Journal of Theoretical and Applied Information Technology
20th July 2014. Vol. 65 No.2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

A BACK PROPAGATION BASED BOTNET DETECTION ALGORITHM FOR ENHANCED NETWORK SECURITY

[1]M.KEMPANNA, [2]DR.R.JAGADEESH KANNAN

[1]Research Scholar, Department of Computer Science & Engg, Karpagam University, Coimbatore-641 021

[2]Professor, Department of Computer Science & Engg,R.M.K Engg.College,Chennai

E-mail: [1] kempsindia@yahoo.com , [2]dr_rjk@hotmail.com

ABSTRACT

The increased network computers makes botnet detection as a challenging one and it makes easier for intruders and attackers to generate mitigation attacks. The centralized propagation nature of botnet floods warms through different botnet clients and questions the network security. To overcome the challenges in identifying the botnet, we propose a new back propagation algorithm for botnet detection. The proposed back propagation method is a learning one, which keeps track of signature of identified warm and the list of hop it traversed. For each warm signature identified, it maintains bot matrix, in which set of hop addresses traversed by the data packet is stored. Whenever a warm packet is identified, its traversal path is tracked and compared with the list of hops present in the bot matrix for the occurrence of hop address present in the traversal path.

Keywords: *Network Security, Botnet, Peer-Peer Networks, Bot Matrix, Hops.*

1. INTRODUCTION

Botnet is recognized as one of the most serious security threats of today. A botnet consists of a network of compromised computers connected to the internet that is controlled by a remote attacker or botmaster. Botnets have become a significant threat to network communications and applications, as they increases the efficiency of network attacks such as denial-of–service (DoS) attacks, scanning, phishing, Email spam, identity theft, click fraud, and espionage . This capability of a botnet is attributed to the large number of hosts that it controls, which ranges from hundreds to thousands that work together in carrying out an attack, as opposed to when only a few number of hosts carry out attack. Compared to other Internet malware, the unique feature of a botnet lies in its control communication network. Most botnets that have appeared until now have had a common centralized architecture. That is, bots in the botnet connect directly to some special hosts (called "command-and-control servers, or "C&C"servers).C&C servers receive commands from their bot master and forward them to the other bots in the network. From now on we will call a botnet with such control communication architecture a "C&C botnet".

A peer-to-Peer network is a network of computers connected in no topology and could be used to transfer data packets. The data packets travel through different networks and computers towards destination. There are hops which act as a bot master which induces other peers to spread spams, the other peers which supports bot master is called as bot client or propagators.

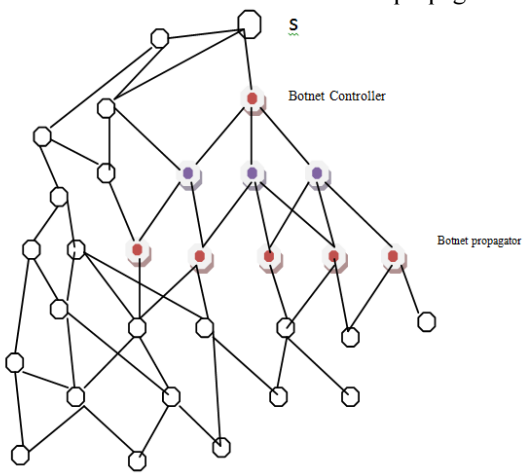

Figure 1: Topology Of Botnet In Peer To Peer Network.

The peers participating in botnet changes its addresses dynamically, so that identifying the peer address is more difficult. There is no such research has been done to identify the nodes participating in the botnet in a peer to peer network

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

due to the dynamic addressing modes of the peers. We take this challenge to identify the peers participating the hosts in peer to peer networks. Whatever be the address changing the network and subnet mask address will be a common and static for any network. This protocol specific property helps us in identifying the botnet nodes.

2. BACKGROUND

There are many researches has been done and many methods have been proposed for the construction of botnet and detection of botnet in peer to peer networks. We explore few of them here as follows:Defense Strategies against Modern Botnets [1]offers several strategies for defense against Botnets with a list and description of measures and activities which should be carried out in order to establish successful defense. They also offers parallel preview of the strategies with their advantages and disadvantages considered in accordance with various criteria.

Using Machine Learning Techniques to Identify Botnet Traffic [2] has presented whereWeb services/applications are in most cases attacked by Botnets. While creating the system, special attention should be paid to protection. Security must be built-in during each phase of the system development and this must be done on everyday basis. In order to be protected, it is necessary to obey the following rules:Usage of Intrusion of Prevention System (IPS); Intrusion Prevention Systems are devices whichmonitor network activity in order to detect vilifications or undesirable activities in real time with the task to block or prevent them from acting.

