Middle-East Journal of Scientific Research 23 (Sensing, Signal Processing and Security): 342-351, 2015
ISSN 1990-9233

DOI: 10.5829/idosi.mejsr.2015.23.ssps.202

Resisting Proxy Based Spoofing Attacks

D. Julia Jemila, D. Sugith Deenan and P. Preethy Rebecca

Department of Computer Science & Engineering,
St. Peter's College of Engineering & Technology, Chennai-54, India

Abstract: Botnet are the serious cause for spiteful bustle in cyberspace recent days. To bear their botnets and to hide their spiteful action, botnet possessor are mimicking legal cyber behavior to pass under the radar. This raise a crucial problem in anomaly detection. In this paper, a popular web site has been selected and web browsing behavior of that web site is taken as an example to manage this problem. To find this browsing behavior semi-markov model has been used. This model shows it is difficult to detect mimicking attack if the number of active bots is larger than the number of legitimate user in the network. Most of the time it is difficult for botnet possessor to fulfill the constrain to carry out a mimicking attack. From the discovery using second order statistical metrics, mimicking attack can be differentiated from genuine flash crowd attack. The main objective of this project is to perform a study of legitimate cyber behavior mimicking attacks from the attackers as well as from defenders side of perspective. And to discriminate mimicking attacks from legitimate user in the network. The theoretical experiments and simulations result confirms the claim. These findings can be widely in different applications and also in other research fields.

Key words: Mimicking attack · DDOS attack · Flash crowd attack · Phishing attack

INTRODUCTION

Cyber-attack is any type of offensive maneuver employed by individuals or whole organizations that targets computer which are information systems, infrastructures, computer networks and personal computer devices by various means of malicious acts usually originating from an anonymous source that steals, alters and will destroys a specified target by hacking into a sensitive system. These are labelled as either a Cyber campaign, cyber war or cyber terrorism in different context. Cyber-attacks can install spyware on a PC to attempts to destroy the infrastructure of entire nations. Cyber-attacks have become increasingly advanced and dangerous as the Stuxnet worm recently demonstrated.

Botnet: A botnet is a collection of Internet-connected programs communicating with other similar programs in order to perform illegal tasks. This can be as terrene as keeping control of an Internet Reply Chat channel, or it could be used to send spam email or participate in DDOS attacks [1]. The abbreviation of the word botnet is a combination of robot and network. The term bot is short for robot. Hackers distribute malicious software (also known as malware) which turn your computer into a bot (also known as a zombie). When the system is hacked, your computer will perform automated tasks over the Internet, without your knowledge. Hackers typically use bots to infect large numbers of computers. All these computers which form a network called as botnet.

Fig. 1: Botnet

Corresponding Author: D. Julia Jemila, Department of Computer Science & Engineering,
St. Peter's College of Engineering & Technology, Chennai-54, India.

Hackers use botnets to send out spam email messages, attack computers and servers, spread viruses and commit other kinds of crime and deceit. When your computer becomes part of a botnet, [2] it might slow down and you might inadvertently be helping criminals.

The term "botnet" [3] can be used to refer to any group of computers, such as Internet Reply Chat (IRC) bots, but the term is generally used to refer to a collection of computers (called zombie computers) that have been recruited by running malicious software. A botnet's originator (known as a "bot herder" or "bot master") can control the group remotely, usually through an IRC and often for wicked purposes. The back bone of this server is known as the command and control (C&C) server. Though rare, more practised botnet operators program command protocols from scratch. These protocols include a server program and a client program for operation, the program that combines the client on the victim's machine. These communicate through a network, using a unique encryption strategy for stealth and protection against detection into the botnet.

