PREVENTING APPLICATION LAYER DDOS ATTACK USING PASSWORD ATTACK RESISTANCE PROTOCOL

[1]Neha G P, [2]Prof. A. M. Prasad
[1]Student, [2]Professor
[1]Computer Science and Engineering,
[1]Dayananda Sagar College of Engineering, Bangalore, India

Abstract: Distributed Denial of service (DDoS) attacks has become a severe threat to the security of Application layer and Network Layer. In this paper the main concern is on Application layer DDoS attacks. These attacks avoid most intrusion prevention systems by sending numerous HTTP requests. Since most of these attacks are launched unexpectedly and severely, a fast intrusion prevention system is desirable to detect and try to lessen these attacks as soon as possible. In proposed system a Honey Pot Technique is used, it is one of the most successful techniques used to collect the sample of malware for the purpose of analysis and identification of attacks. It is an innovative technology which consists of massive energy and possibilities in the field of security. It helps in reading the behavior of the attack and attacker information. It is also a concept for the file security in the cloud. Promising results have been obtained from our proposed system over the existing system.

***IndexTerms* - Application Layer DDoS attacks, Honey Pot Technique, Intrusion Prevention System, Automated Turing Test (ATT), Cloud Storage.**

I. INTRODUCTION

Web sites are badly prone to security risks, and so are any networks to which web servers are connected. Setting aside risks created by employee use or misuse of network resources, web server and the site of hosts present are the most serious sources of security risk. Web servers by design open a window between network and the world. The care taken with server maintenance, web application updates and web site coding will define the size of that window, limit the kind of information that can pass through it and thus establish the degree of web security.

Distributed Denial of Service (DDoS) attacks have been a severe threat to the Internet for decades and numerous defense schemes have been proposed to reduce DDoS attacks at the network layer. Recently, application layer DDoS attacks against web servers are growing rapidly and are bringing victims with great revenue losses. The secrecy of application layer DDoS attacks makes most signature-based intrusion prevention systems ineffective. Since most DDoS attacks are launched unexpectedly and severely, it is desirable to design a defense system that can detect and reduce application layer and network layer DDoS attacks as soon as possible to minimize the losses.

A DDoS attack uses many computers to launch an organized DoS attack against one or more targets. Using client/server technology, the slayer is able to multiply the effectiveness of the DoS significantly by holding the resources of multiple unaware associated computers, which serve as attack platforms. The DDoS attack is the most advanced form of DoS attacks. The main goal of a DDoS attack is to cause damage on a victim either for personal reasons, or for material gain, or for popularity.

Distributed denial-of-service (DDoS) attacks are mainly focused on Network layer and Application Layer. Application-Layer DDoS attack is also called as ‘Layer 7’ DDoS attack. Such attacks are not easy to detect and are even harder to protect against. In fact, they might even fail to notice it until the time the website goes down, and it can also affect many back-end systems.

Since website, its applications, and supporting systems are open to the threats from the external world, they become the key targets for such sophisticated hacks designed to affect the way in which the different systems work or to make the most out of the uncorrected fault. With the development of applications continuing to shift to the cloud, such hacks will turn out to be more difficult to shield against. Instead of spending efforts on protecting our network from such complex and secrecy ways, success is decided based on the smartness of our cloud security technology and how appropriately they can use it.

DDoS Defense Mechanisms

DDoS attacks are tough to solve. So, here they classify some of the DDoS defense mechanisms for the DDoS attacks using two different classifications. The first classification is based on the Activity Deployed. We have the following four categories:

1. Intrusion Prevention,
2. Intrusion Detection,
3. Intrusion Tolerance and Mitigation, and
4. Intrusion Response

The second classification is based on the Location Deployment .We have the following three categories:

1. Victim Network,
2. Intermediate Network, and
3. Source Network.

Based on the Activity Deployed: Intrusion prevention

The best mitigation strategy against any attack is to completely prevent the attack. In this case we try to stop DDoS attacks from being launched in the first place. There are many DDoS defense mechanisms that try to prevent systems from attacks: Among them one of the defense mechanism that prevents from DDoS attack is "Honey Pot Technique".

