Attack me If You Can: Resisting Proxy based Mimicking Attack

Sravia Sivaram#[1] ,S. Swetha#[2] ,S. Suganya#[3]
Final year students
K.Amsavalli,Assistant professor
Department of Computer Science and Engineering

Abstract **- Botnets have become major engines for malicious activities in cyberspace nowadays. In order to sustain their botnets and disguise their malicious action, botnet owners are mimicking legitimate cyber behavior to fly under the radar. This poses a serious challenge in anomaly detection. First of all, we establish a semi-Markov model for browsing behavior. Based on this model, we tend to find that it is impossible to detect mimicking attacks based on statistics if the number of active bots of the attacking botnet is sufficiently large. However, we also find it is hard for botnet owners to satisfy the condition to carry out a mimicking attack most of the time. With this new analysis, we conclude that mimicking attacks can be discriminated from genuine flash crowds using second order statistical metrics. We also define a new fine correntropy metrics and show its effectiveness compared to others .**

Index Terms – Mimicking Attack, Phishing attack, Botnets, Flash crowds, Bot masters.

I. INTRODUCTION

Nowadays more and more critical infrastructure are increasingly reliant on the internet operators Given the widespread use of automated attack tools , attacks against Internet-connected systems are so common- place that Internet crime has become a ubiquitous phenomenon. Although a number of countermeasures , legislations against Internet crime has been proposed and developed, Internet crime is still on the rise .In this paper, We first demonstrate this by proving that legitimate cyber behavior can be successfully simulated, therefore, it is not possible to discriminate mimicking attacks from legitimate cyber events using statistical methods. However, for us to achieve this, attackers need to satisfy one critical condition they have to possess a sufficiently large number of active bots , with no fewer than the number of active legitimate users of the simulated events. By active bots, we mean the bots that botnet owners can manipulate at the time they initiate attacks.

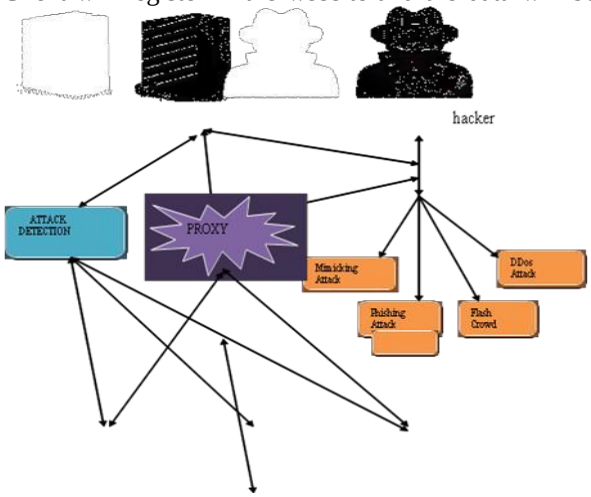
Botnets are the main drivers of cyber -attacks, such as Distributed denial of service (DDoS), information phishing and email spamming. These attacks are pervasive in the Internet, and often cause great financial loss . Motivated by huge financial or political reward, attackers find it worth numerous varities of botnets in cyberspace, such as DSNX bot, evil bot, G-Sys bot, sdbot, and Spy bot . On one hand, researchers have studied botnets in various perspectives, including botnet probing events and internet connectivity , size and domain fluxing.

In this paper, we study mimicking attacks and detections from both sides, as attackers and defenders, are one significant extension based on our preliminary work in from the botnet programmers' view point, so then we simulate the legitimate behavior of web browser then the three key pieces of information ; web page popularity of the target website, web page requesting time interval for a user, and number of pages a user usually browse for one browsing session (referred to as browsing length).
while to organize sophisticated botnets for use as attack tools.

2 RELATED WORKS

Architecture:

Client will register in the website and the data will be



stored in the database. Clients have the right to upload their images to the server. All the information of the client will be send to the server via proxy.

