International Journal of Emerging Technology and Advanced Engineering
Website: www.ijetae.com (ISSN 2250-2459 (Online), An ISO 9001:2008 Certified Journal, Volume 3, Special Issue 1, January 2013)

International Conference on Information Systems and Computing (ICISC-2013), INDIA.

ADAPTIVE DEFENSE STRATEGY: IMMUNIZING SHARED CHANNEL NETWORK FROM DOS ATTACKS

Arshey.M[1], C.Balakrishnan[2]

[1]PG Scholar, Computer Science &Department, S.A Engineering College, Chennai ,Tamil Nadu;
[2]Assistant Professor, Department of PG Studies, S.A Engineering College, Chennai ,Tamil Nadu.

Email: arshi.sm@gmail.com

Abstract

Denial of Service attacks is an attempt to make a machine or network resource inaccessible to intended users. This attack saturates the target machine with a lot of communications requests, in such a way that it cannot respond to traffic from legitimate requests. The Denial of service attacks is massive within the shared channel. In the shared network, the attack rates are large and the client request rates vary. The existing system does not make the client to adapt to the attack rates. In this paper, the server sets a hop sequence for the clients to transmit data packets. The designed system makes clients effectively and dynamically adapt to an attack by increasing their request rate. This is accomplished by utilizing client request time out windows as an indication towards real time attack rates. This does not require any server state or assumptions about network congestion. The server processes client request with high probability while pruning the attack by selective random sampling. Adaptive Selective Verification protocol aids for effective utilization of bandwidth. It is a defense mechanism to impede attackers' effort to deny service to legitimate clients. The goal is to alert the legitimate clients of the attack, eliminate hacker node from the client hop sequence and prevent denial of service to clients due to server overload.

***Keywords*-- Denial of Service, shared channel model, Adaptive Selective Verification**

I. INTRODUCTION

Denial-of-service (DoS) attacks pose an immense threat to the Internet and also the network systems, and there are also many defense mechanisms have been proposed to overcome the problem. Attackers try constantly to modify their attack methods to surpass the security systems and researchers in turn modify their approaches to handle such attacks. The DoS attack is quickly becoming more and more composite. There is variety of known attacks which creates the impression that the problem space is immense, and hard to explore and tackle. The existing systems employ various techniques to counter the problem, and it is difficult to understand their similarities and differences and to evaluate their effectiveness and cost.

The DoS attack aims to disrupt some legitimate activity, such as browsing web pages, or transferring money from bank account, This denial-of-service effect is achieved by sending messages to the target that interfere with its operation, and make it hang, crash, reboot, or do useless work.

The DoS attack dealt here is the application level flooding attack where the attacker tries to flood the network resources, causes a buffer overflow, consumes all available memory or CPU time. In this attack, the attacker sends large amount of packets saturating the bandwidth or tries to deplete the system resources.

The assumption is that the denial of service attacks is large within the shared channel network where legitimate sender and an attacker share a packet communication channel to a receiver. The shared channel model is more appropriate for analyzing denial of service where the attack packets and legitimate packets pass through the same channel.

These types of attacks were dealt by using the currency-based mechanisms, in which a server under attack demands some type of payment from clients in order to raise the bar for provoking work by the server beyond the capacity of the attacker. Classic currency examples in this context are bandwidth and CPU cycles [5]. This aims to limit attackers by making them sacrifice valuable resource like money or CPU cycles. The focus in this paper is on the bandwidth as currency. The clients are encouraged to send repeated requests which consume a lot of bandwidth and server selectively verifies some of the request. Thus the clients here are not able to adapt to the varying attack rates.

This paper concentrates on the adaptive defense strategy to impede the attacker's effort to deny service to legitimate clients. The defense mechanism concentrates on adaptive selective verification protocol [11] which makes the client to effectively and dynamically adapt to the varying attack rates. The client sends request based on the time out windows and the server performs a probabilistic pruning of the request and performs random sampling and acknowledges the request. This helps in using the bandwidth effectively.