Anomaly-based detection techniques attempt to detectbotnets based on several network traffic anomalies such ashigh network latency, high volumes of traffic, traffic onunusual ports, and unusual system behavior that could indicatepresence of malicious bots in the network [3].Although anomaly detection techniques solve the problemof detecting unknown botnets, problems with anomalydetection can include detection of an IRC network that may bea botnet but has not been used yet for attacks, hence there areno anomalies. To solve this, Binkley and Singh [4] proposedan effective algorithm that combines TCP-based anomalydetection with IRC tokenization and IRC message statistics tocreate a system that can clearly detect client botnets.

A wide scale botnet detection and characterization [5] is presented for detection and characterization of botnets using passiveanalysis based on flow data in transport layer. This algorithmcan detect encrypted botnet communications. It helps toquantify size of botnets, identify and characterize theiractivities without joining the botnet. Botsniffer [6] uses network-based anomaly detection to identify botnet C&C channels in a localarea network. Botsniffer is based on observation that botswithin the same botnet will likely demonstrate very strongsynchronization in their responses and activities. Hence, itemploys several correlation analysis algorithms to detectspatial-temporal correlation in network traffic with a very lowfalse positive rate. Modeling Botnet Propagation Using Time Zones [7] presented a botnet monitoringsystem by redirecting the DNS mapping of a C&C server toa botnet monitor. Revealing Botnet Membership Using DNSBL Counter-Intelligence[8] presented howto passively detect botnets by finding botmasters' queries to spam DNS-based black hole list servers (DNSBL).Since most botnets nowadays use IRC for their C&Cservers, many people have studied how to detect them bydetecting their IRC channels or traffic. Binkley and Singh [9] attempted to detect them through abnormal IRC channels. Strayer [10] used machine-learning techniques to detectbotnet IRC-based control traffic and tested the system onrace-driven network data. Chen [11] presented a system to detect botnet IRC traffic on high-speed network routers.

Digital watermarking is often considered for digitalcopyright protection [19]. Digital watermarking is injectedto content file so that when a pirated copy isdiscovered, authorities can find the origin of piracy via aunique watermark in each copy. In a P2P network, allpeers are sharing exactly the same file (if not poisoned), which effectively defeats the purpose of watermarking.Thus, watermarking is not a suitable technology for P2P file-sharing. All the methods we have are based on the features and characteristics and nothing has been discussed to overcome the dynamic addressing of botnet nodes. We propose a new botnet detection algorithm which is a learning technique and works as back propagation.

3. PROPOSED SYSTEM

The proposed model has three operational modules named botnet detection, warm identification, and back propagation. Each has its own functional behavior, botnet detection is to

Journal of Theoretical and Applied Information Technology
20th July 2014. Vol. 65 No.2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

identify the presence of botnet clients in the traversal path of the data packet traversed and warm identification is the initial process of finding malicious data packet from list of packets arrived at a time frame whereas back propagation model is to track the presence of botnet using the list of hops traversed with the help of earlier tracks of packet travel.

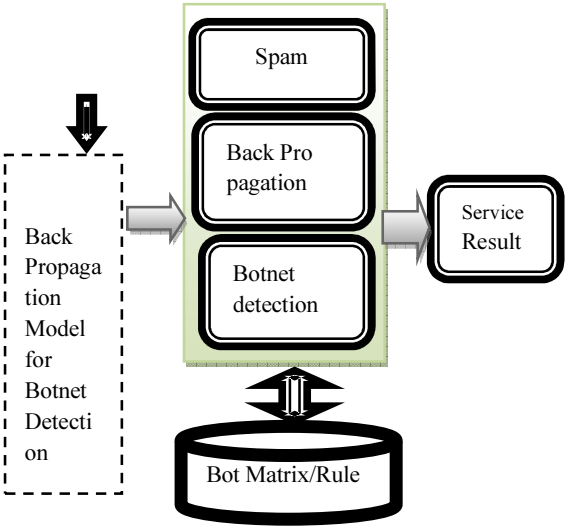

Figure 2: Proposed System Architecture.

4. SPAM FILTERING

The spam filtering is the process of identifying malicious packets which are incoming into the network. The malicious packet may be of any kind, but we consider only the data packets. The malicious packet may be of any kind of attacking packet like DDOS packet. The spam filtering functional model is capable of identifying and classifying the packet into various attacking type. It uses set of rules to identify the packet type and if any of the rules matches with the signature of the packet then it will assign the flag to the packet received. Once it identifies the class of packet then it will hand over the packet to other functional models clubbed with the proposed architecture.

