International Journal of Advanced Research in

Computer Science Engineering and Information Technology

Volume: 3 Issue: 1 26-Jun-2014,ISSN_NO: 2321-3337

DETECTING APPLICATION DENIAL-OF-SERVICE ATTACKS: A GROUP-TESTING-BASED APPROACH

[1] M.LAKSHMI [2]S.DURGAPRIYA

[1]Assitant Professor, Dept.of.Computer science,MCC college.Pattukottai.

[2]Research Scholar, Dept.of.Computer science, MCC college.Pattukottai.

ABSTRACT *−Application DoS attack, which aims at disrupting application service rather than depleting the network resource, has emerged as a larger threat to network services, compared to the classic DoS attack. Owing to its high similarity to legitimate traffic and much lower launching overhead than classic DDoS attack, this new assault type cannot be efficiently detected or prevented by existing detection solutions. To identify application DoS attack, we propose a novel group testing (GT)-based approach deployed on back-end servers, which not only offers a theoretical method to obtain short detection delay and low false positive/negative rate, but also provides an underlying framework against general network attacks.*

Keywords: virus, skip, filtering.

1. INTRODUCTION

More specifically, we first extend classic GT model with size constraints for practice purposes, then redistribute the client service requests to multiple virtual servers embedded within each back-end server machine, according to specific testing matrices. Based on this framework, we propose a two-mode detection mechanism using some dynamic thresholds to efficiently identify the attackers. The focus of this work lies in the detection algorithms proposed and the corresponding theoretical complexity analysis. We also provide preliminary simulation results regarding the efficiency and practicability of this new scheme. Further discussions over implementation issues and performance enhancements are also appended to show its great potentials

2.EXISTING SYSTEM

In Existing System, Denial-Of-Service (DoS) attack, which aims to make a service unavailable to legitimate clients, has become a severe threat to the network security. Traditional DoS attacks mainly abuse the network bandwidth around the Internet subsystems and degrade the quality of service by generating congestions at the network. Consequently, several network-based defense methods have tried to detect these attacks by controlling traffic volume or differentiating traffic patterns at the intermediate routers. However, with the boost in network bandwidth and application service types, recently the target of DoS attacks has shifted from network to server resources and application procedures themselves, forming a new application DoS attack. Malicious traffic is always indistinguishable from normal traffic, adopting automated script to avoid the need for a large amount of "zombie" machines or bandwidth to launch the attack, much harder to be traced due to multiple redirections at proxies. According to these characteristics, the malicious traffic can be classified into legitimate-like requests of two cases: 1) at a high inter arrival rate and 2) consuming more service resources. We call these two cases high-rate and high-workload attacks, respectively.

3.PROPOSED SYSTEM

We propose a novel group testing (GT)-based approach deployed on back-end servers, which not only offers a theoretical method to obtain short detection delay and low false positive/negative rate, but also provides an underlying framework against general network attacks. More specifically, we first extend classic GT model with size constraints for practice purposes, then redistribute the client service requests to multiple virtual servers embedded within each back-end server machine, according to specific testing matrices. Based on this framework, we propose a two-mode detection mechanism using some dynamic thresholds to efficiently identify the attackers. DDoS shield and CAPTCHA-based defense are the representatives of the two major techniques of system-based approaches: session validation based on legitimate behavior profile and authentication using human-solvable puzzles. By enhancing the accuracy of the suspicion assignment for each client session, DDoS shield can provide efficient session schedulers for defending possible DDoS attacks.CAPTCHA-based defenses introduce additional service delays for legitimate clients and are also restricted to human interaction services. This method only counts the number of incoming requests rather then monitoring the server status, it is restricted to defending high-rate DoS attacks and cannot handle high-workload ones.