The Honey Pot Technique is one of the most successful techniques used to collect the sample of malware for the purpose of analysis and identification of attacks. It is an innovative technology which consists of massive energy and possibilities in the field of security. It helps in reading the behavior of the attack and attacker information. It is also a concept for the file security in the cloud, with the help of this technique; User can upload and download the file to or from the cloud storage. Whenever user is uploading a file, for each file, it will generate a unique code. And that Unique code will be sent to user through email. Whenever user is downloading the file, User has to give the corresponding unique code to the respective file. If the user has given the correct code then only, user will be able to get the original file. If user is giving the guessing code or some other wrong code, it will not show the error message but, it gives a duplicate file to the user. This concept is called Honey Pot Technique.

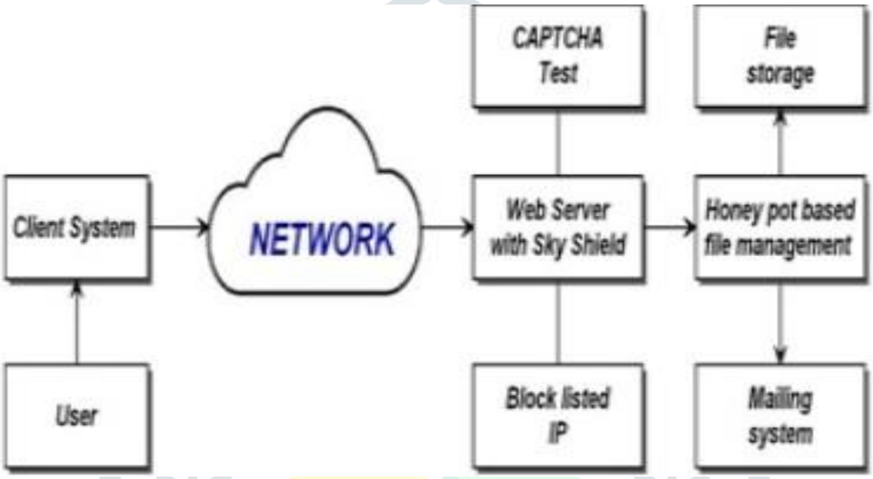


Fig 1: Process of Preventing Application Layer DDOS Attack using Password Attack Resistance Protocol

The challenge of designing a Honey Pot defense system lies in the coordination between the mitigation and detection of attacks.

- First, Mitigation Identification means whenever users will access the web server, System has to identify whether it is mitigation or not. Mitigation means slowing down the server process. It has to identify whether particular IP address is slowing down the service process or not. This is called Mitigation process.

- Second, When there is mitigation happening or mitigation is detected, this system has to identify whether it is an attack or not. For that we can use a captcha test. Based on the captcha test, we will decide that whether it is an attack or not. If it is an attack means, it will be added to the blacklist IP address. If it is not an attack, it will be added to the white list IP address.

Blacklist & White list Management

White list IP addresses are genuine IP addresses, if there are many request coming within the required time or even if the server is slowing down, due to the white list IP address, it is not attack.

Black list IP addresses are suspicious or it is a non-genuine IP address. Whenever request is coming from blacklisted IP address, it will not allow accessing the server.

II. RELATED WORK

Barford et al. [1] found that the detection of a sharp increase in the local variance of the filtered network traffic is an effective way of exposing anomalies. Identifying anomalies fast and accurate is critical to the efficient operation of large computer networks. Accurately characterizing important classes of anomalies greatly facilitates their identification; however, the intricacy and complexities of anomalous traffic.

Tang et al. [2] found that the Voice over IP (VoIP) application uses the Internet to provide voice service; thus it is vulnerable to various security issues common on the IP networks, such as the flooding attack.

Cheng et al. [3] found an efficient online flooding attack detection scheme by combining the sketch techniques with Hellinger distance. The session initiation protocol (SIP) is widely used for controlling multimedia communication sessions over the Internet Protocol (IP).

Salem et al. [4] proposed a flooding attack detection method using a multiple layer reversible sketch.

Sun et al. [5] developed various intrusion detection and prevention systems to detect DDoS attacks and mitigate the caused damage.

