International Journal of Research in Engineering, Science and Management
Volume-2, Issue-11, November-2019
www.ijresm.com | ISSN (Online): 2581-5792

Detection of Distributed Denial of Service Attacks using Machine Learning Techniques

J. Ashok Lawrence[1], L. Alagappan[2], K. Vignesh Varadhan[3], K. Muthamilsudar[4]
[1,2,3]*UG Scholar, Department of Computer Science and Engineering, Kalasalingam Academy of Research and Education, Sivakasi, India*
[4]*Associate Professor, Department of Computer Science and Engineering, Kalasalingam Academy of Research and Education, Sivakasi, India*

***Abstract*: A Denial of Service (DoS) attack is a malicious effort to keep endorsed users of a website or web service from accessing it, or limiting their ability to do so. A Distributed Denial of Service (DDoS) attack is a type of DoS attack in which many computers are used to cripple a web page, website or web- based service. Fault either in users' implementation of a network or in the standard specification of protocols has resulted in gaps that allow various kinds of network attack to be launched of the type of network attacks, denial-of-service flood attacks have reason the most severe impact. This analysis study on flood attacks and Flash Crowd their improvement, classifying such attacks as either high-rate flood or low-rate flood. Finally, the attacks are appraised against principle related to their characteristics, technique and collision. This paper discusses a statistical approach to analysis the distribution of network traffic to recognize the normal network traffic behavior. The EM algorism is discussed to approximate the distribution parameter of Gaussian mixture distribution model. Another time series analysis method is studied. This paper also discusses a method to recognize anomalies in network traffic, based on a non-restricted α-stable first-order model and statistical hypothesis testing.**

***Keywords*: DDoS Impact, Attack Detection.**

1. Introduction

Distributed denial-of-service attacks (DDoS) pose an immense threat to the Internet, and consequently many defense mechanisms have been proposed to combat them. Attackers constantly modify their tools to bypass these security systems, and researchers in turn modify their approaches to handle new attacks. The DDoS field is evolving quickly, and it is becoming increasingly hard to grasp a global view of the problem. This paper strives to introduce some structure to the DDoS field by developing a taxonomy of DDoS attacks and DDoS defense systems. The goal of the paper is to highlight the important features of both attack and security mechanisms and stimulate discussions that might lead to a better understanding of the DDoS problem.

A Denial of Service attack is an attempt by a person or a group of persons to cripple an online service. This can have serious consequences, especially for companies like Amazon and eBay which rely on their online availability to do business. In the not so distant past there have been some large scale attacks targeting high profile internet sites [28, 29, 30,. and 31]. Consequently, there are currently a lot of efforts being made to come up with mechanisms to detect and mitigate such attacks.

Even though the first denial of service attacks did not take place a long time ago (tools that automate setting up of an attack network and launching of attacks, started appearing in 1998), there are a multitude of denial of service attacks that have been used. Broadly speaking the attacks can be of three forms. a) Attacks exploiting some vulnerability or implementation bug in the software implementation of a service to bring that down. b) Attacks that use up all the available resources at the target machine. c) Attacks that consume all the bandwidth available to the victim machine. The third type of attacks is called bandwidth attacks. A distributed framework becomes especially suited for such attacks as a reasonable amount of data directed from a number of hosts can generate a lot of traffic at and near the target machine, clogging all the routes to the victim. Protection against such large scale distributed bandwidth attacks is one of the most difficult (and urgent) problem to address in today's internet. CERT reports bandwidth attacks as increasingly being the most common form of Denial of Service attacks seen in the internet today.

2. Characteristics

It is a java based tool that can run on any platform and can generate TCL scripts for wired and Wireless scenarios for NS2.Main features of NSG are:

- Creating Wired and wireless nodes by drag and drop.
- Creating Simplex and Duplex links for wired network
- Creating Grid, Random and Chain topologies.
- Creating TCP and UDP agents. Also supports TCP
- Tahoe, TCP Reno, TCP New-Reno and TCP Vegas.
- Supports Ad Hoc routing protocols such as DSDV,
- AODV, DSR and TORA.
- Supports FTP and CBR applications.
- Supports node mobility.
- Setting the packet size, start time of simulation, end
- Time of simulation, transmission range and interference

- Range in case of wireless networks, etc.
- Setting other network parameters such as bandwidth, etc for wireless scenarios.

