requests. Often there is no monetary incentive to influence the results, but rather only a psychic benefit such as enjoyment.

Rather than use a network of infected machines, Botmasters can use Cloud services to build botnets. Botmasters purchase a large group of machines from a CSP and install a bot on each machine to form a botnet. Cloud-based botnets, or botclouds, have several advantages over traditional botnets. A

traditional botnet requires substantial time to build, whereas a botcloud can be online in minutes. In addition, while a botnet is unreliable due to the constant threat of infected computers being switched off by their owners, a botcloud is always online and ready. Finally, a botnet cannot fully utilize the processor or bandwidth resources due to the constant threat of detection or computer use by the owner; however, a botcloud can be fully utilized with no fear of interruption.

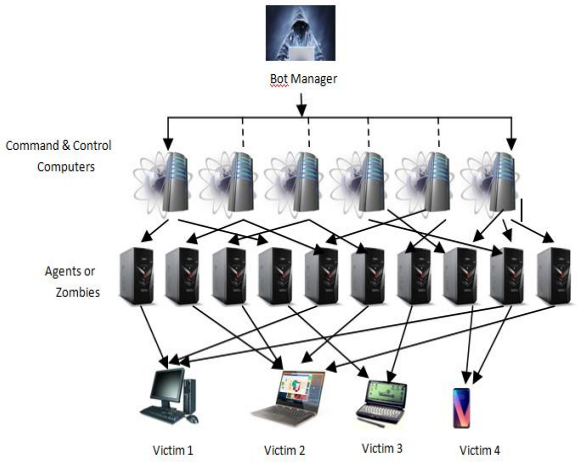

Figure 2 : Agent Based Botnet Attack Scenario

III. DDoS Attack

Nowadays, there are several attacks in the IT world. Basically, as the cloud can give service to legal users it can also service to users that have malicious purposes. A hacker can use a cloud to host a malicious application for achieve his object which may be a DDoS attacks against cloud itself or arranging another user in the cloud. For example an attacker knew that his victim is using cloud vendor with name X, now attacker by using similar cloud provider can sketch an attack against his victim(s). This situation is similar to this scenario that both attacker and victim are in same network but with this difference that they use virtual machines instead of physical network (Figure 2).

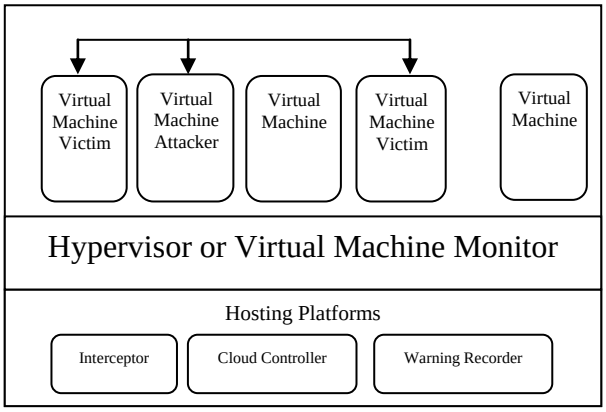

Figure 3 : Attack scenario within cloud

Distributed Denial of Service (DDoS) attacks typically focus high quantity of IP packets at specific network entry elements; usually any form of hardware that operates on a Blacklist pattern is quickly overrun. In cloud computing where infrastructure is shared by large number of VM clients, DDoS attacks have the potential of having much greater impact than against single tenanted architectures. If cloud has not sufficient resource to provide services to its VMs then maybe cause undesirable DDoS attacks. Solution for this event is a traditional solution that is increase number of such critical resources. But serious problem is when a malicious user deliberately done a DDoS attacks using bot-nets.

It may be more accurate to say that DDoS protection is part of the Network Virtualization layer rather than Server Virtualization. For example, cloud systems use virtual machines can be overcome by ARP spoofing at the network layer and it is really about how to layer security across multivendor networks, firewalls and load balances.

