RESEARCH ARTICLE OPEN ACCESS

Impact Analysis of Denial of Service (DoS) due to Packet Flooding

Dr. Deepak Arora[1], Puneet Singh[2] And Vineet Singh[3]
Department of Computer Science & Engineering Amity University, Lucknow

Abstract
In the current scenario, it is noticed that when a server reached to its threshold value then there is situation of packet flooding. Flooding can arise in two ways first when many user make request to the server at same instant of time and other when a single user makes a huge number of requests intentionally to jam the network traffic for that particular server. This intentionally arise situation is called as Denial of Service (DoS) attack. DoS attacks are gigantic threat to internet users and websites. Distributed Denial of Service (DDoS) is an improvised approach for restricting the legitimate users to use intended service. DDoS attacker uses software vulnerabilities of other computer system on the network in order to setup the attacking network for the victim.
In this paper, the focus will be on the behavior of server or victim machine regarding various parameters under DoS attack by using LAN network simulated on OPNET Modeler. It provides model families, that can be used for simulation and record various parameters as realistic value.
Keywords:- DoS Impact, Opnet, Flooding, Retransmission, Delay

I. Introduction

An attack like Denial of Service (DoS) has posed a great threat to the internet resources. In DoS, attacker focuses on the disruption of services running on a machine or service not to be utilized by any other internet user.[1] A DoS attack is achieved by sending data packets in huge amount in order to jam the network traffic of victim machine and victim is unable to process the legitimate user's request. DoS attack holds one to one dimension with attacker and victim which makes it easy to resolve this attack. Whenever attacker implements DoS attack, attackers IP address is spoofed so that the detection of attacker isn't easy. Ongoing deep the DoS attack can be classified to various levels as network level or OS level that includes software bug or exhausting hardware resources [1]. On the basis of no. of machines involved as attacker DoS attack can be classified as: DoS and DDoS. If single system is involved in attacking for making services inaccessible then it is considered as DoS.

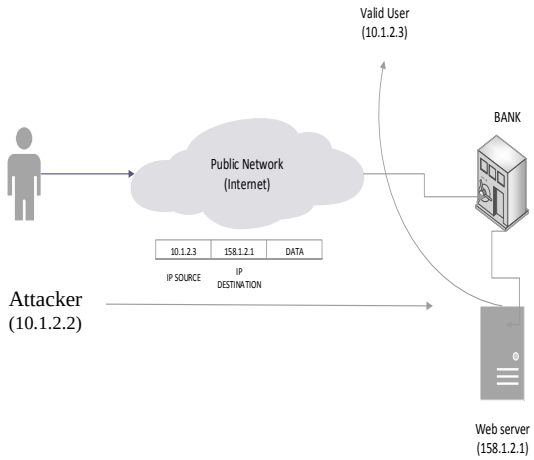

Fig 1. DoS Attack

A Distributed Denial of Service (DDoS) is an improvised approach to make the resources unavailable. When a group of people or computers perform the same task with same intention of blocking the service provider to process legitimate service request by flooding the channel with unusable request packets then the attack scenario is called DDoS. As compared to DoS, DDoS holds many to one dimension that makes the prevention from attack more difficult. For every attack there is a prevention and after prevention a new approach to implement the attack, and hence there are various attacks to implement DoS on network at various layers of network model along with their prevention methods [3].

An approach for prevention of DoS is provided that uses concept of neural network for detection of faulty packets [6].

V.Priyadharshini, Dr.K.Kuppusamy has proposed algorithm for the defense against DDoS attack on the server by filtering packets on the basis of genuine IP address. New algorithm for security works on ideology of creating history of IP addresses visiting the server and IP addresses are matched with blocked list [5]. Checking the route rate of messages requested is helpful in query flood attack's prevention [8]. Yet there exists technology leads to modify the protocols by identifying ideology of attacker so that these attacks can be spoiled. It has been seen that without proper secure protocol the attack can't be avoided, that attack could be proven hazardous [7]. Chang-Lung Tsai, Allen Y. Chang and Huang Ming-Szu has clearly mentioned how these attacks (DoS and DDoS) are implemented and their other essential characteristic.[9]

