Study on Advanced Botnet based on Publicly Available Resources

Jie Yin[1,2], Heyang Lv[3(✉)], Fangjiao Zhang[1,2], Zhihong Tian[4(✉)], Xiang Cui[1,4]

[1] Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China
[2] School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China
[3]Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China
[4]Cyberspace Institute of Advanced Technology, Guangzhou University, Guangzhou, China
heyanglv@126.com,tianzhihong@gzhu.edu.cn

Abstract. In recent years, botnets continue to be an ever-increasing threat on the Internet. To be well prepared for future attacks and ensure the cyberspace security, defenders take more attention on advanced botnet designs that could be used by botmasters in the near future. In this paper, we design an advanced botnet based on publicly available resources, and implement its prototype system, which is named as PR-Bot. Simultaneously, the targeted defense measures are provided. First of all, in terms of system design, PR-Bot is completely constructed based on the third-party publicly available resources and supports the bidirectional communication between the control end and the controlled end. At the same time, the system's command and control (C&C) channel consists of three sub-channels: command control channel (CC channel), command addressing (CA channel) and result feedback (RF channel), making it extremely robust and concealed. Secondly, in terms of defense technology, aiming at the intrinsic weakness of PR-Bot, this paper proposes the targeted defense strategies from the perspective of detection, measurement and tracking, so as to achieve the goal of combating against such botnets. In short, the ultimate purpose of this paper is not to design a highly harmful botnet, but to accurately predict the techniques that the botnet may adopt in the future and assess its new threats from the point of attack and defense.

Keywords: Publicly Available Resource, Command and Control, Bidirectional Communication, Defense Technology.

1 Introduction

1.1 Background

A botnet refers to a group of compromised computers that are remotely controlled by a botmaster via C&C channels [1]. Based on botnets, multiple types of Internet attacks can be initiated, such as: DDoS (Distributed Denial of Service), Email Spam, Bitcoin or Monero Mining, etc. At present, the studies on botnets can be summarized into two aspects: attack technology and defense technology. The purpose of studying attack technology is predicting the attack trends and techniques of future botnets, so as to prevent the possible emerging botnet activities; and the purpose of studying on defense technology is improving the detection efficiency of botnets and discovering the botnets that are already in the cyberspace but not yet exposed in a timely manner, so as to reduce the actual harm caused by them.

In the early days, attackers usually controlled the bot based on the IRC [2, 3] or HTTP [4, 5] protocol. This centralized architecture is simple, efficient and highly interactive. However, its main disadvantage is: once the C&C server is discovered, the defender can close it through technical or coordinated means, causing a serious single point of failure. Although a modified architecture based on the Domain-Flux [6] or Fast-Flux [7] protocol that appeared later can eliminate the problem of single point of failure certainly, it may be attacked by Sinkhole [8]. Once the DGA (Domain Generation Algorithm) adopted by botnets is reversed (which is unavoidable), the defender can register the domain name in advance, making the bots access the forged C&C server built by the defender.

In order to make up for the deficiency of centralized botnets, botnets using P2P protocols as C&C channels have also evolved. In a P2P botnet, each infected host can act as both a client and a server. Based on the distributed features of P2P protocols, the botmaster can issue commands at any node, so it can hide the real address of the C&C server and effectively solve the single point of failure. However, P2P botnets are not perfect, which still have inherent weakness. For example, the structured P2P botnets, such as Storm [9, 10], are vulnerable to Index Pollution attack and Sybil attack, and its scale is easy to be measured by Crawler and Sybil nodes; the unstructured P2P botnets usually communicate by the way of random scanning or peer-list, the former has the inherent weaknesses of flow anomaly, and the latter is vulnerable to Peer-list Pollution attack.

In recent years, the new generation of botnets based on social network have been proposed, such as: Koobface [11], Stegobot [12], etc. Among the social botnet, each social account is a control node, which is equivalent to a C&C server in the traditional botnet, and is used to transfer the commands between the botmaster and individual bots. Although the social botnet can hide the malicious traffic within the normal legitimate traffic, the social platforms are generally only applicable for the botmaster issuing commands and cannot be used by bots to send back harvested information, especially file information. Moreover, for the botnet based solely on the social platform, its C&C channel is relatively simple, and it can be easily detected and destroyed by the defender. Therefore, in order to make up for the inadequacies of social botnets, this paper proposes an advanced botnet based on multiple publicly available resources.

1.2 Contribution

The goal of this paper is to study the development trends of future botnets, increase the defenders' understanding of the advanced botnet, and promote more effective cyber defense to deal with the possible similar cases.

The contributions of this paper mainly include three aspects:

1) Based on the idea of "severless botnet", an advanced botnet based on publicly available resources is designed. The system adopts a three-channel scheme, and each

sub-channel can be supported by multiple publicly available resources and extended in the form of plug-in.

2) We have tested five categories and 37 websites, and the application scenarios of each website when constructing C&C channel are discussed. Meanwhile, the operation flow between the botmaster and individual bots is analyzed and described in detail to verify the feasibility of the proposed model.

3) We have analyzed the attributes and weaknesses of the PR-Bot, and propose a practical targeted defense scheme that covers detection, measurement, and tracing.

2 The Design of PR-Bot

2.1 System Overview

Definition 1 (Publicly available Resource Botnet). In this paper, we believe that all botnets that construct C&C channels based on the publicly available resources (including but not limited to: social network, URL shortener, image hosting, online clipboard or cloud disk, etc.) could be called the Publicly Available Resource Botnet.