3. Existing system

Jamming attacks are much harder to counter and more security problems. They have been shown to actualize severe Denial-of-Service (DoS) attacks against wireless networks [16]. In the simplest form of jamming, the adversary interferes with the reception of messages by transmitting a continuous jamming signal, or several short jamming pulses jamming attacks have been considered under an external threat model, in which the jammer is not part of the network [18]. Under this model, jamming strategies include the continuous or random transmission of high power interference signals.

A. Disadvantage

- The latest types of attacks like HTTP flood, SIDDoS, Smurf and UDP flood etc were not detected.
- The False Positive Rate(FPR) ranges does not get increased above 4.11%.
- The system requirements for this attack detection requires high cost CPUs.

4. Proposed system

In this project, we address the problem of jamming under an internal threat representation. We consider a sophisticated adversary who is aware of network secrets and the implementation details of network protocols at any layer in the network stack [6]. The adversary exploits his internal knowledge for launching selective jamming attacks in which specific messages of "high importance" are targeted. We use random forest and linear regression algorithms, to divide packet information before sending packet to the receiver. Linear regression algorithm is used for plotting the graph.

A. Advantage

1. The adversary exploits his internal knowledge for launching selective jamming attacks in which specific messages of "high importance" are targeted.
2. Random forest algorithm decode the first few bits of a packet for recovering useful packet identifiers such as packet type, source and destination address using decision tree [16].
3. Communication is easy and fast between people because of implementing new scheme.
4. Security is more complicated in many application, but we overcome this problem using Random Forest algorithm.

5. System architecture

A system architecture or systems architecture is the computational design that defines the structure and/or behavior of a system. An architecture description is a formal description of a system, organized in a way that supports reasoning about the structural properties of the system. It defines the system components or building blocks and provides a plan from which products can be procured, and systems developed, that will work together to implement the overall system.

NS2 provides users with an executable command "ns" which takes one input argument, the name of a Tcl simulation scripting file. In most cases, a simulation trace file is created and is used to plot graph and/or to create animation. NS2 consists of two key languages: CCC and Object-oriented Tool Command Language (OTcl). While the CCC defines the internal mechanism (i.e., a backend) of the simulation, the OTcl sets up simulation by assembling and configuring the objects as well as scheduling discrete events (i.e., a frontend). The CCC and the OTcl are linked together using TclCL.

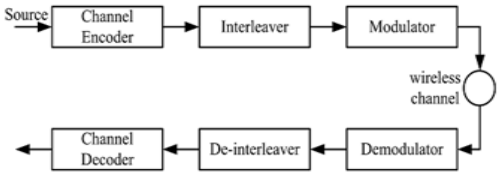

Fig. 1. System architecture

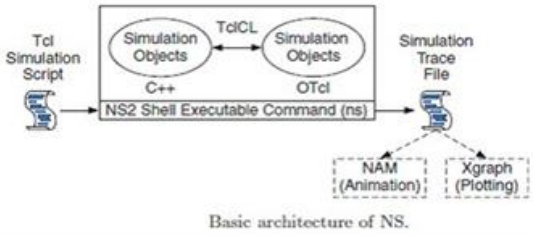

Fig. 2. NS2 architecture

6. Modules description

Real Time Packet Classification: At the PHY layer, a packet m is encoded, interleaved, and modulated before it is transmitted over the wireless channel [13]. At the receiver, the signal is demodulated, deinterleaved, and decoded, to recover the original packet m.

Moreover, even if the encryption key of a hiding scheme were to remain secret, the static portions of a transmitted packet could potentially lead to packet classification. This is because for computationally-efficient encryption methods such as block encryption, the encryption of a prefix plaintext with the same key yields a static ciphertext prefix [16]. Hence, an adversary who is aware of the underlying protocol specifics (structure of the frame) can use the static cipher text portions of a transmitted packet to classify it.

Selective Jamming Module: We illustrate the impact of selective jamming attacks on the network performance. Implement selective jamming attacks in two multi-hop wireless network scenarios [12]. In the first scenario, the attacker targeted a TCP connection established over a multi-hop wireless route. In the second scenario, the jammer targeted

network-layer control messages transmitted during the route establishment process selective jamming would be the encryption of transmitted packets (including headers) with a static key. However, for broadcast communications, this static decryption key must be known to all intended receivers and hence, is susceptible to compromise. An adversary in possession of the decryption key can start decrypting as early as the reception of the first ciphertext block.