DDoS attacks have recently been very successful on cloud computing, where the attackers exploit the "pay-as-you-go" model. There are the three important features of cloud computing : (1) Auto Scaling, (2) Pay-as-you-go accounting, and (3) Multi-tenancy. These important features are the main attraction for the DDoS attackers because the chance of success of attack is also high.

Attackers thoroughly plant bots and trojans on compromised machines over the Internet and target web services with Distributed Denial of Service attacks. DDoS takes the shape of an EDoS attack when the victim service is hosted in the cloud. Organizations exist (also known as "Booters"), which provide a network of bots to their consumers to plan DDoS attacks on their rival websites [18]. Motives of these attacks range from business competition, political rivalry, extortions to cyber wars among countries.

An attacker who plans a DDoS attack would send enough fake requests to achieve "Denial of Service". However, this attack would generate heavy resource utilization on the victim server. "Auto scaling" [16] would take this "overload" situation as feedback and add more CPUs (or other resources) to the active pool of resources of this VM. Once a VM gets deployed, it starts as a "Normal load VM". Now, let us assume that the DDoS attack has started and consequently VM gets overloaded ("Overloaded VM"). The overload condition triggers auto-scaling features of cloud resource allocation, and it will choose one of the many strategies available in the literature for VM resource allocation, VM migration, and VM placement [19]. Overloaded VM may be given some more resources or migrated to a higher resource capacity server or may be supported by another instance started on another server. If there is no mitigation system in place, this process will keep

adding the resources. This situation may last till service provider can pay or cloud service provider consumes all the resources. Finally, it will lead to "Service Denial". In turn, this leads to on-demand resource billing, and thus economic losses over and above the planned budget may occur. One trivial solution is to run VMs on fixed or static resource profile where the SLA does not have any provision for additional resources on demand. In this case, the DDoS will directly result in "Denial of Service" and all the attractive features of the cloud will be lost.

A DDoS attack can be launched by a botcloud and can have the same effect as a DDoS attack launched by a traditional botnet. This attack can be detected and neutralized by a CSP. This will only work if they are monitoring for this type of activity. Therefore, a botcloud perpetrating this attack can be successful in shutting down the target domain, but perhaps only for a short period of time.

Let us discuss some of the DDoS attacks encountere in the last one decade, in Dec 2007 during the riots in Russia, government sites suffered severe DDoS attcks. Access to IP addresses outside Estonia was removed by many of them for serverral days.In Nov 2008, the conficker worm used vulnerabilities found in Microsoft OS. It uses vulnerable machine and other machines are unwillingly connected to it, to make a large botnet. On 4th Jul 2009, 27 websites of White House, Federal Trade commission, Deptt of Transportation, and the Department of the Treasury were attacked. On 1st august, Blogging pages of many social networking sites (Twitter, Facebook etc.) were affected by DDoS attack, aimed at "Cyxymu" Georgian blogger.

In 2010, Operation Payback in which DDoS attacks launched on websites of MasterCard, PayPal and Visa, as they decide to stop giving service to WikiLeaks. In 2011, LulzSec hacktivist group attacked website of CIA (cia.gov). In 2012 Many attacks at us banks involve use of itsoknoproblembro DDoS tool. Many such do-it-yourself toolkits are available. In 2013, 150 Gbps DDoS attacks, hence DDoS are increasing regularly [1].

IV. SOLUTIONS FOR DETECTION OF DDOS ATTACK

Detecting and combating botnets remains a difficult task, but some progress has been made in this direction. Different botnet detection techniques correspond to different network structures and communication protocols. Based on these two attributes, two general classifications are recognized : Internet Relay Chat (IRC) based botnets and Peer to Peer (P2P) based botnets.

IRC is a text-based messaging protocol based on the traditional client/server model. An infected computer acts as an IRC client and contacts the IRC server to join the network. The botmaster then uses the IRC server to send instructions and data to the client for execution. This structure is highly centralized around the IRC server(s). If a server is taken offline, all bots that depend on it will no longer be able to receive or execute further instructions.