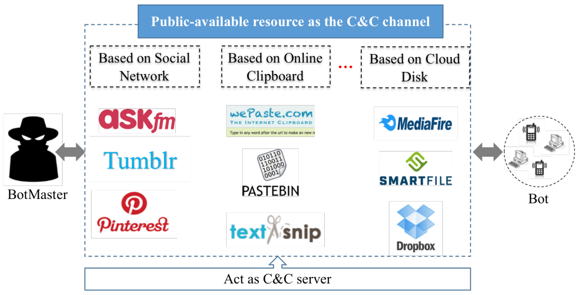


Fig. 1. Basic characteristics of Publicly Available Resource Botnet

The basic characteristics of this type of botnets are: the botmaster no longer relies on the self-built C&C server to control the bots, but uses the open and free website on the Internet to act as the C&C server. All communication flows are transferred through the Internet publicly available resources.

2.2 System Design

In botnets, no matter how complex the control model of botnet is or how powerful the bot program is, the interaction between the control end and the controlled end usually involves only the transfer of text information (that is: string content) or file information (that is: binary content), as shown in Table 1.

Table 1. Interactive information between the control end and the controlled end

	Text information	File information
Control end	command or others	malicious program
Controlled end	callhome or others	stolen files

Although there are many kinds of publicly available resources on the Internet, as long as the information can be released, there is a possibility that an attacker can abuse it to act as the C&C server. However, due to the limitations of the nature of the publicly available resources, not all publicly available resources are suitable for issuing both text information and file information. For example, the social platforms used by social botnets are generally only applicable to store the commands issued by the botmaster, but not applicable to store the harvested information sent back by individual bots, making it a one-way communication channel.

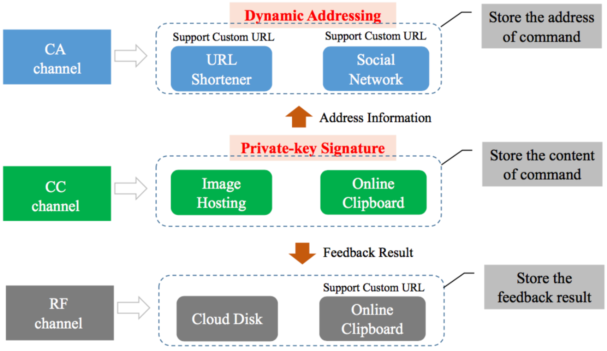


Fig. 2. The control model of PR-Bot

The PR-Bot, an advanced botnet designed in this paper, takes into account the limitations of a purely social platform as C&C channels, so it combines multiple publicly available resources and uses their respective advantages to construct C&C channels. The PR-Bot is suitable for transmitting both text information and file information, as well as supporting two-way communication between the botmaster and bots. All in all, the PR-Bot adopts the three-channel scheme, and the interaction information in each channel is distributed in different locations of cyberspace, as shown in Figure 2.

2.3 System Architecture

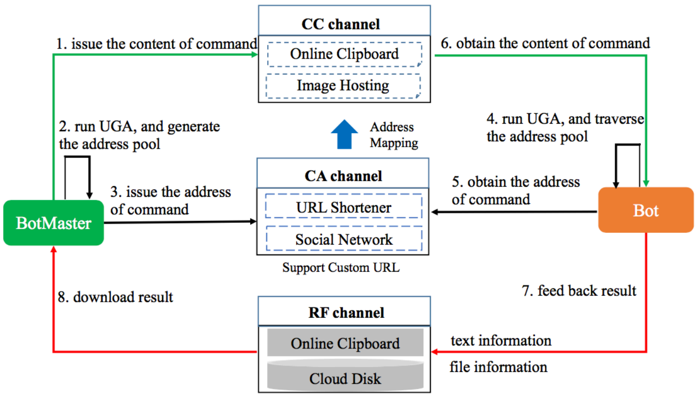


Fig. 3. The system architecture of PR-Bot

The architecture of PR-Bot is shown in Figure 3, whose communication between the botmaster and bots includes the following six stages:

1) Botmaster issues the content of command: The botmaster issues the content of command to the publicly available resources, such as an online clipboard or image hosting website, and records the URL of the website (abbreviated as PR_Address_A) where the command is located. Among them, for the online clipboard, the command is issued in the form of string; for the image hosting, the command is first converted into a picture and then issued in the form of picture.

2) Botmaster issues the address of command: First, the botmaster selects a URL shortener or social network that supports customized URL, designs and runs the Username Generation Algorithm (UGA) [13], followed by selecting several candidate addresses (abbreviated as PR_Address_B) from the URL address pool. And then, the botmaster issues the address PR_Address_A recorded in Stage 1 as content to the website corresponding to the address PR_Address_B.

3) Bot obtains the address of command: After the bot infects the controlled end, to establish the communication with the control end, it will first run the UGA algorithm consistent with the botmaster, and then traverse the URL address pool one by one. When an address (take PR_Address_B as an example here) is found to be accessible, it is considered that address of command is stored at this place, and then the address PR_Address_A will be extracted.

4) Bot obtains the content of command: After the bot has obtained the address PR_Address_A in Stage 3, the content of command is then obtained from this address. After obtaining the content of command, the bot first verifies its validity and availability, and only the command that passes verification can be executed. Otherwise, the control logic of the bot program will jump to Stage 3 and run again. Among them, “validity” refers to whether the command is within the validity period specified by the botmaster; “availability” refers to whether the command is signed by the private key of the botmaster.

5) Bot feeds back the result information: For the command that needs to feed back the result information, the botmaster is required to specify the receiving address as a parameter in the issued command. The bot will feed back the relevant information based on this parameter in the command and the customized protocol with the control end. Among them, for the “callhome” command, the bot needs to feed back the basic information of the infected device, such as: system type, internal and external network IP, and host name, etc; for the “file stealing” command, the bot needs to feed back the files of the infected device.