Random Forest: A random forest consists of multiple random decision trees. Two types of randomness are built into the trees. First, each tree is built on a random sample from the original data. Second, at each tree node, a subset of features is randomly selected to generate the best split.

Linear Regression: Linear regression is useful for finding relationship between two continuous variables. One is predictor or independent variable and other is response or dependent variable. It looks for statistical relationship but not deterministic relationship. Relationship between two variables is said to be deterministic if one variable can be accurately expressed by the other. For example, using temperature in degree Celsius it is possible to accurately predict Fahrenhe it.

7. System implementation

Network Simulator (Version 2), widely known as NS2, is simply an event-driven simulation tool that has proved useful in studying the dynamic nature of communication networks. Simulation of wired as well as wireless network functions and protocols (e.g., routing algorithms, TCP, UDP) can be done using NS2. In general, NS2 provides users with a way of specifying such network protocols and simulating their corresponding behaviors. Due to its flexibility and modular nature, NS2 has gained constant popularity in the networking research community since its birth in 1989. Ever since, several revolutions and revisions have marked the growing maturity of the tool, thanks to substantial contributions from the players in the field. Among these are the University of California and Cornell University who developed the REAL network simulator,1 the foundation on which NS is invented. Since 1995 the Defense Advanced Research Projects Agency (DARPA) supported the development of NS through the Virtual Inter Network Testbed (VINT) project [10].2 Currently the National Science Foundation (NSF) has joined the ride in development. Last but not the least, the group of researchers and developers in the community are constantly working to keep NS2 strong and versatile. Again, the main objective of this book is to provide the readers with insights into the NS2 architecture. This chapter gives a brief introduction to NS2. NS2 Beginners are recommended to go through the detailed introductory online resources.

Network simulation (NS) is one of the types of simulation, which is used to simulate the networks such as in MANETs, VANETs etc. It provides simulation for routing and multicast protocols for both wired and wireless networks. NS is licensed for use under version 2 of the GNU (General Public License) and is popularly known as NS2. It is an object-oriented, discrete event-driven simulator written in C++ and Otcl/tcl.NS-2 can be used to implement network protocols such as TCP and UPD, traffic source behavior such as FTP, Telnet, Web, CBR and VBR, router queue management mechanism such as Drop Tail, RED and CBQ, routing algorithms and many more. In ns2, C++ is used for detailed protocol implementation and Otcl is used for the setup. The compiled C++ objects are made available to the Otcl interpreter and in this way, the ready-made C++ objects can be controlled from the OTcl level.

A. Design and considerations

1. It is a discrete event simulator for networking research.
2. It provides substantial support to simulate bunch of protocols like TCP, FTP, UDP, https and DSR.
3. It simulates wired and wireless network.
4. It is primarily Unix based.
5. Uses TCL as its scripting language.
6. Otcl: Object oriented support.
7. Tclcl: C++ and otcl linkage.
8. Discrete event scheduler.

B. Design

NS2 stands for Network Simulator Version 2. It is an open-source event-driven simulator designed specifically for research in computer communication networks [19]. NS2 uses OTcl to create and configure a network, and uses C++ to run simulation. All C++ codes need to be compiled and linked to create an executable file.

C. Use OTcl

- For configuration, setup, or one-time simulation, or
- To run simulation with existing NS2 modules.
- This option is preferable for most beginners, since it does not involve complicated internal mechanism of NS2.
- Unfortunately, existing NS2 modules are fairly limited. This option is perhaps not sufficient for most researchers.
- Use C++
- When you are dealing with a packet, or - when you need to modify existing NS2 modules.

This option perhaps discourages most of the beginners from using NS2. This book particularly aims at helping the readers understand the structure of NS2 and feel more comfortable in modifying NS2 modules [26].

D. Considerations

The following constraints must be kept in mind while developing the design:

- Power supply should be cut off when module not in use.
- Module has to be installed in a place where there is availability of strong network for notifications.
- The food items have to be placed in their respective

slots.

- The design must be applicable any existing refrigerator.
- "Embedded C" was the effective option among others and is chosen as programming language.
- The output is shown on a circuit board which consists of the smart refrigeration module. LED's are placed at various places to indicate the flow of execution.

