IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Vol.-6, Issue-11, Nov. 2018 E-ISSN: 2347-2693

Multilayered Framework for Mitigating (MLFM) EDoS Attack

A. Somasundaram

[1]UG Dept of Computer Applications, Sree Saraswathi Thiyagaraja College, Coimbatore, India

[*]*Corresponding Author: somasundaram.a@gmail.com, Tel.: +9865518811*

Available online at: www.ijcseonline.org

Accepted: 25/Nov/2018, Published: 30/Nov/2018

***Abstract*—** "Cloud Computing" is an attractive model for enterprise businesses because of its on-demand, openness, reduced cost, scalability and pay-by-use business model. The DDoS attack on metered resources of Cloud environment is termed as Fraudulent Resource Consumption (FRC) attack. The FRC Attack leads to EDoS (economic Distributed Denial of Service) attack which aims to consume the cloud resource by attacker and impose financial burden to the legitimate user, where integrity, availability and confidentiality of the cloud services are never compromised but affects the accountability which leads to inaccurate billing. This paper surveys different techniques that generate, detect and mitigate the EDoS Attack on Cloud and proposes a Multilayered Framework for Mitigating EDoS Attack.

***Keywords*—**Cloud computing; Distributed Denial of Service (DDoS) attack; Fraudulent Resource Consumption (FRC) attack,; Economic Distributed Denial of Service (EDoS) Attack

I. INTRODUCTION

Cloud computing is offering on-demand, virtually infinite computing infrastructure, information services and entertainment services to the customer's. Elasticity of resource availability at the service provider is the key driving aspect for the growing popularity of the cloud paradigm. Cloud computing enables services to be deployed and accessed globally with little maintenance by providing QoS as per service level agreement (SLA) of customer. The users of the Cloud pay as they use (PAYG), based on application needs. The cloud services can be categorized as metered services and unmetered services. The payment for the Cloud services is calculated based on metering of resource elements, application and services. The basic billable infrastructure elements are listed in Table1.

Table 1- Metered Resources

S. No	Resource	Metric
1	CPU	CPU cycles consumed (vs Number of CPU instances allocated.) by number of hours provisioned
2	Memory	The actual amount of RAM consumed (vs allocated) multiplied by the number of hours.
3	Storage	The average or peak amount of storage consumed per hour
4	Bandwidth	the average bandwidth utilization required by different kinds of application

According to the authors [1], pricing for a cloud service can be applied based on parameters like configuration and duration of use of resources. Cloud pricing is based on the dynamic operational cost of running the service. In this pricing model, the base cost of running the service is specified by the service provider, base operational cost - C_{base}.The pricing rules which define the pricing overhead for running the service under various load conditions are specified by the service provider. This pricing overhead is given by β (l, t), where l is the load at time instance of operation t. The current price, Pt for the interval at a given load condition is given as,

$$P_t = C_{base} \times \beta (l, t) \quad (1)$$

Where,
P_t - operational price at time t
C_{base} -base operational cost
β (l, t) - cost of running the service under load l at time t.
The overall load Lt on the cloud infrastructure at time t is the sum of the load on every node as given by:

$$L_t = \sum_{i=1}^{n} l_i \quad (2)$$

Where,
L_t - total load on the cloud at time t
l_i - load index of the individual cloud component.
The load obtained in (2) is mapped to a corresponding pricing value at a given interval of time, t. The bill amount is computed as a summation of the product of instantaneous

pricing obtained in (1) and the utilization of the consumer, u_t The total bill amount is obtained as,

$$\text{Bill} = \sum_{t=1}^{n} P_t \text{ X } u_t \quad (3)$$

where,
P_t - operational price at time t
u_t - resource utilization of the consumer at time t.

The remainder of the paper is organized as follows:
Section II provides a brief overview of FRC Attack. Section III describes the EDOS Attack. Section IV surveys EDOS attack generation methods. Section V surveys EDOS attack detection. Section VI surveys EDOS attack mitigation. Section VII presents Multilayered Framework for Mitigating (MLFM) EDOS Attack. Finally, Section VIII summarizes and draws the conclusions.