6) Botmaster downloads the result information: After a certain period of time, for the command that will feed back the result information, the corresponding result acquisition module will be run. During the operation of the module, it will download the text information (such as callhome information) or file information (such as stolen files) returned by the bot for further analysis, which is based on the address parameter in the issued command and the customized protocol with the controlled end.

What needs to be explained here is, during the acquisition of the command, the bot does not first locate the address (that is PR_Address_A) where the content of command is located, but the address (that is PR_Address_B) where the address of command is located. Based on this address PR_Address_B, the bot will know the address PR_Address_A and further obtains the actual content of the command. In other words, the bot will obtain the content of command through the "Secondary Addressing Mechanism", which will be described in detail in 3.2. To a certain extent, this mechanism can effectively improve the robustness and flexibility of the C&C channel.

2.4 The Standard for Selecting Publicly Available Resources

As shown in Table 2, this paper tests five major types of publicly available resources and analyzes the specific application scenarios of each when constructing C&C channel. Among them, PR-Bot stores the content of command based on online clipboard or image hosting website; stores the address of command based on a URL shortener or social network that can customized URL address, stores the stolen files based on public cloud disk website, stores the callhome information based on online clipboard that can customized URL address, so as to provide support for the requirements of CC channel, CA channel, and RF channel.

Table 2. Application scenarios of publicly available resources

Type	Is it suitable for issuing text?	Is it suitable for issuing file?	Is it suitable for issuing picture?	Application scenarios
Online Clipboard	Y	N	N	store the content of command store the callhome information
Image Hosting	N	N	Y	store the content of command
URL Shortener	Y	N	N	store the address of command
Social Network	Y	N	Y	store the address of command
Cloud Disk	N	Y	Y	store the stolen files from bot

1）Publicly available resource for storing the content of command

Table 3. Online clipboard and its selection standard

Name	Site	Customized URL	Storage Space	Storage Time
dpaste	http://dpaste.com/	N	unknown	1 year
pasted	http://pasted.co/	N	unknown	unknown
pastebin	http://pastebin.com/	N	512KB	unlimited
wepaste	http://www.wepaste.com	Y	unknown	unlimited
cl1p	https://www.cl1p.net/	Y	unknown	1 time
textsnip	http://www.textsnip.com	Y	70000 characters	unknown
showtxt	http://showtxt.cn/	Y	unknown	unknown

In the CC channel, when selecting an online clipboard, as shown in Table 3, PR-Bot only considers two factors: one is the size of the space for storing information, and the other is the length of time for storing information. And at this stage, it does not consider whether the website supports customized URL. As long as the storage space can reach 200KB and the storage time can reach 1 month, it means that the requirements is met.

In addition, for the image hosting website, as shown in Table 4, PR-Bot only selects the websites that have no registration required and do not compress the pictures. And the size of signal picture allowed to upload should meet the requirements of 1MB. Moreover, because the storage time supported by the image hosting website is usually unlimited, so this factor was not taken into account here.

Table 4. Image hosting and its selection standard

Name	Site	Registration	File Size Limit	Is it compressed?
baidu-pic	http://image.baidu.com	N	5MB	N
360-pic	https://st.so.com/	N	2MB	Y
Imgbb	https://imgbb.com/	N	16M	N
sm.ms	https://sm.ms/	N	5MB	N
upload.cc	https://upload.cc/	N	5MB	N
Imgur	https://imgur.com/	N	1MB	N
sina	http://photo.weibo.com/	Y	20MB	Y
qiniu	https://www.qiniu.com/	Y	unlimited	N

2) Publicly available resource for storing the address of command

Table 5. URL Shortener and its selection standard

Name	Site	Customized URL	Storage Time
tinyurl	https://tinyurl.com/	Y	unlimited
is.gd	https://is.gd/	Y	unlimited
yep.it	http://yep.it/	Y	unlimited
shorturl	https://shorturl.com/	N	unlimited
shorl	http://shorl.com/	N	unlimited
bit.ly	https://bitly.com/	N	unlimited

Table 6. Social network and its selection standard

Name	Site	Customized URL	Temporary Mailbox
tumblr	https://www.tumblr.com	Y (Revisable)	Y
pinterest	https://www.pinterest.com	Y (Revisable)	Y
ask.fm	https://ask.fm	Y	Y
twitter	https://twitter.com	Y (Revisable)	N
facebook	https://www.facebook.com	N	N
weibo	https://weibo.com	N	N
qzone	https://qzone.qq.com/	N	N
renren	http://sns.renren.com/	N	N

In the CA channel, we test some services and show selection standard in Table 5 and Table 6. For the URL shortener website, because it itself stores the mapping relationship between the long URL and the short URL, and the storage time is usually unlimited, PR-Bot does not have too many restrictions when selecting such publicly available resources, as long as it supports customized URL for dynamic addressing. Similarly, for social network website, in order to achieve the dynamic addressing, PR-Bot only selects the social platform that can customized homepage URL based on the user name. In addition, priority is given to websites that support temporary mailbox registration in order to avoid exposing too much real information about the botmaster.

3) Publicly available resource for storing the feedback result information

Table 7. Public cloud disk and its selection standard

Name	Site	File Size Limit	Storage Time
sendspace	https://www.sendspace.com	300M	30
fileden	http://fileden.net/	100M	60
senduit	http://www.senduit.com/	100M	7
zippyshare	https://www.zippyshare.com/	500M	30
rapidshare	http://www.rapidshare.com.cn/	100M	30

In the RF channel, for the online clipboard website, except for the storage space of up to 200 KB and the storage time up to 1 month, the website that supports customized URL is required, for making that the botmaster can find the result information fed back from bot by a certain rule (URL+ numeric string). For the public cloud disk, as shown in Table 7, PR-Bot mainly considers two factors: one is the file size limit, and the other is the storage time, which is similar to the image hosting. If the website allows 10M-sized files to be uploaded and storage time can be up to 1 month, it is enough. In addition, the public cloud disk is available without registration.