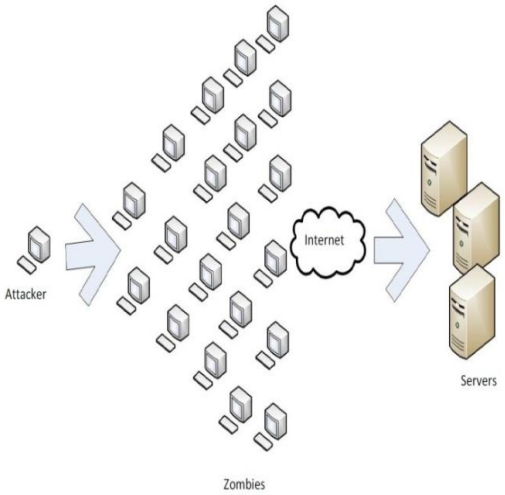

Fig 2. DDoS Attack

DoS attack is intentional approach to prevent access of network resources. An attacker or hacker regularly insert attack program on insecure machine. The compromised machines are commonly called as Master / Handlers or Zombies which collectively called as Bots and attack in network is called as Botnet [2]. DDoS attack can be classified as Bandwidth depletion or resource Depletion. A bandwidth depletion attack is designed to flood the victim network with unwanted traffic that prevents legitimate traffic from reaching the victim system. Bandwidth attacks can be divided to flood attacks and amplification attacks. [3] DoS attack can be implemented by various ways on different layers such as jamming and node destruction on physical layer and spoofing, Replay and Hello floods on network layer [3]. For implementing these attacks there are various tools present and that are compared and shown how these attacks can even lead to crash of nodes [4]. A lot of work is discussed for implementation or prevention of this attack but during the attack how a victim behaves in terms of network resources, which will be focused in this work.

II. Background

The whole project is designed on the basis of simulation. For this project simulation is very cost effective idea, hence OPNET tool is used. It is one of leading research oriented tool in industries. Due to Graphical user interface this tool is very easy for configuring and designing the networks. It provides editors to edit the given devices and design for any custom configuration. Any protocols, packet formats and other network related codes can be done with OPNET. For any scenario simulated using this tool, there is ease in recording the various parameters with the imaging the problem as of real time and produces the results in Graphs. The obtained result can be presented in any image format such graphs, bar charts, histogram etc. It provides the component in hierarchical manner on top network followed by nodes then process at the bottom. At each hierarchical level an editor is provided so that a required network can be designed using various topologies. To design the network various networking devices which act as nodes are provided along with the connecting links. For connecting two nodes, specific link will be used. All the link and nodes have their own properties which make them different from others and present the scenario as of real time. To test that network devices are correctly connected NetDoctor is used. It checks the designed network and if there is an error, it rectifies the network so that required network can be designed easily for simulation. If any specific node with specific configuration is needed then it can be designed using existing nodes it helps in creating new devices by providing internal structure and test them without any cost. If any set of devices are modified with new feature they can be saved as model family and used in other projects by adding that model in project. Random traffic can be created in order to check the performance of the created network. If it is needed for a device to perform a required task then using process editor a node can be updated with new features such as protocols, algorithms etc in form of codes added as process in required device. Various scenarios are contained in same problem and to know the effect of variation of a single parameter a new scenario can be rebuild and the results of required scenarios can be compared to know the effect on network for the variation of single parameter. In a single project various scenarios can be designed. It helps at the result processing time. It can be easily configured that for how long a simulation should run to achieve the target effect. In this project LAN

Technology is used, as simulation is done to know attack's impact not to implement the attack, so it is assumed that security measure kept has been successfully breached.

III. Experimental Setup

A small network has been setup based on client server architecture in Opnet Modeler 14.5. For setting up the whole network a model family is being used known as "Network Security labs". It is collection of workstations, routers, firewalls, Ethernet server, switch, internet and config objects. Users and compromised hosts both are workstations. In the designed network no additional traffic is created. For performing the operations User (genuine service request) and compromised hosts or zombies are those which are making huge requests in order to implement attack. In different scenarios the user is using different applications (HTTP, FTP, Database) and bots are implementing attack using the same application. Profile objects are configured so that all the hosts request to server in required manner. Two profiles have been designed one for the user known as User profile which supports application profile named user. The user profile is for genuine user. The second is Zombie profile which support attack application. In different scenario the profile supports different application. During HTTP application both users and bots are using image browsing, with object size of 1000 bytes and 10KB, 50KB respectively. With FTP application, user and bots are uploading files, with file size of 1000 bytes and 10KB, 50KB respectively in various scenarios. The inter-request time for user and bots in above applications are 1 second and 0.1 second. For Database application transactions are executed by user and bots. The inter-request time, inter-transaction time for user in all the application will be 1 second, where for bots its value will be 0.1 second. The size of object, file and transaction for user will be 1000 bytes and for bots it will be 10KB and 50KB.

All the 5 zombies machines are using same zombie profile due to which all of them attack target in serial manner and no. of zombies(CH) remain constant i.e. 5 in all scenarios.