Mimicking Attack: Mimicking attack is the Application layer DDOS attack. It mimics human browsing behavior. False negative is always a problem for intrusion detection system in real practice. e.g,page request interval, numbering of browsing page in a session. Study of mimicking attacks and detections from both sides, as attackers as well as defenders, which is a significant extension based on our preliminary work in. From the botnet programmers' perspective, to observe the legitimate behavior of a web browser, three key pieces of information needed: web page popularity of the target user website, web page requesting time interval for a user and number of pages a user usually browses for one browsing session. Based on the research on web browsing dynamics, there are three distributions in place for the three key pieces of information. If botmasters have a sufficient number of active, then each bot can simulate one legitimate user using the three statistical distributions. However, it is hard for botnet owners to meet the sufficient number condition for certain mimicking attacks, such as flash crowd attacks. Demonstrate that botmasters can simulate a flash crowd successfully in terms of statistics. With a adequate number of active bots, a botmaster can use one bot to simulate one legitimate user using the knowledge of web browsing dynamics.

Web browsing is very popular nowadays and the web becomes a major media for information dissemination for governments, companies and individuals. DDoS attacks on web sites reward attackers financially or politically. Have witnessed increasing number of this kind of attacks. A few detection and mitigation strategies are in place for application layer DDoS attacks [1].

Related Work: In this section, we briefly describe some of the approach of spoffing attacks and its detection.

A new type of bot called "TORPIG" is discovered. Investigated for a period of ten days. More than 180 thousand infection and almost 70GB of data which has been infected. While botnets have been "hijacked" and studied previously, the Torpig botnet exhibits certain properties that make the analysis of the data particularly interesting [4]. To estimate the size of Botnet. Botnet Size can reach 350,000 members. To find the size of the botnet, there are two methods explained in this paper [5]. (a)Foot Print-Overall size of the infected population in a network can be estimated. But it does not captures the actual capacity, CCDF method is used. (b) Live population-It shows the live bots present in C&C channel, It shows the actual capacity. Botnet Infiltration, Infiltrate botnet by joining C&C channel. Military force moves as individuals through enemy position without detection [6]. IP trace back follows FDPM method, which is able to find the real source of the attacking packets. It shows, (a) How many resources can be traced in one trace back, (b) How large is the false positive rate. (c) How many packets are needed to trace one source.

FDPM Characteristic 1. Flexibility 2. Adaptively changes its marking rate [7]. Due to the memory less features of the internet routing mechanism makes difficult to trackback the source of the attacks. Source of the attack is found out by using "Entropy Variation". It Calculates Packet size. Shows the difference between Normal and DDOS attacks traffic [8]. HCF(Hop-Count-Filtering) which builds an accurate IP-to-hop-count (IP2HC) mapping table to detect and discard spoofed IP packets. HCF is easy to deploy, as it does not require any support from the underlying network. Through analysis using network measurement data, [9] HCF can identify close to 90% of spoofed IP packets and then discard them with little collateral damage [10]. Kill-Bots provides authentication using graphical tests but is different from other systems that use graphical tests. First, Kill-Bots uses an intermediate stage to identify the IP addresses that ignore the test and persistently bombard the server with requests despite repeated failures at solving the tests. Second, Kill-Bots sends a test and checks the client's answer without allowing unauthenticated clients access to sockets, TCBs and worker processes [11].

Flash-crowd attacks are the most vicious form of distributed denial of service (DDoS). defenses against flash-crowd attacks via human behavior modeling, which differentiate DDoS bots from human users. Current approaches to human-vs-bot differentiation, such as graphical puzzles [12], are insufficient and annoying to humans, whereas our defenses are highly transparent. Three aspects of human behavior: a)request dynamics, by learning several chosen features of human interaction dynamics and detecting bots that exhibit higher aggressiveness in one or more of these features. b) request semantics, by learning transitional probabilities of user requests and detecting bots that generate valid but low-probability sequences. c) ability to process visual cues, by embedding into server replies human-invisible objects, which cannot be detected by automated analysis and flagging users that visit them as bots [13]. DDoS is a spy-on-spy game between attackers and detectors. Attackers are mimicking network traffic patterns to disable the detection algorithms which are based on these features. zombies use controlled function(s) to pump attack packages to the victim, therefore, the attack flows to the victim are always share some properties. If distance is less than Threshold, then it is DDOS Attack. If distance greater than Threshold, then it is Legitimate access.

