International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395 -0056
Volume: 03 Issue: 11 | Nov -2016 www.irjet.net p-ISSN: 2395-0072

Survey on Cumulative Study on DOS attack prevention system in Web application using Software puzzle

Prof.Rupali.S.Shishupal[1], Sayli.P.Waghavale[2], Anuradha.D.Kalekar[3] , Jyoti.B.Ugile[4]

[1] *Professor, Dept. of Computer Engineering, Sinhgad Institute of Technology, Lonavala, Maharashtra, India*
[2,3,4] *Student, Dept. of Computer Engineering, Sinhgad Institute of Technology, Lonavala, Maharashtra, India*

Abstract – *Denial of Service (DoS) and Distribute Denial of Service (DoS) attacks are one of the major risk and among the hardest security problems in today's Internet. Which request a client to perform computationally very high operations before producing services from a server to client, is a well-known countermeasure to them. An attacker increase his puzzle solving ability by using construct in graphics processing unit hardware to extremely weaken the effectiveness of client puzzles. In this paper, analyze how to halt DoS/DDoS attackers from boost their puzzle-solving abilities. To this end, suggest a new client puzzle labeled as software puzzle. Unlike the existing client puzzle strategy, which publish their puzzle algorithms early and generate software puzzle for client request, a puzzle algorithm in the present software puzzle scheme is developed only after threshold value of client request top the CPU's request handling capacity by using technique like decision tree, fuzzy logic and generate algorithm such as: 1) an attacker is not able to prepare an implementation to solve the puzzle early and 2) the attacker needs extensive efforts required for translating a central processing unit puzzle software to its functionally identical GPU version.*

KeyWords: **Software puzzle,Code obfuscation,Distributed denial of service,Clustering**

1.INTRODUCTION:

Requests for a web transaction are received from all the clients by the web server with the parameter like date, time and client IP to save in the database. Then all this data from the database will be retrieved in a vector for preprocessing, where chosen data like IP is fetched in a single dimension vector for clustering process. Single dimension vector of IP addresses of the client that was fetched in the past step is been set to fuzzy- C Means clustering process. The clustered IP are then consider for their higher priority using the entropy sharing factor of Shannon information gain. Information gain is used to classify fluent IP address in the clusters which frequently affecting the web server for its performance. Further information is send to decision tree to finding DOS attack. Decision tree takes a two dimensional vector which is loaded with the attributes like IP address and their information gain values. Each of the indices of the vector is feed to the tree to form the nodes and at every levels of the tree with respect to the Shannon information gain values. Then these values are keep collecting the at the corresponding nodes to get the weighted decisions for judging attack level. Then this attack level is normalized in between the range 0 to 100 to get the desired level of softwarepuzzle. This level of attack is been send to proxy server for puzzle generation process with a reference key generate. Once the proxy server receives the attack level it determines the expression desired to tackle the attack in its raw form. Then the variables in the expression is set to change the variables by assigning random number from 1 to 9.Once the expression is having the real numbers, then it is been evaluated.

2.DOS AND DDOS Attack

2.1 DoS attacks

DoS attack [1] can be described as an attack create to render a computer or network incapable of producing normal services. A DoS attack is considered to take place only when access to a computer or network resource is intentionally blocked or degraded as a result of injurious action taken by another user. These attacks don't necessarily waste data directly or permanently, but they intentionally compromise the availability of the resources. The most common DoS attacks target the computer networks bandwidth or connectivity.

Bandwidth attacks overmuch the network with such a high volume of traffic that all available network resources are consumed and reliable user requests cannot get through, resulting in degraded productivity. Connectivity attacks overmuch a computer with such a high volume of connection requests, that all available operating system resources are absorbed and the computer can no longer process reliable user requests.

2.1.1 DoS attack classification

DoS attacks can be classified into five categories based on the attacked protocol level, as illustrated in fig [1].

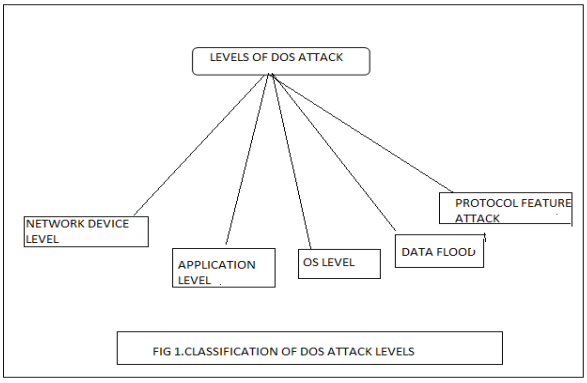

FIG 1.CLASSIFICATION OF DOS ATTACK LEVELS

DoS attacks in the Network device level include attacks that might be induced either by capture advantage of virus or weaknesses in software or by trying to overextend the hardware resources of network devices.

