TELKOMNIKA Indonesian Journal of Electrical Engineering
Vol. 12, No. 11, November 2014, pp. 7963 ~ 7969
DOI: 10.11591/telkomnika.v12i11.6318

Research of the Communication Model of Botnet Based on P2P

Gao Jian*, Yang Ming, Guo Chengqing
Peple's Public Security Univercity of China, China
National computer network and information security management center, China
Corresponding author, e-mail: gaojianbeijing2006@163.com

Abstract

The communication mechanism of botnets is a concern of security scholars, especially based on peer to peer. Botnet has gradually formed some mature and covert communication channels. The communication mechanism for P2P botnet existing is classified into two models: the Send communication model and the Request communication model. We propose an evaluation index including concealment, effectiveness, efficiency and robustness and its calculation method. At the same time we using these evaluation index to simulate, evaluate and analysis the two kinds of models, and study the relationship between them and the botnet basic characteristics.

Keywords: *Botnet, peer to peer, evaluation, communication model*

1. Introduction

A "botnet" is a network of compromised computers (bots) that are controlled by an attacker (botmasters). Botnets are one of the most serious threats to today's Internet; they are the root cause of many current Internet attacks, such as email spam, distributed denial of service (DDoS) attacks, click fraud, etc. Early Botnet mainly used a centralized command and control mechanism. Such Botnet built command and control channel based on IRC protocol, this kind of Botnet is relatively mature, and has a weak security. Therefore, presently Botnet control technology is gradually transformed to P2P; they explored distributed command and control via P2P protocol to against the single point failure problem and increase robustness and concealment.

2. P2P Botnet Communication Mechanism

Communication mechanism is an important function module in botnet, it also determines the network topology, network stability and the ability against attacks of botnet.

The communication mechanisms of existing P2P botnet can be summarized for the Request mechanism and Send mechanism. The Request mechanism is based on "publish/subscribe", attacker send commands to a server definite in advance, all the bot will access command from the server. The Send mechanism is a kind of active send command mechanism, all the bot just passively waiting for orders from other bot, when a bot received the order, it will send command to other bots.

In concentrated botnet, Request mechanism is widely used. In botnet based on HTTP, the command will be issued on a web site, all the bots are predefined, periodically visit the site to access command, which is the typical request mechanism.

Compare with the Request mechanism, Send mechanism is more complex. In Send mechanism botnet, all the bot just wait passively for receiving command, when receiving the order, the order is forwarded to all of its neighbor nodes.

We can see that Request mechanism and Send mechanism botnet have the following different:

(1) In normal state, all the Request type bots periodically send query message to get command; and the Send type bots is just waiting to receive the command.

Received May 23, 2014; Revised June 29, 2014; Accepted July 10, 2014

(2) When commands are received, all the Request type bots execute the command immediately, they don't do other operations; send type bots execute the command too, at the same time they also put forward the command to all of its neighbor nodes.

(3) Request type bots may send query message with the normal P2P nodes, and Send type bots send command only with other bots.

3. P2P Botnet Communication Model

(1) Send Communication model

The P2P botnet using Send communication model mostly based on independent P2P protocol. Because the main purpose of the existing P2P botnet is to share file, P2P nodes send information to the P2P network and query the keyword that user interested in to access files. This communication mechanism does not meet the Send communication model, so P2P botnet using Send communication model need to design P2P communication protocol in order to adapt its own architecture. The hybrid P2P botnet putted forward by Wang [1], Super botnet proposed by Ryan Vogt et al [2]. and the P2P botnet proposed in the second chapter are independent protocol itself, divide nodes into two categories: the super node and the ordinary node. The Send communication mechanism is used between the super nodes, so this paper called Send communication model.

The following main characteristics of Send communication model is,

a) Use independent protocol.

b) The classification of nodes.

c) The Send communication mechanism is used in the super node layer.

All nodes in the Send-Communication-Model are divided into two types as shown in Figure 1. Red means a Super-Node, white means an Ordinary-Node. Super-node in the model is both as a server and a client. The network has a double-layer structure characteristic, each Ordinary-Node only need maintenance and a small amount of super-node connections.

This is very beneficial to improve the flexibility of the botnets, reduces the message processing time, and reduces routing number of nodes involved in the process, also reduces network traffic between nodes. In the network topology of Send-Communication-Model, Ordinary-Nodes only communicate with Super-Nodes, and the Super-Nodes are responsible for forwarding the task. Since the formation of different communication mechanisms and node selection, these botnets make up different topologies.

