International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395 -0056
Volume: 04 Issue: 02 | Feb -2017 www.irjet.net p-ISSN: 2395-0072

Cryptographic Countermeasure Against Prevention Of Dos and Distributed DOS Attack

Pawar Dipali[1], Shinde Sonali[2], Agawane Aakanksha , Prof. Smita Khot

student[1] Department of Computer Engineering ,D.Y.P.I.E.T. college Maharastra, India.

student[2] Department of Computer Engineering ,D.Y.P.I.E.T. college Maharastra, India.

student[3] Department of Computer Engineering ,D.Y.P.I.E.T. college Maharastra, India.

Abstract - *DOS and DDOS are the major problems in cyber security. DOS attack is action to make machine and network resource unavailable to its intended users. The point of such DOS attack is to overload the targeted server bandwidth and other resources. Thereby blocking the website or server by DDOS attack. Need of classification of DDoS attacks and DDoS defense mechanisms to defend Denial of Service attack are the major problem. Cyber security that allow a client to perform very expensive and vital operation.So to avoid this problem we are going to implement system called puzzle solving software. In this project whenever client send request to server then puzzle generator generate puzzle to requested client (authorized client).By solving that puzzle, client being granted service from server and the task of server is to check whether the puzzle solved correctly or not. If puzzle is not solve by client then access is not given to client. Sometimes performance of system will decreased by attacker to prevent this we implement algorithm such that an attacker is unable to solve puzzle in time.*

Key Words: *Software Puzzle, Denial of Service(DoS), Code Protection, GPU Programming, Distributed Denial Of Service (DDoS).*

1.INTRODUCTION

As todays internet becomes need of peoples life, without internet people can't live for long period. There is need to keep server available becomes more important. There is need to provide prevention from the unwanted user on the server., These type of users are sever threat to the system and it affects to the availability and reliability of Internet. Denial-of-service attack (DoS attack) is a cyber-attack where the attacker look for to make a machine or network resource unavailable to authorized users by temporarily or indefinitely interrupting services of a host connected to the Internet. Denial of service is typically accomplished by unavailable the targeted machine or resource with superfluous requests in an attempt to overload systems and prevent some or all legitimate requests from being fulfilled. A DoS attack is correspondent to a group of people crowding to the entry door or gate to a shop or business, and not letting legal parties enter into the shop or business, disrupting normal operations. This attack makes server temporarily unavailable and service connected to the internet and blocks the network. There are two types of DOS attack Local and Remote. DoS attacks target the network bandwidth or the connectivity. DDOS does not attack on the computer but it is intentionally want access to the personal information on the server. DDOS attack is different from the other attacks which is distributed in form and create network bandwidth. Mainly focus on a defense mechanism for the transport layer, particularly for the Transmission Control Protocol (TCP). Client puzzle protocol is essentially an end-to-end protocol, it can be readily implemented and integrated into TCP. This assumes that malicious client can solves the puzzle using legacy CPU resource only. However this condition not always true. Now a days multi-core GPU is integrated in the computer so attacker utilize CPU and GPU to pressure on computational capacity.

2. LITERATURE SURVEY

"Client puzzles: A cryptographic countermeasure against connection depletion attacks" this system, introduces a new approach that we refer to as the client puzzle protocol, the aim of which is to fight against connection depletion attacks. The idea is quite simple, when there isno witness of attack, a server accepts connections request normally, that is aimlessly. When a server comes under attack, it accepts connections selectively. In particular, the server gives to each client wishing to make a connection a unique client puzzle. A client puzzle is a quickly computable cryptographic problem formulated using the time, a server secret, and additional client request information. The server resource allocated to it for a connection, the client must submit to itself for a connection, the client must submit to the server a accurate solution to the puzzle it has been given. Client puzzle are deployed in union with conventional timeouts on server resources. Thus, while genuine client will experience only a small degradation in connection time when a server comes under attack, an attacker must have access to large computational resource to create breach in service. Cryptographic puzzles have been used for several task, such as fighting against junk e-mail, creating digital time capsules, and metering Web site usage[1].

"Reconstructing Hash Reversal based Proof of Work Schemes" this method , elaborated an idea of Proof of Work (PoW) mechanisms, in which a server request that clients prove they have done work previously it commits resources to their requests. Most PoW mechanisms are puzzle-based techniques in which clients solve processing thorough puzzles.As attacks use more resources, and therefore the puzzle difficulties increase, weaker legitimate clients may experience unacceptable requirements to obtain service. While computationally weaker clients would experience longer latencies during an attack, it would be extremely more functional than a protocol without the PoW based defense. Using Graphical Processing Units (GPUs) provides a powerful technique for launching resource inflation attacks. The attackers can use cheap and widely available GPUs to boost their ability to solve typical hash reversal based puzzles by a factor of more than 600. This paper is the calculation of Hash- Reversal PoW schemes in the presence of resource-inflated attackers. In this show that client-based adaptation is necessary for providing satisfactory service to genuine clients in this situation. Additionally, it show that an robust hash reversal PoW scheme based only on server load will fail to provide service, and can create a novel DoS attack against fair clients. Given these results, hash reversal PoW strategy proposed for DoS protection mechanisms should keep track of client behavior given the developing threat of GPGPU based attacks[2].