Kandula et al. [6] proposed a system to protect web clusters from application layer DDoS attacks by using the CAPTCHA techniques.

Rangasamy et al. [7] designed a graphical puzzle authentication mechanism to determine whether a client is suspicious or not.

Yang et al. [8] developed a Filter-based approach which filters to block unwanted traffic Capability based mechanisms that focus on controlling resource usage by clients.

Wang et al. [9] proposed a moving target defense mechanism that defends authenticated clients against Internet service DDoS attacks.

Joldzic et al. [10] proposed a fully transparent system monitoring network traffic for DoS attacks. The system offers an effective way of separating the attack sources during a DDoS attack.

Sivabalan et al. [11] developed mitigation of application layer using Filter-based approaches.

Li et al. [12] developed a lightweight DDoS attacks detection mechanism for web server using TCM-KNN (Transductive Confidence Machines for K-Nearest Neighbors) and genetic algorithm based instance selection methods.

S.Yu et al. [13] found that the Distributed Denial of Service (DDoS) attack is a severe threat to the Internet, and Botnets are usually the engines behind them. Weaken botmasters attempt to disable detectors by faking the traffic patterns of flash crowds. This poses a critical challenge to those who defend against DDoS attacks.

Thapngam et al. [14] proposed a behavior-based detection that can separate Distributed Denial of Service (DDoS) attack traffic from genuine traffic regardless to various types of the attack packets and methods.

S.Rangan et al.[15] found that Countering distributed denial of service (DDoS) attacks is getting ever more challenging with the huge resources and techniques increasingly available to attackers.

III. Proposed Work

The proposed system uses an effective defense system, Honey Pot Technique is one of the most successful techniques used to collect the sample of malware for the purpose of analysis and identification of attacks. It is an innovative technology which consists of massive energy and possibilities in the field of security.

It helps in reading the behavior of the attack and attacker information. It is also a concept for the file security in the cloud, with the help of this technique; User can upload and download the file into or from the cloud storage.
The System architecture is shown below.

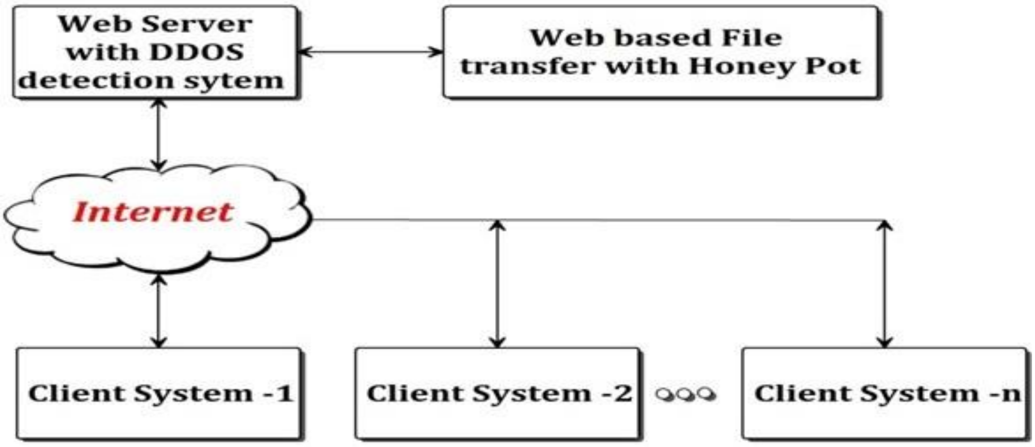

Web Server with DDOS detection sytem
Web based File transfer with Honey Pot
Internet
Client System -1
Client System -2
Client System -n

Fig 2: System Architecture

Mitigation Phase Module

Mitigation Identification means whenever users will access the web server, System has to identify whether it is a mitigation or not. Mitigation means slowing down the server process. It has to identify whether particular IP address is slowing down the service process or not. This is called Mitigation process.

Detection Phase Module

When there is mitigation happening or mitigation is detected, this system has to identify whether it is an attack or not. For that we are using a captcha test. Based on the captcha test, we will decide that whether it is a attack or not. If it is a attack means, it will be added to the blacklist IP address. If it is not a attack, it will be added to the white list IP address.