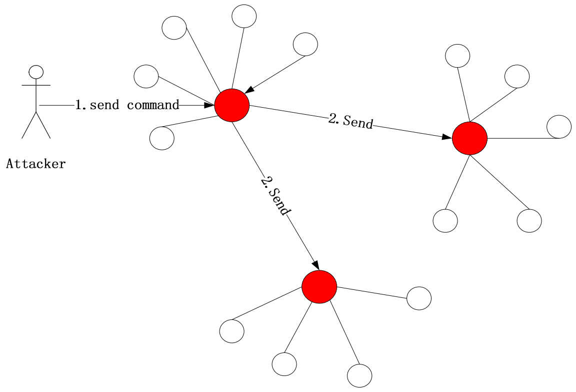

Figure 1. Send Network Model

(2) Request Communication model

The P2P botnet using type of communication mostly based on existing P2P protocol. Distributed structured P2P networks existing are: Pastry, Tapestry, Chord, CAN and Kademlia [8]. Structured solve the management mode of the network, using DHT algorithm for routing. DHT (Distributed Hash Table) algorithm is through distributed hash function map keyword input to a node, and then connected the node through the specific routing algorithm. Network node is assigned a unique node identifies (Node ID), the resource object generates a unique resource identifier by hash algorithm (Object ID), and the resources are stored in node with the same or similar NID, when querying, positioning the same method to node storing the resource.

The designers of P2P botnet [7] can easily use the P2P protocol mechanism to realize the Request communication. Designer can insert the botnet command, some predefined file name or the hash value into the records associated with the keyword. All nodes periodically find the hash value, or the file, when find the corresponding record, download the record and related commands to the local. Most P2P network using the Kademlia protocol or Kademlia protocol on similar. Overbot [3] designed by Guenther Starnberger et al, is using the Kademlia protocol to realize the communication and control function. The first version of the Storm botnet [4] described in the preceding chapters using Overnet protocol to communication, Overnet is a routing protocol based on P2P distributed hash table (DHT) of Kademlia.

In the Request communication model, there are many protocols that nodes can use: Pastry, Tapestry, Chord, CAN and Kademlia, currently most P2P botnet using the Kademlia protocol [10], so this chapter only research the network communication model based on Kademlia protocol [11, 12]. The basic architecture of Request communication model shows in Figure 2, in Figure 2 red means zombie nodes, and white means normal P2P nodes, node is normal P2P. The attacker will insert command and some predefined file name into record related with keywords of a node. Other zombie nodes periodically query the name of the file to get the command [9].

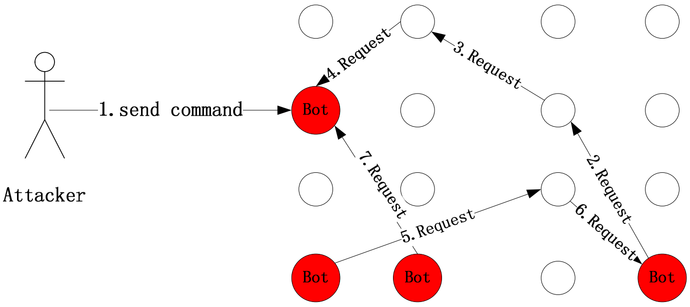

Figure 2. Request Communication Model

4. Comparison and Analysis of Model

(1) Simulation tools

Use the Peersim simulator to simulate the Request communication model on the use of Kademlia protocol. Peersim is part of the BISON [5] project, its goal is common P2P botnet simulator, to simulate the dynamic P2P protocol network, according to GPL. It supports structured and unstructured P2P network simulation, using Java development, supports two simulation ways: discrete event simulation (event-based) and cycle simulation (Cycle-based), the discrete event simulation can simulate the underlying transport layer, with high simulation precision, the cycle simulation does not take into account the layer beneath the cover, high simulation efficiency and large scale, have a good scalability. Cycle-based mode is based on CDSimulator class in the peersim.cdsim package, simplified and omit details; with good scalability and can support up to ten million level nodes; does not support transmission layer simulation (direct dialogue with nodes and protocols); does not support concurrent processing. The event-based model is based on the EDSimulator class in the peersim.edsim package, strong practicability; support the transport layer simulation; cycle-based model development can be run under the event-based engine; the efficiency is not high, supports up to one hundred thousand levels nodes.

Peersim does not have its own Implementation of any P2P protocol. But on the Peersim home page, there is a lot of Peersim source P2P protocol provided by the user, so it is simply achieve protocol and replaced protocol because of extensible and pluggable component structure characteristics. Round-robin simulation mode can achieve 1000000 overlay node scales, do better statistical functions, round-robin simulation document in more detail, discrete event simulation, and distributed simulation is not supported.