3 The Implementation of PR-Bot

3.1 CC Channel

PR-Bot mainly supports two types of commands, the "callhome" and "file stealing", which parameters are shown in Tables 8.

Table 8. The parameters of commands

Command	Key	Value	Remark
callhome	cmd_type	callhome	command type
	paste_name	wepaste	website name
	address	http://www.wepaste.com/abcde	url address
file stealing	cmd_type	upload_file	command type
	cloud_name	sendspace	website name
	paste_name	wepaste	website name
	file_type	doc	file type
	address	http://www.wepaste.com/abcde	url address

In order to prevent C&C hijacking and replay attacks, before issuing the command, the botmaster will specify the validity period of the command and sign the command based on the private key. The format of the command to be issued is as follows:

Base64 (Base64 (private key signature (original command ^ validity period))#original command ^ validity period)

Take the “callhome” as an example, its original content is a string in JSON format, which is as follows:

```
{
    "cmd_type":"callhome",
    "paste_name":"wepaste",
    "address":"http://www.wepaste.com/abcde"
}
```

First, after specify validity period, private key signature, base64 encoding and string splicing, the corresponding content is as follows:

```
bqi+YmecF62Li+INvT4xRdO8d6Z7GXY74E8qVKYwHNrCYUY7wGEuLHrm5bdfyeuQ4S7BH5fx
zIQmsQn9xY/+iPjzXv9ap/mZefCY5JCpzQ6X/uLsQPYulvihTJZ52deiQZvPRwWZtPwSCBu3si1Pga
N/mxs8eWSg17PGuWD1L37/TT7BvG+IrdozFqBQF5kILlX3hahEIqsh7DRJpDpwyfCH5Nz2K5zm7
X3apA34Sz1OL1vNxfHML2TZtNBR2LTCgtaXWa9JNiszHulPEF7pgHdTxWLuqQ/lG5Tc9/5LN04y
7evFnJETMPK+WLV4mRaFwkjlx3c6kHkZJ64eyI5qiQ==#{"cmd_type":"callhome","paste_name":"w
epaste","address":"http://www.wepaste.com/abcde"}^2018_05_12-2018_06_12
```

Next, if selecting the online clipboard website to store the command, it only needs to issue the above content in the form of text to the online clipboard website, which corresponding address looks like http://dpaste.com/0SV8NS5. But if selecting the image hosting website to store the command, the above content shall be converted into picture firstly and then issued, which corresponding address looks like http://h.hiphotos.baidu.com/image/pic/item/c8177f3e6709c93db7a0d055933df8dcd00054c6.jpg. In the process of storing data based on picture, PR-Bot does not embed the command into an existing picture, but directly converts it into pixels, and stores the original content in the form of pixels. The converted image style is shown in Figure 4.

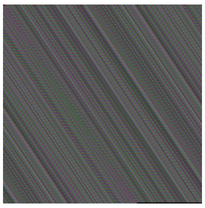


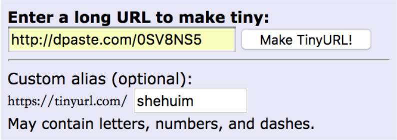
Fig. 4. Image style generated from text

Normally, each pixel in a colored image is composed of three color information of RGB. Each color information occupies 8 bits, and the three colors are 24 bits, which means that each pixel can store 3 bytes of data. For a 500*500 RGB picture, it can store 75000 bytes of data, about 730KB, which is enough to meet the space required for storing the command. Take the string “callhome” as an example: its hexadecimal representation is “0x63, 0x61, 0x6c, 0x6c, 0x68, 0x6f, 0x6d, 0x65”. First, it is divided into groups and each group consists of three units, if there are less than three, add 0 at the end. In this way, the original hexadecimal string is divided into three groups of {0x63, 0x61, 0x6c}, {0x6c, 0x68, 0x6f} and {0x6d, 0x65, 0x00}, and the corresponding RGB can produce three pixels. That is, the string "callhome" is converted into three pixel values. In addition, in the process of generating a picture, in addition to recording the content of the original data, the size of the original data also needs to be recorded to restore it normally. For PR-Bot, it uses two pixel units at the beginning of the picture, which is the six-byte space, for recording the data size. Finally, the pixel information in the picture consists of "*data size (2 fixed pixels) + data content + 0 (may exist)*".

3.2 CA Channel

When issuing the address of command, the botmaster designs a UGA algorithm, which seed is based on the current date and hottest topics. The method is to prevent the address list generated by the bot from being predicted prematurely by the defender. For the hottest topics, it may change every day and has no fixed pattern. Even if defenders have mastered the UGA algorithm, they can know the address list only waiting until that day, making it impossible to predict prematurely. Although unavoidable, it increases the working costs of defender to some extent. The generated address list is as follows:

URL Shortener:
https://tinyurl.com/shehuim
https://tinyurl.com/ixniimli
https://tinyurl.com/vyowinfc
…
Social Network:
https://www.tumblr.com/yfvqwvvi
https://www.tumblr.com/nshynnuu
https://www.tumblr.com/ldtctknm
…



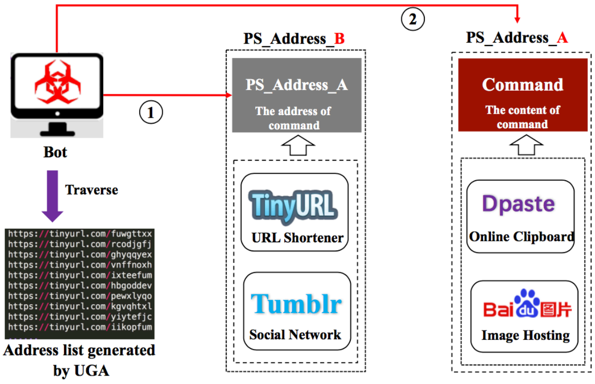
Enter a long URL to make tiny:
http://dpaste.com/0SV8NS5

Custom alias (optional):
https://tinyurl.com/ shehuim
May contain letters, numbers, and dashes.