II. FRC ATTACK

A denial-of-service (DoS) attack is an attempt to make a computer resource (e.g. the network bandwidth, CPU time, etc.) unavailable to its intended users. To overload the necessary network and CPU resources, attackers tend to use a large number of machines to launch the Distributed DoS (DDoS) attacks [8].The effects of Denial of Service (DDoS) attacks in Cloud environment, involve not only the reduction of quality of the service, but also the service maintenance costs in terms of resource consumption by exploiting the cloud flexibility and elastic behavior.

The DDoS attack can be divided into two major categories
(1) Resource focused attack (e.g. network bandwidth, memory, and CPU)
(2) Application-focused attack (e.g. web applications, database service)

The DDoS Attack degrades the main Cloud features such as Auto Scaling, Pay-as-you-go accounting and Multi-tenancy. A DDoS attack becomes a Fraudulent Resource Consumption (FRC) attack when consuming the metered resources of Web based vices and increasing the cloud consumer's financial burden [12]. The motive of an FRC attacker could range from ego and hacktivism to monetary gain, extortion, revenge, competitive advantage, or economic espionage [9].

FRC attacks are stealthy in nature and invisible to the detection mechanism. They are sophisticated attacks aim at exploiting the cloud flexibility, forcing the application to consume more resources than needed, affecting the cloud customer more on financial aspects than on the service availability [16]. The root causes for FRC attack is auto scaling and elastic load balance nature of the Cloud feature. The FRC attack creates negative impacts to the cloud service provider as well as customer. The successful FRC attack will affect the business model as follows,

- Integrity of CSP may lost which leads to lose of business
- Operating cost of CSP increased and decreases the profit
- Leads financial burden to the cloud users.

The FRC attack causes direct and indirect impacts to the cloud users. The Economic Denial of Sustainability (EDoS) attack and Energy oriented distributed denial of service (e-DDoS) are the two major impacts of FRC which are fraudulently consumes resources bandwidth and electrical energy respectively.

III. EDOS ATTACK

The EDoS in cloud are the results of the DDoS attack, where the service to the legitimate user is never restricted. But the service provider who is using cloud will incur a debilitating bill by using highly elastic (auto-Scaling) capacity to unsuspectingly serve a large amount of undesired traffic in order to maintain the QoS as per the SLA. This leads to Economic Denial of Sustainability (EDoS)[6]. "Auto-scaling automates the expansion or contraction of system capacity that is available for applications and is a commonly desired feature in cloud IaaS and PaaS offerings. When feasible, technology buyers should use it to match provisioned capacity to application demand and save costs [7].

The EDoS attack caused by fraudulent consumption of Network Bandwidth, Processing Power Exhaustion and Disk Hardware Solicitation. An EDoS is attack generated by inject large amount of malicious traffic such as HTTP and XML based requests to the Cloud in a constant rate , which will exploit the Cloud's scalability and gear up the usage cost of the legitimate user. The Fig 1 depicts the scenario that, the attacker accesses the cloud services fraudulently and leads to auto scale of metered resources. The fraudulently consumed resources in turn misleads to inaccurate billing.