(2) Robustness

There are a lot of tools that attack network by malicious software, in the early it rarely consider the robustness of the network.For example, an attacker uses trinoo to control a large number of infected hosts, launch attack distributed denial of service, but the robust of the network is very poor. The network based on Send communication model has good robustness because of the P2P architecture. S. Saroiu et al found in the research of Gnutella network that the network has a strong robustness, when shut down 60% of the nodes, the entire network will be destroyed.

In a simulated environment, super node in Send communication model is still used random selection of its neighbor list, assuming that all nodes do not have self-repair function, namely in the closed will no longer attempt to connect to other nodes. A number of super nodes still accounted for all the nodes 15%, ordinary nodes accounted for 85%, the botnet size 10000. Random closed 1000-8000 nodes, and calculate the remaining nodes of the degree. The degree of super node directly affects the entire network topology, in the research of robustness; we have verified the differences between nodes and average degree of node directly affect the robustness of the whole network. On average size of a particular botnet, this chapter does five experiments; get the average values of five experimental results. In Figure 3, there are the average degree is respectively 2, 4, 8.

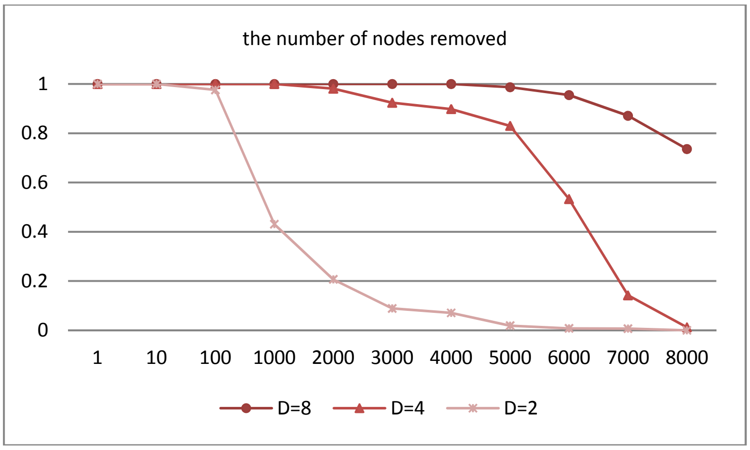

Figure 3. Robustness Analysis of Send Communication Model

As seen from Figure 3, when the size of network is constant, the robustness become strengthen with the increase of node average degree. In the Send communication model, when the node removal 80%, the achievable rate of the remaining nodes still can reach more than 70%.

In the simulation process of task communication, maintenance of communication and efficiency, every simulation process is relatively independently, and the last simulation has no effect to the simulation this time. In the robust analysis process to the Request communication model using Kademlia protocol, first set the configuration file, p_idle=0, p_rem=1, p_add=0, generates the Kademlia network with 8000 nodes. Among them

The default value of p_idle is 0; means the possibility of the node maintain the original state in a single execution process;

The default value of p_rem is 0.5; means the possibility of existing node failure in a single execution process;

The default value of p_add is 0.5; means the possibility of the new nodes join in the single execution process.

There is node failure in the course of the execution, this paper record the number of inquiry hops in the experimental implementation, as shown in Figure 4.

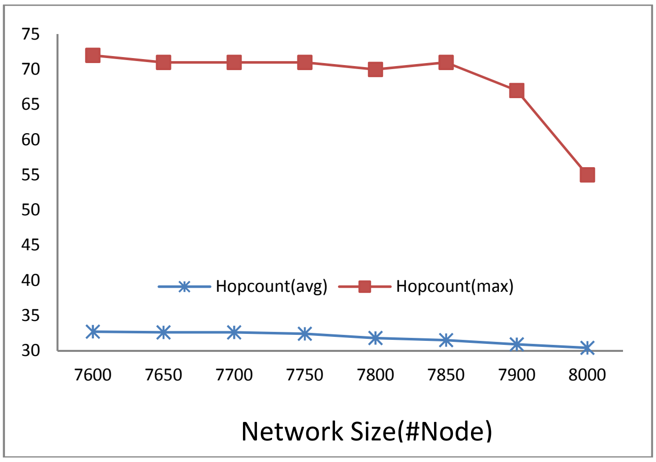

Figure 4. Robustness of Send Communication Model-query Hop

The latency of query is shown in Figure 5.