P2P based botnets use a more decentralized, distributed model without a centralized server. Each bot has a list of known peers comprising only a small part of the entire botnet. A botmaster sends data and instructions to one or more peers, that in turn pass these on to all other peers they know. These messages then traverse the entire network. As there is no central server, there is also no central point of failure, thus making this class of botnet more robust against attack.

Botnet creators have found that truly decentralized networks are relatively slow to organize and react to new commands, so most P2P based botnets employ a hybrid structure. Hybrid structures use a layer of super-peers that maintain long lists of known peers. When a new peer joins the network, it first contacts a super-peer and downloads this list. This download process and the role of these super-peers makes the hybrid structure more prone to detection by system users and administrators (Schoof and Koning, 2007).

The main methods for traditional botnet detection are honeypots and intrusion detection. Honeypots are unprotected computers that are intentionally (allowed to be) infected by botnets. Honeypots are of two types: (1) Low Interaction Honeypots and (2) High Interaction Honeypots. Honeyd is the example of Low Interaction Honeypot and Honeynets are the example of High Interaction Honeypots.

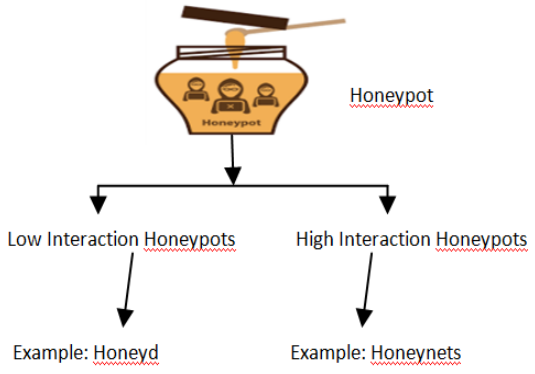

Figure 4 : Honeypot and its types

Intrusion detection is the process of monitoring a host or network and analyzing the incoming network traffic. Traffic can be analyzed for known botnet activity or for general suspicious anomalies. DNS tracking is a type of intrusion detection that analyzes DNS queries between bots and their server (Jing et al., 2009). As most bots must first contact a server (or super-peer) to join the network, researchers look for these messages to learn which computers are infected.

As botnet detection techniques continue to improve and the price of Cloud services continues to drop, botmasters will move their activities to the Cloud. However, most current botnet detection methods will not easily port to the Cloud. The CSP is in the best position to detect botclouds and should be responsible for maintaining intrusion detection mechanisms to detect not only incoming attacks (as they do now) but also outgoing attacks. The CSP has complete control of the Cloud and thus complete control of any botclouds they detect. In general, the CSP is in a better position to detect attacks better than other actors. For instance, an extrusion detection system designed to identify click fraud could be deployed across all nodes under the control of a single CSP. To achieve the same coverage outside of the Cloud would require multiple Internet Service Providers (ISP) to implement and coordinate the same system.

The existing botnet detection techniques are categorize into two main groups given as Honeynets Based Detection Technique and Intrusion Detection System[12]. Researchers focus on the cyber-security to detect botnets attacks and prevent cloud servers from the botnet attacks. But still research on botnet detection is immature, and need more research to improve data security in cloud computing.

The record universal cases performed by botnets are DDoS, click fraud, phishing fraud, key logging, bitcoins fraud, spamming, sniffing traffic, spreading new malware, google AdSense abuse, password stealer and mass identity theft with bots[6]. Like worms propagation the botnet also propagate itself, similarly like virus, botnet also keep it hidden from detection. Botnet has an integrated control and command system that's why it attack similar numerous standardization unaccompanied tools. It spurs with a very high infected by botnet, bots are also known as a zombie, that's why a botnet is also called zombie network.

Botnet detection is the most important task to improve the cyber-security against various cyber-attacks occurs in internet nowadays. According to the previous research botnet detection techniques can be classified into two categories honeynets detection techniques and intrusion detection techniques [12][25][26]. Intrusion detection system is further divided into sub-categories.