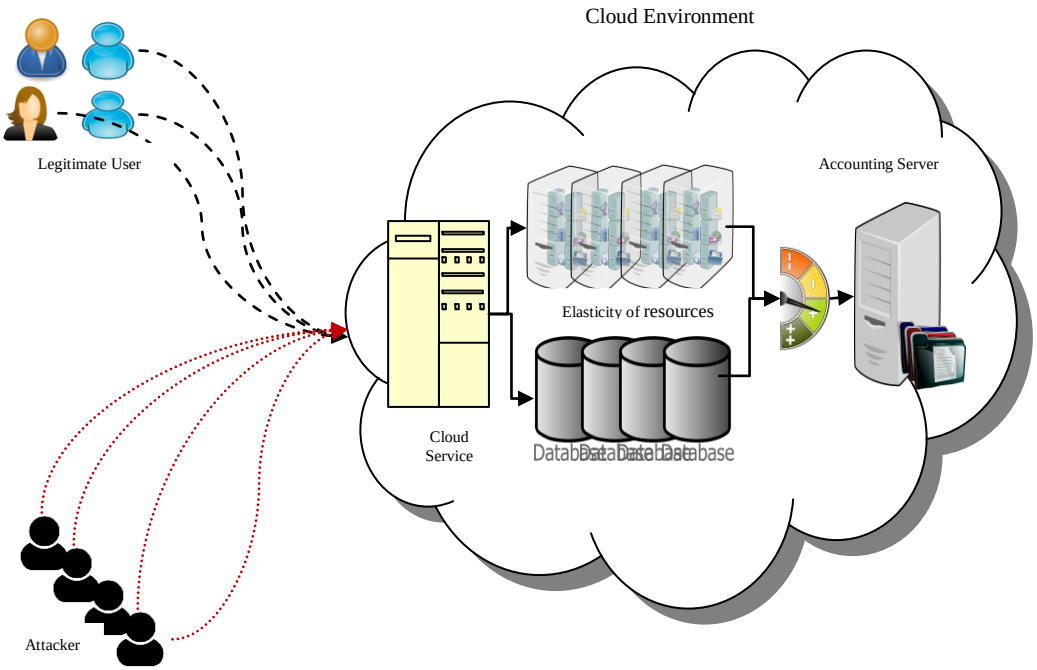

Fig 1: EDOS Attack

It was found in an experiment, by sending 1000 requests/second with 1000 Megabits/second data transfer on a web-service hosted on Amazon CloudFront for 30 days incurred an additional cost of $42,000 to the cloud user. In a similar experiment, this incurred the additional cost to the customer, by an attacker just sending one web request (size of 320KB) per minute for one month, which accumulates total 13GB of data transfer [7]. The Fig 1 shows the EDoS operations and its countermeasures.

IV. ATTACK GENERATION

The EDoS attack executing at the various layers by generating large number of fake requests to the Cloud Server to the consumption of Cloud resource. The EDoS attack can be generated by using botnets or cloud originated DDoS attacks. Table 2 describes the attacks that aim at causing EDoS attack.

Table 2 : EDoS Attack generation Methods

S.No	Layer	Type of Attack	Description
1	Network /Transport Layer	Spoofing	Hide the source of an attack to gain access to restricted resources or services.
2		Port Scanning	Misusing the protocols such as TCP SYN, TCP ACK, TCP ECHO, TCMP SWEEP
3		Web Service-addressing Spoofing	This is an extension of the spoofing attack where the ReplyTo or FaultTo address in a SOAP header is falsified leading to a reflective attack
4		Reflective attack	Request messages are sent to reflector machines via zombie machines containing the spoofed source IP address of the victim. The genuine replies to these requests are then sent to the victim causing flooding.
5	Application Layer	Oversized XML	The attacker sends very large XML document which contains nested elements, with the intention of increases memory requirements of the server.
6		Flooding	Make use of protocols like HTTP, ICMP to saturate the network bandwidth on a network using zombies
7		Coercive Parsing	The attacker sends malformed XML aimed a clogging up CPU cycles by incorporating many namespace declarations or by simply using very deeply nested XML structures[5]

Index Page Based EDoS on Infrastructure Cloud- Index page of any website is available freely without any authentication credentials so employing bulky and concurrent HTTP-GET requests to index page of a website to generate resource consumption overhead on server [6].These attack consumes good amount of bandwidth and leads to heavy economic loss to the cloud user.

Web-Bugs- A Web Bug is embedded in a spam-email of legitimate user's browsers will generate an HTTP-GET Request to attack the Cloud Server [4]. These intelligent attacks are planned by constructing bots behaving like a real user based on the web service flow and behavior.