"Time-lock puzzles and timed-release crypto" this system narrate the notion of timed-release crypto where the goal is to encrypt a message so that it can not be decrypted by anyone, not even the sender, until a prearranged amount of time has passed. The goal is to send information into the future. We study the problem of creating computational puzzles, called time-lock puzzles that require a precise amount of time to solve. The solution to the puzzle reveals a key that can be used to decrypt the encrypted information. This approach has the obvious problem of trying to make CPU time and real time agree as closely as possible but is nonetheless interesting. The more computational resources might be able to solve the time lock puzzle more quickly, by using large parallel computers. Another approach is the puzzle doesn't automatically become solvable at a given time; slightly, a computer needs work continuously on the puzzle until it is solved[3].

"mod kaPoW: Mitigating DoS with transparent proof of-work" this technique described a approach of mod kaPoW system that has the efficiency and human transparency of proof-of-work strategy and also having the software backwards compatibility. There are several disadvantages of

using CAPTCHAs. One drawback is the user-interface problem they create; users with visual disabilities are unable to access content legitimately while natural users find it increasingly difficult to solve CAPTCHAs correctly as the images have become less readable in order to thwart sophisticated attacker that have developed automated solvers for simple CAPTCHAs. Another drawback is the static nature of the problems being given out. A proof-of-work scheme alters the operation of a network protocol so that a client must rebound their challenge along with a correct answer before being granted service. The challenge acts as a refine for clients based on their willingness to solve a computational task of varying difficulty. This paper describes the design, performance, and evaluation of a novel web based proof-ofwork system that provides the benefit of configurable PoW protocols in a portable manner. Unlike CAPTCHAs, the system is transparent toits users and supports backwards compatibility for traditional clients. The basic approach only requires changes to web servers and is similar to the URL rewriting approach employed by content-distribution networks such as Akamai. In the approach, the web server dynamically rewrites URL references by attaching a computational puzzle to them[4].

" Proofs of work and bread pudding protocols" this protocol introduces an idea of bread pudding protocol. Bread pudding is a dish that originated with the purpose of reusing bread that has gone stale. In the same manner, a bread pudding protocol to be reused by the verifier to achieve a separate, useful, and verifiable correct computation. In this paper, we deviate from the standard cryptographic aim of proving knowledge of a secret, or the truth of a mathematical statement. POW is a protocol not defined or treated formally, POWs have been defined as a mechanism for a number of security goals, including server access metering, construction of digital time capsules, uncheatable benchmarks and denial of service. This paper contributes bread pudding protocol to be a POW such that the computing effort invested in the proof may be harvested to achieve a separate, useful and verifiably correct computation. These POWs can serve in their own right as mechanisms for security protocols as well as harvested in order to outsource the Micro Mintminting operation to a large group of untrusted computational devices[5].

"Avoiding Permanent disabling of Wireless Sensor Network from the attack of malicious Vampire Nodes" this method explores resource depletion attacks, which permanently disable networks by draining node's battery power. Here we propose a new mechanism to alleviate the attack from our network. The open nature of the wireless links attracts many security threats to the network. These attackers compromise them with the link and deplete the battery energy resource. One such attack is Vampire Attack[6].

3. PROBLEM CONTEXT

In second section we explain our work briefly to set of methodology and Counter measures of DDOS attack.

3.1 Scenario

Denial of service is typically achieved by inflating the targeted machine or resource with superfluous requests in an attempt to overload systems and prevent some or all legitimate requests from being fulfilled. Clients are considered adversaries of varying maliciousness that need to be throttled accordingly. In general, a client's maliciousness is measured from the load they have placed upon the system in the past and their contribution to the current load. Server issuer creates and returns a work function to the client while aborting the connection to avoid storing per-client state. After receiving and solving the function, the client solver attaches both the function and solution when resending the request. If the challenge solution pair is valid, the server verifier allows the request to proceed; otherwise the request is denied and another work function may be sent. Issuing challenges and verifying answers must add minimal overhead; otherwise the system becomes a target for attack. Work must be bound to the client server connection to ensure that the specific client is throttled. The system must