Proposed Work: A study of mimicking attacks and detections from both sides, as attackers and defenders has been made, which is a significant extension based on our preliminary work in. From the botnet programmers' perspective, in order to simulate the legitimate behavior of a web browser, we need 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 browses for one browsing session. Based on the research on web browsing dynamics, there are three distributions in place for the three key pieces of information. If botmasters have a sufficient number of active, then each bot can simulate one legitimate user using the three statistical distributions. However, it is hard for botnet owners to meet the sufficient number condition for certain mimicking attacks, such as flash crowd attacks [12, 14]. We demonstrate that botmasters can simulate a flash crowd successfully in terms of statistics. With a sufficient number of active bots, a botmaster can use one bot to simulate one legitimate user using the knowledge of web browsing dynamics.

Botnet Architecture: The primary intent of cybercriminals and botmasters is to reach a wide audience of users remaining hidden to principal security firms, it's natural that they are exploring the possibility to exploit social media platforms. Social have monopolized the majority of user's internet experience; the principal factor of attraction for cyber criminals is the huge number of services from gaming to payments, that are developing for these platforms that could be exploited to realize more or less complex fraud schemas. The relationship between social networks and botnet is strict. This second scenario is becoming very common, [15] botnet authors are using various social network platforms to control the infected machines, typically the create fake accounts that send encrypted messages to malware on victims. The principal advantage of this approach is that the traffic related to botnet based on a social network is very hard to detect.

Client Architecture and Updates: 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 .Count the number of HTTP requests of each flow for the given time intervals and to describe the browsing behavior of a legitimate web viewer or user.

Client Browsing Behavior: The client browsing behavior is captured. Proxy setting in the web browser is changed. IP address of the victim system is given in the proxy setting. All HTTP request of the victim is observed at the server side. So, now all the HTTP request, which the user types in the URL will be shown in the botnet server. The URL will then be stored in the Botmaster Database.

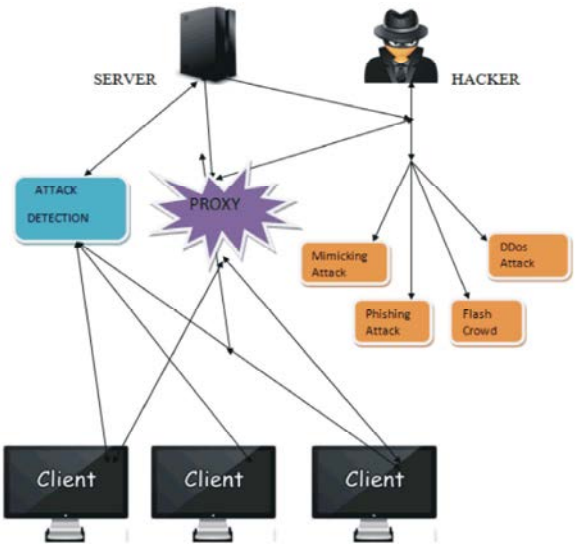


Fig. 2: Botnet Architecture diagram

BotMaster Architecture: Design 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.

Mimicking Attack and its Detection: Using the collected details about victim in the botmaster will successfully generate flash crowd attack and mimicking attack. If any modification done in the botmaster page it will automatically reflect in the victim client website. After analyzing the client response from the server, can able to detect the mimicking attack.

Implementation: The legitimate user browsing Behavior is observed by the Botmaster. It fairly easy to spoof an Ip address that is the "referrer" or "referring IP" and since most users are on a shared IP for an entire city/ hosting provider the number of "referring IP's" [16] is a finite number with some ip's carrying much more weight in terms of visitors than others.Many bots use IRC for Command and Control, So detect IRC Bot commands by Off ramp TCP port 6667, Insoect payload and IRC behavior. web accessing behavior and page popularity follows the Zipf-like distribution. For a given website, assume that there are N(N > 0) web pages in total and they are sorted in terms of popularity from the most to the least as w1, w2, . . . , wN. Let random variable W be the requested web page and Pr [W = wi] be the request probability of page wi. Then the Zipf-Mandelbrot distribution can be formulated as.