In the OS level DoS attacks take advantage of the ideas operating systems implement protocols. One example of this kind of DoS attacks is the Ping of Death attack. In this attack, ICMP echo requests having total data sizes greater than the maximum IP standard size are transmit to the targeted victim.

This attack often has the effect of crashing the victims machine. Application-based attacks try to settle a machine or a service out of order either by taking advantage of specific bugs in network applications that are running on the target host or by using such applications to drain the resources of their victim. In data flooding attacks, an attacker attempts to use the bandwidth available to a network, host or device at its greatest extent, by sending huge quantities of data and so causing it to process extremely large amounts of data. An attacker could attempt to use up the available bandwidth of a network by simply blasting the targeted victim with normal, but meaningless packets with spoofed source addresses. DoS attacks based on protocol features take benefit of certain standard protocol features.

2.2 DDoS Attacks

Distributed Denial of Service (DDoS) attacks[1]: ‘‘A DDoS attack uses numerous computers to launch a coordinated DoS attack against one or more targets. Using client/server technology, the enforcer is capable to multiply the effeciency of the DoS extremely by utilize the resources of multiple uninformed associate computers, which deliver as attack platforms’’. The DDoS attack is the most advanced mode of DoS attacks. It is illustrious from other attacks by its capability to extend its weapons in a ‘‘distributed’’ form over the Internet and to collective these effort to form fatal traffic. The main goal of a DDoS attack is to object suffering on a victim either for personal reasons, either for material gain, or for popularity.

2.2.1 DDOS Strategy

A Distributed Denial of Service Attack is form of four elements, as shown in actual attacker.

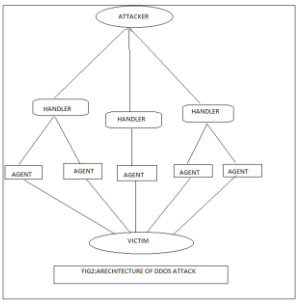

FIG2:ARECHITECTURE OF DDOS ATTACK

• The handlers or masters, which are arbitrated hosts with a special program running on them, efficientof controlling multiple agents.

• The attack daemon agents or zombie hosts, who are arbitrated hosts that are running a special program and are important for generating a stream of packets towards the affianced victim. Those machines are commonly external to the victims own network, to avoid efficient response from the victim, and external to the network of the attacker, to avoid burden if the attack is traced back.

• A victim or target host.

3.Currency-based DoS defence mechanisms

In currency-based DoS defense mechanisms[2] a server under attack demands some form of payment from all clients in order to raise the bar for provoking work by the server. In this section describe some of the current work on two classes of currency-based mechanisms: puzzle-based and bandwidth-based. Also describe resource fairness as a goal that currency mechanisms aim to achieve.

3.1 Puzzle-Based Mechanism

Puzzle based defense mechanisms such has try to correct the inequality between the cost to the attacker for generating are request and cost to the server for processing are request by demanding a payment, in the form of a puzzle solution, from each client. In a typical puzzle-based scheme, a request must be followed by a proof of payment from the client. The payment may be in the form of computation or memory accesses that the client needs to perform to solve the puzzle. Since the amount of resources available to the attacker is limited (even if it is much more than that of the legitimate

clients), the attacker will not be capable to exhausting magnify his attack.

3.2 Bandwidth-Based Mechanism

In a bandwidth-based currency scheme clients use additional bandwidth to get access. It is often assumed that attackers are using all of the bandwidth available to them to execute an attack, whereas normal clients are using only the resources they require to accomplish their less-demanding objectives. Hence normal clients have bandwidth to unused and can use this fact to reduce the attacker's chances of success. Two general scenario have been analyzed by researchers in this domain. In selective verification clients send extra requests and the server selects from them probabilistically. The extra requests serve as a mode of bandwidth payment that could adaptively change according to the severity of the attack. Bandwidth auctions allow clients to build credit by sending bytes to auction system from which the server periodically takes requests from clients that have built the most credit.