II. RELATED WORK

The existing system deals with currency based mechanism [5]. The related work of dealing with DoS attacks is the Capability-based mechanism. In this the legitimate clients are allowed to prove their legitimacy to gain priority [9]. The counter-intuitive currency based mechanism is the use of bandwidth as payment. In this the clients try to spend excess bandwidth to get access. The main aim is that the attackers use all the available bandwidth to execute an attack but on the other hand the legitimate clients use only the resources available with them to accomplish their task. This fact helps to differentiate attackers from the legitimate clients [1].

In [2] selective verification with bins is implemented which allows clients to send extra requests and the server selects the request from them by a probabilistic approach. The idea is illustrated in **Fig 1** where the requests from the attacker and client requests are mixed together and filtered before being addressed by the server. Bandwidth auction is yet another strategy which allows clients to build credit by sending bytes to an accounting system and the server takes requests from clients that have built the most credit. Bandwidth auctions provide a natural adaptive mechanism since clients send packets until they are serviced.

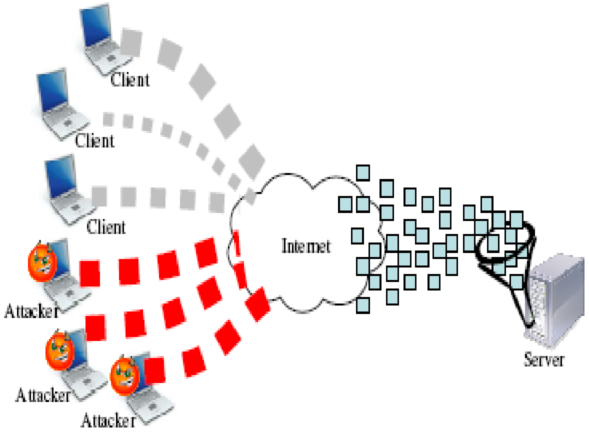

Fig. 1. Selective verification

All the mechanism provided above are not adaptive approach to the arising DoS attacks. [3] Provides ideas that are effective for filter schemes and that which are adaptive to the varying attack rates. This scheme generates alarm based on the attack rates. This does not give an idea about dealing with the bandwidth payments. [13] Shows how to make use of the information available in the application layer to differentiate between low and high utility clients so as to provide better service to legitimate clients. [15] proposes an adaptive solution for installing router throttles in the network.

The main focus in this paper is to deal with the network resource attacks and the server resource attacks. These types of attacks mainly concentrate on using up of the network and the server resources.

III. PROBLEM STATEMENT

The proposal here is to prevent the DoS attacks which use packet floods to consume network and server resources. The types of attack can be classified as Network Resource attack and Server Resource attack.

Network Resource Attack: In this type of attack, the attacker sends many useless packets to the victim server with the aim of exhausting the network bandwidth connecting to server. If the attack succeeds, the network bandwidth is successfully depleted. This causes the legitimate servers to experience complete service degradation because the packets are unable to reach the server.

Server Processing Attack: In this the attacker sends useless packets with the aim of devasting the victim server ability to process the increased load of packets. The server is thus forced to drop the incoming packets and the legitimate users experience service degradation or failure.

To deal with these types of attack, the adaptive protocol can be used in which the client is made to dynamically adapt to the varying attack rates by using the time out windows up to a threshold. The server implements a reservoir- based sampling to effectively sample from the sequence of incoming packets. This helps effectively in reducing the usage of bandwidth and also the server processing overhead.

IV. SYSTEM ANALYSIS

1.1. *Existing System*

Currency-based mechanisms are ones in which a server under attack demands some type of payment from clients in order to raise the bar for provoking work by the server beyond the capacity of the attacker. Classic currency examples in this context are bandwidth and CPU cycles. Our focus in this paper is to use bandwidth as the currency. In order to get service, the clients are encouraged to spend more bandwidth by either sending repeated requests from which the server selectively verifies some or dummy bytes on a separate channel to enable a bandwidth auction. The selective verification algorithm requires no server state, and does not implement any adaptive mechanism.

