A Study on Social Network based P2P Botnet

Jian Gao[1], Meilin Liu[2]
[1, 2]People's Public Security University of China, Beijing 100038, China
Email address: gaojian[AT]ppsuc.edu.cn, 15650733742[AT]gmail.com

***Abstract*— *In order to control and detect Botnets more effectively, combined the characteristics of social network with P2P network, we proposes a new Botnet that combines the characteristics of social network with P2P network and uses microblog to replace the role of traditional Botnet command server. Firstly, we conduct a thorough research on the command and control mechanisms. The nodes in Botnet are categorized as conmmon nodes and super nodes. They have different functionalities respectively; Secondly, the topology of Botnet and the encryption mechanism of command propagation are studied. We also proposee a dynamic generation algorithm for micro-blog, and usee hard coded method to integrate it in the zombie nodes. The public key and symmetric key encryption algorithms are both used in the communication. When the attackers distribute commands, they use public key encryption. When super nodes and conmmon nodes forward commands, they use symmetric key, which will improve invisibility of botnet communication; Finally, we evaluate this botnet from efficiency, robustness and covert communication. We conclude that the efficiency and diameter of the botnet are proportional, and the robustness mainly rely on the size of the list of neighbor nodes of the super nodes. The communication concealment is based on the amount of communication in the process of sending a command.*

***Keywords*— *Botnet, P2P, Social network, Super nodes, Command and Control.*

I. Introduction

Botnets is a universal computing platform that can be remotely controlled by attackers through the construction of several non-cooperative user terminals in the network [1]. Attackers control multiple user terminals by command and control channel to carry out malicious activities, such as distributed denial of service (DDoS), spam, stealing sensitive information, which will cause a great threat to network security.

According to the basic principle of Botnet communication, we design a P2P botnet command and control model based on social network. In this paper, we use micro-blog to issue command, combining with anti-single point failure characteristics of the P2P network. When nodes of botnet is missing or shut down, the botnet that has strong robustness can communicate well. Finally, simulation analysis for the robustness and efficiency of the botnet was done. Through the analysis of the results, we can see that when the number of nodes is fixed, robustness of the entire botnet increases as the average degree of nodes increases and the efficiency increases as the nodes access cycle shrinks.

II. Background and Related Work

Traditional botnets are classified into IRC botnet, HTTP botnet and P2P botnet according to their control and command mechanisms. In the P2P botnet, relationship between the nodes is equal. It effectively overcomes the problem of single point failure. So P2P botnet has better robustness and secrecy. In [1], the author gave us the definition of botnet , key indicators, and characteristics. As the same time, it classified the botnets, and studied the propagation model and defense model. In [2], it proposed a new P2P Botnet, and the command and control channel were studied in detail. Finally, the botnet is simulated on three aspects, including robustness, efficiency and effectiveness. In my previous research [3], a botnet based on P2P is designed, which increased the robustness and concealment by using the P2P communication mode in the super nodes layer.

Recent years, with the development of social networks, botnets based on social networks are becoming popular. Koobface [4] is one of the most active botnets based on social networks. It can use Facebook, Myspace and many other social networks as its command control channel. Nazbot5 and Twebot6 are two botnets using Twitter as their command control channels. Stegobot [5] combines data steganography with the command process. It communicate by means of hiding information in pictures and distribute it via social networks. ASP2P [6] botnet adopts the communication mechanism of social network and P2P. It divided botnet nodes into two types: Servent bot, and Client bot. On the Servent bot layer, the P2P communication mechanism is used.

Social networks are widely used in China. Such as Renren, Post Bar, QQ space and Sina Micro-blog. In the morning of May 12, 2016, Sina micro-blog monthly active users reached 261 million. There are three important reasons for selecting micro-blog as the command control channel. First, the number of users is relatively large and Sina micro-blog is widely used. Secondly, users can read micro-blog content from other users without logging in. Third, Sina micro-blog has released its two development interface, facilitating the operation of botnet nodes through opening API.

The differences between the social botnet in this paper and other social botnets are mainly embodied in the following aspects:

Not every botnet node has a micro-blog account alone. Only the controller registers a micro-blog account and issues the command content through the account.