Captcha Test Module

When there is mitigation happening or mitigation is detected, this system has to identify whether it is an attack or not. For that we are using a captcha test. Based on the captcha test, we will decide that whether it is a attack or not. If it's not a attack then after captcha test move to the homepage.

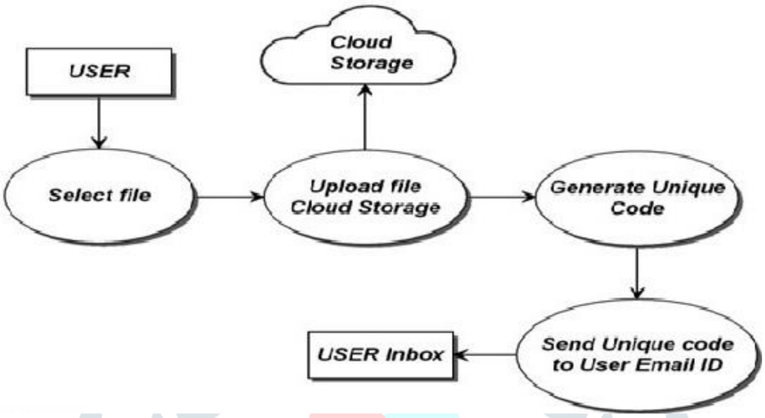


Fig 3: File Uploading Process

Fig. 3 shows how user can upload the files to the cloud with security. There are mainly three cases to upload the files to the cloud:

Case 1: User needs to register with their valid details. Now, User can login with the given user id and password. If login credentials are same as given in the details then user can login to the page and upload and download the files. User needs to upload the file using a unique code that is generated and sent to user's mail. With the appropriate code only the user can be able to upload and download the files from and to the cloud.

Case 2: If the user is able to enter right user id and fails to give the right password then a threshold is set to determine whether user is a Trusted user or not. Based on the threshold user is given a chance to login by entering his password, if user forgets password then the user becomes eligible for CAPTCHA test. If the user is success in the test then user can upload and download files to and from the cloud, then the user will be considered in the whitelist.

Case 3: If the login credentials given by the user is incorrect then user is considered as untrusted if in case user knows the user id and doesn't know the password then a threshold is set and based on this user need to take the CAPTCHA test and if user success in the test then user will be considered in the blacklist and cannot be able to enter into the login page this is based on the trusted user System IP address.