1.1.1. *Drawbacks Of The Existing System*

- There is a high probability of exposure to attack in the shared channel network since the legitimate clients and attackers share the same channel.
- Clients are allowed to send repeated requests to server till an acknowledgment is received. So they send repeated requests and are not concerned if there is an attack or not. They are not able to dynamically adapt to attack. Thus there is increased bandwidth usage. Server overhead increases due to flooding of request packets.
- Clients assume that the attackers send requests at a certain rate and clients always try to send requests at a rate more than the attacker rate. Thus knowledge of attack rates is a pre requirement.
- Server does not perform any node verification. It chooses request randomly and processes them.
- Client does not really know whether there is attacker in the network or not.

1.2. *Proposed System*

The proposed system uses a protocol which is highly adaptive to the varying attack rates. The protocol used in this paper is the Adaptive Selective Verification (ASV) which is a distributed adaptive mechanism for impeding attackers' efforts to deny service to legitimate clients. This scheme uses bandwidth as currency. The level of protection employed by the clients is that they dynamically adjust to the current level of attack rates. At a high level of the attack, the clients exponentially ramp-up the number of requests they send in consecutive time-windows, up to a threshold which is maintained at the client. The server implements a reservoir-based random sampling to effectively sample from a sequence of incoming packets using bounded space.

This enables adaptive bandwidth payments with server state whose size remains small and constant regardless of the actions of the attacker. Thus the main aim of the proposed system is to intimate the legitimate clients about attackers in the network thereby not allocating the bandwidth to the attacker. This helps to reduce the bandwidth usage and also reduces the server overhead.

1.1.2. *System Architecture*

The **Fig 2** represents a shared channel network model which consists of a server and n nodes sending request packets to the server in which one of the nodes is a hacker node.

The server performs a node credibility verification of all the clients which are connected to the network and allots the hop sequence for the request to be transmitted to server. The Server maintains a database with the hop sequence for each node. The server on identifying the hacker alerts the other legitimate clients about the hacker in the particular node and updates the hop sequence. The clients also perform node verification before transmitting the request packet to the next hop sequence. This verification results in eliminating the attacker node from the hop sequence and thus prevents the flooding of the channel.

1.1.3. *The Adaptive Mechanism*

The Adaptive Selective Verification (ASV) protocol is a cost-based, DoS-resistant-protocol in which bandwidth is the currency. ASV imagines the shared channel model as its fundamental attack model. The key idea of the protocol is for clients to spend more bandwidth to compete with attacker's bandwidth usage, and for the server to selectively process incoming requests. A client attempts to adapt to the current level of attack by exponentially replicating its requests up to a threshold as the severity of attack increases. The server implements a reservoir sampling algorithm to collect a random sample of the incoming requests and process them at its mean processing rate.

1.1.4. *ASV Protocol Of The Clients*

The client first identifies the attack rate and adaptively increases the number of Request that it sends in succeeding time out window.

1. *Initialize count:* Set x to 0 and let X be the threshold based on the logarithmic calculation of the maximum attack rate of client and attacker.
2. *Double count of the Request:* Send 2^x Request packets to the server.
3. *Check for Time out:* If no ACK packet is received within time T seconds, set x to x+1; if an ACK packet is received, exit the protocol and proceed to the next phase of communication.
4. *Iterate:* If x is less than the threshold X, go back to step 2; else exit without communicating with the server.

