ISSN No. 0976-5697

Volume 6, No. 2, March-April 2015

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

A Theoretical Model for Real-Time Resource Monitoring for Securing Computing Infrastructure against DoS and DDoS Attacks

Emmanuel C. Ogu[1], Idowu S.A.[2], Adesegun O.A.[3]
Department of Computer Science and Information Technology,
School of Computing and Engineering Sciences,
Babcock University, Ilishan-Remo, Ogun State. Nigeria.

Abstract **–**The challenge of denial of service (DoS) attacks and its distributed (DDoS) variants have immensely clogged the pathway of growth and development of the Internet and its reliant technologies, as well as computing infrastructure in general. This type of attacks have gradually carved a niche for itself as one of the most obnoxious forms of attacks to computing infrastructure in recent times. Many existing techniques for detecting, and mitigating the impact and extent of damage of this kind of attacks already exist. Most of them focus on monitoring and classifying every traffic that goes through the network as either "genuine" or "malicious". However, due to the speed and overwhelming pressure of this attack, it is becoming increasingly difficult for most of these techniques to stand in the face of real-world attacks. This research proposes a more resource-centric technique for monitoring computing resources against DoS and DDoS attacks which focuses on monitoring the rate of consumption of critical computing resources, in real-time, by various processes, tasks and traffic that bother on them, creating room for prescribed actions to be taken in order to forestall full DoS and DDoS breaches before they occur.

Keywords **–** DoS, DDoS, Computing Infrastructure, Real-Time Monitoring, Cloud Computing

I. INTRODUCTION

With the successful delivery of the first message over the Advanced Research Projects Agency Network (ARPANet) at the peak of the American Cold War, it was instilled in the minds of the over 1000 witnesses present that a technological revolution was soon to be born. The possibility of remote access to files was confirmed. On January 1, 1983, with the successful establishment of the first TCP/IP communication, the internet was birthed. [1]

In the early years of the Internet, there became a massive interest and intrigue in trying to see whether systems and communication infrastructure could be brought down. For instance, an attacker might want to get control of an IRC channel by performing a Denial of Service (DoS) attack against the channel owner just out of "intrigue"; or try to get recognition in underground hacker communities for taking down popular web sites [2] [3]. Prospects hint that this interest evolved into the various cybercrimes we now know in the field of Information Technology and the Internet.

In those early years, there were not many penalties for such crimes, so these were perpetrated at will, either for 'interest', 'intrigue', or petty revenge. [4] Because the Internet was welcomed and found its first use in the hands of researchers and academicians, the issue of cybercrimes was not so much to bother about. But as the use of the internet grew over the years, it gradually moved from the domain of research and academics to playing vital roles in healthcare and medicine, as well as in the workings of various governments and economies. This meant that the issue of Cybercrimes had moved beyond being a mere issue of individual wrongdoings to matters of national security. [5] [6] [7] [8]

A 2002 Computer Security Institute/Federal Bureau of Investigation (CSI/FBI) survey on Computer Crime and Security found that 90% of respondents (who were majorly large corporations and government agencies) detected computer security breaches within the last 12 months. The report documented that 80% of respondents acknowledged financial losses due to computer breaches, a total of approximately $455,848,000 in financial losses, rising from $377,828,700 reported in 2001. Respondents citing their Internet connections as a frequent point of attack rose from 70% in 2001 to 74% in 2002 [9] [10]. In recent times, Cybercrimes in the United States alone have been estimated to cost losses of up to $100 billion annually [11].

A Denial of Service (DoS) attack is a class of cybercrime attacks in which an attacker of a group of attackers attempt to cripple and online (Internet-reliant) service [12]. This crippling effect is usually achieved by flooding computing infrastructure and resources with useless or malicious network traffic, processing requests or bogus data such that the infrastructure and its resident resourcesbecome either too busy attending to or trying to ward off such requests, traffic or data that it is unable to process legitimate requests from users who have subscribed to use these resources; or unable to bear the load of the flood, and goes offline [13].Computing resources that are referred to include CPU time, memory space, and network bandwidth [14].

In the distributed variant of this attack, known as the Distributed Denial of Service (DDoS) attack, an attacker may seek the assistance of the computing resources of various compromised machines; and by remote central control and coordination, the attacker is able to harness the computing resources of this various compromised sources and channel them in the direction of the target machine or server with the flooding it as in a regular DoS attack. This practice of distributed coordination greatly amplifies the natural strength and fatality of a DDoS attack and greatly complicates the task of detection and defence [15].