All the users and zombie workstations are connected to internet from client side. On server side there is a router which is connecting the server to the internet. The server is set to its default configuration. The server is highly efficient with default data forward rate of 100,000 packets / second. In order to determine the performance of server, its efficiency has been reduced to three different values of 100, 50 and 15 packets per second.

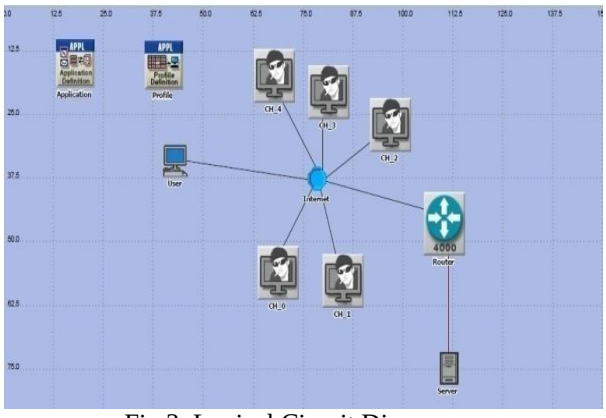

Fig 3. Logical Circuit Diagram

The whole simulation is run for 1 hour and after constant (1) second of start of simulation, user profile is activated and it remains active till end of simulation. After constant (300) seconds of simulation passes Zombie profile get activated for constant (2000) seconds. There exist nine more scenarios called as Baseline one for every server efficiency and application, in which there is no zombie only user is making requests.

IV. Results and Discussion

As discussed how various scenarios are designed, what constraints are imposed during simulation of attack. In this section the impact of attack will be drawn regarding various parameters.

(A) Traffic Drop:

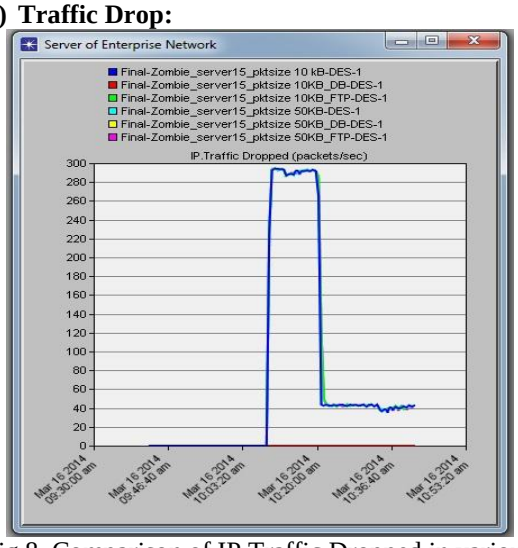

Fig 8. Comparison of IP Traffic Dropped in various cases

There can be various reasons for dropping a packet by a node.

This can happen due to increase in no. of packets arriving at server with pass of time. As unprocessed packets are waiting in queue due to reduced server efficiency, its memory got consumed and traffic drop

is noticed. Keeping the server efficiency constant then among all three applications HTTP and FTP application shows similar behavior as there is no effect on increasing the size of page and file. Although there is no traffic dropped in scenario with Database application for both transaction sizes. The traffic drop can also occur due to collision of packets as a result of congestion in network. As no traffic is generated in any scenario, so drop in Database application is not noticed due to existence of priority queue. On increasing the server efficiency and keeping the application and size of page and file constant, it is found packet drop before the time than that of lowest efficiency and range of packet drop also increases first then show downfall. Due to increase in server efficiency the channel allocation for request and response packets are decreased and increased respectively due to which congestion in incoming packets increases which lead to earlier traffic drop with increased efficiency scenario.

(B) Memory Free size:

It shows how quicker the server memory got consumed. Among the various scenarios, a scenario with HTTP and FTP application shows similar behavior and pattern in consuming the server memory. As Zombie profile gets active the packets start arriving the server in quick manner that leads to decrease in amount of forwarding memory free size. As soon as the graph reaches X axis the server got out of memory and start dropping the incoming packets. For Database application the graph remains a straight line parallel to X axis which indicates traffic drop can't be achieved using Database application.