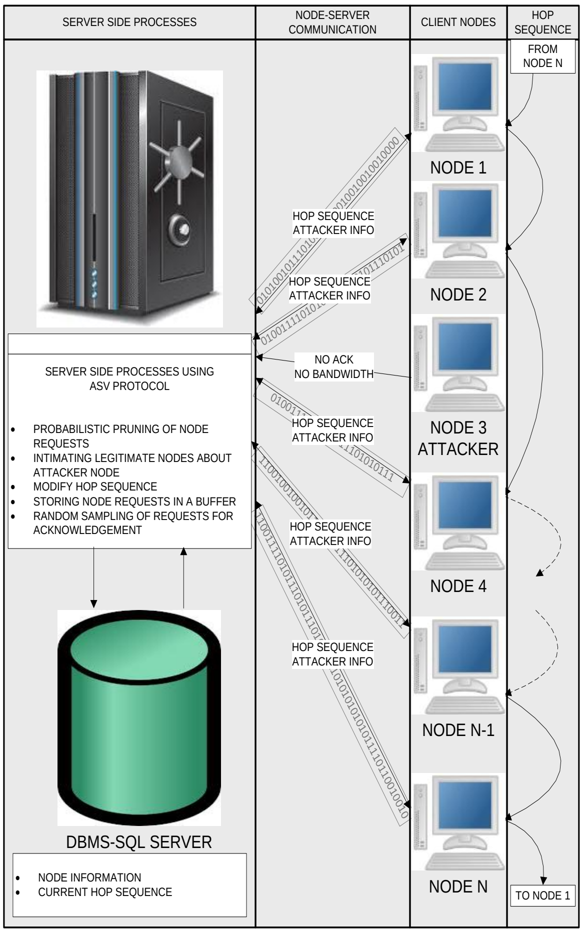
SERVER SIDE PROCESSES
NODE-SERVER COMMUNICATION
CLIENT NODES
HOP SEQUENCE
FROM NODE N
NODE 1
NODE 2
NODE 3 ATTACKER
NODE 4
NODE N-1
NODE N
TO NODE 1
HOP SEQUENCE ATTACKER INFO
HOP SEQUENCE ATTACKER INFO
NO ACK NO BANDWIDTH
HOP SEQUENCE ATTACKER INFO
HOP SEQUENCE ATTACKER INFO
HOP SEQUENCE ATTACKER INFO
SERVER SIDE PROCESSES USING ASV PROTOCOL
PROBABILISTIC PRUNING OF NODE REQUESTS
INTIMATING LEGITIMATE NODES ABOUT ATTACKER NODE
MODIFY HOP SEQUENCE
STORING NODE REQUESTS IN A BUFFER
RANDOM SAMPLING OF REQUESTS FOR ACKNOWLEDGEMENT
DBMS-SQL SERVER
NODE INFORMATION
CURRENT HOP SEQUENCE

Fig.2. System Architecture

1.1.5. *ASV Protocol Of The Server*

The server performs reservoir sampling on incoming REQ packets during each time-out window. The server processes a random subset of the arriving requests at a rate not exceeding the given amount of all packets per second.

1. *Initialize window count:* Set y to 1.
2. *Establish reservoir:* Store the Request packets arriving in window in a reservoir. If reservoir is not filled up before the time out, go to step 4. Else, set Request packet count to point to the next request.
3. *Random Sampling:* If there is an incoming REQ numbered x, accept it for placement into the reservoir with a probability and discard the others. Set x to x+1 and iterate till the time out window expires.
4. *Check for time-out:* Accept the packets in the reservoir and send acknowledgement to all the accepted Requests.
5. *Iterate:* Empty the reservoir and set the window count y to y+1 and return to step 2.

V. SYSTEM IMPLEMENTATION

1.3. *Shared Channel Model*

In the shared channel model, legitimate clients and the attackers share a packet communication channel to the server by transmitting and receiving packets through the same channel. This model is more appropriate for analyzing denial of service attacks. Here the individual client interacts with the server and establishes a connection. The server maintains the list of the clients connected their port number, their node name and their IP address in its database.

1.4. *Hop Sequence (Unidirectional)*

The server sets a unidirectional hop sequence for the client node to transmit the packets. The server assigns each client its node to which it has to transmit data. If any client deviates from the set hop sequence, the server identifies that the node is an adversary and tries to remove the node from the network. The legitimate clients always follow the hop sequence set by the server to transmit the data packets.