Fig. 5. How to use the URL Shortener

For the URL Shortener, when storing the address of command, only the URL address to be converted and the suffix of the alternative URL is needed, as shown in Figure 5. And for the social network website, the user's personal homepage address needs to be configured based on the alternative URL suffix, and then the address of command can be issued as a new message. Of course, all operations are automated by the program.

As shown in Figure 6, for the bot, it undergoes two steps when obtaining the content of command, that is the "Secondary Addressing Mechanism" described above. First, the bot will traverse the generated address list based on the hard-coded UGA algorithm. When finding the address PS_Address_B, it will obtain the address PS_Address_A and



```
https://tinyurl.com/fuwgttxx
https://tinyurl.com/rcodjgfj
https://tinyurl.com/ghyqqyex
https://tinyurl.com/vnffnoxh
https://tinyurl.com/ixteefum
https://tinyurl.com/hbgoddev
https://tinyurl.com/pewxlyqo
https://tinyurl.com/kgvqhtxl
https://tinyurl.com/yiytefjc
https://tinyurl.com/iikopfum
```

Fig. 6. Secondary Addressing Mechanism

further extract the content of command. The reason for adopting the "Secondary Addressing Mechanism" is to improve the flexibility and scalability of the PR-Bot. For some publicly available resources that are suitable for storing commands but not support customized URL, the jump relationship provided by the "Secondary Addressing Mechanism" can make it become "a publicly available resource that supports customized URL". In addition, this mechanism can also improve the robustness and concealment of the C&C channel to some extent.

3.3 RF Channel

Callhome Module. Obviously, if all bots upload the information of controlled end to the unique URL specified in the command, there is a problem of information loss due to the limited storage space of the online clipboard website. In order to avoid this problem, this paper adopts the strategy of "URL + numeric string".

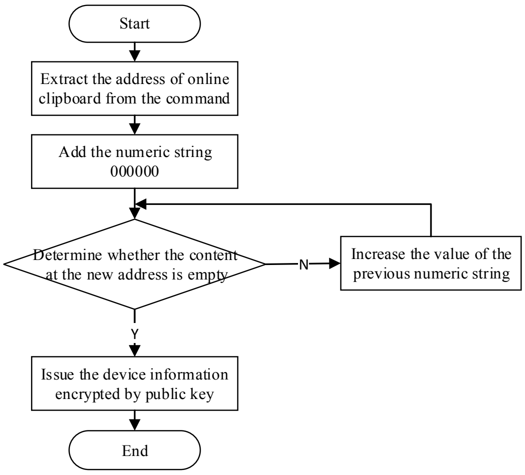

Start
Extract the address of online clipboard from the command
Add the numeric string 000000
Determine whether the content at the new address is empty
N
Increase the value of the previous numeric string
Y
Issue the device information encrypted by public key
End

Fig. 7. The brief solution for result feedback

As shown in Figure 7, after the bot extracts the address parameter from the command, it does not upload the device information to the address directly. Instead, the bot will add a numeric string of the specified digits (according to the law from small to large) behind the address, and then it sends the device information to the new address. Take the address http://www.wepaste.com/abcde as an example: After the bot obtains the address parameter, it will add a numeric string from 000000 to 999999 behind it and then traverse the URLs from http://www.wepaste.com/abcde000000 to http://www.w epaste.com/abcde999999. The device information is not fed back until an address with empty content is found.

However, if only the above brief solution is adopted, although the problem of the limited storage space can be solved, there is a problem of information coverage due to concurrent operation of bots. That is, if an address with blank content is found by two bots at the same time, they will upload device information to the address, no matter who comes first, there must be a case where one bot overwrites information uploaded by another bot. It is because the content on the online clipboard website is readable and writable.

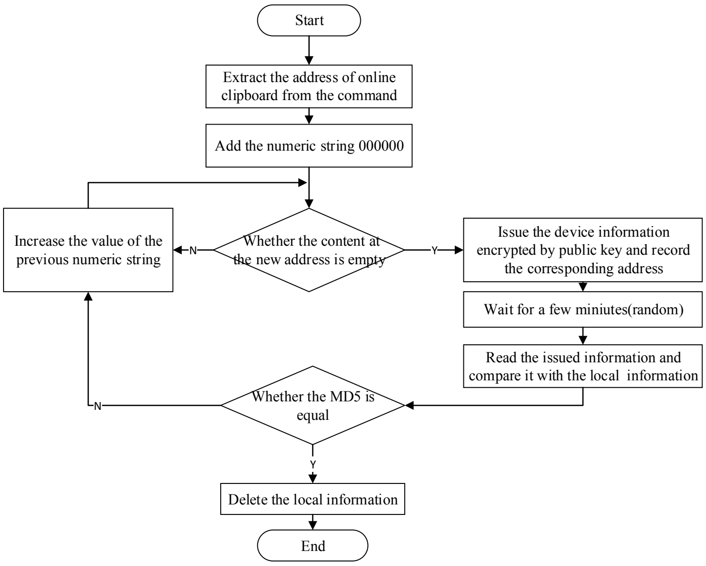

Start
Extract the address of online clipboard from the command
Add the numeric string 000000
Increase the value of the previous numeric string
N
Whether the content at the new address is empty
Y
Issue the device information encrypted by public key and record the corresponding address
Wait for a few miniutes(random)
Read the issued information and compare it with the local information
N
Whether the MD5 is equal
Y
Delete the local information
End

Fig. 8. The enhancement solution for result feedback

In order to solve this new problem, this paper uses an enhanced solution, as is shown in Figure 8. Specifically, after several minutes of uploading device information, the bot reads the uploaded information again and compares it with the locally stored information to verify whether the information is uploaded by itself. If the MD5 of the two are the same, it is considered that the device information is successfully uploaded. Otherwise, the new address is traversed sequentially and the device information is re-uploaded. In this way, even if multiple bots find an address with empty content at the same time, there will be no problem of information coverage.