Four methods of attacks is initiated by the botnet. First one is mimicking attack where the URL typed by the user is redirected to a malicious site. Secondly DDOS, attack creates disturbance to a networking environment. Thirdly, in flash crowd attack , the image which was sent by the user is displayed multiple times to the receiver.

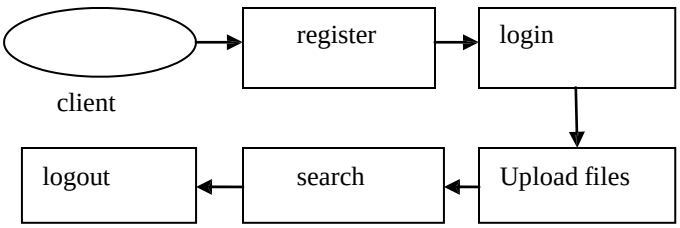
Finally , in phishing attack, when the user types the URL , the hacker sends the fake site to the user and then extracts the user name and the password of the genuine user. Then the user is redirected to the original site.

Server identifies these kinds of attacks and blocks the client and intimates the user via a warning message.

2.1 Client registration and updates:

Cyber attackers organize more and more botnets to carry out their illegal tasks , such as launching DDoS attacks, sending spam mails perform information phishing and collect sensitive information.

In this section, the client will register in the website and that data will be stored in database. Clients have the right to upload their images to the server . Here the client is the victim . The web pages he accessing is the target victim web sites. This work is done for the observation point. We count the number of HTTP requests of each flow for the given time intervals and to describe the browsing behavior of legitimate web viewer or user.



2.2 Botmaster's observation and updation:

Botmasters can simulate a flash crowd successfully in terms of statistics. With sufficient numbers of active bots, a botmaster can use one bot to simulate one legitimate user using the knowledge of web browsing dynamics.

In this section, a web page to observe the potential victim for sufficient time in attack free cases. This training should be taken periodically to update the parameters to reflect the ever changing web browsing behavior. The client Browsing details will be collected in this BotMaster web page. All the web page that the client accessing will be collected in this BotMaster page.

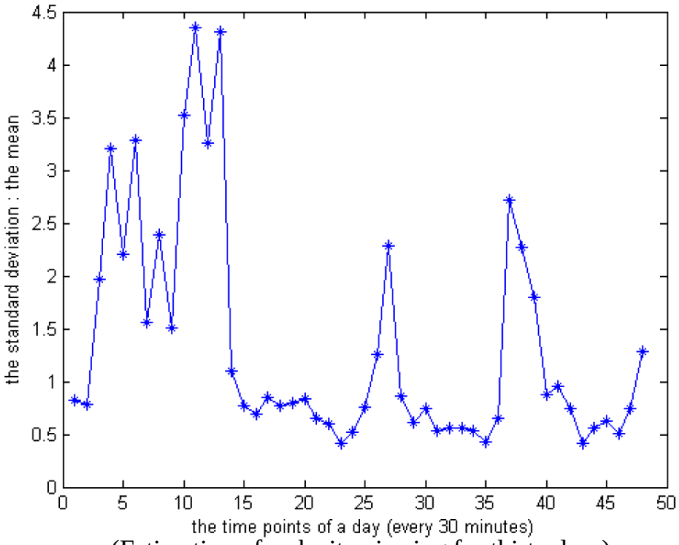
2.3 Mimicking attacks and its detection:

A botnet is usually established by a botnet writer developing a program, called a bot, and installing the program on Internet using various methods. All the bots from a botnet are controlled by a botmaster . The hosts running these programs are known as zombies . For a botnet , there is one or a number of command and control (C&C) servers to communicate with bots. To protect the C&C servers and sustain the botnet, the IP address of C&C is rapidly changed by botmasters.

In this section , using the collected details about victim the botmaster will successfully generate flashcrowd attack and mimicking attack. If we perform any modification in the botmaster page it will automatically reflect in the victim client website. After analyzing the client response from the server, we can able to detect the mimicking attack.

