Poster: ServerLess C&C channel

Jianjun Zhao[1], Fangjiao Zhang[1,2], Chaoge Liu[1,2]
1 (Institute of Information Engineering, Chinese Academy of Sciences)
2 (School of Cyber Security, University of Chinese Academy of Sciences)
zhangfangjiao@iie.ac.cn

***Abstract*—With the development of botnet detection technology, it is becoming more and more difficult to build a robust botnet command and control(C&C) channel. Botnet has the problem of single point of failure(SPOF), and the owner can be found easily considering the tracing technologies. Consequently, we propose a command and control channel model containing three subchannels, each of which is consists of several public services. SPOF is solved due to the complexity and diversity of public services. The use of public services as botnet infrastructure of communication, eliminates the association between the communication node and the owner, and mitigates the risk that owner being traced.**

I. Introduction

A botnet is a group of compromised computers or IoT devices that are remotely controlled by botmasters via command and control channels. In the process of continuous development of offensive and defensive technology, the botmaster implements more robust and more concealed communication channels against detection mechanisms. For one thing, the botmaster has evolved from centralized C&C into decentralized structures in order to evade the detection and block of ISP and security companies. For another， botmasters are more likely to use public services instead of their own servers to deploy their C&C channel in order to avoid being traced.

Public services such as social network, URL shortening service(USS) and online clipboard are becoming more and more popular in people's daily life. The lightweight public service reduces the time cost of using the network because of its accessibility and simplicity. Unfortunately, the botmaster are using these services to build their C&C channels at the same time [1]. Researchers have found that some malwares get command and response addresses via USS (e.g. bit.ly, goo.gl.) and social network (e.g. twitter, pinterest), or using Internet clipboard (e.g. pastebin.com, cl1p.net) and Internet infrastructure (e.g. DNS TXT record). These kinds of C&C channel do not require botmaster to deploy its own servers, but instead uses the public services as its communication infrastructures, so we named it as ServerLess C&C channel. The ServerLess C&C channel has become a popular trend and it is difficult to be completely shut down.

Due to these features of ServerLess C&C channel, we design a botnet using this structure and make three contributions. Firstly, we improve the multi-channel botnet model [2] and add an addressing phase before the command is issued. The advantage is that the address where command stored is more flexible and more difficult to detect and close. We also merged the registration and data uploading into one channel, because the essence of both is send data from bot to botmaster. Secondly, botmaster does not need to use its own servers when building a botnet using the channel model we designed, but instead uses the public service as a command and control infrastructure, which prevents the botmaster from being traced, and saves much cost definitely. Thirdly, we summarize the current public services that can be used in ServerLess C&C channel model and propose two new approaches: using APIs of data synchronization services and video danmaku.

II. Design

In our work, we mainly discuss the process of message transmission of botnet using ServerLess C&C channel. Assuming that we have a group of compromised devices as bots, and each bot has the ability to communicate with the public services. The command and control process of the botnet mainly consists of three parts: address, issue commands, and receive results.

A. Address (“①” and “②” in Figure 1)

We design a subchannel, which is used to release the address of the stored command, rather than issue the command directly. This subchannel includes several public services, such as USS, online clipboard service, disposable e-mail service. All public services in this subchannel must have the ability to customize an alias or URL so that botmaster and bot can share a series of addresses by means of an address generation algorithm.

First, botmaster and bot need to share an initial public service list so that botmaster and bot can know which public services are used to exchange information at this stage. Next, botmaster randomly selects some services in the public service list, and then use the address generation algorithm to generate an address for each service, the address of the command encrypted with botmaster private key released here. Similarly, the bot also randomly selects some services, using the same algorithm to generate a series of addresses, and try to get the content. If the content exists and can be successfully decrypted with the botmaster public key, then prove that the address is correct. During this phase, bot also complete the authentication of the botmaster.