The controller and the botnet nodes use the same algorithm to generate the micro-blog account. The controller registers the account via an anonymous network. The botnet nodes read encrypted commands from the micro-blog through an open network.

Not every botnet node regularly requests content from the micro-blog account. Only super nodes query it according to fixed cycles.

Command transfers between super nodes are carried out through the P2P network. Common nodes get commands through the super nodes.

The botnet designed in this paper has a complete encryption mechanism. The controller issues an command through the micro-blog account that is encrypted by public key. Command transferred between botnet nodes are encrypted by symmetric keys.

III. Command Control Channel Design

Command and Control is the most critical link of Botnet communication, referred to as C&C. The topology of the entire botnet, classification of nodes, encryption and decryption mechanism, nodes state transition, command transfer format, and so on all need to be implemented in the command control channel design.

A. Design of C&C Communication Protocol

The botnet designed in this paper replaces the command server with random micro-blog. It greatly improves the concealment of the entire botnet command sending. It is difficult to trace the origin. The previous design of the botnet divided nodes into two categories, super nodes and common nodes.

The common nodes get the command process as shown. The common nodes select its member request in the list of the super nodes first when it gets the command, if all the super nodes cannot connect, it would obtain it from the designated micro-blog using the micro-blog algorithm. This method prevents all botnet nodes from accessing one or more micro-blog addresses and increase the invisibility.

The type of command of botnets can be divided into two categories. One kind of them needs feedback. For example, it needs a botnet to send back the collected word files or a keystroke record. This command requires additional feedback addresses (temporary mailbox, FTP server, seed) after the command signature; The other kind does not need feedback. For example, DDoS attack. The botnet nodes get the command and perform attack at the specified time, or downloads the update command and downloads the function module or the main update module on the specified server. For commands with feedback information, the function modules can be published online in BT mode, and then generated short addresses. Then, it will be encrypted and released in micro-blog. In the whole process, the short address generation and the release of function modules are realized by the third party platform, so as to reduce the possibility of traceability.

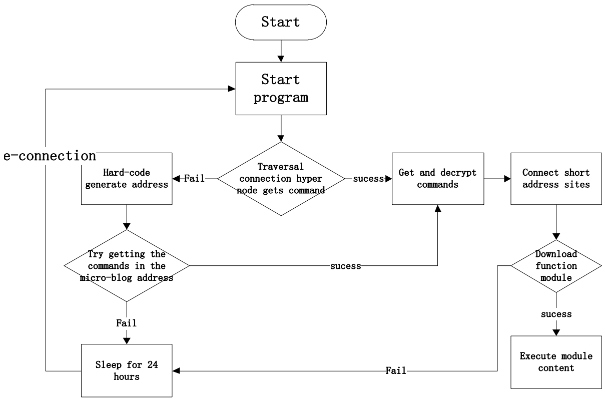

Fig. 1. Flow chart of common nodes request command.

The process of super nodes getting the command as shown. The super nodes not only have the function of getting commands from micro-blog, but also undertakes the responsibility of getting the command request from the common nodes and the function of forwarding the command to other super nodes. Therefore, in the process of nodes selection, there are more requirements, such as fixed IP, open ports, online time, configuration and so on.

When the super nodes get the command, it gets preferentially from the random micro-blog address. If it receives the commands forwarded by other super nodes within a given time, it can be in a sleep state and wait for the next time to get a command. The process of downloading function module after the obtained command of super nodes is the same as the common nodes.

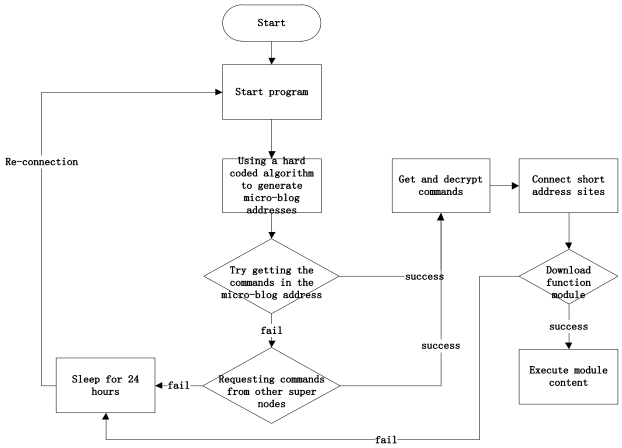
Start
Start program
Re-connection
Using a hard coded algorithm to generate micro-blog addresses
Get and decrypt commands
Connect short address sites
Try getting the commands in the micro-blog address
success
Download function module
fail
success
success
Sleep for 24 hours
fail
Requesting commands from other super nodes
Execute module content
fail