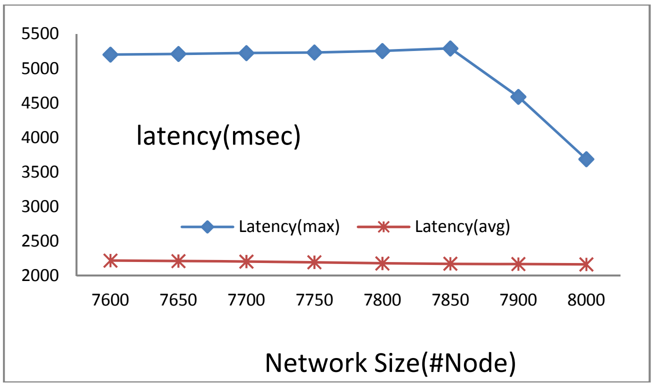

Figure 5. Robustness of Send Communication Model-query Latency

As shown in the experiments, with the increasing number of failure node, the hops of node query and query latency also increase.

(3) Efficiency

In the Send communication model, when receive the command, the super node will forward them to the neighbor, while the ordinary nodes periodically visit the super nodes to obtain the command. Therefore, its efficiency depends on two factors, the first is the number of super nodes forwarding, and the second is the cycle of ordinary nodes access. In study of efficiency evaluation index in this chapter, we found that the botnet diameter is one of the important factors affecting the super node forwarding times. In the analysis results of Gnutella topological analyzed by D.Stutzbach et al. [6], the size of existing Gnutella network can reach 800000, diameter is 11. Due to the structure of Gnutella model, most of the nodes can be reached any nodes within 6 hops. Most of the Gnutella super node contains about 30 super nodes; ordinary node generally contains 3 super nodes.

In the simulated environment, super node in Send communication model is still used random selection of its neighbor list, each super node has 30 neighbors, and ordinary node contains only 3 super nodes. A number of super nodes still accounted for 15%, ordinary nodes accounted for 85%, the number of nodes from 1000 to 10000. For size of a particular botnet, this paper does three experiments, for the three experimental; we calculate the maximum network diameter and average diameter. The experimental result is shown in Figure 6.

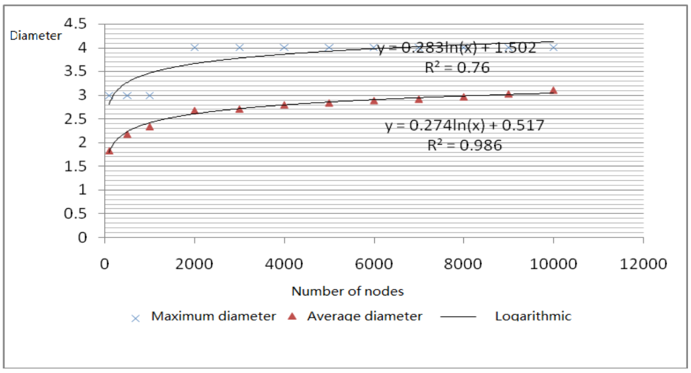

Figure 6. Network Diameter of Send Communication Model

In Request communication model using Kademlia protocol, all nodes get the command by periodically requesting the keyword information. Efficiency refers to time that spend on the command sent to every node in the network and each node obtains command, which is the maximum time of each node to search keywords. In the experiment, we still use network with 1000 to 8000 nodes, the maximum query latency and average is shown in Figure 7.