YO-YO ATTACK - Exploiting the auto-scaling mechanism to perform an efficient attack that impacts the cost of a service and the response time of standard users [25].This is also called as Reduction of Quality (RoQ) attack It cycles between two phases repeatedly: In the on-attack phase, the attacker sends a short burst of traffic that causes the auto-scaling mechanism to perform a scale up. In the off-attack phase, the attacker stops sending the excess traffic.

V. EDOS ATTACK DETECTION

EDOS attack imposes exhaustive computation tasks to the server on the Cloud by exploiting its system vulnerability or flooding it with huge amount of useless packets. This causes serious damages to the services running on the Cloud server [1]. EDoS detection aims to identify the suspicious traffic pattern which will consume the billable resources of the Cloud. EDoS attacks are specific to Cloud service and are not easy to detect, because cloud services don't have any mechanism to provide the correlation between "requests" and "successful transactions" [10].Attack detection systems are based on monitoring the traffic transmitted over the protected networks to provide quality services with minimum delay in response.

The attack can be detected based on various metrics such as pattern in web access behavior of a client, session duration and thresholds based filtering. Patterns are recognized from web access logs or request headers of each transaction. The specific pattern to identify in the log, is decided by attack traces and other past historic behaviors [13].

Signature-based detection: It detects traffic anomalies by looking for patterns that match signatures of known anomalies. It's based on a firewall, which is working as a filter. It receives the request from the client, and redirected to a Puzzle-Server. The Puzzle-Server sends a puzzle to the client, who either sends a correct or false answer of the puzzle. If the answer is correct, the server will send a positive acknowledgment to the firewall which will add the client to its white list and will forward the request to the protected server to get services. Otherwise, the firewall will receive a negative acknowledgment and put the client in its black list [23].

Time Spent on a Page (TSP) based Detection: Time Spent on a Web Page (TSP) is defined as time spent on viewing a web page. The TSP of the attack traffic differs from the mean TSP of a web page. This deviation of TSP from the mean is calculated taking the exponential distribution of the TSPs and the calculated value is used to detect the surreptitious behavior [14].

Threshold-based detection: The threshold is used to differentiate between normal traffic and abnormal traffic in the network. Dynamic threshold value is based on training or priori knowledge of the network activity, after that the threshold is selected [15].

VI. EDOS ATTACK MITIGATION

There are varieties of EDoS attack Mitigation techniques are available. The Fig 3. Shows taxonomy of the EDoS attack mitigation. EDoS mitigation schemes can be classified into two categories; reactive and proactive solutions. Reactive solutions systems are waiting the attack to occur then try to mitigate its impacts. It works in three steps. First step, the use traffic monitoring to identify attacks in progress, the second step triggered the and equence to locate the source of attack and in the third step, mitigation methods are implemented to eliminate or reduce the impact of the attack. The proactive solution treating the source of packets before reaching to the secured server [17]. The filtering systems are considered as reactive solutions. However, Overlay-based techniques are considered as proactive solutions. There are many mechanisms available to mitigate EDoS attacks. Few of these methods are discussed in this section.

Secure Overlay Services (SOS): SOS architecture consists of a set of nodes which are classified into four groups. The first group is the Secure Overlay Access Points (SOAP), while the second collection is the overlay nodes which connect SOAP nodes with the third group .i.e., Beacon nodes. The last group is the Secret Servlets. It reduces the possibility of harmful attacks by "performing intensive filtering near protected network edges", and by "introducing randomness and anonymity into the architecture, making it difficult for an attacker to target nodes along the path to a specific SOS-protected destination" [22].

EDoS Shield: This mechanism has two main components, the cloud verifier node and virtual firewall. Firewall does the packet filtering based on the White list and Black list method. The service provider uses CAPTCHA (Graphic turning test) to identify that the request is coming from a legitimate user or from a malicious machine [19]. If request is coming from an attacker (machine) then request is add in black list and we block the request i.e. request cannot pass through virtual firewall. Otherwise request passes through

virtual firewall and starts the service in cloud infrastructure. The limitation of this scheme is that the time delay, due to Turing test performed on every incoming request