4.LITERATURE SURVEY

This section of literature survey eventually reveals some facts of "Enriched Framework For Software Puzzle For Avoiding Dos Attack on web server" based on analysis of many authors work as follow:

[3] Proposed idea of Client puzzles: A cryptographic countermeasure against connection depletion attacks as When server comes under attack , it distribute cryptographic spuzzle to client whom want service from server and it is having limitation of it requires special client side software and client already have a program capable of solving a client puzzle. where the author expresses the view of future enhancement is most robustness in stronger attack, capable of handling attacks mounted at very high speed. This protocol can be built straightforward or can be layered on top.

[4] Introduced idea of Reconstructing Hash Reversal based Proof of Work Schemes as DoS protection mechanisms which keep track of client behavior given the emerging threat of GPGPU based attacks. A server orders that the clients submit a proof of the work they have performed before processing their request. and it is having limitation as attacks use more resources, and the puzzle difficulties increases, weaker legitimate client may experience where the author expresses the view of future upgrade can successfully restrict a resource scaling attacker's capabilities by adjusting puzzle unnecessary requirements to obtain service puzzle difficulty based on past client behavior.

[5] Describes idea of Time-lock puzzles and timed-release crypto as narrates Encrypt a message it can't be decrypted by anyone, not even sender until a pre arranged time has elapsed and it is having limitation of the CPU time required to solve a problem can depend on the amount and nature of the hardware used to solve the problem and the parallelizability of the computational problem being solved. where the author expresses the view of future enhancement Computational problems that can't be solved without running a computer continuously for at least a certain amount of time, use trusted agents who don't reveal certain information until a specified date.

[6]Explained idea of mod_kaPoW: Mitigating DoS with transparent proof-of work as present mod kaPoW a novel system that has the efficiency and human transparency of proof-of work schemes as well as the software backwards compatibility of CAPTCHA scheme and it is having limitation of This technique has a overhead when processing files containing a variable number of URLs where the author expresses the view of future enhancement is transparent to the end users and gives backward compatible to end users. It doesn't extraordinary special client software. In the system, a web server powerfully changes URLs

[7] Narrate idea of Proofs of work and bread pudding protocol as a bread pudding protocol to be a POW such that the computational effort invested in the proof may be reused by the verifier to achieve a helpful, verifiable correct calculations. and it is having limitation of the highly computationally intensive operation of minting in the MicroMint strategy where the author expresses the view of future enhancement can achieve security goal and client pay for access to a resource by offering small amount of its computational power

5.CONCLUSION

Due to enormous growth of the internet and thereby its user most of the applications are suffering from the DOS attack problem. This eventually slow down the systems or sometimes even they cause server failures also, due to this businesses are suffering huge setbacks. To resolve this many systems are been using like captcha and simple mathematical equation solving etc. By studying many of the past work this paper learns the fact that no system is existed which is providing complete solution for prevention of the DOS attack. So Software puzzles are seems to be promising aspect towards this which can easily handle the long standing queues of the user's request.

REFERENCE

[1] C. Douligeris and A. Mitrokotsa, "DDoS attacks and defense mechanisms: Classification and state-of-the-art," Comput. Netw., vol. 44, no. 5, pp. 643–666, 2004.

[2] R. Shankesi, O. Fatemieh, and C. A. Gunter, "Resource inflation threats to denial of service countermeasures," Dept. Comput. Sci., UIUC, Champaign, IL, USA, Tech. Rep., Oct. 2010. [Online]. Available: http://hdl.handle.net/2142/17372

[3] A. Juels and J. Brainard, "Client puzzles: A cryptographic countermeasure against connection depletion attacks," in Proc. Netw. Distrib. Syst. Secur. Symp., 1999, pp. 151–165.

[4] J. Green, J. Juen, O. Fatemieh, R. Shankesi, D. Jin, and C. A. Gunter, "Reconstructing Hash Reversal based Proof of Work Schemes," in Proc. 4th USENIX Workshop Large-Scale Exploits Emergent Threats, 2011

[5] R. L. Rivest, A. Shamir, and D. A. Wagner, "Time-lock puzzles and timed-release crypto," Dept. Comput. Sci., Massachusetts Inst. Technol., Cambridge, MA, USA, Tech. Rep. MIT/LCS/TR-684, Feb. 1996. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.110.5709.

[6] E. Kaiser and W.-C. Feng, "mod_kaPoW: Mitigating DoS with transparent proof-of-work," in Proc. ACM CoNEXT Conf., 2007, p. 74.