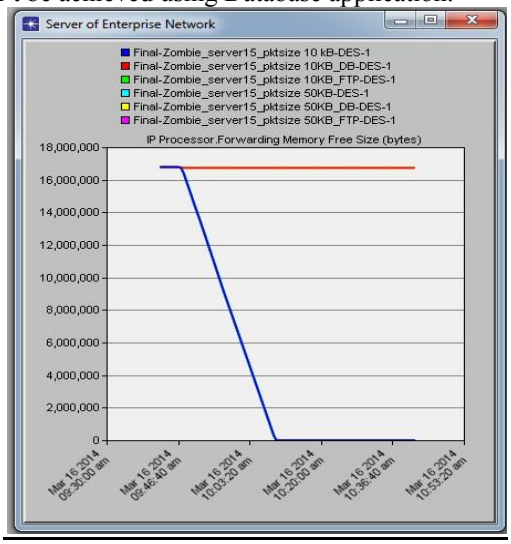

Fig 5. Comparison of IP Processor Forwarding Free Memory in various cases

On increasing the server efficiency and keeping the application and size of page and file constant, it is found that the rate of consumption of memory first increases and then decreases.

(C) CPU Utilization:

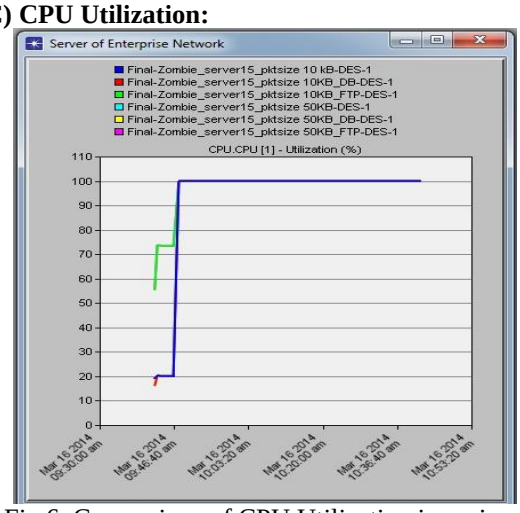

Fig 6. Comparison of CPU Utilization in various cases

As the server efficiency was reduced to 15 packets per second, the CPU utilization remains constant when there is no attack but as the attack is implemented the no of packets start arriving in large amount at server and get stored buffer memory that makes CPU of server busy and is utilized up to 100%. When FTP application is used for the attack by bots, CPU utilization of the server starts increasing from 70% and goes on increasing with time in random manner and become constant at 100%. In Database application and HTTP application parameters value increases from 20% to top value within no time. This indicates due to increase in the no. of packets arriving at the server and getting buffered the CPU of server is busy in processing the data. But if the server efficiency is increased and keeping other constraints unmodified then initial point value keeps decreasing.

(C) **IP Processing Delay:** The delay can be faced if packets received are waiting for a long time in queue to be processed.

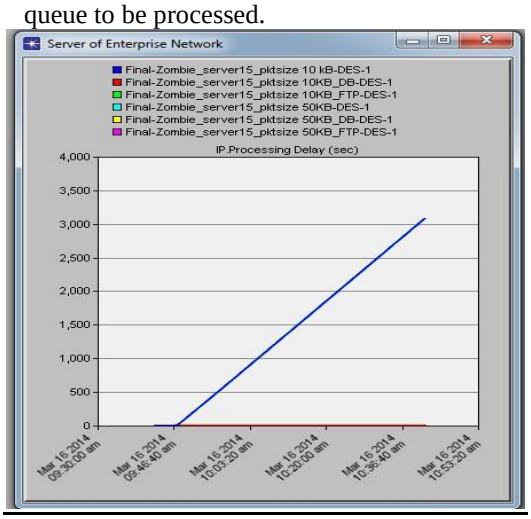

Fig 7. Comparison of IP Processing Delay in various cases

If the server efficiency is kept constant then it is noticed that in scenarios with Database application for both transaction size the delay is very less and nature of its graph is curve which increases with time. While on other hand, FTP and HTTP application show similar behavior for processing delay on different file and page size. On increasing the server efficiency for same application and file size it is found that there is decrease in inclination of graph for IP Processing Delay which indicates delay rate decreases.

(E) TCP Retransmission Count: TCP Retransmission count indicates the generated response is not received on the client side and packet is destroyed due to collision with other packet in heavy network traffic or packet face timeout. On considering the Database application there is decrease in average retransmission count with time and it become constant throughout the simulation. Large transaction size scenario shows little more retransmission count as compared to smaller transaction size scenario. On other hand HTTP and FTP application shows a large retransmission count as compared to Database but on comparing these two it is noticed that FTP application is little quicker in retransmitting the unacknowledged packets than that of HTTP. The retransmission count is little higher for FTP, yet nature of graphs are same for these two applications. First start increasing and then become constant.