Fig. 2. Flow chart of super nodes request command.

B. Topological Structure

This paper designs a P2P botnet based on social network, which uses the multi-layer command communication mechanism that the controllers send command anonymously and the botnet nodes requests command actively. When a controller needs to send a command, it uses the microblogging generation algorithm prepackaged, and uses time as the seed to generate the day's weibo account, and registering it by the anonymous network. And then their encrypted command will be released into the accounts.

The super nodes use the communication mechanism of P2P. It uses the same micro-blog name generation algorithm and generates the same micro-blog name as the controller. At the same time, it takes the initiative to obtain the encryption command from a micro-blog account through the HTTP request. Thus, the micro-blog accounts that every time controllers send a command and the botnet nodes request are not immutable. It increases the difficulty of tracking and tracking the entire botnet by the defender and will make up for the lack of command servers in the P2P botnet model [3], which was proposed previously. The super nodes forward the command after they get the command. The forwarding, encryption and decryption mechanisms of nodes are described in detail in the next section.

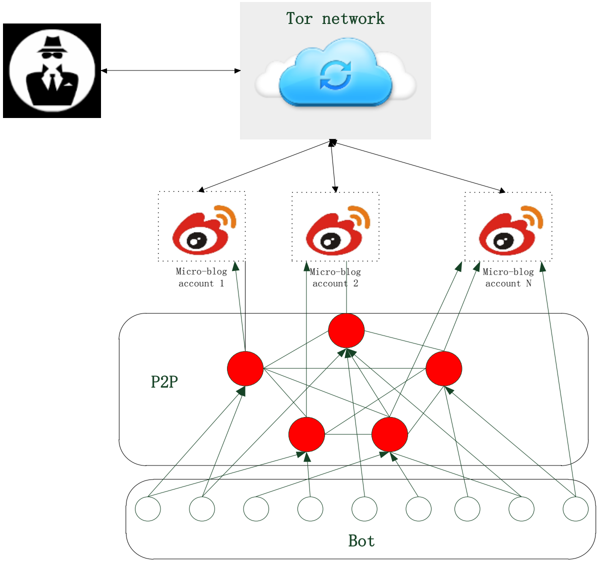

Fig. 3. Botnet topology.

C. Communication Encryption Mechanism

The controller generates a pair of keys through the RSA algorithm. PuK is the public key and PrK is the private key. PuK is encapsulated in the botnet by hard coding. The controllers and the botnet nodes have the same micro-blog account generation algorithm, which takes the date as seed and generates an account.

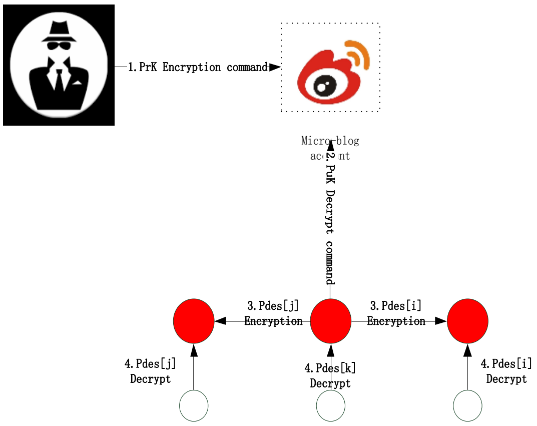

Fig. 4. Botnet communication encryption mechanism.