Honeynets& Honeypots Based Detection System: Honeynets and Honeypots both are denoting the end user devices. These end users PC's are the best way to collect critical information about the cyber-attacks. This end user PC is very easy for botmaster to attack and compromise, because it's very vulnerable to malicious attacks. The cyber-security group will be able to make good detection techniques under the collected information about the botnet attacks through these honeynets.

IDS (Intrusion Detection System): Intrusion detection system is using for monitoring the traffic flow for the malicious activities of a network. During the traffic if it found some malicious attack it directly inform the computer system or the administrator of the system. IDS have also the capabilities to take action against such malicious activities to block the traffic coming from the virus infected system. There are two types of intrusion detection system one is signature based and the other is anomaly based.

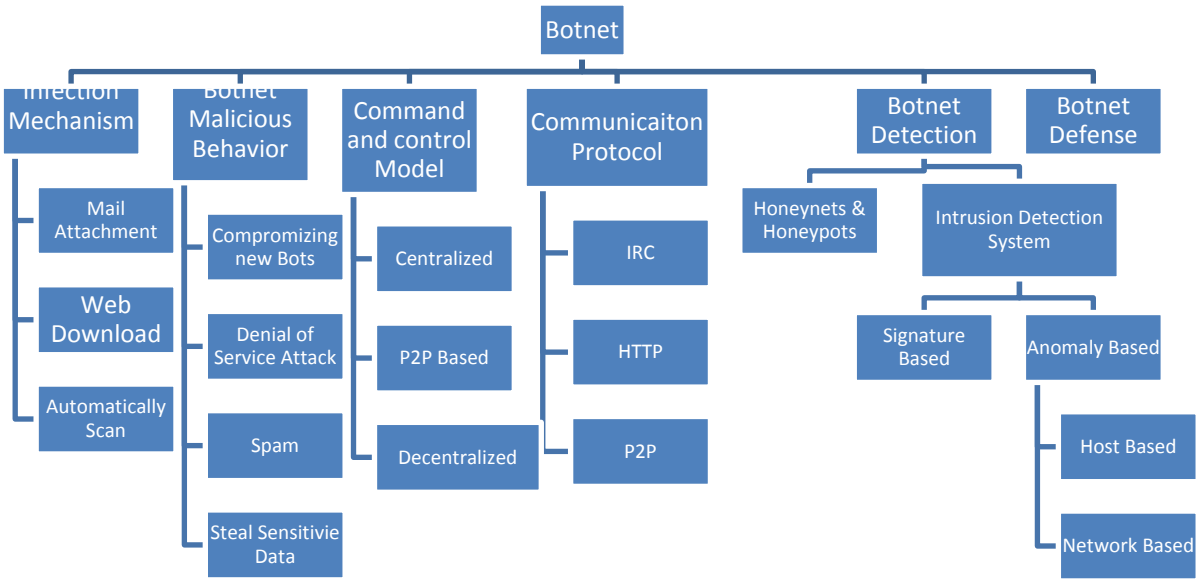

Figure 3: Botnet Taxonomy

V. CONCLUSION

Botnets and Botclouds are the major threats for the future of cloud computing, mainly because they attack the system by being a part of the cloud computing or cloud network architecture. Identification of such bots or bot-master is really a challenging task. The main methods for tradition botnet detection are Honeypots and Intrusion Detection. Honeypots are unprotected computers that are intentionally allowed to be infected by botnets and Intrusion detection is the process of monitoring a host or network and analyzing the incoming network traffic and handle malicious process accordingly and trace the Bot attacks with proper handlers. DNS tracking is a type of intrusion detection that analyzes DNS queries between bots and their servers. More research is required to implement new detection techniques at the CSP, as CSP is in the best position to detect and block the botnet attacks. Every network call is gone through the CSP and if there is strong bonding between CSP and client, then it will be beneficial for detecting and taking appropriate action against the attackers.

ACKNOWLEDGMENT

The authors are thankful for all the people, who supported us directly or indirectly for the preparation of this research paper. We are thankful for Dr. K.K. Goyal for his continuous support and motivation.