$$P_{r[w=w_i]} = \frac{\Omega}{(i+q)^{\propto z}}$$

where αz(αz > 0) is the skewness factor, which dominates the skewness of the distribution and q(q = 0) is the plateau factor. Once a browsing page has been decided, a bot submits the page request to the victim and downloads the page to the host computer without displaying it (e.g. discarding it or depositing it to the cache). When the requested page has been downloaded, the bot decides a “reading” time interval following the Pareto distribution before requesting another web page. A flow is a group of HTTP requests that share the same source IP and destination IP addresses(IP spoofing) [17]. For a given observation point,count the number of HTTP requests of each flow for the given time intervals. As a result, a flow is a sequence of numbers. Denoted as F(μ, σ), where μ is the mean of the flow and σ is the standard deviation of the flow. Physically, μ is the average number of HTTP requests for a web page over the observation time intervals.To describe the browsing behavior of a legitimate web viewer, the classical Markov model to a four parameter semi-Markov model [18] as follows Λ = (P, T,L, π), where P, T,L, π represents the state transition matrix, duration at the current state, browsing length and the initial probability distribution of the states, respectively. State transition matrix can be represented as P = {pij}, 0 = i, j = N, T, represents the time duration a viewer stays at the current state. T = {ti}, i = 0, 1, _E _E _E, N; 0 = ti = +8, L, represents the browsing length of the current session. L = {li} = {0, 1, _E _E _E, N}. π is the probability that a viewer selects a page as the first page of his browsing session.The number of active web viewers for a given time point t, which denoted as n(t). n(t) varies against the time point of a day. Intuitively, there are more web viewers during working time than early morning.A 30 days observation on n(t) for every 30 minutes and found that n(t) was stable day after day. The duration of a browsing session for a user is dominated by $P_r[D = t.l] = P_r$ [18].

Algorithm 1: The mimicking attack algorithm

1. Observe the target web site and extract the related browsing dynamic parameter *αz, q, αp, λ, μl,* n(*t*).
2. Initialize the parameter 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. For each bot □ { *bots* }t do
5. Generate a random number *rnd*.
6. Identify an initial page according to equation (1) with *rnd*.
7. Decide the browsing length L for this bot using equation (3) with *rnd*.
8. if j ≤ 1;
While j = L do

- Submit the request and discard
- the downloaded content.
- Wait for a time interval decided
- by equation(2) and *rnd*.

c. j = j+ 1;
d. Identify a new page request following
the semi-Markov model □.
end
9. Remove the current bot from set { bots}t.
end

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.

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 the flow fine correntropy of page request flows against R(t),and denote as Vf (n(t)).
3. While {true} do
Monitor the volume of the page requests of the web site, denote as R' (t)
While {R'(t) ≥ R(t)} do
a. Following statistical methodology, sample request flows for sufficient sample points
b. Calculate the flow fine correntropy V'f (t);
c. ΔVf (t)=|Vf (t) -V'f (t)|;
d. if ΔVf (t) is sufficient then
it is mimicking attack;
else
do nothing;
end
end
end

In order to improve the detection effectiveness of this method, we will use the fine correntropy to replace the standard deviation as a metric to repeat the same experiments as conducted.It is impossible to discriminate the attack when ρ = 1, but can clearly differentiate them when ρ = 0.98. Combined with the variation of the number of legitimate users it is found that the obtained threshold for effective detection is around 19.6% = 0.98, which means a botmaster has to possess more than 19.6% of the number of legitimate users to fly under the radar. In other words, the detection accuracy improves around 3 times using the proposed fine correntropy metric compared to using the standard deviation as the metric.

Performance Analysis: Reliance on detecting Bot Communication degenerates into arms race between bot authors and defenders. Communication is very flexible, easy to encrypt/obfuscate, Relying on detecting bot communication is not viable in the long term, Leverage all available bot characteristic and build detectors for each

Topology	Design Complexity	Detectablity	Message Latency	Survivability
Centralizes	Low	Medium	Low	Low
Peer- to - Peer	Medium	Low	Medium	Medium
Random	Low	High	High	High