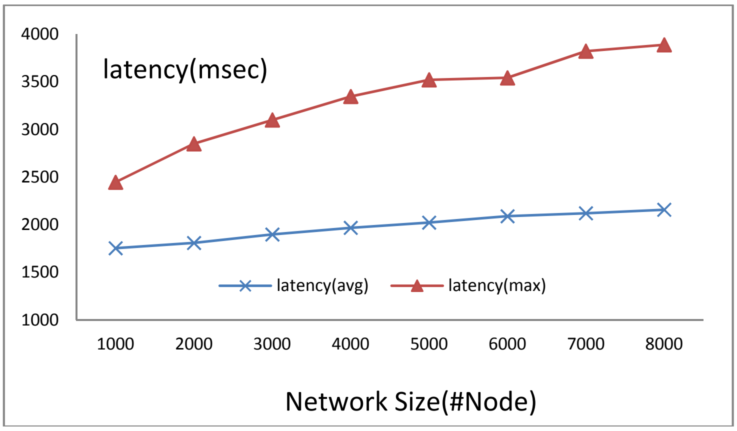

Figure 7. Request Communication Model Query Time

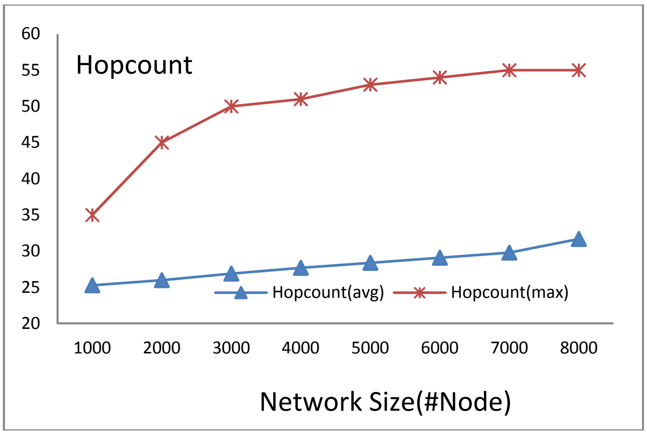

Figure 8. Request Communication Model Query Hops

During the experiment, the query hops that the nodes required are recorded, the query hops that the nodes required to some extent also show the efficiency of the network, the hop countless and efficiency is higher, the hop is more and efficiency is lower. We use the above parameters, the relationship between hops and network size is shown in Figure 8.

5. Conclusion

Based on the research and study of botnets on how to evaluate a botnet, and what a botnet assessment conducted in-depth research, we propose a comprehensive set of P2P botnetsthe evaluation index system: hidden, efficiency, effectiveness and robustness. And the two main types of P2P botnets models: Send Request communication model and communication model has been evaluated and analyzed. In the simulation process mainly analyzes the efficiency and robustness of two aspects, combined with botnet important characteristics: the botnet size, diameter, average degree, respectively, were studied for some important indexes of botnet based on peer to peer, and concern of mitigating the destructive effect of botnets [13, 14].

Acknowledgements

This work was supported by 12th fivetechnology support Program of China (Grant No. 2013BAK02B05), the People’s Public Security Univercity of China research project (Grant No. 2013JXDY22), Construction of network security online learning platform (Grant No. 2014JKF01143).

References

[1] Ping Wang, Lei Wu, Ryan Cunningham, and Cliff C. Zou. *Honeypot Detection in Advanced Botnet Attacks*. In International Journal of Information and Computer Security (IJICS). 2010; 4(1): 30-51.
[2] Ryan Vogt, John Aycock, Michael Jacobson. *Army of Botnets*. In Proc. of the 2007 Network and Distributed System Security Symposium (NDSS). 2007.
[3] LasseTrolleBorup. *Peer-to-Peer botnet: a case study on Waledac*. Mathematical Modelling. 2009
[4] Wei Yu, Philip Coyer Boyer, SriramChellappan, and Dong Xuan. *Peer-to-Peer System-based Active Worm Attacks: Modeling and Analysis*. Proc. of the IEEE International Conference on Communications (ICC). 2005.
[5] http://www.gnu.org/software/bison/
[6] Bittorrent. http://www.bittorrent.com/
[7] Moheeb Abu Rajab, Jay Zarfoss, Fabian Monrose, and Andreas Terzis. *A multifaceted approach to understanding the botnet phenomenon*. Proc. of the 6th ACM SIG- COMM Conference on Internet Measuremen, Rio de Janeiro, Brazil. 2006.
[8] Eric Rescorla. *Introduction to Distributed Hash Tables*. IAB Plenary, IETF 65.
[9] G GU, J Zhang, W Lee. BotSniffer: *Detecting Botnet Commandand Control Channels in Network Traffic*. Proc. Of NDSS, 2008.
[10] Julian B Grizzard, Vikram Sharma, Chris Nunnery, Brent ByungHoon Kang, David Dagon. *Peer-to-Peer Botnets: Overview and Case Study*. Proc. of the 1st USENIX Workshop on Hot Topics in Understanding Botnets (HotBots ’07), Cambridge, MA. 2007.
[11] Ping Wang, Lei Wu, Baber Aslam, Cliff C. Zou. *A Systematic Study on Peer-to- Peer Botnets*. In Proc. of the International Conference on Computer Communications and Networks (ICCCN ’09), San Francisco, CA. 2009.
[12] ZHUGE Jian-Wei, HAN Xin-Hui, ZHOU Yong-Lin, YEZhi-Yuan, ZOU Wei*. Research and Development of Botnets*. *Journal Of Software*. 2005; 37(1): 31-37.
[13] Somayeh Soltani, Seyed Amin Hosseini Seno, Maryam Nezhadkamali, Rahmat Budiarto. A survey on real world botnets and detection mechanisms. *International Journal of Information and Network Security (IJINS)*. 2014; 3(2): 116-127
[14] Nishikant C Dhande. Botnet Prevention Strategies for Social Network users: Cases and Remedies. *International Journal of Informatics and Communication Technology (IJ-ICT).* 2013; 2(1): 46-50