4.1Algorithm

Step1: start

Step2: read bot matrix bm , Rule set R_s, initialize spam flag sf=0.

Step3: read incoming packet P_i.

Step4: extract features of packet F_p = ${Ipaddress, port, payload,ttl,hosts}.

Step5: for each rule R_i from R_s

Compute rule match $R_r = ø(F_{p(A}i\text{-}n_j\text{-}R_{i(A}i\text{-}n_j)$.

If (R_r==0)

{

$B_m=B_m+\$.$

Sf=1.

}

End

Step6: if (sf==1)

Initialize botnet detection functional module.

End

Step7: stop.

5 BACK PROPAGATION

The identified malicious packet details will be the trigger to invoke this functional model. The extracted packet feature and bot matrix is used to traverse the source of spam from where it injected into the network. Once its feature is identified and the proposed system generates a Botnet Handling Packet (BHP) and sends the packet to the source node. While forwarding the packet it uses source routing mechanism, so that the packet path can be predefined. The selection of path to be traversed by the BHP packet is done using the bot matrix Bm. From the packet feature of the received packet it extracts the list of hops and for each hop from the traversal path it check for the occurrence in the bot matrix. Once it has an occurrence in the bot matrix the hop id will be removed from the list. Finally it selects a different path which does not pass through any of the node from bot matrix.

5.1 Algorithm

Input : extracted feature $ of incoming packet F_p

Step1: start

Step2: read bot matrix B_m

Step3: Hop list H_l = $(host).

Step4: for each hop h_i from H_l

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Check for the presence in B_m.

If ($\mathbf{B_m}\Sigma h_i$)

Remove h_i from list.

$H_l = \hat{O}(H_l(h_i))$.

End.

End.

Step5: compute BHP message.

Step6: select source routing path Srp exclusive of B_m.

Step7: send packet.

Step8: stop.

5.2.1 Botnet detection

The identification of botnet is done using pattern mining technique. Whenever a spam or warm identified by the system it stores the whole packet features to the data base. From the store data we extract the common pattern or frequent pattern present in the traversal path is identified. We compute the sub set of patterns and compute the frequency of each subset of pattern. The patterns which have more support value is identified as the members of botnet or botnet clients. The occurrence of a node in more traversal path shows definite guilty.

5.2.2 Algorithm

Step1: start

Step2: initialize bot matrix B_m.

Step3: read warm data base D_w.

Step4: for each record D_i from D_w

Convert record into pattern P.

$P_i = D_{i(hops)}$.

$P = P+P_i$.

End

Step5: for each node from pattern P_i compute support

$Sup_{node} = §/ë$.

§= no of times it occurs in the patterns

Ë= total no pattern that other nodes present in overall.

If $Sup_{node} >$ threshold

Add node id to Bot matrix B_m.

$B_m = B_m$+nodeid.

End.

End.

Step6: stop.

6. RESULTS AND DISCUSSION

The proposed back propagation model has been implemented in Network simulator Ns2. It produced efficient results and the identification of botnet has performed efficiently. The Figure 3 shows the accuracy of botnet detection according to the number of spam packets received. It is clear that the detection accuracy and frequency is increasing according to the number of warm or spam packets received. The proposed system is a learning system so that if the number of spam packets received is increases then the frequency of botnet detection also gets increased. The Figure 4 shows the packet delivery ratio of different algorithms. it is very clear that the proposed method has more delivery ratio than others.

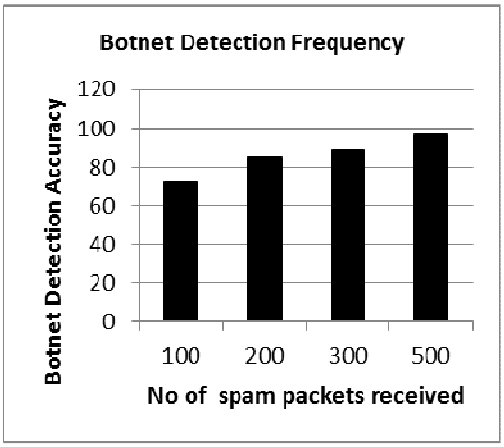

Figure 3: Botnet Detection Accuracy.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