B. Mimicking attack and its algorithm:

In order to detect the mimicking attacks , we have to establish a profile of the fine correntropy of flows for the non-attack cases , and identify an anomaly when the variation of flow fine correntropy is sufficiently different from the normal value. We can manually supervisor the network traffic of the web site, which we try to protect, for a number of given periods (we use twenty four hours as one period) . We take the periods that are attack free as benchmark for anomaly detection. As a result, we can establish a map of the number of page request against time for a twenty four hour period. The granularity of time could be at the second or minute level, as in order to detect attacks in time.



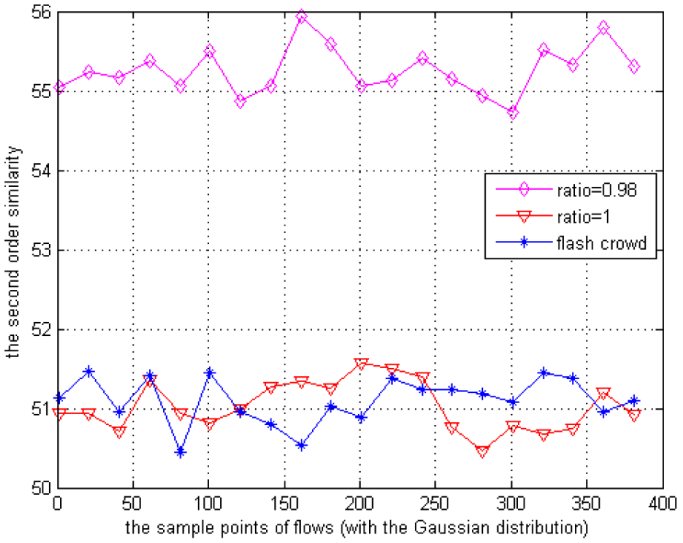
(Estimation of web site viewing for thirty days)

Fig(2.3a)

The effectiveness of the proposed mimicking attack algorithm, we collected the web dynamic data of a popular news web site for thirty days to at a major backbone network center. The data for two days (the 1st and the 30th of June 2010) has been explicitly extracted for our experiments. We used the data from day 1 as a training data set, and extracted the key parameters from the data set to populate the parameters of the semi-Markov model. We call these as training data and target data, respectively. We arrange the data sets into a matrix for both days: each web page of the web site is a row in the matrix, and every column denotes the number of requests in a thirty minute duration shown in the fig (2.3a)

C. Effectiveness of detection method:

A mimicking attack is possible to proven that, the effectiveness of the proposed detection method using simulations in this section. We note that the proposed detection method is independent of any specific flow distribution because the parameters that we use are the second order statistical data, e.g. the standard deviation. Without loss generality shown in the fig(2.3b) , we use the Gaussian distribution for the traffic flows in the following experiments.



(Traffic flow detection graph)

Fig(2.3b)

III. LEGITIMATE CYBER BEHAVIOR

In this section, we act as attackers in order to study how to mimic the browsing behavior of a legitimate web viewer. A mathematical model based on the semi-Markov Chain process is established to describe a mimicking algorithm

3.1 An Individual View of Web Browsing Behavior

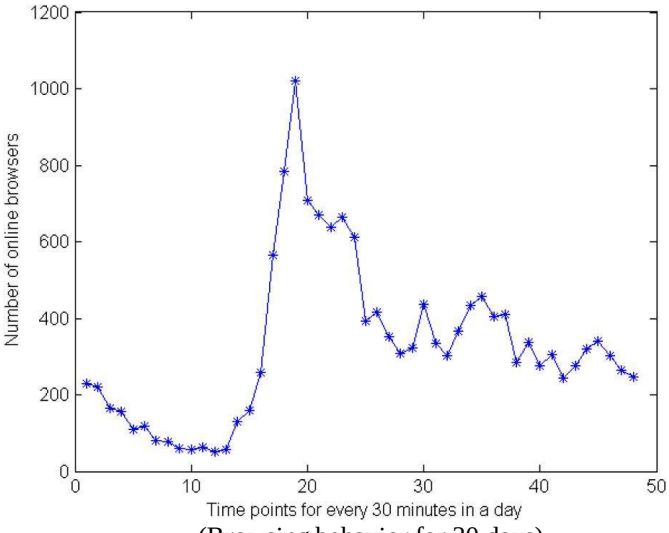
If a botnet owner has a sufficient number of active bots (the sufficient number condition holds), then he can us one bot to act as one legitimate viewer. The problem for botnet writers is how to statistically simulate the behavior of a legitimate browser. A browsing session for a user is shaped by three factors: which pages to request , time duration of viewing a page and how many pages to go through.