Fig. 3: Botnet communication

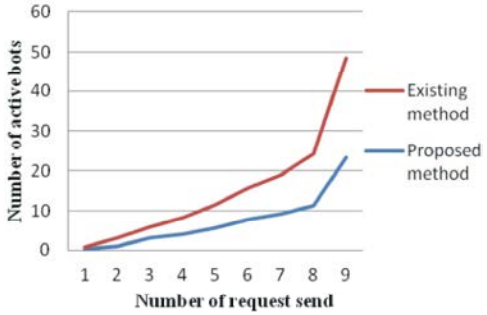


Fig. 4: Ratio of Active bots in the network

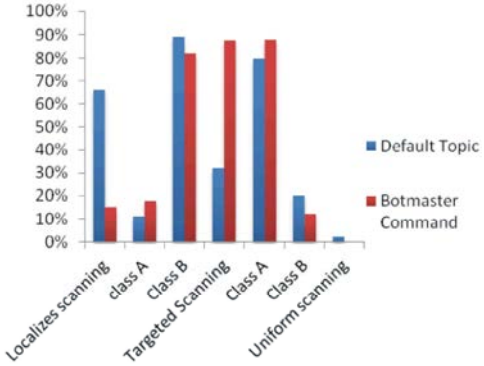


Fig. 5: Botnet scan result

bot behavior. Preliminary evidence very promising, It has strong correlation between bot communication and bot propagation. Correlating data sources from a large live network.

The comparison of existing and proposed graph in Fig. 4 shows, that the number of active bots in the network is more in the existing system. whereas the number of active bots are less in proposed system. This is due to the mimicking attack done by the server in the proposed system on frequent basis.

It is impossible to find the legitimate user inside the network when the active bot is more than the legitimate users. The ratio shows there is less active bots in the network when compared to existing system. Thus, early finding of active bots in the network will prevent legitimate users inside the network from the attack of

Botmaster. The Botnet scan in the Fig. 5 shows the localizes scanning and targeted scanning. From this both scanning the Botnet can be found out . The class A and class B in the in Localizes scanning shows, at class A the default topic is less when compared to the Botmaster command and the class B shows default topic is more than the Botmaster command. In the Targeted scanning for class A and class B the default is more than the Botmaster command. Finally, the uniform scanning shows only the default topic.

There are two methods to scan bot in the botnet.

- Immediately start scanning the IP space looking for new victims after infection.
- Scan when issued some command by botmaster.

RESULT AND DISCUSSION

A web page is created and that is the victim web page, if the client is new to that web page, they have to register in that web page. After registration, the client will login with the user name and the password. The web page will be opened. All the client registration details will be stored in the database. In that web page, the client have the rights to upload, download, search, update and retrieve the files in the website.

Botmaster will observe the browsing behaviour of the target victim and store all the behaviours in the Botmaster database. Here from the existing two more attacks has been included in proposed method. So, here the Botmaster will perform folowing four attacks namely.

- Flash Crowd attack
- Mimicking Attack
- Suspending DDOS attack
- Phishing Attack

Client system is hacked by flash crowd attack, instead of retrieving one file from the server, the files will be retrieved n number of times, where the value n is given by the Botmaster in the Botmaster page.

Here the phishing attack shows the sensitive information of the user, Based the client web page the sensitive information will be their user name and password. here the user name and password is hacked using phishing attack.

When the server finds there is some change in the browsing behavior of the legitimate user, The browsing behavior has been analyzed and the server found that phishing attack has been taken place. Server will notify the legitimate user about the attack by sending the message.