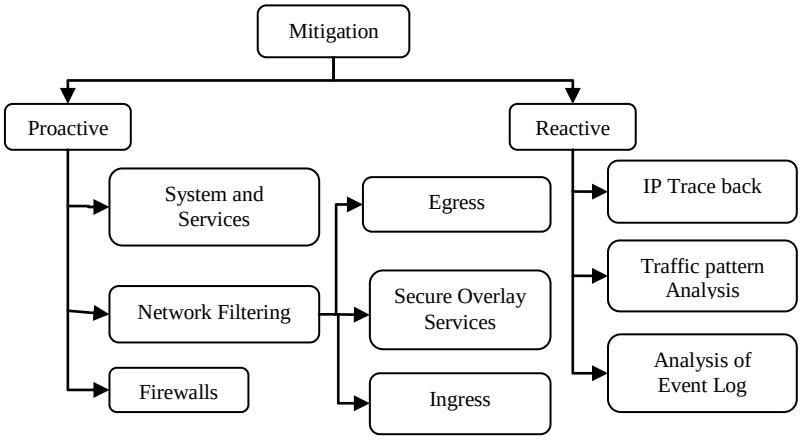

Figure 3: Taxonomy of EDoS Attack Mitigation

Enhanced EDoS-Shield: This is used to mitigate the EDoS attacks originating from spoofed IP addresses [3]. When user registers into cloud for the first time, the request goes to Verifier node and TTL value is recorded related to source IP address. When user sends request, the Verifier node check the request against source IP address and corresponding TTL value range. If both values match, then requester is added to white list and request pass through virtual firewall otherwise added in a black list and request is blocked at virtual firewall. This method fails to find the attacker with-in network vulnerability to IP spoofing.

sPoW: self-verifying Proof of Work (sPoW) is a On Demand Cloud based and application layer mitigation scheme. The main function of this method is to filter the attack traffic before it start over committing of resources. It transforms the network level traffic to distinguishable traffic that can be filtered using pattern matching. In second phase it sends crypto puzzles to client to resolve by brute force method. Here client solves a sPoW puzzle to discover a hidden channel to communicate with the serve[26].This framework requires high computation power to solve crypto-puzzles for client, which can create overheads on the machine to brute force harder puzzles, which makes this method not suitable for mobile devices

In-Cloud Scrubber Service: It Generates and verifies the Client puzzle (crypto puzzle) to authenticate the clients. The generated puzzle solved by the consumer by brute force method. Cloud-service is switched between normal and suspected modes, it depends on server and network bandwidth. During the normal mode, the incoming requests will be immediately directed to cloud-service and otherwise it will be directed to In-Cloud Scrubber Service for verification process during the suspected mode. The limitation of this technique is that Client-puzzles provide weak access guarantees to customer/users.[20].

Digital signature based architecture: This framework used to differentiate the legitimate user from the attacker. The client request goes to cloud infrastructure and it is verified at verifying node using public key infrastructure (PKI). Request is send to certify authority (CA) to check that request is coming from legitimate user or an attacker. Certify authority tries to decrypts the request with his private key. If request is decrypted by CA private key, it proves that it's coming from a legitimate user; otherwise it is originated from an attacker. If request is coming from legitimate user, it is passed through the firewall and is forwarded to cloud infrastructure for service while other requests are blocked [18].

VivinSandar and Shenai Framework : This frameworks is based on firewall. It receives the request from the client, and redirect to a Puzzle-Server. The Puzzle-Server sends a puzzle to the client, who either sends a correct or false answer of the puzzle. If the answer is correct, the server will send a positive acknowledgment to the firewall which will add the client to its white list and will forward the request to the protected server to get services. Otherwise, the firewall will receive a negative acknowledgment and put the client in its black list [21].