However, what gives DoS and DDoS attacks the reputation they now have is that most DoS attacks target and consume resources rapidly at the network and transport layers (and also at the infrastructure layer through the Application-Layer DoS that is currently gaining rapid popularity), where it is difficult to authenticate whether an access, data, packet or connection is genuine or illegitimate and malicious [16].

It is important to note, however, that DoS does not only occur as a result of malicious attacks. A phenomenon known as flash crowds could also result in a DoS. Flash Crowds occur when a large crowd of legitimate users try to gain access to a server resource or service at the same time [15]; and this, as a matter of fact, goes a long way to further complicate the task of detecting and controlling DoS attacks because flash crowd traffic and DoS attack traffic have certain characteristics in common (such as their impacts on the network and server resources), and distinguishing them under the rush and load of DoS traffic can be a really difficult task.

As an example, a flash crowd DoS could occur when the website or servers of a big and popular university begins to experience intermittent unavailability during periods of admissions and registration as a result of the massive influx of new, prospective and returning students to the site for various reasons.

II. PROBLEM STATEMENT

The Internet has today evolved to be man's most indispensable resource; garnering controlling powers in various aspects of human endeavours. Security has been a nagging challenge with an exponential growth in its devastating power. Denial of service (DoS) attacks have become the nemesis of the Internet and a bane of various other technologies, business and operations that are reliant of the Internet. Having been around now for almost two decades [5], DoS attacks have grown to become the most popular and most dreaded of cybercrimes by businesses and it only keeps getting worse, leaving great losses and damages in its wake for businesses and organizations.

This problem has been further exacerbated by free availability of many sophisticated tools that are easy to use even for unskilled users, which can be used to gain root access to other peoples' machines. Once a machine is cracked, it becomes a "zombie" under the control of a "master" machine that is usually operated by the attacker. The attacker can then instruct, through the master, all its zombies to flood a particular destination. This simultaneous resulting traffic can clog links, servers, and cause routers near and around the victim, or the victim itself to fail under the overwhelming load [12].

A. Classes Of Dos Attacks:

[17], identified two general classes of DoS attacks:

a. Network-Layer DoS attacks – in this class of attacks, attackers send large bogus packets towards the victim server to overwhelm them, normally using IP spoofing.
b. Application-Layer DoS attacks – in this class of attacks, attackers use a flood of legitimate HTTP GET requests to overwhelm Web Servers by pulling large files from the victim server in massive numbers or running a large number of (search or database) queries through the victim's machine.

These classes of DoS attacks are further classified as bandwidth-exhausting / bandwidth (HTTP Flooding) attacks and resources-exhausting / resource attacks depending on what critical resource(s) they target.

III. PAST RELATED RESEARCHES

A lot of researches have attempted to proffer solutions to the problem of DoS and DDoS attacks. Most of these solutions broadly either rely on various forms of monitoring of traffic, requests and data that pass through to servers and computing infrastructure for processing, or on various forms of analysis in trying to determine and classify these traffic, requests and data as "good" or "bad / malicious"; notable amongst these are researches and solutions are those by [18], [19], [20], [21], [12], [16], [22], [23], [24], [25] and [26], [27], amongst many notable others.

However, a critical investigation into these solutions seems to suggest that they may still not be able to stand the test of real-world DoS attacks. This is perceived so because a good amount of useful resources are either spent on trying to analyse content to determine whether they are malicious or not; trying to process and compute relatively complex, sometimes very sophisticated algorithms and heuristics; or just kept lying in idleness while useful, legitimate content either continue to wait, sometimes indefinitely, behind floods of malicious content still waiting to be processed and analysed; or are summarily discarded or timed out as a result of prolonged waiting.

[28], proposed a novel approach to solutions for securing computing infrastructure against DoS attacks that directs more focus to the rate at which these accesses, packets, processes, applications and connections consume critical
computing resources right from the stage of acquiring or provisioning these resources; the unavailability of which results in a denial of service, as against analysing through and authenticating the various contents of these. This is agreed to be a more exigent objective because as pointed out by [15], the strength of a DoS attack, especially those that are orchestrated over networks, doesn't rely so much on the content of the packets in the attack traffic as it does on the overwhelming nature of the volume of the attack traffic. As a result, this research continues in this trail by proposing a model for monitoring, in real-time, the resource consumption levels of various applications and processes that run on computing infrastructure. This is an extension to the model for resource partitioning as proposed by [28].

This research, however, may also be considered an extension of the work by [29] in which a solution was proposed for combating SlowPOST Denial of Service attacks that target the OSI application layer, to be in line with the propositions of [28].