1.5. *Node Verification And Data transmission*

The client node before transmitting the data packets to next client node verifies with the server's hop sequence to identify if the node is legitimate. Similarly a client node before receiving the data packets from other nodes also verifies its legitimacy. The server always monitors the data transmission which occurs in its network. The server is updated about the time of sending the packet to and receiving the packets from the client nodes so as to identify any delay in data transfer.

1.6. *Attack Intimation*

The hacker node is the one which does not allow other legitimate clients to make usage of the resources available with the server. Here the hacker tries to modify the hop sequence set by the server thus not allowing the legitimate clients to get the resources from other clients. The server, on identifying the hacker in a particular node, alerts all the other legitimate clients by sending an alarm message to them. By this the clients can understand that there is a hacker in their communication network.

1.7. *Protocol Implementation*

The adaptive selective verification is then implemented for the server and the client. The client sends a single request at a time and if it does not get acknowledgement within the time period, it starts sending message at an exponential rate.

The server maintains a buffer where it stores the entire client request. The server then performs a probabilistic pruning of the requests and performs random sampling of the request to be processed. The protocol implemented helps to impede the attacker's effort.

1.8. *Calculating Lifetime Of Attacker*

The server, on identifying the attacker in the network sets a timer to determine the length of time the attacker persists in a particular node. The server identifies the time of entry of the attacker and the time the attacker exists from the network. By identifying this server can allocate the node to legitimate clients so that node can be used for forwarding the packets and thus the server adds the entry of that node in the hop sequence.

VI. CONCLUSION AND FUTURE ENHANCEMENT

ASV is one of the mechanisms by which DoS can be prevented. In this, the clients exponentially increase the number of requests they send in successive time out windows, up to a threshold. The server implements a reservoir based random sampling to effectively sample from a sequence of incoming packets Attackers are effectively recognized by probabilistic methods. This also provides an effective utilization of available bandwidth.

The adaptive protocol can be extended by encrypting the data packets before transmitting it to the next node. By determining the life time of an attacker in the node the attack pattern can be identified and thus can prevent the network from the denial of service attacks.

Acknowledgements

I thank my project coordinator Mr. Muthukumaraswamy M.E, my project guide Mr. C.Balakrishnan M.E (PhD), who are members of faculty with the Department of Computer Science and Engineering, S.A Engineering College, without whose guidance, this paper would not have been possible. I also wish to record my thanks to our Head of the Department Mrs.Umarani Srikanth M.E (PhD) for her consistent encouragement and ideas. I would like to express my gratitude to all those who helped me make this paper a reality and gave me the opportunity to publish this paper.

REFERENCES

1. Journal Papers

[1] Charalampos Patrikakis, Michalis Masikos, and Olga Zouraraki National technical university of athens "distributed denial of service attacks" the internet protocol journal - volume 7, number 4

[2] Akash Mittal1, Prof. Ajit Kumar shrivastava, Dr. Manish manorial. a review of ddos attack and its countermeasures in tcp based networks International journal of computer science & engineering survey (ijcses) vol.2, no.4, november 2011

[3] Nirbhay Ahlawat1and Chetan Sharma2 Classification And Prevention Of Distributed Denial Of Service Attacks. International journal of advanced engineering sciences and technologies vol no. 3, issue no. 1, 052 – 060

[4] Georgios Loukas and Gülay Öke Protection Against Denial Of Service Attacks: A Survey. Computer Journal Volume 53, Issue 7 Pp. 1020-1037.

[5] B. B. Gupta, , R. C. Joshi, and Manoj Misra Distributed Denial of Service Prevention Techniques International Journal of Computer and Electrical Engineering, Vol. 2, No. 2, April, 2010 1793-8163

2. Text Books

[1] Internet Denial of Service: Attack and Defense Mechanism by Jelena Mirkovic, Sven Dietrich, David Dittrich, Peter Reiher.

[2] Hacking Exposed: Network Security Secrets &Solutions, 5th Edition by Stuart McClure, Joel Scambray and George Kurtz.