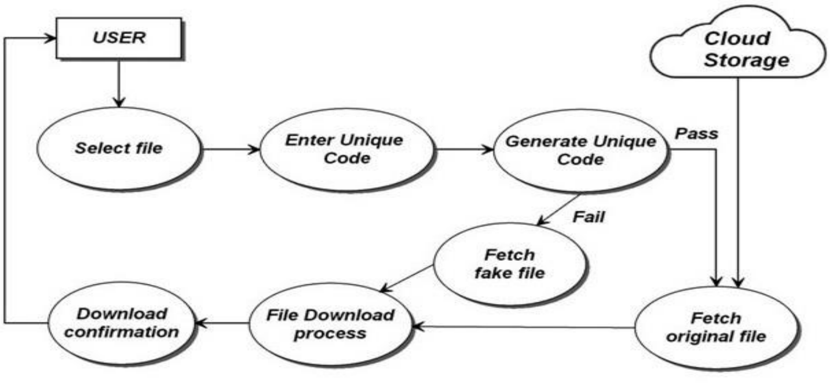


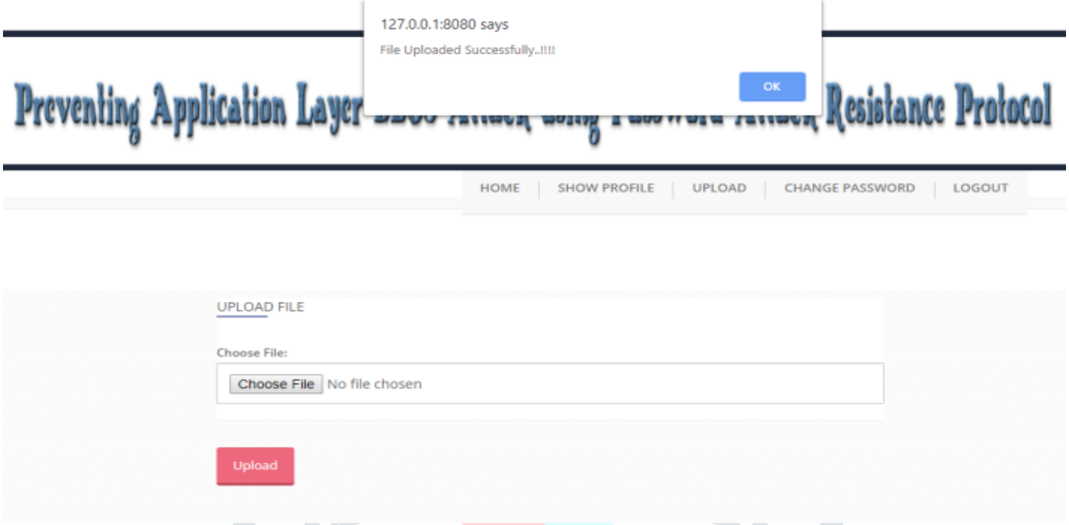
Fig 4: File Downloading Process

Fig. 4 shows how user can download the files from the cloud with security. In this case the user needs to select the file which they want to download from the cloud, and then the user can use the same unique code which is generated and sent to their mail during uploading process using that code only they can be able to download the files from the cloud. If in case user enters correct generated unique code while uploading but fails to enter the same unique code while downloading then also user will receive the file but it doesn't contain the same information as in the original file.

IV. Experimental Analysis

In this section we describe the results of how untrusted user tries to access the trusted user's application and the solution to overcome that problem.

First user can login using login credentials provided by the admin. Once the user is authenticated user can upload their respective files and while downloading those files they need to provide the valid unique code which is sent to their mail id. Below figure show file uploading and downloading process respectively.



127.0.0.1:8080 says

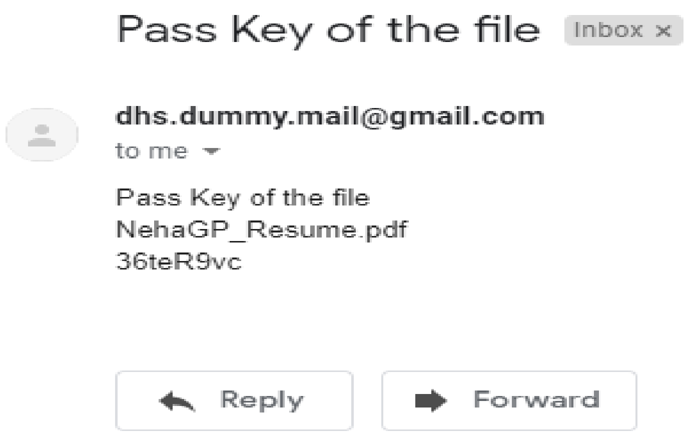
File Uploaded Successfully..!!!!