To describe the browsing behavior of a legitimate web viewer, we extend the classical Markov model to a four parameter semi-Markov In order to find the four parameters for the semi-Markov model , we should observe the potential victim for sufficient time in attack free cases and based on the data collected , we can extract the four parameters. Of course , this training should be taken periodically to update the parameters to reflect the ever changing web browsing behavior .With this four parameter semi-Markov model in place , every bot can independently simulate a legitimate web viewer's browsing behavior.

3.2 The system view of legitimate behavior:

In this scenario , we are interested to see the various phenomenon in a system view point . For example , for a given point of time , we expect to know the number of total page requests to a web site, and number of requests for a specific web page of the web site.

In order to answer these questions n, we need one more Parameter : the number of active web viewers for a given time point , varies against the time point of a day. Intuitively, there are more web viewers during working time than early morning. We have conducted a 30 days observation on for every 30 minutes shown in the fig(3.2), and found that was stable day after day.



(Browsing behavior for 30 days)

Fig(3.2)

3.3 The Legitimate Behavior Mimicking Algorithm:

In order to simulate legitimate flows or a flash crowd of a website, attackers have to firstly study the subject and extract related browsing dynamic parameters. There are practical methods to obtain the parameters.

When all the parameters are in hand, we can arrange active bots to carry out a mimicking attack. We present the implementation detail of the mimicking attack in Algorithm 1.

Algorithm 1: The mimicking attack algorithm

1. Observe the target web site, and extract the related browsing dynamic parameters $\alpha_z, q, \alpha_p, \mu_l, n(t)$.

2. Initialize the parameters of the semi-Markov model.

3. Take $n(t)$ bots from a set of active bots, $\{bots\}_t$, and instruct these bots to run independently.

4. **foreach** bot ϵ $\{bots\}_t$ **do**

4.1. Generate a random number *rnd*.

4.2. Identify an initial page according to equation (1) With *rnd* ;

4.3. Decide the browsing length L for this bot using equation (3) with *rnd*;

4.4. j = 1 ;

while j ≤ L **do**

a. Submit the request and discard the downloaded content;

b. Wait for a time interval decided by equation (2) and *rnd*;

c. j = j + 1;

d. Identify the a new page to request following the semi-Markov model .

end

4.5. Remove the current bot from the set $\{bots\}_v$;

end

5. Introduce new bots and update $n(t)$;

6. Go to step 4.

This algorithm can be used to launch a flash crowd mimicking attack if we have a target flash crowd to obtain the browsing dynamic parameters. This methodology can be applied to other types of mimicking attacks, such as email spamming, botnet membership recruitment or virus spreading.

3.4 The Mimicking Attack Detection Algorithm

In order to detect the flash crowd mimicking attacks, We can manually supervisor the network traffic of the web site, which we try to protect, for a number of given periods (we use twenty four hours as one period).
.

The granularity of time could be at the second or minute level in order to detect attacks in time.

The detection algorithm is shown in detail in Algorithm 2.