For the botmaster, when downloading the device information, it is also in accordance with the law of " URL+ numeric string", sequentially traversing the address from "URL + 000000" to " URL + 999999". Theoretically, as long as the background program traverses the first address with blank content, it means that all the device information uploaded by the bots has been downloaded. However, in the specific implementation process of PR-Bot, in order to improve reliability, the background program only terminates when it traverses ten consecutive addresses with empty content.

File Stealing Module. Cloud disk website, also known as cloud storage website, is mainly classified into two categories: the public cloud disk and the private cloud disk. Among them, the public cloud disk can be used without registering, and the information on it is public; and the private cloud disk needs to be registered before they can be used, and the information on it is private and cannot be seen by other users unless the owner actively shares them. In addition, most private cloud disks provide the API interfaces for users to operate data in it.

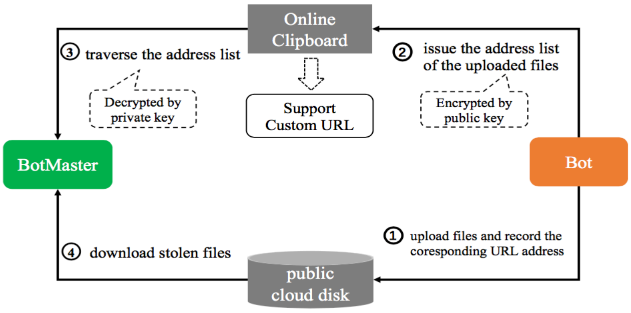


Fig. 9. File feedback process based on public cloud disk

In the process of feeding back files, PR-Bot selects a public cloud disk to store the stolen files, which is shown in Figure 9. First, the bot traverses the specific types of files in the controlled end and uploads them to public cloud disks one by one, and records the corresponding URL address at the same time. Then, all URL addresses are issued to the online clipboard website in the form of string. When issuing the address, it is basically the same as the flow of issuing callhome information, including the strategy of “URL + numeric string” and the mechanism of secondary verification.

Table 9. Parameters of API interface of private cloud disk

Name	Site
Mediafire	{email}, {password}, {app_id}, {folder_key}
Mega	{email}, {password}
Smartfile	{key}, {password}

Here, we will discuss the reason why PR-Bot does not choose private cloud storage to store stolen files. If PR-Bot selects the private cloud disk to store stolen files, it is necessary to specify the parameters required by the private cloud disk API interface in the command, which is used for authentication and as shown in Table 9. It will cause two problems:

1) Once the defender has mastered the identification information, it is equivalent to obtaining the control authority of the account corresponding to the private cloud disk, so the files in the cloud disk can be viewed and deleted, and the bots are allowed to upload spam data deliberately;

2) Because only one or a few account parameters are specified in the command, with the increase in the number of bots, the behavior of sharing the same account can easily cause abnormalities on the website, and may expose the entire botnet's activities.

For the public cloud disk, the above two problems are inexistent. First, even if the defender obtains the command and masters the law of "URL address+ numeric string" adopted by bots, it cannot locate the network location where the stolen file is located. It is because the uploaded address list is encrypted by the public key, and it can only be decrypted by the private key of the botmaster. Second, when a file is uploaded through a public cloud disk, each bot is equivalent to an independent user and has no necessary correlation, so there is no problem that multiple bots share the same account. Therefore, PR-Bot selects the public cloud disk to store stolen files, whose purpose is to ensure the security of the C&C channel.

4 The Defense Measures

Accurately finding botnets similar to PR-Bot and taking targeted measures to contain them is the ultimate goal of this paper. For the PR-Bot botnet proposed in this paper, this section describes the contents of PR-Bot defense from the aspects of detection, measurement and tracking, in order to take over the control of the botnet or reduce its availability.

4.1 Detection

In the CA channel, PR-Bot uses the UGA algorithm to generate an address pool, and the bot obtains commands by connecting the pseudo-random addresses. This process is similar to DGA. Therefore, some methods for detecting DGA also apply to UGA detection: (1) Character feature detection based on domain name [14]. There are still differences between the addresses generated by UGA and the normal addresses, such as: the use of a large number of URL Shortening services, the use of unusual user names, the use of fixed social network and etc. Therefore, the rules of distribution of domain name strings can be found by constructing the semantic rules and feature vectors, and they can be identified by the methods such as data mining and machine learning. (2) Detection based on domain name activity. In order to obtain commands, the bot will constantly address, and the addressing time will show some regularity, such as addressing once every 1 hour, or addressing at a fixed point of time and even early in the morning. These features all show the non-human characteristics, so the domain name activity and spatio-temporal features can be used to detect the malicious addresses [15].

In the CC channel, PR-Bot mainly uses the online clipboard website and image hosting website. Therefore, the detection method based on communication content and network layer anomalies can be used. (1) For the commands issued on an online clipboard website, the bot will obtain the commands in the form of text, which are encoded and have the specificity, as well as identifiability. Among them, for the transmission content of the HTTP protocol, a feature matching rule may be configured in advance, such as Snort and other intrusion detection systems, to quickly and accurately discover such botnet. (2) For the commands issued on the map bed website, the bots will obtain commands from the downloaded pictures. It can also be detected through the abnormality of the transmitted information. However, the detection method based on the communication content is only applicable to botnets with specific characteristics. The disadvantage is that the unknown botnets cannot be detected, and the signature of the bot program needs to be continuously maintained and updated. The network layer anomaly detection method assumes that the communication mode between the botmaster and bots is quite different from the normal user communication, so that the trail of the botnet can be found through the flow analysis [16]. In the RF channel, PR-Bot uploads the text information according to the address specified by the botmaster, which is similar to the CC channel, so the detection method based on the communication content and the network layer anomaly may also be used.