An Enhanced EDoS Mitigation System: This system tests the legitimacy of the request by testing the first packet from the source of requests during each session to distinguish the human user from the botnet. The test is done by the verifier node(s), which use the Graphical Turing Test (GTT) in

verifying the packets. After that, the users' requests will be examined by the IPS device. If IPS detects malware in the contents of packets, the source IP address will be placed in the Malicious List. The last layer of the monitoring process tools will be done by the Reverse Proxy (RP) which performs several tasks including detecting the suspicious users who try to overwhelm the system by sending a huge amount of requests without drawing the attention of the previous monitoring layers. If there are suspicious users detected, the client puzzle server will send a crypto puzzle to them to delay their requests [24]

Damask: This is based on Software-Defined Networking .The DaMask architecture has three layers, network switches, network controller, and network applications. The main functions of the DaMask are DDoS detection and reaction. There are two separate modules in the DaMask, DaMask-D, a network attack detection system, and DaMask-M, an attack reaction module. It requires little effort from the cloud provider which means few changes are required from the current cloud computing service architecture [11]. Table 3 shows the comparison between EDoS mitigation mechanisms.

VII. MULTILAYERED FRAMEWORK FOR MITIGATING (MLFM) EDOS ATTACK

The following section discusses the proposed architecture of Multilayered Framework for Mitigating EDoS Attack defense in a cloud environment. Figure 3 shows the MLFM Architecture and how it is filtering the traffic**.** To secure the cloud environment from EDoS attack, Traffic Analyzer is placed as first layer. The Tuning Test layer provides the first level of traffic filtering by challenging the user with the verifiable client puzzle to authenticate the client. If the request found to be malicious it is simply rejected else the request will pass to the Traffic Analyzer which acts as second lever filter of traffic. The Traffic Analyzer layer encompasses the User Behavior Analyzer (UBA), Threshold Monitor (TM) and Attack Pattern Analyzer (ABA).The User Behavior Analyzer detect the attack by comparing the traffic with the statistical data stored in Client Log and Attack Pattern Directory. At the same time the Threshold Monitor compares the utilization of resource with the threshold level and the Attack Pattern Analyzer monitors the normal usage profile for each web service or web service operation. The Attack Pattern Analyzer works based on the parameters such as session duration, sequence of access of resources. Based on the analysis the TA may allow the traffic into the access the service or let the user to face the advanced authentication.