1. The controller encrypts command by PrK and issues it to micro-blog.
2. Super nodes to obtain the cipher text in micro-blog by Http. The super nodes decrypts the command by PuK encapsulated in the code and approves this command (using public-key cryptography algorithms can prevent other attackers from using a fake command to host the botnet [6]). When a super node distributes commands to other super nodes, or a common nodes requests a command from a super nodes, the DES algorithm is used to encrypt and decrypt commands. The botnet nodes will generate a random secret key in the first run. The key is used to encrypt and decrypt commands. The list of neighbor nodes of a super nodes and the list of super nodes of an common nodes contain three main elements $< IP_i, Port_i, Pdes_i >$, IP address, port number and nodes key.
3. The super nodes use the other's secret key to transmit the commands when they distribute commands to other super nodes.
4. When a normal node requests a command from a super nodes, the command is decrypted using the secret key of that super nodes.

The secret key of every super node are randomly generated, and not the same. It can prevent other attacker from obtaining entire network communication after getting a secret key of a nodes.

IV. Experimental Evaluation

Many scholars have studied the evaluation index of Botnets.

Andbot and SUbot were studied in the literature [7], [8]. The botnet was evaluated in resilience, power consumption, and stability. Document [9] drawn on the advantages of CVSS proposed a new evaluation model for botnet CBSS. Document [10] simulates the P2P botnet from a complex network perspective. It used robustness as an index to evaluate the condition of a nodes when it fails. In my previous research [11], I proposed a botnet evaluation model and its evaluation indicators, including robustness, concealment, efficiency and effectiveness.

The effectiveness is largely dependent on the size of botnets. The quality of command control channels cannot be embodied. Therefore, this article mainly takes the other three indexes as the indicators of evaluation.

A. Experimental Environment

In this paper, we use Visual Studio 2010 as the development environment, and use C# language to simulate the efficiency, robustness and communication concealment of Botnet communication. Defines the member of the common nodes as the reference class , such as: heartbeat time, member list, public key. Super nodes inherit from common nodes and have active functions of active sending and passive requested. To save system resources, all nodes only have analog communication functions, and system function is not available, such as: send traffic, collect documents and so on.

B. Efficiency Evaluation

Efficiency refers to the time of each nodes received commands sent from the controller. It shows how fast the nodes in the botnet get commands. Efficiency is closely related to communication concealment. The higher the efficiency, the shorter the nodes access cycle, the greater the amount of traffic generated. The longer the node's access cycle, the lower the efficiency, and the smaller the amount of traffic generated simultaneously.

Assume that the time from controller sending command to all nodes to all nodes accept the command is T. Diameter represents the maximum distance between any two super nodes. Use D to represent the diameter of the entire botnet. The diameter D can be expressed as:

$$D = \max(d(N_i, N_j)) \qquad (1)$$

N_i and N_j are any two points in super nodes.

In P2P Botnet, diameter is used to measure efficiency. We can ignore the encryption and decryption time and forwarding time of the command. At the same time, it's safe to ignore the difference in heartbeat time between nodes. It's more scientific only to consider it taking how many jumps between nodes to each other.

In the process of simulation, the number of nodes varies from 1000 to 10000. Super nodes account for 20% of the total. Common nodes account for 80%. All super nodes can randomly select their neighbor nodes. The size of neighbor nodes is represented by L. L can be 30 or 20. The number of nodes in the list of super nodes in common nodes is 3-5.

In the process of simulation, super nodes selection strategy can lead to larger changes in jumps. Stochastic model selection method is adopted in this paper. As we can see, with the increasing of the L, the number of jumps required for messaging is decreasing. At the same time, as the botnet grows, the number of jumps is increasing. Therefore, during the designing of the P2P botnet, increasing the list size of the super nodes in moderation can improve the efficiency of the entire network to obtain commands. However, if the list size of each super nodes is increased simply for increasing efficiency, it will increase the traffic, and the concealment of botnet will be weakened.