In addition, public resource service providers should actively improve the security protection of the website to prevent normal services from being abused by attackers. PR-Bot needs to automatically register a large number of accounts and automatically issue control commands. Therefore, service providers can use the verification code-based or speed-limiting method to prevent the account from being registered in batches. Although this method will degrade legitimate users' experience, it increases the cost of the attacker and can effectively avoid creating a potential target for attackers. Besides, the content of the account on the platform can be monitored in real time, and the release of the suspicious character string shall be further traced or handled by the security personnel.

4.2 Measurement

By measuring the PR-Bot botnet, it can portray its topological structure and corresponding scale, so that the defender can understand more about the outline and characteristics of the PR-Bot. However, due to the mechanism characteristics of PR-Bot itself, it is difficult to measure the PR-Bot, and the traditional measurement methods based on Crawler and Sybil cannot be applied. However, in the RF channel, the bot adopts the strategy of “URL + numeric string” to upload the callhome information or the address list of stolen files. The defender can find out the pattern adopted by PR-Bot through reverse analysis or flow monitoring, so that the entire scale of the botnet can be measured through the method of address traversal. Although the PR-Bot measurements are affected by various factors, such as time zone, startup/shutdown, it is difficult to accurately estimate the scale of the entire botnet, but it can estimate the number of bots as much as possible.

4.3 Tracking

If the defenders have mastered the botnet C&C channel, they can run the bot in a controlled environment or join the botnet in an infiltrated form to understand the internal activity of the botnet. However, because the existing bots usually check the operating environment more strictly, the former method can be easily detected by the attackers, and this is true for the PR-Bot introduced in this paper. In this section, we focus on how to track botnets by means of infiltration, and the infiltrating agent is called “Infiltrator”. Infiltrator can disguise as an infected controlled device to join the botnet and simulate the real communication protocol of PR-Bot to communicate with the botmaster to observe the internal activities of PR-Bot. Among them, in the RF channel, the infiltrator can intentionally submit a decoy file with tracking watermark or other payloads, so as to track the botmaster. For example: the infiltrator embeds a hidden remote picture URL in a Word document, so if the botmaster downloads and opens the file, it will actively request the URL and load the remote picture, and then the defender can trace the position of the botmaster based on the source of the request.

5 Related Work

To be well prepared for future botnet attacks, security researchers have done many works on studying advanced botnet models and defense technologies.

Sanatinia et al. [17] presented a robust, stealthy botnet that named OnionBots. The botnet use Tor privacy infrastructures for cyber-attacks by completely decoupling their operation from the infected host IP address and by carrying traffic that does not leak information about its source, destination, and nature. Ali et al. [18] presented ZombieCoin which used Bitcoin network for botnet C&C. ZombieCoin is robustness, because common takedown techniques of confiscating suspect web domains, seizing C&C servers or poisoning P2P networks, would not be effective. Yan et al. [19] proposed an anti-pollution P2P botnet called AntBot, which used a tree-like structure to propagate commands in P2P networks. The tree-like structure with the randomness and redundancy in its design, renders it possible that individual bots, when captured, reveal only limited information.

Besides, there are a number of botnet designs are based on publicly available resources. Artturi et al. [20] explores the multitude of ways in which modern malware abuses third-party web services as C&C channels, including Google Docs, Tumblr, Twitter and so on. Lee et al. [21] explore botnets based on USS, and propose alias flux methods that frequently change shortened URLs of C&C servers to hide their existence, which is similar to the domain flux method. Nagaraja et al. [22] exploit image steganography techniques to set up a communication channel within the social network, and use it as the botnet’s C&C channel. However, none of these research works have studied how to design a resilient and efficient bidirectional communication channel. Our study focuses on constructing a three-channel botnet based on multiple publicly available resources and is complementary to the existing research works to some degree.

On the defensive side, there have been many types of approaches to detect botnets, including signature-based, anomaly-based, DNS-based and data mining, machine learning techniques. For public service-based botnets, Chen et al. [23] design an unsupervised system to detect Twitter spam campaigns that use botnets to send duplicate content with embedded URLs. The unsupervised detection approach allows to build a blacklist of malicious email addresses, URLs and Twitter accounts, and to share threat intelligence with the research community in real-time. Guo et al. [24] explore the currently typical C&C server finding schemes as three types: dedicated IP address, Internet infrastructure and third-party service from a new perspective. Their work indicates that third-party service based C&C presents a better approach in terms of complexity, flexibility, traffic covertness and scale. In this paper, for PR-Bot, we propose the targeted defense strategies from the perspective of detection, measurement and tracking, so as to achieve the goal of combating against such botnets.

6 Conclusions

This paper introduces an advanced botnet based on publicly available resources, which is named PR-Bot. The PR-Bot is constructed by a three-channel scheme, which includes three sub-channels: CC channel, CA channel and RF channel. Each sub-channel can be supported by multiple publicly available resources and can be extended in the form of plug-in. Meanwhile, PR-Bot also uses the technologies, such as information hiding, content encryption and digital signature, to improve the robustness and concealment of C&C channels. In addition, in the face of new challenges, this paper proposes the defense strategies against PR-Bot in terms of detection, measurement and tracking to deal with possible similar cyber threats. We believe that it is of great practical significance to study how to construct a highly antagonistic botnet from the perspective of the attackers and propose the effective defense strategies before the attackers deploy them in practice. In the next step, we will conduct an in-depth study on this type of botnets, and design a rapid and effective detection system.