Fig. 6: Client Registration

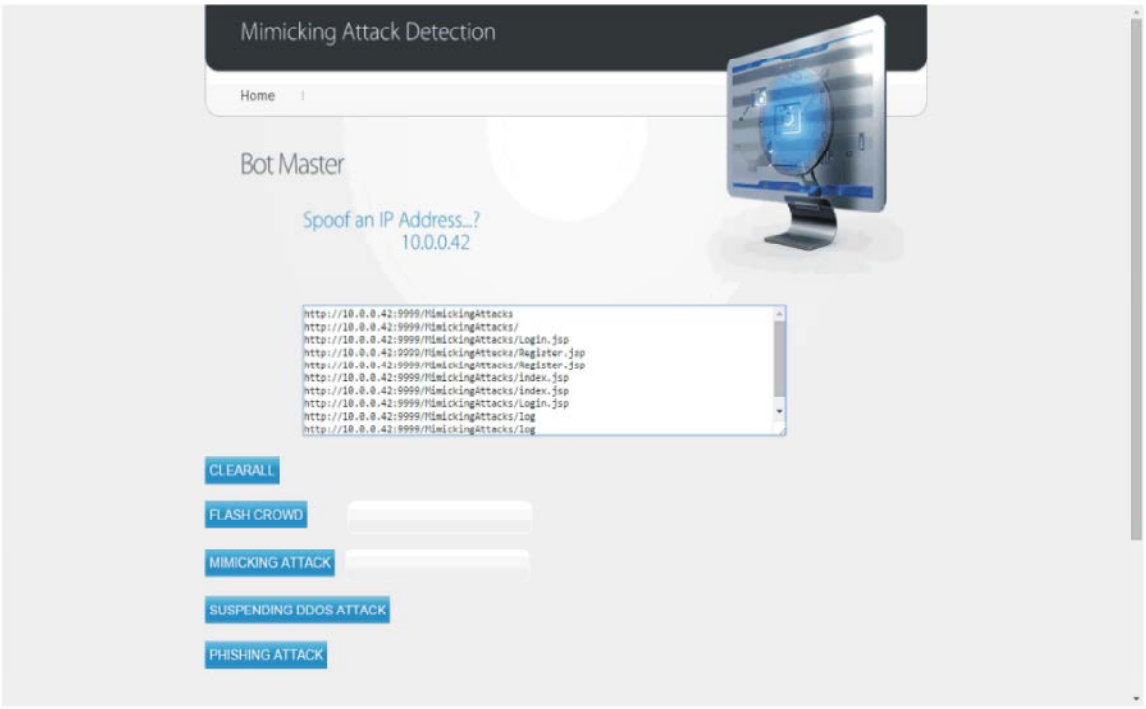
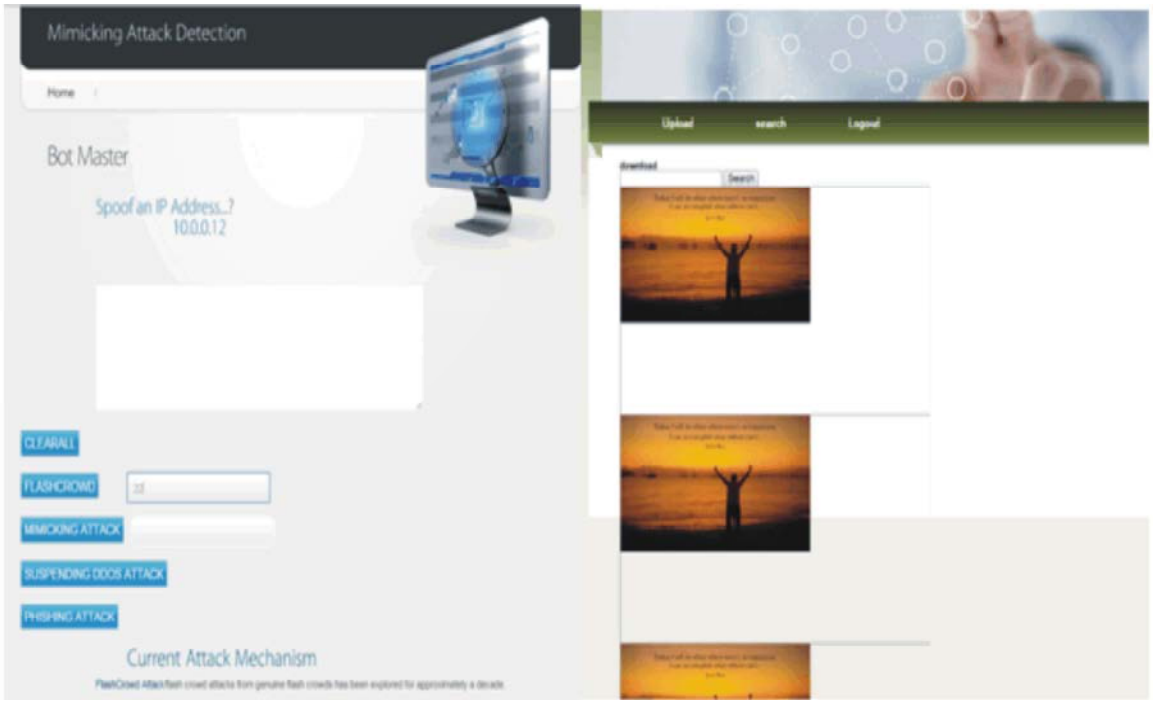