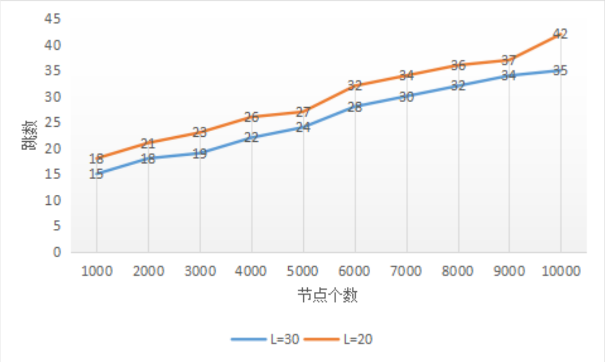

Fig. 5. Command efficiency.

C. Robustness Evaluation

Robustness refers to the extent that some nodes lost for some reasons affect the entire botnet communication, and whether the communication of other normal nodes in the botnet is interrupted because of some lost nodes.

We use connection rates to represent the robustness of the entire botnet. Connection rate is the proportion of the remaining nodes in the connected state and all the remaining nodes after the botnet deletes a part of the nodes.

This paper uses $C(p)$ representative connectivity, $C(p)$ is as follows:

$$C(p) = \frac{\#number \cdot of \cdot bots \cdot connected \cdot in \cdot botnet}{number \cdot of \cdot remaining \cdot bots} \qquad (2)$$

In the process of simulation, we generate 5000 botnet nodes, The proportion of super nodes and regular nodes is still 1:4. The neighbor nodes list size of the super nodes is 8. For fully randomized networks and two layer P2P network our proposed, we randomly delete part of the nodes and test the connection rate of the whole network.

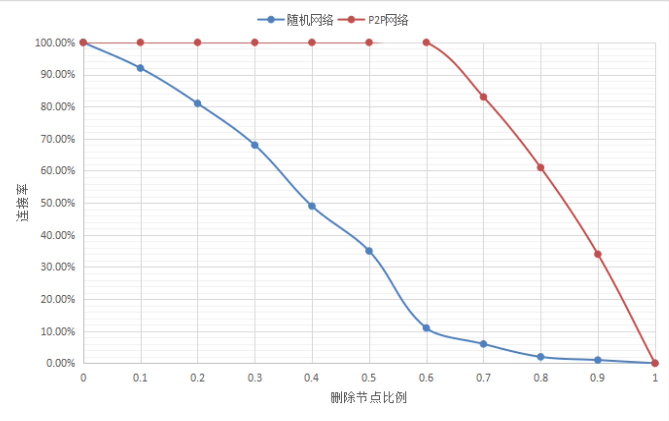

Fig. 6. Botnet robustness.

As we can see from the graph, in the random network, as the nodes deleted, the connection rate of the whole network is gradually decreasing. However, layered P2P networks can still show good robustness in the face of massive loss of nodes. The main reason is that the deletion of common nodes has no effect on the connectivity of the whole network. The entire network connectivity is affected only when enough scale super nodes are removed.

D. Communication Concealment Evaluation

Communication concealment depends on the size of the traffic generated by the entire botnet and the encryption mode of the communication. It represents the amount of traffic generated by a command, that is, the traffic generated from the controllers issuing an command to all nodes obtaining the command.

Suppose the number of nodes in botnet is N. Among them, the number of common nodes accounted for 80%. Therefore, the number of common nodes is $80\% * N$. According to the command communication mechanism of the common nodes, only one request command is selected randomly from the list of its own super nodes, so no additional traffic is generated. At the same time, on the assumption that the number of super nodes is $20\%*N$. Because the super nodes follow the principle that the same command does not forward or return, and the common nodes command communication mechanism is different. So the super nodes layer formed an undirected graph. According to the theorem of graph theory, in undirected graphs, the sum of the nodes is two times the number of edges. At the same time, the average degree of the super nodes is set to D. In practice, the number of super nodes can be adjusted dynamically. The degree of each super nodes is not necessarily the same. As follows:

$$D = \frac{\sum_{i=1}^{N} d_i}{N} \qquad (3)$$

d_i indicates the degree of the super nodes i. According to the previous setting, we can calculate the amount of communication when the task is delivered. The amount of communication is:

$$T = 80\%*N + 20\%*N*D/2 \qquad (4)$$

The figure shows the amount of traffic that a botnet issue a command when the number of nodes between is 1, 000 and 10, 000 and the average number of super nodes is 30 and 20. As can be seen from the diagram, the common nodes have little influence on the overall traffic, and the degree of the super nodes is the main contributor that affects the total amount of communication.