resist precomputation and replay attacks for responsive real-time throttling. A GPU processor has fast but small shared memory. GPU contains man Streaming Multiprocessors (SMs) consisting of 'n' number of identical processing cores. A client wants to obtain a service; he/she sends a request to the server. After receiving the client request, the server responds with a puzzle challenge x. existing client puzzle schemes ,[7] is fixed and disclosed in advance, the puzzle is called a data puzzle or it is referred to as a software puzzle. The attackers intend to exhaust scarce resources, including memory or disk space ,CPU cycles, and bandwidth by generating too many requests. Such an attack is feasible because the attacker often pays very little for requesting a System replace the linear chain of pending and incomplete connection. Layout obfuscation, data obfuscation, control-flow obfuscation, and preventive transformation SOFTWARE puzzle, recap its rival GPU-inflated Denial of Service attack in advance. When a client wants to obtain a service, she sends a request to the server. Opaque predicates into a program to disturb and conceal the real control flow. An opaque predicate is a Boolean-valued expression whose value is known a priori to an obfuscator but is difficult for a obfuscator to deduce. Attacker can easily utilize the GPUs or integrated CPU-GPU to inflate his computational capacity. Existing client puzzle technique is not much effective over inflated attacks due increasing computational cost. Hacker amortize one puzzle solving task to hundreds of GPU cores if the client puzzle function is parallelizable, and simultaneously send request to graphical processing units to solve puzzles which function is not parallelizable. Focus on defending a server solely against a flooding attack. After sending Acknowledgment information removed to increase the attack efficiency the main parallelism strategy is used to reduce the total puzzle-solving time. Existing client puzzle scheme not support on some browser such internet explorer, Firefox etc. Scheme dynamically embeds client-specific challenges, transparently delivers server challenges and client responses. This state is created by applying algorithms. Secrete information maintained by the server that changes every minute. Client puzzles were also devised to help prevent Denial of service attacks on authentication protocol. When a server is under attack, it sends out a cryptographic puzzle for the user to solve before allocating resources for that client. Their difficult problems taken while creating the cryptographic puzzles and make it feasible by providing required information in finding the solution. Puzzle should be easy for the server to create and verify and for client it should be difficult to solve. If Server is not under attack then sometimes allowing access without solving puzzle. This puzzle is possible for an attacker to keep a table of known puzzle because puzzles are changing randomly. Server has to know what puzzles it has generated and which ones to verify. Now, DDoS attack is the most advanced form of the dos attack. DDOS distinguished from the other attack by its ability to deploy on the internet distributed form. DDOS attacker not trying to access user's machine they are trying to make server inefficient. Main goal of attacker is to damage on a victim either for personal reasons, either for material gain, or for popularity. They can TAKE advantage of the Internet architecture and internet is designed with functionality with security in mind. The attacker extracts the security holes and vulnerabilities of the user machines and attack code. Furthermore he tries to protect the code from discovery and deactivation. Software puzzle mechanism is introduced here to prevent GPU inflated DoS attacks.

CPU code block contains different instruction set that are required for puzzle generation. There are different activities performed during puzzle generation are presented in CPU code block. Algorithm code block contains all operations related to encryption algorithm and stores all mathematical operations. Obfuscation technique used for the code protection code obfuscation known as creation something which is simple and hard to understand. After deploying puzzle on the client browser which is solved by user, verified by the puzzle solver to save the server time.

4. SYSTEM ARCHITECTURE

Interaction between client and server is described as follows. Various graphical passwords schemes have been suggest as alternatives to text-based passwords. Research and experience have shown that text-based passwords are difficult to remember. Psychology studies have proved that the human brain is better at recognizing and recalling images than text. graphical passwords are intended to easily keep in mind and reducing the memory burden on users so we use graphical puzzle as follows.