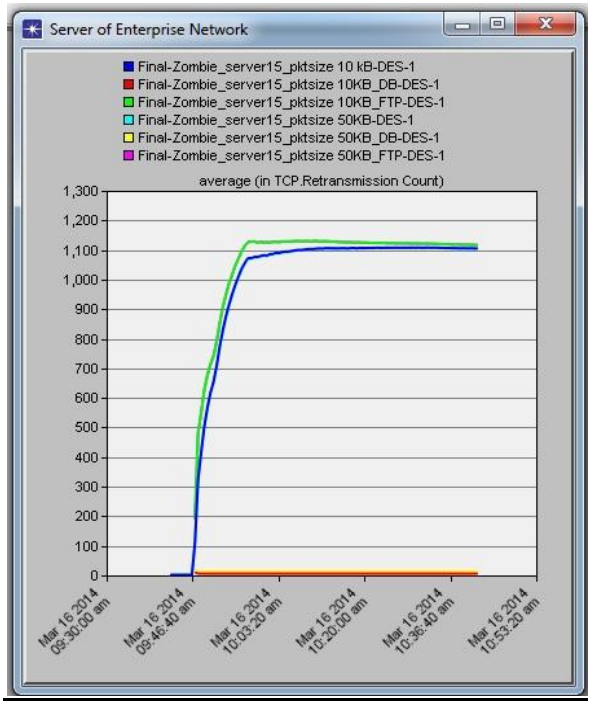

Fig 8. Comparison of average TCP Retransmission count in various cases

On increasing the server efficiency and keeping application and size of page, file constant it is found that with increase in server efficiency the retransmission count get increases and become constant later on.

V. Conclusion

On implementing the DDoS attack with different applications commonly used it can be said that attack can be easily implemented by using HTTP and FTP application but not with Database application. As traffic drop is noticed in both the applications except scenarios with Database one. File and page size doesn't show any effect in traffic drop. Considering server's forwarding Memory it can be said that Database application doesn't consumes much memory where as HTTP and FTP application can do it very easily. In this also File size and page size doesn't play any important role. Due to heavy no. of requests in all application scenario the CPU utilization is 100% that clears server is facing a great load and that makes CPU busy. In every application delay is noticed but the delay with HTTP and FTP applications are much higher as compared to Database application. Each application is facing problem in sending responses to its users but Database application user are getting response quicker than that of HTTP and FTP applications.

VI. Acknowledgement

We are very thankful to their respected Mr. Aseem Chauhan, Chairman, Amity University, Lucknow, Maj. Gen. K.K. Ohri, AVSM (Retd.), Pro-Vice Chancellor, Amity University, Lucknow, India, for providing excellent computation facilities in the University campus. Authors also pay their regards to Prof. S.T.H. Abidi, Director and Brig. U.K. Chopra, Deputy Director, Amity School of Engineering, Amity University, Lucknow for giving their moral support and help to carry out this research work.

REFERENCES

[1] Douligeris, C. and Mitrokotsa, A. (2009). *DDoS attacks and defense mechanisms: a classification*.

[2] Bhange, A. et al. (2012). *DDoS Attacks Impact on Network Traffic and its Detection Approach*. In Proc of International Journal of Computer Applications. ISSN 0975–8887 Vol. 40 No.11.

[3] Krishna Chaitanya, D. and Arindam, G. (2007). *Analysis of Denial-of-Service attacks on Wireless Sensor Networks Using Simulation*. Middlesex University.

[4] Arun Raj Kumar, P. and S. Selvakumar (2009). *Distributed Denial-of-Service (DDoS) Threat in Collaborative Environment- A Survey on DDoS Attack*

Tools and Traceback Mechanisms. In Proc of IEEE International Advance Computing Conference (IACC 2009) Patiala, India, 6-7 March 2009.

[5] Priyadharshini, V. and Kuppusamy, K. (2012). *Prevention of DDoS Attacks using New Cracking Algorithm*. International Journal of Engineering Research and Applications (IJERA), Vol. 2, Issue 3, May-Jun 2012, pp.2263-2267 ISSN: 2248-9622.

[6] Chonka, A., et al (2010). *Chaos Theory Based Detection against Network Mimicking DDoS Attacks*. IEEE COMMUNICATIONS LETTERS Vol. 13, No. 9, SEPTEMBER 2009.

[7] Aad I., et al. *Impact of Denial of Service Attacks on Ad Hoc Networks*.

[8] C. Hu, Y., et al (2002), *Ariadne: A secure on demand routing protocol for ad hoc networks*. In Proc of MobiCom 2002, September 2002.

[9] Lung Tsai, C., et al (2010). *Early Warning System for DDoS Attacking Based on Multilayer Deployment of Time Delay Neural Network*. In Proc of Sixth International Conference on Intelligent Information Hiding and Multimedia Signal Processing.