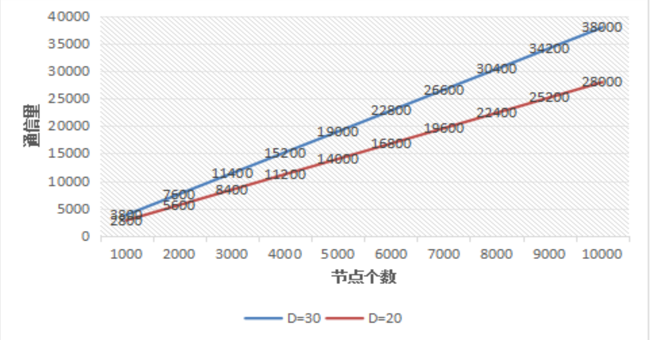

Fig. 7. Botnet traffic.

V. Conclusion

Botnets are at a stage of rapid development, often with DDoS attacks and stealing personal private information, which is very harmful to the society and individuals. In order to avoid detection and traceability, botnet is changing its topology and command control mechanism. This paper studies a new type of P2P botnet based on social networks and its communication mode. Besides, we evaluated its efficiency, robustness and invisibility. The next phase of the research focuses on: (1) Analyze and identify Domain-Flux botnet through traffic, and propose the detection scheme; (2) Investiage the defense and blocking method of command control channel based on mobile botnet; (3) Study Mirai botnet samples and network data streams and analyze its detection method .

Reference

[1] Fang Binxing, Cui Xiang, Wang Wei, "A survey of Botnet," *Journal of Computer Research and Development*, vol. 48, issue 08, pp. 1315-1331, 2011.

[2] P. Wang, S. Sparks, and C. C. Zou, "An advanced hybrid peer-to-peer botnet," *Proceeding HotBots'07 Proceedings of the first conference on First Workshop on Hot Topics in Understanding Botnets*, USENIX Association, pp. 2-2, 2007.

[3] Gao Jian, Lu Tianliang, and Wang Wei, "Botnet based on P2P C&C design and simulation," [J]. People's Public Security University of Chian: Natural Science Edition, vol. 21, issue 3, pp. 65-70, 2015.

[4] K. Thomas and D. M. Nicol "The Koobface botnet and the rise of social malware," *IEEE 5th International Conference on Malicious and Unwanted Software*, pp. 63-70, 2010.

[5] S. Nagaraja, A. Houmansadr, P. Piyawongwisal, Vijit Singh, Pragya Agarwal, and Nikita Borisov, "Stegobot: A covert social network Botnet," *International Workshop on Information Hiding*, vol. 6958, pp. 299-313, 2011.

[6] L. Cao and X. Qiu, "ASP2P: An advanced botnet based on social networks over hybrid P2P," 22nd *Wireless and Optical Communication Conference*, pp. 677-682, 2013.

[7] Cui Xiang, Fang Binxing, Yin Lihua, Liu Xiaoyi, and Zang Tianning, "Andbot: Towards advanced mobile Botnets," *LEET'11 Proceedings of the 4th USENIX conference on Large-scale exploits and emergent threats*, 2011.

[8] Wang Shuai, Cui Xiang, Liao Peng, Li Dan, "S-URL flux: A novel C&C protocol for mobile Botnets," *International Conference on Trustworthy Computing and Services*, pp. 412-419, 2013.

[9] Li Xuefeng. "Research on botnet architecture of P2P," [D]., Tsinghua University, 2011.

[10] Xu Xiaodong, Cheng Jianguo, Zhu Shirui. "Robustness analysis of unstructured P2P botnet," *Journal of Computer Applications*, vol. 31, issue 12, pp. 3343-3345, 2011.

[11] Gao Jian, "Research on botnet based on P2P and key technology," [D]. Beijing University of Posts and Telecommunications, 2011.

GAO Jian, born in 1982, Ph. D. professor. He research interests include Botnet, Malware, Cyber crime, and ne