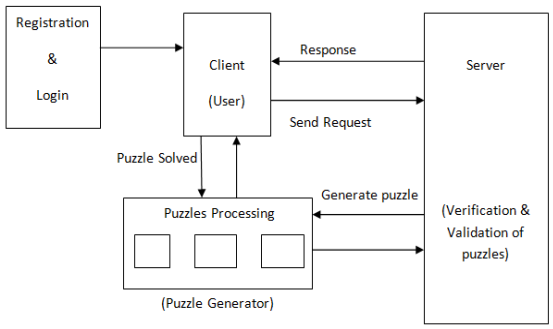

Fig. Software Puzzle Architecture

4.1 Types of Puzzles –

4.1.1 Sudoku:

In 9 x 9 Sudoku puzzles as they appear daily in our local newspaper. Although becoming fairly proficient in solving these, it still takes me at least thirty minutes to complete the typical 81 element Sudoku Puzzle. In one sense such efforts can be examine a waste of time since the time could be better spent on other endeavors. During these efforts (which can become addictive) come up with a new approach of looking at Sudoku problems in reverse. To generate Sudoku Squares we begin with a simple 4 x 4 case using just the four letters A, B, C, and D as our elements. We delegate the elements in the first row as A-B-C-D and the first column by A-C-B-D and then proceed to write the remaining elements using the four letters in cyclic fashion. The resultant generic square reads- As required, all rows and columns contain the base elements A, B, C, and D just once, as is also the case for the four sub matrixes. In any Sudoku Square of n*n total elements there will always be n sub-matrixes each containing the base letters just once. Thus for the present case of n=4, there are four sub matrixes. Similar to all other Backtracking problems, we can solve Sudoku by one by one assigning numbers to empty cells. Before assigning a number, we check whether it is safe to assign. We basically check that the same number is not present in current row, current column and current 4*4 grid matrix. After checking for safety, we assign the number, and recursively check whether this assignment points to a solution or not. If the assignment doesn't points to a solution, then we try next number for current empty cell. And if none of number (1 to 4) points to solution, we return false.

4.1.2 Cued Click Point:

Cued Click Points (CCP) is used for graphical password authentication. A password consists of one click-point per image for a sequence of images. The next image displayed is based on the previous click-point so users receive next implicit feedback as to whether they are on the correct path when logging in. CCP indicate both improved usability and security. Various graphical password schemes have been planned as alternatives to text-based passwords. Research has shown that text-based passwords are filled with both usability and security problems that make them less fetching solutions.

4.1.3 jigsaw

A jigsaw puzzle is a tiling puzzle that often requires oddly shaped interlocking and tessellating pieces. Each piece usually has a small part of a picture on it; when complete, a jigsaw puzzle generate a complete picture. In some cases more advanced types have appeared on the market, such as spherical jigsaws and puzzles showing optical illusions. Jigsaw puzzles were created by painting a picture rectangular piece of wood, and then cutting that picture into small pieces with a jigsaw, hence the name. Typical images found on jigsaw puzzles contain scenes from nature, buildings, and repetitive designs. Castles and mountains are two traditional subjects. However, any kind

of picture can be used to make a jigsaw puzzle, some companies examine to turn personal photographs into puzzles.

4.2 Software Puzzle Generation:

There are three steps in the puzzle core generation, puzzle challenge generation, software puzzle encrypting/obfuscating.

4.2.1 Puzzle Core generation:

Code block storeroom, the server head chooses n code chunks based on hash utilities and a secret key is the server's secret chunks are accumulated into a puzzle core.

4.2.2 Puzzle Challenge Generation:

Given certain secondary input messages such as IP addresses, and in line coefficients, the server computes a message m from open data such as their IP addresses, port numbers and cookies, and produces a Puzzle challenge alike to encoding plaintext m with key y to produce cryptograph text.

4.2.3 Puzzle Verification.

Returns a puzzle answer and if the server checks, the client is able to find the facility from the server. No secure station between the client and the server until puzzle confirmation conclusion.

5. APPLICATION

5.1.1. Online shopping site

5.1.2 0nline form filling

CONCLUSION

DDOS attacks represent a critical problem in the Internet and challenge its rate of growth and wide acceptance by the general public, skeptical government and businesses Software puzzle scheme is prepaid for destroying GPU inflated DOS attack. It hires software protection technologies to ensure challenge data confidentiality and code security for a respective time period, e.g., 10-20 seconds. It has different security requirement from the conventional cipher which demands long-term confidentiality only, and code coverage which focuses on long-term robustness against reverseengineering only. Since the software puzzle may be well-built upon a data puzzle, it can be composed with any existing server-side data puzzle scheme, and easily deployed as the present client puzzle schemes do.

REFERENCES

[1] 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.

[2] 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 Paper ID: ART20161447 281International Journal of Science and Research (IJSR) ISSN (Online): 2319-7064 Index Copernicus Value (2013): 6.14 | Impact Factor (2015):6.391 Volume 5 Issue 9, September 2016 www.ijsr.net Licensed Under Creative Commons Attribution CC BY Workshop Large-Scale Exploits Emergent Threats, 2011.

[3] 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.

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

[5] M. Jakobsson and A. Juels "Proofsof work and bread pudding protocols,"in Proc. IFIP TC6/TC11 Joint Working Conf. Secure Inf. N

[6] Rashmi Nayakawadi, Md Abdul Waheed, Rekha Patil "Avoiding Permanent disabling of Wireless Sensor Network from the attack of malicious Vampire Nodes", in International Journal of Advanced Scientific and Technical Research Issue 4 volume 3, May-June 2014