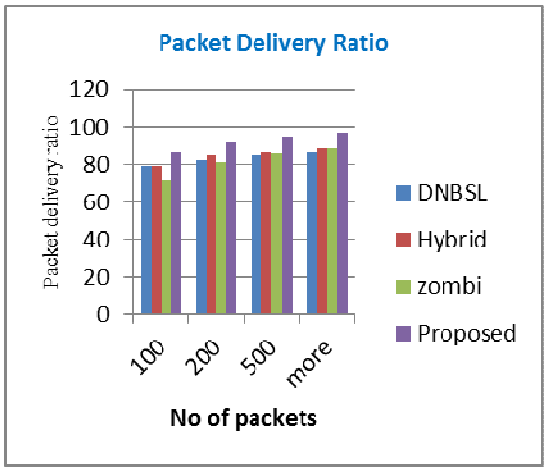

Figure 4: Packet Delivery Ratio

7. CONCLUSION

The proposed system has produced very good results. As the warm data base increases the size of bot matrix also gets increased. This helps the proposed system to identify more botnet clients and botnet nodes. So that the packet delivery and throughput of the overall network gets increased. The back propagation model helps the peer to peer networks to identify the botnet nodes efficiently and it makes easier to send data packets in some other way to overlook the botnet nodes. The proposed system has proved well in all directions of the quality of service.

REFERENCES

[1] Srdjan Stanković, Defense Strategies Against Modern Botnets, (IJCSIS) International Journal of Computer Science and Information Security,Vol. 2, No. 1, 2009

[2] C. Livadas, R. Walsh, D. Lapsley, and T. Strayer, Using Machine Learning Techniques to Identify Botnet Traffic," Submitted to 2nd IEEE LCN Workshop on Network Security, 2006.

[3] B.Saha and A.Gairola, "Botnet: An overview," CERT-In WhitePaperCIWP-2005-05, 2005.

[4] J.R.Binkley and S.Singh, "An algorithm for anomaly-based botnetdetection," in Proc. USENIX Steps to Reducing Unwanted Traffic on theInternet Workshop (SRUTI'06), , 2006, pp 43–48.

[5] Karasaridis, B. Rexroad, and D. Hoeflin, "Wide-scale botnetdetection and characterization," in Proc. 1st Workshop on Hot Topics inUnderstanding Botnets, 2007.

[6] G. Gu, J. Zhang, and W. Lee, "Botsniffer: Detecting botnet commandand control channels in network traffic," in Proc. 15th Annual Networkand distributed System Security Symposium (NDSS'08), 2008.

[7] D. Dagon, C. Zou, and W. Lee, "Modeling Botnet PropagationUsing Time Zones," Proc. 13th Ann. Network and Distributed SystemSecurity Symp. (NDSS '06), pp. 235-249, Feb. 2006.

[8] A.Ramachandran,N.Feamster, and D.Dagon, "Revealing BotnetMembership Using DNSBL Counter-Intelligence," Proc. USENIXSecond Workshop Steps to Reducing Unwanted Traffic on the Internet(SRUTI '06), June 2006.

[9] Arce and E. Levy, "An Analysis of the Slapper Worm," IEEESecurity & Privacy Magazine, vol. 1, no. 1, pp. 82-87, Jan.-Feb. 2003

[10] T.Strayer,"Detecting Botnets with Tight Command and Control,"ARO/DARPA/DHS Special Workshop Botnet, 2006.

[11] Y.Chen,"IRC-Based Botnet Detection on High-Speed Routers,"ARO/DARPA/DHS Special Workshop Botnet, 2006.

[12] S.H. Kwok, "Watermark-Based Copyright Protection SystemSecurity," Comm. ACM, pp. 98-101, Oct. 2003.

[13] D.P. Majoras, O. Swindle, T.B. Leary, and J. Harbour, "Peer-to-

[14] Peer File-Sharing Technology: Consumer Protection and CompetitionIssues," Federal Trade Commission Report, June 2005.

[15] N.Mook, "P2P Flooder Overpeer Cease Operation," Beta News, http://www.betanews.com/article/P2P_Flooder_Overpeer_Ceases_Operation/1134249644, Dec. 2005.

[16] N.Mook, "P2P Future Darkens as eDonkey Closes," http://www.betanews.com/article/P2P_Future_Darkens_as_eDonkey_Closes/1127953242, Sept. 2005.

[17] G. Pallis and A. Vakali, "Insight and Perspectives for ContentDelivery Networks," Comm. ACM, pp. 101-106, Jan. 2006.

[18] P. Rodriguez et al., "On the Feasibility of Commercial Legal P2PContent Distribution," SIGCOMM Computer Comm. Rev., vol. 36,no. 1, pp. 75-78, Jan. 2006.