REFERENCES

[1] R.V. Deshmukh, K.K. Devadkar, "Understanding DDoS Attack & Its Effect In Cloud Environment", Procedia Computer Science 49 (2015) 202 – 210, 4th International Conference on Advances in Computing, Communication and Control (ICAC3'15).

[2] M. D. Firoozjaei, J. Jeong, Hoon Ko, H. Kim, "Security challenges with network functions virtualization", Future Generation Computer Systems (2016), pp. 1-15.

[3] Jakóbik, A., Palmieri, F., Kołodziej, J., Stackelberg games for modeling defense scenarios against cloud security threats, Journal of Network and Computer Applications (2018), doi: 10.1016/j.jnca.2018.02.015

[4] Kassidy Clark Martijn Warnier Frances M. T. Brazier, BOTCLOUDS : The Future of Cloud-based Botnets?

[5] L. Coppolino, S. D'Antonio ,G.Mazzeo ,L. Romano, "Cloud security: Emerging threats and current solutions" , Computers and Electrical Engineering (2016) pp. 1–15, ISSN :0045-7906, Elsevier

[6] W. Doua, Qi Chena, J. Chenb "A confidence-based filtering method for DDoS attack defense in cloud environment", Future Generation Computer Systems 29 (2013) pp. 1838–1850

[7] N. Khana, A. Al-Yasirib, "Identifying Cloud Security Threats to Strengthen Cloud Computing Adoption Framework", The 2nd International Workshop on Internet of Thing: Networking Applications and Technologies (IoTNAT' 2016), Elsevier, Procedia Computer Science 94 (2016) pp. 485 – 490

[8] F. Lombardi, Roberto Di Pietro, "Secure virtualization for cloud computing", Journal of Network and Computer Applications 34 (2011) pp. 1113–1122

[9] S. K. Majhia, S. K. Dhalb, "Threat Modelling of Virtual Machine Migration Auction", International Conference on Information Security & Privacy (ICISP2015), 11-12 December 2015, Nagpur, INDIA, Procedia Computer Science 78 (2016) pp. 107 – 113

[10] O. Osanaiye, Kim-Kwang R. Choo and M. Dlodlo, "Distributed Denial of Service (DDoS) Resilience in Cloud: Review and Conceptual Cloud DDoS Mitigation Framework", Journal of Network and Computer Applications, 2016

[11] A. Thilakarathne, J. I Wijayanayake, "Security Challenges Of Cloud Computing", INTERNATIONAL JOURNAL OF SCIENTIFIC & TECHNOLOGY RESEARCH VOLUME 3, ISSUE 11, NOVEMBER 2014 ISSN 2277-8616

[12] S. Singh, Young-Sik Jeong and Jong Hyuk park, "A Survey on Cloud Computing Security: Issues, Threats, and Solutions", Journal of Network and Computer Applications, 2016

[13] K. J. Singh, Tanmay De, "MLP-GA based algorithm to detect application layer DDoS attack", Journal of Information Security and Applications 36 (2017) 145–153

[14] G. Somani, M. S. Gaur, D. Sanghi, M. Conti, M. Rajarajan," DDoS victim service containment to minimize the internal collateral damages in cloud computing", Computers and Electrical Engineering (2016), pp. 1–15

[15] G. Somani, M. Gaur, D. Sanghi, M. Conti, Rajkumar Buyya, "DDoS Attacks in Cloud Computing: Issues, Taxonomy, and Future Directions", Computer Communications (2017), pp. 1-22

[16] R. Trapero, J. Modic, M. Stopar, A. Tahaa, N. Suri, "A novel approach to manage cloud security SLA incidents", Future Generation Computer Systems 72 (2017), pp. 193–205

[17] B. Wang, Y. Zheng, W. Lou, Y. T. Hou, "DDoS attack protection in the era of cloud computing and Software-Defined Networking", Computer Networks 81 (2015), pp. 308–319

[18] Z. Wang, "An elastic and resiliency defense against DDoS attacks onthecritical DNS authoritative infrastructure", Journal of Computer and System Sciences, 2017.