Preventing Application Layer ... Resistance Protocol

UPLOAD FILE

Choose File: No file chosen

Fig 5: File uploading



Pass Key of the file

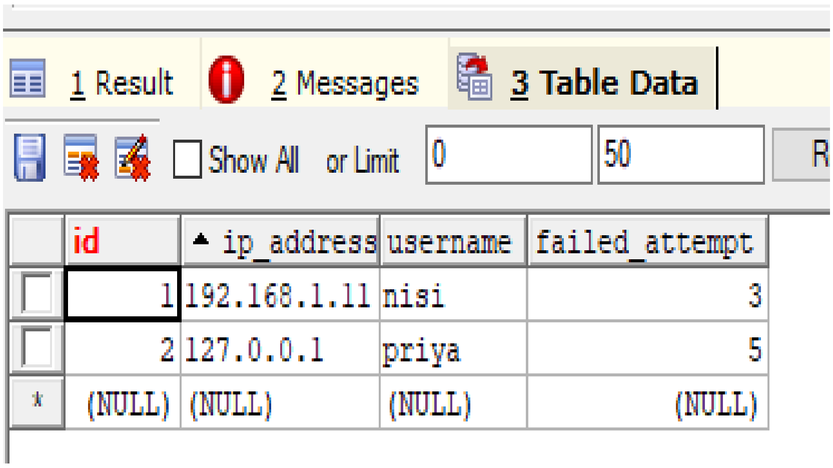
dhs.dummy.mail@gmail.com

to me

Pass Key of the file
NehaGP_Resume.pdf
36teR9vc

Fig 6: File Downloading

Second, If user is able to enter the correct user id and fails to give the correct password then a threshold is set to determine whether user is a Trusted user or not. From the below figure second user is a valid user and if the user forgets the password then the threshold is set to 5 based on this threshold user is given a chance to login by entering the correct password, if in case user forgets password then the user becomes eligible for CAPTCHA test. If the user is success in the test then user can upload and download files to and from the cloud, then the user will be considered in the whitelist based on the system's IP address which is considered during registration process.

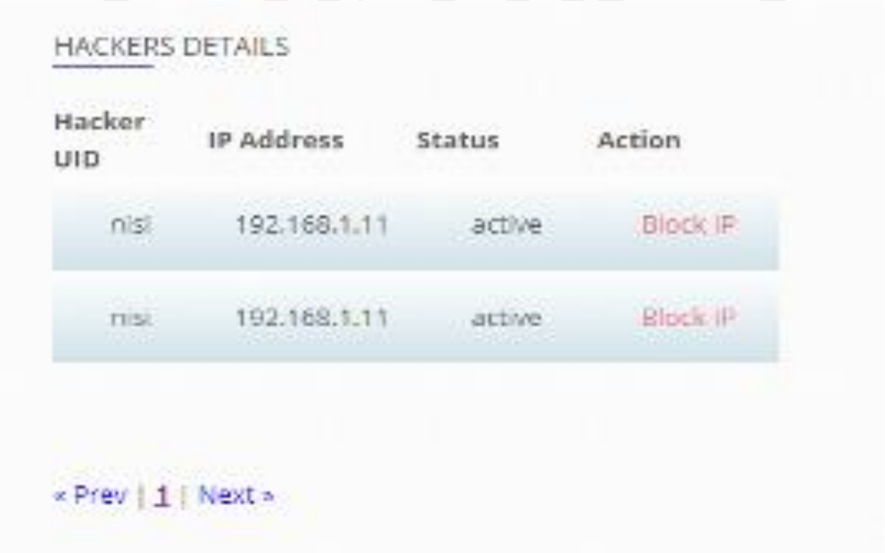


	id	ip_address	username	failed_attempt
□	1	192.168.1.11	nisi	3
□	2	127.0.0.1	priya	5
*	(NULL)	(NULL)	(NULL)	(NULL)

Fig 7: Untrusted and Trusted user

Third, If the login credentials given by the user are incorrect then user is considered as untrusted. From the above figure first user is trying to access the second user's account but the user doesn't know the password so the threshold is set to 3 based on this threshold user is given a chance to login by entering the password, if user forgets password then the user becomes eligible for CAPTCHA test. based on this user need to take the CAPTCHA test and if user success in the CAPTCHA test then also user will be considered in the blacklist and cannot be able to enter into the login page this is based on the trusted user System IP address which doesn't match the IP address which is used during the registration process and the trusted user gets the mail saying someone is trying to access your account.

.


HACKERS DETAILS

Hacker UID	IP Address	Status	Action
nisi	192.168.1.11	active	Block IP
nisi	192.168.1.11	active	Block IP

Fig 8: Untrusted User's Details

In this above figure we describe how IP address is blocked when untrusted user is trying to access the trusted users account. We have developed a prototype of Preventing Application Layer DDoS attacks using Java and evaluated it on a 64-bit Windows 10 system with an Intel (inside) Core (i5) and 8.0GB RAM.