Algorithm 2: The mimicking attack detection algorithm

1. Establish the profile of $R(t)$ for a 24 hour period;

2. Establish a mapping of the variation of flow fine correntropy
of page request flows against $R(t)$, and denote as $V_f(n(t))$;

3. **while** {true} **do**

Monitor the volume of page requests of the web site, denote as $R'(t)$;

while { $R'(t) \geq R(t)$ } **do**

a. Following statistical methodology, sample request flows for sufficient sample points;

b. Calculate the flow fine correntropy $V'_f(t)$;

c. $delV_f(t) = | V_f(t) - V'_f(t) |$;

d. **if** $delV_f(t)$ is sufficient **then**

it is mimicking attack

else

do nothing

end

end

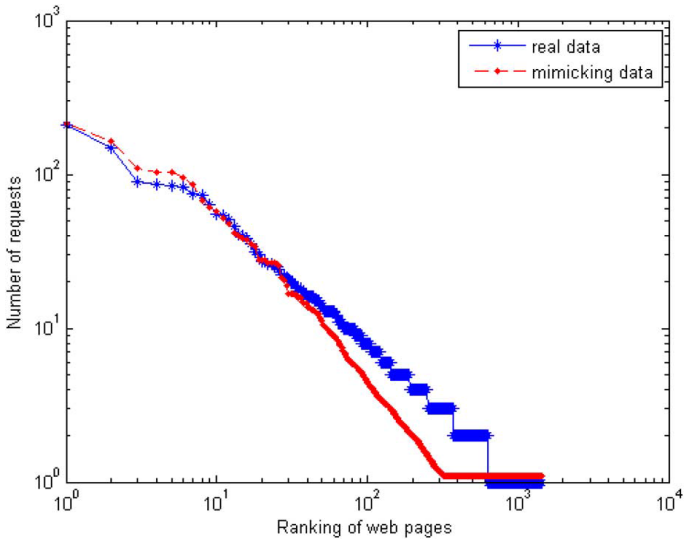
end

The goal of the proposed method is to detect flash crowd mimicking attacks, rather than identify attack sources, which is referred to as traceback.

4. Effectiveness of mimicking attack model:

The effectiveness of the proposed mimicking attack algorithm, we collected the web dynamic data of a popular news web site for thirty days.

.



(comparison of web pages and target data)

Fig(4.1)

We used the data from day 1 as a training data set, and extracted the key parameters from the data set to populate the parameters of the semi-Markov model. We call these as training data and target data, respectively.

We arrange the data sets into a matrix for both days: each web page of the web site is a row in the matrix, and every column denotes the number of requests in a thirty minute duration

IV. CONCLUSION AND FUTURE ENHANCEMENT

We have established a mathematical model to simulate the browsing dynamics of legitimate web browsers . However ,There is a critical condition for a successful mimicking attack: the number of active bots of the botnet must not be lower than the number of active legitimate users. Based on this new finding , we therefore proposed a second order statistics based discrimination algorithm to detect this kind of attack. Our theoretical analysis and simulations confirmed the effectiveness of the proposed detection.

Our future work will follow a lot of legitimate network events that do not involve a large number of users. Therefore , botnet owners do have the capability to perform perfect mimicking attacks, likes membership recruitment, performance degradation attacks, and so on. We have a significant interest in addressing this problem by finding new methodologies.

REFERENCES

[1] B. A. Huberman, P. L. T. Pirolli, J. E. Pitkow, and R. M. Lukose, "Strong regularities in world wide web surfing," Science, vol. 280, no. 3, 1998.

[2] M. Mitzenmacher, "A brief history of generative models for power law and lognornal distributions," Internet Math., vol. 1, 2004.

[3] S. Yu, W. Zhou, S. Guo, and M. Guo, "A dynamical deterministic packet marking scheme for DDoS traceback," in Proc. IEEE Global Telecommun. Conf. (Globecoms), 2013.

[4] Y. Xiang, W. Zhou, and M. Guo, "Flexible deterministic packet marking: An IP traceback system to find the real source of attacks," IEEETrans. Parallel Distrib. Syst., vol. 20, no. 4, pp. 567–580, June 2009.

[5] Y. Xie and S.-Z. Yu, "A large-scale hidden semi-Markov model for anomaly detection on user browsing behaviors," IEEE/ACM Trans. Netw., vol. 17, no. 1, pp. 54–65, Feb. 2009.

[6] G. Carl, G. Kesidis, R. Brooks, and S. Rai, "Denial-of-service attack-detection techniques," IEEE Internet Comput., vol. 10, no. 1, pp. 82–89, Jan./Feb. 2006.