The possible source of error includes replacing of food items without using them leading false notification. To recover from such an error, the module could be reset or dedicated application could be developed.

E. Block diagram

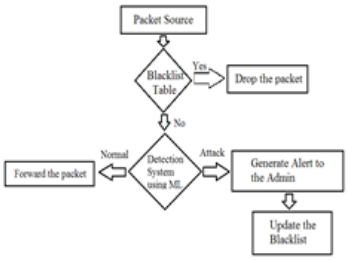

Fig. 3. Block diagram

8. Results

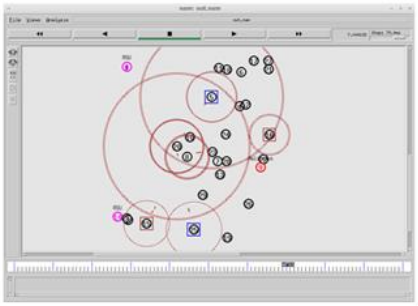

Fig. 4. Output of DDoS

Fig. 4. shows that there are sender, receiver and an attacker. The sender sends data to the receiver without any interruption of the attacker.

9. Conclusion

We addressed the problem of selective jamming attacks in wireless networks. We considered an internal adversary model in which the jammer is part of the network under attack, thus being aware of the protocol specifications and shared network secrets. We showed that the jammer can classify transmitted packets in real time by decoding the first few symbols of an ongoing transmission. We evaluated the impact of selective jamming attacks on network protocols such as TCP and routing. Our findings show that a selective jammer can significantly impact performance with very low effort. We developed three schemes that transform a selective jammer to a random one by preventing real-time packet classification. Our schemes combine cryptographic primitives such as commitment schemes, cryptographic puzzles, and all-or-nothing transformations (AONTs) with physical layer characteristics. We analyzed the security of our schemes and quantified their computational and communication overhead.

10. Future work

In future, an external selective jammer who targets various control packets at the MAC layer. To perform packet classification, the adversary exploits inter-packet timing information to infer eminent packet transmissions. We proposed the estimation of the probability distribution of inter-packet transmission times for different packet types based on network traffic analysis. Future transmissions at various layers were predicted using estimated timing information. Using their model, the authors proposed selective jamming strategies for well-known sensor network MAC protocols. We may consider several packet identifiers for encrypted packets such as packet size, precise timing information of different protocols, and physical signal sensing. To prevent selectivity, the unification of packet characteristics such as the minimum length and inter-packet timing was proposed. Similar packet classification techniques were investigated.

References

[1] CERT, http://www.cert.com
[2] M. Li. An approach to reliably identifying signs of DDOS flood attacks based on LRD traffic pattern Recognition. Computers & Security, 23(7): 549-558, 2004.
[3] C.S. Sastry, S. Rawat and A.K. Pujari. Network traffic analysis using singular value decomposition and multiscale transforms. Information Sciences, 177(23): 5275-5291, 2007.
[4] M.F. Rohani, M.A. Maarof and A. Selamat. Continuous LoSS detection using iterative window based on SOSS model and MLS approach. In Proceedings of the International Conference on Computer and Communication Engineering, Kuala Lumpur, Malaysia, May 2000
[5] H. Hajji. Statistical analysis of network traffic for adaptive faults detection. IEEE Transactions on Neural Networks, 16(5):1053–1063, September 2005.
[6] J. D. Brutlag. Aberrant behavior detection in time series for network monitoring. In LISA '00: Proceedings of the 14th USENIX conference on System administration, pages 139–146, Berkeley, CA, USA, 2000.
[7] D. Rincón and S. Sallent. On-line segmentation of non- stationary fractal network traffic with wavelet transforms and Log-likelihood-based statistics. LNCS, 3375: 110- 123, 2005.
[8] C. Douligeris and A. Mitrokotsa. DDoS attacks and defense mechanisms: classification and state of- the-art. Computer Networks, 44(5): 643-666, 2004.
[9] P. García-Teodoro, J. Díaz-Verdejo and G. Maciá- Fernández. Anomaly-based network intrusion detection: techniques, systems and challenges. Computers & Security, 28(1-2): 18-28, 2009.
[10] V. A. SIRIS and F. PAPAGALOU. Application of anomaly detection algorithms for detecting syn flooding attacks. In Proceedings of IEEE Global Telecommunications Conference (GLOBECOM '04), volume 4, pages 2050–2054, Dallas, USA, 2004
[11] H. Wang, D. Zhang, and K. G. Shin. Syn-dog: Sniffing syn flooding sources. In Proceedings of the 22th International Conference on Distributed Computing Systems (ICDCS'02), pages 421–429, Washington, DC, USA, 2002. IEEE Computer Society.