V. Conclusion

This study discusses an overview of how to defend against application layer DDoS attacks, it is essential to have a fast response system that can automatically detect and mitigate malicious requests as soon as possible. In this paper, we design and implement such a system named Honey Pot Technique which collects the sample of malware for the purpose of analysis and identification of attacks. And used in the field of security. It helps in knowing the behavior of the attack and attacker information. In this case user can upload and download the file to or from the cloud storage. Whenever user is uploading a file, for each file, it will generate a unique code. And that Unique code will be sent to user through email. Whenever user is downloading the file, User has to give the corresponding unique code to the respective file. If the user has given the correct code then only, user will be able to get the original file. If user is giving the guessing code or some other wrong code, it will not show the error message but, it gives a duplicate file to the user.

VI. REFERENCES

[1] P. Barford, J. Kline, D. Plonka, and A. Ron, “A signal analysis of network traffic anomalies,” in Proc. IMC, 2002, pp. 71–82.

[2] J. Tang, Y. Cheng, and C. Zhou, “Sketch-based sip flooding detection using Hellinger distance,” in Proc. GLOBECOM, Nov./Dec. 2009, pp. 1–6.

[3] J. Tang, Y. Cheng, Y. Hao, and W. Song, “SIP flooding attack detection with a multi-dimensional sketch design,” IEEE Trans. Depend. Sec. Comput., vol. 11, no. 6, pp. 582–595, Nov. /Dec. 2014.

[4] O. Salem, S. Vaton, and A. Gravey, “A scalable, efficient and informative approach for anomaly-based intrusion detection systems: Theory and practice,” Int. J. Netw. Manage., vol. 20, no. 5, pp. 271–293, 2010.

[5] H. Liu, Y. Sun, and M. S. Kim, “Fine-grained DDoS detection scheme based on bidirectional count sketch,” in Proc. ICCCN, 2011, pp. 1–6.

[6] S. Kandula, D. Katabi, M. Jacob, and A. Berger, “Botz-4-sale: Surviving organized DDoS attacks that mimic flash crowds,” in Proc. NSDI, 2005, pp. 287–300.

[7] J. Rangasamy, D. Stebila, C. Boyd, and J. G. Nieto, “An integrated approach to cryptographic mitigation of denial-of-service attacks,” in Proc. ASIACCS, 2011, pp. 114–123.

[8] X. Liu, X. Yang, and Y. Xia, “NetFence: Preventing Internet denial of service from inside out,” in Proc. SIGCOMM, 2010, pp. 255–266.

[9] H. Wang, Q. Jia, D. Fleck, W. Powell, F. Li, and A. Stavrou, “A moving target DDoS defense mechanism,” Comput. Commun., vol. 46, pp. 10–21, Jun. 2014.

[10] O. Joldzic, Z. Djuric, and P. Vuletic, “A transparent and scalable anomaly-based DoS detection method,” Comput. Netw., vol. 104, pp. 27–42, Jul. 2016.

[11] S. Sivabalan and P. J. Radcliffe, “A novel framework to detect and block DDoS attack at the application layer,” in Proc. TENCON, 2013, pp. 578–582.

[12] Y. Li, T.-B. Lu, L. Guo, Z.-H. Tian, and Q.-W. Nie, “Towards lightweight and efficient DDoS attacks detection for Web server,” in Proc. WWW, 2009, pp. 1139–1140.

[13] S. Yu, W. Zhou, W. Jia, S. Guo, Y. Xiang, and F. Tang, “Discriminating DDoS attacks from flash crowds using flow correlation coefficient,” IEEE Trans. Parallel Distrib. Syst., vol. 23, no. 6, pp. 1073–1080, Jun. 2012.

[14] T. Thapngam, S. Yu, W. Zhou, and S. K. Makki, “Distributed denial of service (DDoS) detection by traffic pattern analysis,” Peer-Peer Netw. Appl., vol. 7, no. 4, pp. 346–358, 2014.

[15] S. Ranjan, R. Swaminathan, M. Uysal, A. Nucci, and E. Knightly, “DDoS-shield: DDoS-resilient scheduling to counter application layer attacks,” IEEE/ACM Trans. Netw., vol. 17, no. 1, pp. 26–39, Feb. 2009.