Fig. 7: Client Browsing Behavior



a)Attacker perform flash crowd attack b)Flash crowd attack in victim webpage

Fig. 8: Flash Crowd attack

Once the legitimate user found that their system has been hacked by the message send by the server, they will send the IP address and the port number and ask the server to release their IP address.

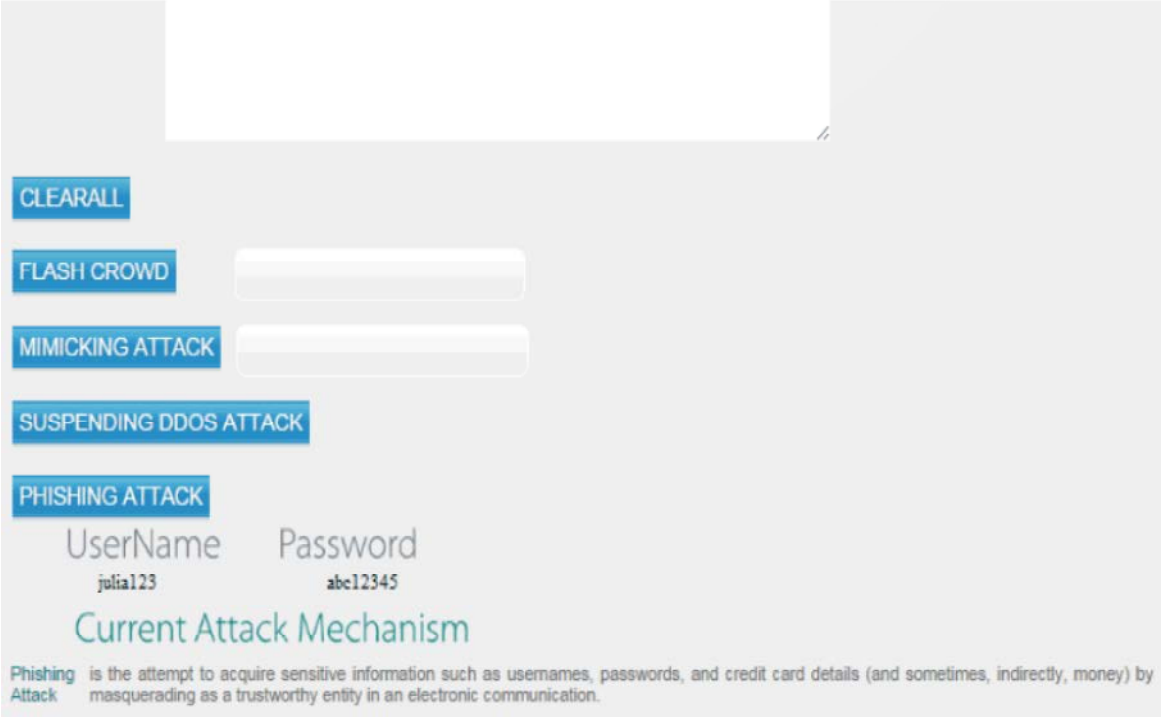


Fig. 9: Phishing Attack

Fig. 10: Client Browsing Behaviour analysis

Fig. 11: Client IP address is banned

Fig 12 Client page is secure

When the server gets the response from the legitimate user. The server will check the IP address and the port number, if that matches the legitimate user. IP address is released and the user will be back to the secure web page.