IV. THE METHODOLOGY

This real-time resource monitor is responsible for monitoring and keeping track of processes, packet traffic and applications and their resource consumption rates as they utilize resources in both the main resource partition and the reserve resource partition as proposed by [28].

When a process, packet traffic or application begins to consume resources in the major resource partition at a high rate that tends to suggest the imminence of a denial of

service, the monitoring unit senses this level of consumption and immediately traps the culprit, then moves the culprit into the reserve partition where it would be monitored more closely and serviced at a much slower rate using limited resource provisions until the consumption levels normalize.

For example: If application A alone, running in the main resource partition, begins to consume say up to 60% and above of the resource provisions in that partition; the monitor immediately traps application A (suspecting that application A may soon become the cause of an imminent denial of service). Application A may then be halted or terminated immediately, or diverted to the reserve partition where it may either be refreshed steadily (by repeated reclaiming and re-allocation of resources in the reserve partition) or may be left to keep running in the reserve partition and consuming resources, but at a minimal rate, while it is being monitored until it's consumption levels normalize (say below 20%). It is then moved back into the main resource partition when its resource consumption has reached safe levels.

A. Algorithm For The Resource Monitor[i]:

The algorithm below outlines this process:

Step 1: **Start** resource_monitor

Step 2: **Input** the maximum allowed resource consumption (expressed in percentage, e.g. 60%): *consumption*

Step 3: **Input** the interval of checking the process consumption levels (expressed in milliseconds, e.g. 15): *interval*

Step 4: **Monitor** resource consumption of all processes used by applications or packet traffic in the major resource partition and reserve resource partition

Step 5: **Output**: return void. ***Trap*** all processes which resource consumption is greater than #var *consumption*

Step 6: **Continue**to Step 4

B. Flowchart For The Resource Monitor:

The flowchart in Figure 1 illustrates this algorithm:

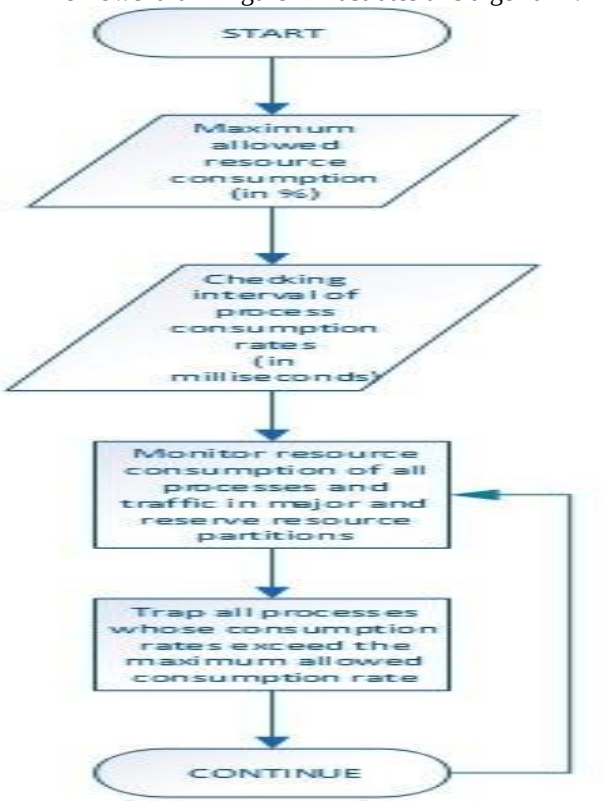

Figure 1: Flowchart for the Resource Monitor

C. Function Flow Block Diagram For The Resource Monitor:

The functional flow block diagram (FFBD) for the resource monitor is given in Figure 2 below:

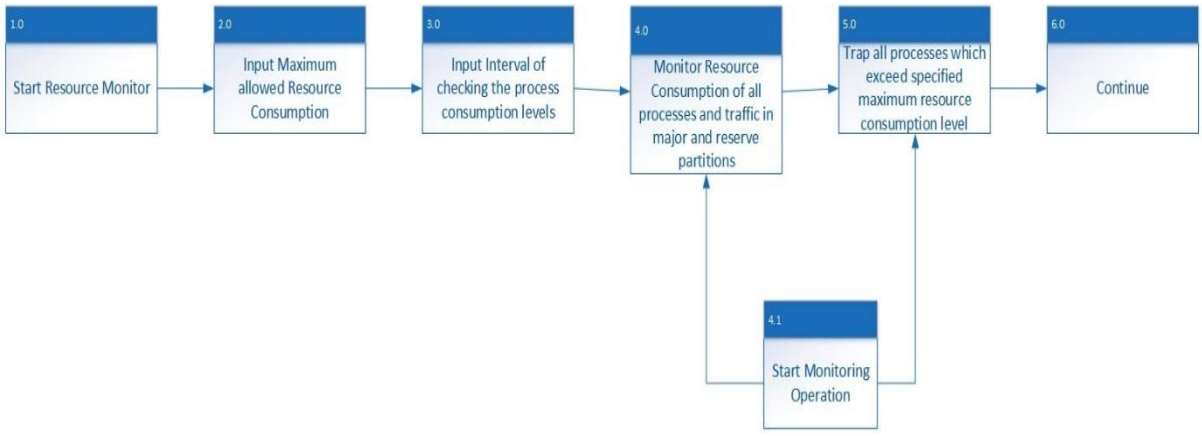

Figure 2: Functional Flow Block Diagram of the Resource Monitor

V. SUMMARY

This model of resource monitoring would ensure that denial of service does not occur even by flash crowds. Resources would be more preserved and more efficiently utilized even under the strains of denial of service attacks. Legitimate traffic would neither be kept waiting nor summarily discarded but would still be serviced, processed and responded to, only at a slower and more controlled rate.

VI. CONCLUSION

For best results to be obtained using this resource monitor, it should be implemented on computing infrastructure that has more than one resource pool or partition, such as that proposed by [28]. By this, overbearing requests, processes, traffic and applications would be serviced separately from less-resource-demanding requests.

VII. REFERENCES

[1]. Ruthfield, S. (1995, September). The Internet's History and Development From Wartime Tool to the Fish-Cam. Crossroads - Special issue on networks, 2(1), pp. 2-4. doi:10.1145/332198.332202

[2]. Dittrich, D. (1999). The DoS Project's "trinoo" Distributed Denial of Service Attack Tool. Retrieved February 1, 2014, from http://staff.washington.edu/dittrich/misc/trinoo.analysis

[3]. Qijun, G., & Liu, P. (2007, June). Denial of Service Attacks. San Marcos. Retrieved from http://s2.ist.psu.edu/paper/DDoS-Chap-Gu-June-07.pdf

[4]. Network Box UK Ltd. (2010). Denial of Service Attacks (DoS). London, United Kingdom: Network Box: Managed Security Services. Retrieved from http://www.network-box.co.uk/sites/default/files/Denial%20of%20Service.pdf

[5]. Dennis, M. A. (2012, March 2). Denial of Service Attack (DoS Attack). (Encyclopaedia Britannica Online Academic Edition). Encyclopaedia Britannica. Retrieved January 29, 2014, from http://www.britannica.com/EBchecked/topic/1055468/denial-of-service-attack

[6]. Murphy, D. M. (February 2010). War is War? The utility of cyberspace operations in the contemporary operational environment. Proceedings of the workshop for the center for strategic leadership (pp. 1-4). Pennsylvania, USA.: U.S. Army War College.

[7]. Internet Security Systems. (March 2005).

[8]. Ophardt, J. A. (2010). Cyber Warfare and the Crime of Aggression: The Need for Individual Accountability on Tomorrow's Battlefield. Duke Law and Technology Review(3), 1-27.

[9]. Whitman, M. E. (2003, August). Enemy at the gate: threats to information security. Communications of the ACM - Program compaction, 46(8), pp. 91-95. doi:10.1145/859670.859675

[10]. Power, R. (2002). CSI/FBI computer crime and security survey. Computer Security Issues & Trends, 8(1), pp. 1-24.

[11]. Gorman, S. (2013, July 22). Annual U.S. Cybercrime Costs Estimated at $100 Billion; Study Casts Doubt on Previous, Higher Figures. Wall Street Journal Publications.

[12]. Peng, T., Leckie, C., & Ramamohanarao, K. (2003a). Detecting Distributed Denial of Service Attacks Using Source IP Address Monitoring. The University of Melbourne, Australia, Department of Electrical and Electronic Engineering. Victoria 3010, Australia: ARC Special Research Center for Ultra-Broadband Information Networks. Retrieved January 30, 2014, from http://www.cs.mu.oz.au/~tpeng/mudguard/research/detection.pdf

[13]. Sabahi, F. (2011). Virtualization-Level Security in Cloud Computing. Communication Software and Networks (ICCSN), IEEE, 250-254.