Acknowledgements. This work is supported by the Key Laboratory of Network Asses sment Technology at Chinese Academy of Sciences, Beijing Key Laboratory of Netwo rk Security and Protection Technology, and the National Key Research and Developm ent Program of China (NO.2016YFB0801604, NO.2016QY08D1602, No.2016QY06 X1204).

References

1. Xiang C, Binxing F, Jinqiao S, et al. Botnet triple-channel model: Towards resilient and efficient bidirectional communication botnets[C]//International Conference on Security and Privacy in Communication Systems. Springer, Cham, 2013: 53-68. (2013)
2. Li C, Jiang W, Zou X. Botnet: Survey and case study[C]//innovative computing, information and control (icicic), 2009 fourth international conference on. IEEE, 1184-1187. (2009)
3. Bailey M, Cooke E, Jahanian F, et al. A survey of botnet technology and defenses[C]//Conference For Homeland Security, 2009. CATCH'09. Cybersecurity Applications & Technology. IEEE, 2009: 299-304. (2009)
4. Amini P, Pierce C, Kraken Botnet Infiltration [EB]. Blog on DVLabs, 2008[2011-06-10]. http://dvlabs.tippingpoint. Com. (2011)
5. Jeff Williams. "Operation b107 - Rustock Botnet Takedown". http://blogs.technet.com/b/mmpc/archive/2011/03/18/operation-b107-rustock-botnet-takedown.aspx. (2011)
6. Sharifnya R, Abadi M. DFBotKiller: domain-flux botnet detection based on the history of group activities and failures in DNS traffic[J]. Digital Investigation, 12: 15-26. (2015)
7. Nazario J, Holz T. As the net churns: Fast-flux botnet observations[C]//Malicious and Unwanted Software, 2008. MALWARE 2008. 3rd International Conference on. IEEE, 2008: 24-31. (2008)

8. Stone-Gross B, Cova M, Cavallaro L, et al. Your botnet is my botnet: analysis of a botnet takeover[C]//Proceedings of the 16th ACM conference on Computer and communications security. ACM, 2009: 635-647. (2009)
9. Holz T, Steiner M, Dahl F, et al. Measurements and Mitigation of Peer-to-Peer-based Botnets: A Case Study on Storm Worm[J]. LEET, 2008, 8(1): 1-9. (2008)
10. Davis C R, Fernandez J M, Neville S, et al. Sybil attacks as a mitigation strategy against the storm botnet[C]//Malicious and Unwanted Software, 2008. MALWARE 2008. 3rd International Conference on. IEEE, 2008: 32-40. (2008)
11. Thomas K, Nicol D M. The Koobface botnet and the rise of social malware[C]//Malicious and Unwanted Software (MALWARE), 2010 5th International Conference on. IEEE, 2010: 63-70. (2010)
12. Nagaraja S, Houmansadr A, Piyawongwisal P, et al. Stegobot: a covert social network botnet[C]//International Workshop on Information Hiding. Springer, Berlin, Heidelberg, 299-313. (2011)
13. Cui, X., Fang, B.X., Yin, L.H., Liu, X.Y.: Andbot: Towards Advanced Mobile Botnets. In: Proceedings of the 4th Usenix Workshop on Large-scale Exploits and Emergent Threats, LEET 2011 (2011)
14. Yadav S, Reddy A K K, Reddy A L, et al. Detecting algorithmically generated malicious domain names[C]//Proceedings of the 10th ACM SIGCOMM conference on Internet measurement. ACM, 48-61. (2010)
15. Gu G, Perdisci R, Zhang J, et al. BotMiner: Clustering Analysis of Network Traffic for Protocol-and Structure-Independent Botnet Detection[C]//USENIX security symposium. 2008, 5(2): 139-154. (2008)
16. Silva S S C, Silva R M P, Pinto R C G, et al. Botnets: A survey [J].Computer Networks, 2013, 57(2): 378-403. (2013)
17. Sanatinia, Amirali, and Guevara Noubir. "Onionbots: Subverting privacy infrastructure for cyber attacks." In Dependable Systems and Networks (DSN), 2015 45th Annual IEEE/IFIP International Conference on, pp. 69-80. IEEE (2015)
18. Ali, Syed Taha, Patrick McCorry, Peter Hyun-Jeen Lee, and Feng Hao. ZombieCoin 2.0: managing next-generation botnets using Bitcoin. International Journal of Information Security: 1-12. (2017)
19. Yan G, Ha D T, Eidenbenz S. AntBot: Anti-pollution peer-to-peer botnets[J]. Computer Networks, 55(8): 1941-1956 (2011)
20. Lehtiö, Artturi. "C&C-as-a-Service: Abusing Third-party Web Services as C&C Channels." (2015)
21. Lee, Sangho, and Jong Kim. "Fluxing botnet command and control channels with URL shortening services." Computer Communications 36, no. 3: 320-332. (2013)
22. Nagaraja, Shishir, Amir Houmansadr, Pratch Piyawongwisal, Vijit Singh, Pragya Agarwal, and Nikita Borisov. "Stegobot: a covert social network botnet." In International Workshop on Information Hiding, pp. 299-313. Springer, Berlin, Heidelberg (2011)
23. Chen Z, Subramanian D. An Unsupervised Approach to Detect Spam Campaigns that Use Botnets on Twitter[J]. arXiv preprint arXiv:1804.05232, (2018)
24. Guo X, Cheng G, Hu Y, et al. Progress in Command and Control Server Finding Schemes of Botnet[C]//Trustcom/BigDataSE/I SPA, 2016 IEEE. 1723-1727, (2016)