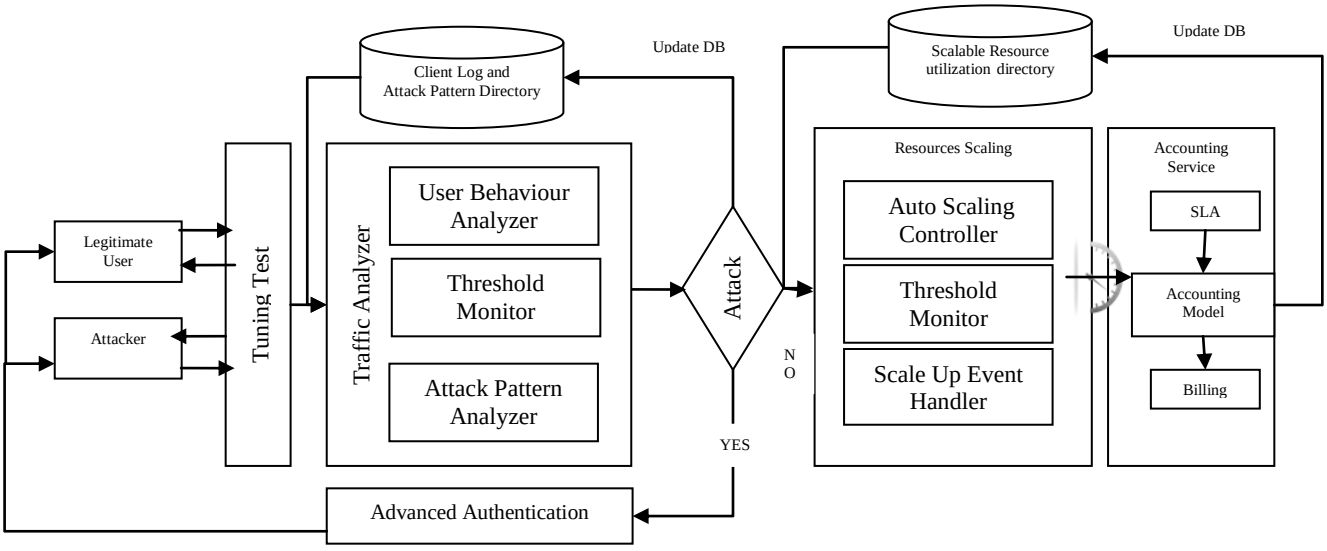

Fig 3: Multilayered Framework For Mitigating (MLFM) EDOS Attack

The Scale Up Event Handler in Resource scaling layer notifies the Threshold Monitor whenever there is an increase in the Virtual instance. The Threshold Monitor ensures the resource consumptions of running virtual instances do not exceed the normal utilization level. If it exceeds it triggers Auto Scaling Controller for scale down or sustains the instances based on the demand. The Accounting Services Layer generates the billing based on the SLA and Accounting Model of the Cloud Consumer.

VIII. CONCLUSION

This paper provides a comprehensive and detailed survey about EDoS attack and proposes architecture Multilayered

Framework for Mitigating (MLFM) EDoS Attack., which provides efficient traffic filtering, Controlled resource access, Verifiable Resource Accounting based on the SLA. The proposed MLFM architecture mitigates the EDoS attack and provides high quality of service to the cloud users.

REFERENCES

[1] Antunes, J.; Neves, N.; Verissimo, P. Detection and Prediction of Resource-Exhaustion Vulnerabilities. In Proceedings of the 19th International Symposium on Software Reliability Engineering, Seattle, WA, USA, 10–14 November 2008; pp. 87–96.

[2] Akshay Narayan, Shrisha Rao, Gaurav Ranjan and Kumar Dheenadayalan," Smart Metering of Cloud Services", 6th Annual IEEE International Systems Conference (IEEE SysCon 2012), Vancouver, Canada, March 2012.

[3] Al-Haidari, Fahd, Mohammed H. Sqalli, and Khaled Salah. "Enhanced EDoS-Shield for Mitigating EDoS Attacks Originating from Spoofed IP Addresses."Trust, Security and Privacy in Computing and Communications (TrustCom), 2012 IEEE 11th International Conference on. IEEE, 2012.

[4] Armin Slopek and Natalija Vlajic, "Economic Denial of Sustainability (EDoS) Attack in the Cloud Using Web-Bugs", in 17th International Symposium on Research in Attacks, Intrusions and Defenses (RAID 2014), Gothenburg, Sweden , September 2014.

[5] Ashley Chonka,YangXiang n, WanleiZhou,AlessioBonti ."Cloud security to protect cloud computing against HTTP-DoS and XML-DoS attacks". Journal of Computer Applications. Volume 34, Issue 4, July 2011, Pages 1097–1107.

[6] B Saini, G Somani , "Index Page Based EDoS Attacks in Infrastructure Cloud" , in Recent Trends in Computer Networks and Distributed Systems Security(2014), 382-395

[7] Bing Wang , Yao Zheng, Wenjing Lou, Y. Thomas Hou.DDoS attack protection in the era of cloud computing and Software-Defined Networking. Computer Networks 81 (2015) 308–319.

[8] Denial-of-service attack ,Wikipedia, http://en.wikipedia.org/wiki/Denial-of-service_attack

[9] G. Stonebumer, A. Goguen, and A. Feringa, "Risk Management Guide for Information Technology Systems," NIST Special Publication 800-30, July 2002.

[10] http://www.rationalsurvivability.com/blog/2009/01/a-couple-of-follow-ups-on-the-edos-economic -denial-of-sustainability-concept

[11] Joseph Idziorek and Mark Tannian, "Exploiting Cloud Utility Models for Profit and Ruin", Proceedings of IEEE International Conference on Cloud Computing, pp. 33-40, 2011.

[12] Joseph Idziorek, Mark F. Tannian, and Doug Jacobson," The Insecurity of Cloud Utility Models", IEEE Computer Society, March/April 2013.