[14]. Carthy, J. (2006, December 18). UCD School of Computer Science and Informatics, Dublin. Retrieved October 20, 2014, from COMP 1001: IT & Architecture (Lecture Notes): http://www.csi.ucd.ie/staff/jcarthy/home/FirstYear/Comp1001-L12.pdf

[15]. Peng, T., Leckie, C., & Ramamohanarao, K. (2007). Survey of network-based defense mechanisms countering the DoS and DDoS problems. ACM Computing Surveys (CSUR): Article No. 3, 39(1). doi:10.1145/1216370.1216373

[16]. Peng, T., Leckie, C., & Ramamohanarao, K. (August 2003b). Protection from Distributed Denial of Service Attack Using History-based IP Filtering. The University of Melbourne, Australia, Department of Electrical and Electronic Engineering. Victoria 3010, Australia: ARC Special Research Center for Ultra-Broadband Information Networks. Retrieved January 31, 2014, from http://ww2.cs.mu.oz.au/~tpeng/mudguard/research/icc2003.pdf

[17]. Ni, T., Gu, X., Wang, H., & Li, Y. (2013). Real-time detection of application-layer DDoS attack using time series analysis. Journal of Control Science and Engineering - Special issue on Advances in Methods for Networked and Cyber-Physical System, Article No. 4 .

[18]. Paxson, V. (1998). Bro: a system for detecting network intruders in real-time. Proceedings of the 7th USENIX Security Symposium.7, pp. 1-22. San Antonio, Texas, USA.: USENIX Association Berkeley, CA, USA.

[19]. Mahajan, R., Bellovin, S. M., Floyd, S., Ioannidis, J., Paxson, V., & Shenker, S. (2001, February). Controlling High Bandwidth Aggregates in the Network. ACM SIGCOMM Computer Communications Review, 32(3), 62-72.

[20]. Gil, T. M., & Poletto, M. (August, 2001). MULTOPS: a data-structure for bandwidth attack detection. Proceedings of the 10th USENIX Security Symposium, (pp. 23-38). Washington, D.C., USA.

[21]. Yau, D. K., Lui, J. C., & Liang, F. (2002, May). Defending against distributed denial-of-service attacks with max-min fair server-centric router throttles. Proceedings of IEEE International Workshop on Quality of Service (IWQoS), 29-41.

[22]. Verkaik, P., Spatscheck, O., Van der Merwe, J., & Snoeren, A. C. (September 2006). PRIMED: Community-of-Interest-Based DDoS Mitigation. Proceedings of the 2006 SIGCOMM workshop on Large-scale attack defense (pp. 147-154). New York, NY, USA: Association for Computing Machinery.

[23]. Ranjan, S., Swaminathan, R., Uysal, M., Nucci, A., & Knightly, E. (2009, February). DDoS-shield: DDoS-resilient scheduling to counter application layer attacks. IEEE/ACM Transactions on Networking (TON), 17(1), 26-39.

[24]. Xie, Y., & Yu, S.-Z. (2009, February). Monitoring the application-layer DDoS attacks for popular websites. IEEE/ACM Transactions on Networking (TON), 17(1), 15-25.

[25]. Walfish, M., Vutukuru, M., Balakrishnan, H., Karger, D., & Shenker, S. (2010, March). DDoS defense by offense. ACM Transactions on Computer Systems (TOCS), 28(1), Article No. 3 (54 pages). doi:10.1145/1731060.1731063

[26]. Das, D., Sharma, U., & Bhattacharyya, D. K. (2011). Detection of HTTP flooding attacks in multiple scenarios. Proceedings of the 2011 International Conference on Communication, Computing & Security (pp. 517-522). Rourkela, Odisha, India: Association for Computing Machinery New York, NY, USA. doi:10.1145/1947940.1948047

[27]. François, J., Aib, I., & Boutaba, R. (2012, December). FireCol: A Collaborative Protection Network for the Detection of Flooding DDoS Attacks. IEEE/ACM TRANSACTIONS ON NETWORKING, Volume 20(Issue 6), 1828-1841.

[28]. Ogu, E. C., Alao, O. D., Omotunde, A. A., Ogbonna, A. C., & Izang, A. A. (2014, October). Partitioning of Resource Provisions for Cloud Computing Infrastructure against DoS and DDoS Attacks. International Journal of Advanced Research in Computer Science, 5(7).

[29]. Raghunath, A., Ramachandran, S., Vaidyanathan, S., & Subramania, S. (2013, September). Data Rate Based Adaptive Thread Assignment Solution for Combating the SlowPOST Denial of Service Attack. ACM SIGSOFT Software Engineering Notes, Volume 38(Issue 5), 1-5.

[i] A prototype implementation of this resource monitor was implemented during this research, and also experimented upon. However, the implementation cannot be published due to copyright restrictions. Interested individuals and organization could contact the corresponding author through the email address: ecoxd1@yahoo.com for deals to provide source codes and experimentation results.