4.MODULES

Node Details Declaration:

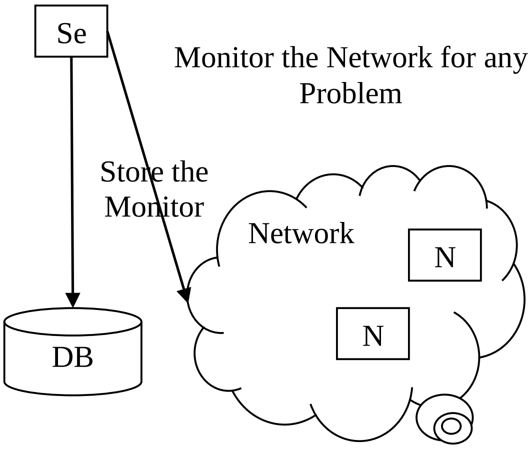
In node details declaration, the node is register to network topology. That is specified the node IP address, Port Number and status. Node login to the network topology while it check the user authentication Then only server system, allows the node in to the transmission .Node can send the packets to the destination or otherwise can send to server system. Node can add and relive is very easy in the network. Status also monitor by server system.

Server Creation:

In server creation, the centralized server system design for whole network. It has one centralized database and collect the details of each node. And store in to the centralized database. Server maintains these details, it very useful for node calculation and node details identification. Server can receive the request from all clients and the provide the corresponding response.

Server Monitoring:

In Server Monitoring, describe the Server monitoring, In Server monitoring if have any problem in network it will be take the action. The action is particular packet is discard and also the particular node details collect from database then that particular node remove from the network .Server system can identify the node by using the captcha. Monitoring process also detect the attacker node in the whole network. Monitoring result also store in the server side.



Captcha Generation:

In Captcha generation , each request notified by using this unique captcha. This captch unique for all system. Captcha has two parts one is node id and another one is process id. Each node has the node id as node name and port number combination. And each Process id started from the process name and combine with process count. It used for identify the node and type of process from DOS attacking node.

CONCLUSIONS

This paper is to apply group testing principles to application DoS attacks, and provide an underlying framework for the detection against a general case of network assaults, where malicious requests are indistinguishable from normal ones. For the future work, we will continue to investigate the potentials of this scheme and improve this proposed system to enhance the detection efficiency

REFERENCES

[1] Chieh-Jen Cheng, Chao-Ching Wang, Wei-Chun Ku, Tien-Fu Chen , and Jinn-Shyan Wang, “Scalable High-Performance Virus Detection Processor Against a

Large Pattern Set for Embedded Network Security" Commun. vol. 51, pp. 62–70,2011.

[2] O. Villa, D. P. Scarpazza, and F. Petrini, "Accelerating real-time string searching with multicore processors," Computer, vol. 41, pp. 42–50,2008.

[3] D. P. Scarpazza, O. Villa, and F. Petrini, "High-speed string searching against large dictionaries on the Cell/B.E. processor," in Proc. IEEE Int. Symp. Parallel Distrib. Process., 2008, pp. 1–8.

[4] D. P. Scarpazza, O. Villa, and F. Petrini, "Peak-performance DFA based string matching on the Cell processor," in Proc. IEEE Int. Symp. Parallel Distrib. Process., 2007, pp. 1–8.

[5] L. Tan and T. Sherwood, "A high throughput string matching architecture for intrusion detection and prevention,"in Proc. 32nd Annu. Int. Symp. Comput. Arch., 2005, pp. 112–122.

[6] S. Dharmapurikar, P. Krishnamurthy, and T. S. Sproull, "Deep packet inspection using parallel bloom filters," IEEE Micro, vol. 24, no. 1, pp.52–61, Jan. 2004.

[7] R.-T. Liu, N.-F. Huang, C.-N. Kao, and C.-H. Chen, "A fast string matching algorithm for network processor-based intrusion detection system," ACMTrans. Embed. Comput. Syst., vol. 3, pp. 614–633, 2004.

[8] F. Yu, R. H. Katz, and T. V. Lakshman, "Gigabit rate packet pattern matching using TCAM," in Proc. 12th IEEE Int. Conf. Netw. Protocols, 2004, pp. 174–178.intrusion detection system," ACMTrans. Embed. Comput. Syst., vol. 3, pp. 614–633, 2004.

[9] R. S. Boyer and J. S. Moore, "A fast string searching algorithm,"Commun. ACM, vol. 20, pp. 762–772, 1977.