[13] Keromytis, A., Misra, V., AND Rubenstein, D., 2002. SOS : Secure Overlay Services. SIGCOMM, pp. 61–72.

[14] Koduru, A. ; Comput. Sci. & Eng., Nat. Inst. of Technol., Trichy, India ; Neelakantam, T. ; Saira Bhanu, S.M. Detection of Economic Denial of Sustainability Using Time Spent on a Web Page in Cloud, IEEE International Conference on Cloud Computing in Emerging Markets (CCEM) 2013

[15] L. Jun-Ho et.al. "Multi-level intrusion detection system and log management in cloud computing," Proceedings of the13th International Conf. pp. 552–555, Feb. 2011

[16] Massimo Ficco and Massimiliano Rak, Stealthy Denial of Service Strategy in Cloud Computing, IEEE Transactions On Cloud Computing, January 2015.

[17] Mirkovic, G. Prier, and P. Reiher. 2002a. Attacking DDoS at the source. In Net-work Protocols, 2002. Proceedings.10th IEEE International Conference on. 312–321.DOI:http://dx.doi.org/10.1109/ICNP.2002.1181418

[18] Mohit Kumar, Nirmal Roberts."A Technique to reduce the Economic Denial of Sustainability (EDoS) Attack in Cloud". Fourth International Conference on Recent Trends in Information, Telecommunication and Computing – ITC 2013

[19] Mor Sides , Anat Bremler-Barr, Elisha Rosensweig in " Yo-Yo Attack - Vulnerability in auto-scaling mechanism " in SIGCOMM '15 August 17-21, 2015, London, United Kingdom

[20] Naresh Kumar, M. , Sujatha, P. ; Kalva, V. ; Nagori, R. ; Katukojwala, A.K. ; Kumar, M., Mitigating Economic Denial of Sustainability (EDoS) in Cloud Computing Using In-cloud Scrubber Service, 2012 Fourth International Conference on Computational Intelligence and Communication Networks (CICN),IEEE 2012.

[21] Naresh Kumar, M., Sujatha, P., Kalva, V., Nagori, R., Katukojwala, A., Kumar, M. "Mitigating economic denial of sustainability (edos) in cloud computing using in-cloud scrubber service", In: Computational Intelligence and Communication Networks (CICN), 2012 Fourth International Conference on. 2012, p. 535–539. doi:10.1109/CICN.2012.149.

[22] ReviewMyLife.co.uk. 2011. Amazon CloudFront and S3 maximum cost. http://www.reviewmylife .co.uk/blog /2011/05/19/amazon-cloudfront-and-s3-maximum-cost/. (2011).

[23] S VivinSandar, SudhirShenai "Economic Denial of Sustainability (EDoS) in Cloud Services using HTTP and XML based DDoS Attacks". International Journal of Computer Applications (0975 – 8887) Volume 41– No.20, March 2012.

[24] Soon HinKhor Akihiro Nakao, "sPow On-Demand Cloud-based eDDoS Mitigation Mechanism"Fifth Workshop on Hot Topics in System Dependability

[25] Sqalli, Mohammed H., Fahd Al-Haidari, and Khaled Salah. "EDoS Shield-A Two-Steps Mitigation Technique against EDoS Attacks in Cloud Computing." Utility and Cloud Computing (UCC), 2011 Fourth IEEE International Conference on. IEEE, 2011.

[26] Wael Alosaimi and Khalid Al-Begain. An Enhanced Economical Denial of Sustainability Mitigation System for the Cloud. 2013 Seventh International Conference on Next Generation Mobile Apps, Services and Technologies.

Authors Profile

Mr.A.Somasundaram pursed Bachelor of Science from Bharathiyar University in 2001 and Master of Computer Apllications from Bharathiyar University in year 2004. He is currently pursuing Ph.D. and currently working as Assistant Professor in UG Department of Computer Applications at Sree Saraswathi Thiyagaraja College since 2016. He is a Life member of ISTE. His main research work focuses on Network Security and Cloud Security. He has 14 years of teaching experience.