[3] Communication Networks by S.Hekmat.

3. Conference proceedings

[1] C. A. Gunter, S. Khanna, K. Tan, and S. S. Venkatesh. "DoS protection for reliably authenticated broadcast". In NDSS'04: Network and Distributed System Security Symposium. The Internet Society, 2004.

[2] M. Sherr, M. B. Greenwald, C. A. Gunter, S. Khanna, and S. S.Venkatesh. "Mitigating DoS attacks through selective bin verification." In IEEE ICNP Workshop on Secure Network Protocols (NPSec).

[3] C. C. Zou, N. Duffield, D. Towsley, and W. Gong. "Adaptive defense against various network attacks." In IEEE Journal on Selected Areas in Communications,

[4] R. Mahajan, S. M. Bellovin, S. Floyd, J. Ioannidis, V. Paxson, and S. Shenker, "Controlling high bandwidth aggregates in the network," Comput. Commun.

[5] D. Mankins, R. Krishnan, C. Boyd, J. Zao, and M. Frentz. Mitigating distributed denial of service attacks with dynamic resource pricing. In ACSAC '01: Proceedings of the 17th Annual Computer Security Applications Conference.

[6] Gul Agha, Carl Gunter, Michael Greenwald, Sanjeev Khanna, Jose Meseguer, Koushik Sen, and Prasanna Thati. " Formal modeling and analysis of DoS using probabilistic rewrite theories".

[7] C. Jin, H. Wang, and K. G. Shin. "Hop-count filtering: an effective defense against spoofed DDoS traffic ." In CCS '03: Proceedings of the 10th ACM conference on Computer and communications security, ACM Press.

[8] A. Yaar, A. Perrig, and D. Song, "Pi: a Path Identification Mechanism to Defend against DDoS Attacks," Proc. IEEE Symp. Security and Privacy, May 2003.

[9] X. Yang, D. Wetherall, and T. Anderson. "A DoS-limiting network architecture". In SIGCOMM '05: Proceedings of the 2005 conference on Applications, technologies, architectures, and protocols for computer communications, ACM Press.

[10] Che-Fn Yu and V. D. Gligor. "A specification and verification method for preventing denial of service".IEEE Trans. Software. Eng., 16(6):581{592, 1990.

[11] Sanjeev Khanna, Santosh S. Venkatesh, Omid Fatemieh, Fariba Khan, and Carl A. Gunter. “Adaptive selective verification.” In IEEE Conference on Computer Communications (INFOCOM '08).

[12] D. Moore, G. Voelker, and S. Savage. Inferring internet denial of service activity. In Proceedings of the 10th USENIX Security Symposium.

[13] M. Srivatsa, A. Iyengar, J. Yin, and L. Liu. A middleware system for protecting against application level denial of service attacks. In Middleware.

[14] A. Yaar, A. Perrig, and D. X. Song. SIFF: A stateless internet flow filter to mitigate DDoS flooding attacks.

[15] D. K. Y. Yau, J. C. S. Lui, F. Liang, and Y. Yam. Defending against distributed denial-of-service attacks with max-min fair server-centric router throttles.

4.Generic Websites

[1] A crime research on the various DoS attacks by Terrance A. Roebuck http://www.crime-research.org/articles/network-security-dos-ddos-attacks/ (Accessed date: 8-Aug-2012)

[2] Prevention of DoS attacks in organization networks. http://www.esecurityplanet.com/network-security/how-to-prevent-dos-attacks.html (Accessed date: 25-Aug-2012)

[3] Denial of Service attacks http://www.denial-of-service-attacks.com/(Accessed date : 9-Sep-2012)

[4] Defeating distributed denial of service in a step by step fashion. http://www.sans.org/dosstep/(Accessed date: 20-Oct-2012)

[5] A report on global DDoS attack statistics for 2012 from Prolexic. http://www.prolexic.com/(Accessed date : 20-Oct-2012)