[12] M. Charikar, K. Chen, and M. Farach-Colton. Finding frequent items in data streams. In Proceedings of the 29th International Colloquium on Automata, Languages and Programming (ICALP '02), pages 693–703, London, UK, 2002. Springer-Verlag.
[13] J. D. Brutlag. Aberrant behavior detection in time series for network monitoring. In LISA '00: Proceedings of the 14th USENIX conference on System administration, pages 139–146, Berkeley, CA, USA, 2000.
[14] V. Paxson. Bro: A System for Detecting Network Intruders in Real-Time. In Computer Networks, volume 31 (23–24), pages 2435–2463, 1999.
[15] E. S. Page. Continuous inspection schemes. Biometrika, 41:100–115, 1954.
[16] S.X. Wu and W. Banzhaf. The use of computational intelligence in intrusion detection systems: A review. Applied Soft Computing, 10: 1-35, 2010.
[17] O. Salem, S. Vaton, and A. Gravey. An efficient online anomalies detection mechanism for high-speed networks. In IEEE Workshop on Monitoring, Attack Detection and Migitation (MonAM 2007), November 2007.
[18] W. Leland, M. Taqqu, W. Willinger, and D. Wilson, "On the Self-Similar Nature of Ethernet Traffic (Extended Version)," IEEE/ACM Trans. Networking, vol. 2, no. 1, pp. 1-15, Feb. 1994
[19] Y. Guan, A. A. Ghorbani, and N. Belacel, An unsupervised clustering algorithm for intrusion detection. In Proc. of the Sixteenth Canadian Conference on Artificial Intel ligancy (AI 2003), pages 616-617, Halifax, Canada, May 2003. Springer.
[20] Huang Kai, Qi Zhengwei, Liu Bo" Network Anomaly Detection Based on Statistical Approach and Time Series Analysis" 2009 International Conference on Advanced Information Networking and Applications Workshops.
[21] Federico Simmross, Juan Ignacio, Pablo Cassia-de-la- Higuera, Ioannis A. Dimitriadis" Anomaly Detection in Network Traffic Based on Statistical Inference and α- Stable Modeling" IEEE transactions on dependable and secure computing, vol. 8, no. 4, July/August 2011.
[22] Khadijah Wan Mohd Ghazali and Rosilah Hassan:" Flooding Distributed Denial of Service Attacks-A Review "Journal of Computer Science, 7(8): 1218-1223, 2011.
[23] W. Leland, M. Taqqu, W. Willinger, and D. Wilson, "On the Self-Similar Nature of Ethernet Traffic (Extended Version)," IEEE/ACM Trans. Networking, vol. 2, no. 1, pp. 1-15, Feb. 1994.
[24] A. Papoulis, Probability, Random Variables, and Stochastic Processes, third ed., McGraw-Hill, 1991.
[25] G.R. Arce, Nonlinear Signal Processing: A Statistical Approach. John Wiley and Sons, 2005.
[26] Monika Sachdeva, Gurvinder Singh, Krishan Kumar, Kuldeep Singh, "DDoS Incident and their Impact," IAJIT 2010.
[27] Khadijah Wan, Mohd. Ghazali Rosilah Hassan" Flooding Distributed Denial of Service Attacks-A Review "
[28] CNN. Cyber-attacks batter Web heavyweights, February 2000. www.cnn.com/2000/TECH/computing/02/ 09/cyber attacks
[29] CNN. Immense. Network assault takes down Yahoo, February http://www.cnn.com
[30] Netscape. Leading web sites under attack, February 2000 technews.netscape.com "Journal of Computer Science.
[31] CERT coordination center. Denial of Service attacks http://www.cert.org/tech_tips/denial_of_service.html
[32] Distributed Denial of Service (DDoS) Attacks/tools. http://staff.washington.edu/dittrich/misc/ddos