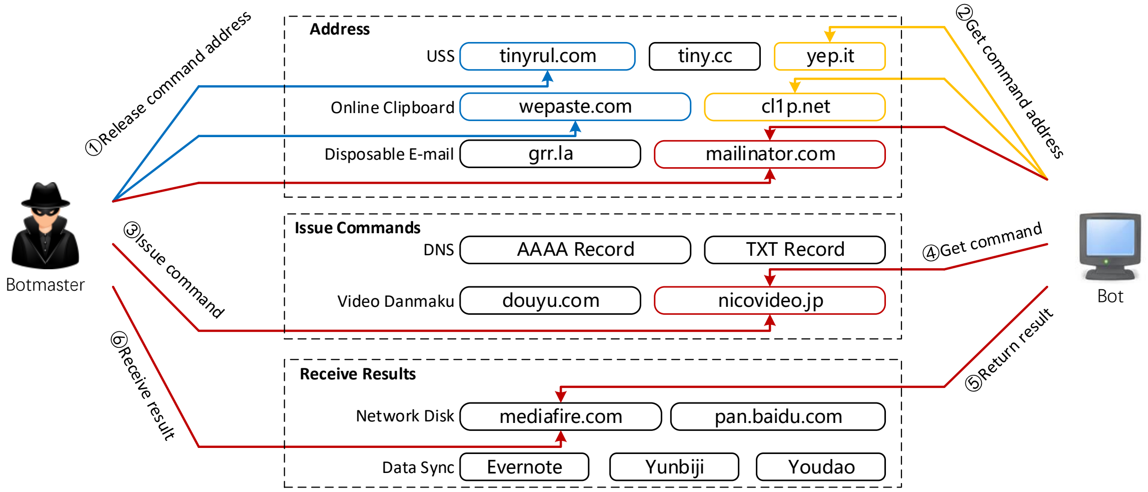

Figure 1. The process of command and control

B. Issue commands ("③" and "④" in Figure 1)

We design another subchannel to store the real command, public services in this subchannel do not require additional conditions, but it is better to use common services to avoid traffic detection. Public services such as DNS and video danmaku can be used. Video danmaku is a kind of video comment emerged in recent years that can be directly displayed on the video by scrolling, staying or even more action effects, and it is popular in China and Japan.

First, botmaster selects a symmetric key and encrypts the key and the command with the public key of the bot, and issue them to the public service of the subchannel. The bot has acquired the command address from the previous part, so the bot can obtain command and symmetric key by decrypting with its private key. It should be noted that the command should contain a receiving address so that the bot can knows where to send the result. If the initialized public service list needs to be changed, a new one can also be issued in the command. The reason for using a symmetric key is to protect the result from being stolen.

C. Receive results ("⑤" and "⑥" in Figure 1)

The third subchannel is used for botmaster to receive results. We use the public data synchronization services, such as network disk and sync notes, to build this subchannel. These services have privacy protection, and remote login will not be identified as an abnormal. Typically, the network disk and sync notes services will provide a series of API, which makes the file and data can be uploaded without the client, so that the bot can easily upload the results.

After the execution of the command, bot uses the symmetric key to encrypt the result and then encrypts the ciphertext again with the private key. Next, bot sends the result to the address specified in command. If necessary, the additional information, like account and password required to use APIs, should also be included in the command issued previously. In the end, botmaster accesses the result address, receives the content and decrypts it with the bot's public key and symmetric key and gets the original result. Besides, the botmaster authenticate the identity of the bot in this phase, and symmetric encryption protect the result from being stolen by faked botmaster.

III. COUNTERMEASURE

For this botnet, the address is the most important part of the whole process, so the countermeasures should focus on this part. We recommend the public services use captcha to avoid the abuse. Moreover, the communication content in the channel is ciphertext without meaning, so the public services should investigate and remove these meaningless content, and can also create a blacklist to avoid repeatedly use.

IV. CONCLUSION

Using public services to build botnet will be a new trend. The model we proposed in this paper integrate the current implementation and only use public services as botnet infrastructure of communication to solve the SPOF problem. Meanwhile, many new public services on the Internet can be found and used, making the ServerLess C&C channel extensible. In addition, our approach mitigate the risk that botmaster being traced, because he does not need to purchase a domain name or a server.

V. ACKNOWLEDGMENTS

The authors would like to thank the anonymous reviewers for their helpful comments for improving this paper. This work is supported by the Key Laboratory of Network Assessment Technology, Chinese Academy of Sciences and Beijing Key Laboratory of Network security and Protection Technology.

REFERENCES

[1] 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. IEEE, 2016: 1723-1727.

[2] Cui X, Shi J, Liao P, et al. The Triple-Channel Model: Toward Robust and Efficient Advanced Botnets (Poster Abstract)[C]//RAID. 2012: 376-377.