CONCLUSION

By using the taxonomy and accurately identifying what type of botnet, it will be easier to use the correct evasion technique. Bot provide support infracture for a

large range of devasting internet attacks. IRC base notnet detection are effective. This project shows, how the client will register in the website and that data will be stored in database. Clients have the right to upload their files 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. Count the number of HTTP requests of each flow for the given time intervals and to describe the browsing behavior of a legitimate web viewer or user. Here, the user browsing behaviour is monitored using Spoofing of IP address [16].

There are many legitimate events with small number of active users in cyberspace, which make it easy for botnet owners to meet the critical number condition to successfully mimicking those kind of events to carry out their malicious goals. Based on this analysis, Design a web page to observe the potential victim for sufficient time in attack free cases. This will 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. Using the collected details about victim in the botmaster will successfully generate flash crowd attack and mimicking attack. If the bot perform any modification in the botmaster page it will automatically reflect in the victim client website. After analyzing the client response from the server, can able to detect the mimicking attack.

REFERENCES

1. Carl, G., G. Kesidis, R. Brooks and S. Rai, 2006. "Denial-of-service attack detection techniques," IEEE Internet Computing, 10(1): 82-89.
2. http://en.wikipedia.org/wiki/Botnet.
3. Ianelli, N. and A. Hackworth, 2006. "Botnets as vehicle for online crime," in Proceedings of the 18th Annual FIRST Conference.
4. Stone-Gross, B., M. Cova, L. Cavallaro, B. Gilbert, M. Szydlowski, R. Kemmerer, C. Kruegel and G. Vigna, 2009. "Your botnet is my botnet: Analysis of a botnet takeover," in Proceedings of the 2009 ACM Conference on Computer Communication Security,
5. Rajab, M.A., J. Zarfoss, F. Monrose and A. Terzis, 2007. "My botnet is bigger than yours (maybe, better than yours): why size estimates remain challenging," in HotBots'07: Proceedings of the first conference on First Workshop on Hot Topics in Understanding Botnets.
6. A.P., 2007. ACM Computing Survey, "An IP Trace back System to Find the Real Source of Attacks". 39(1).
7. International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, "Entropy Based Detection of DDOS Attacks". Volume-1, Issue-5, November 2011.
8. Yu, S., W. Zhou, S. Guo and M. Guo, 2013. "A dynamical deterministic packet marking scheme for ddos traceback," in Proceedings of the IEEE Globecom.
9. Wang, H., C. Jin and K.G. Shin, 2007. IEEE/ACM Trans. Netw.,."Defense against spoofed ip traffic using hop-count filtering," 15(1): 40-53.
10. Srikanth Kandula Dina Katabi, MIT {kandula,dina} @csail. mit .edu, "Botz4Sale: Surviving Organized DDoS Attacks That Mimic Flash Crowds" Matthias Jacob Princetonmjacob@princeton.edu, Arthur Berger MIT/Akamai awberger@mit.edu.
11. Oikonomou, G. and J. Mirkovic, 2009. "Modeling human behavior for defense against flash-crowd attacks," in Proceedings of the 2009 IEEE Conference on Computer Communication.
12. Jung, J., B. Krishnamurthy and M. Rabinovich, 2002. "Flash crowds and denial of service attacks: Characterization and implications for cdns and web sites," in Proceedings of the WWW. IEEE, pp: 252-262.
13. Yu, S., W. Zhou and R. Doss, 2008. IEEE CommunicationsLetters, "Information Theory Based Detection Against Network Behavior Mimicking Ddos Attack," 12(4): 319-321.
14. Yu, S., G. Zhao, S. Guo, Y. Xiang and A. Vasilakos, 2011. "Browsing behaviour mimicking attacks on popular websites," in INFOCOM Workshops.
15. Yu, S., S. Guo and I. Stojmenovic, 2012. "Can we beat legitimate cyber behavior mimicking attacks from botnets," in Proceedings of INFOCOM, pp: 3133-3137.
16. Duan, Z., X. Yuan and J. Chandrashekar, 2008. "Controlling ip spoofing through interdomain packet filters," IEEE Trans. Dependable Sec. Comput., 5(1): 22-36.
17. Peng, T., C. Leckie and K. Ramamohanarao, 2007. "Survey of network-based defense mechanisms countering the dos and ddos problems," ACM Computing Survey, 39(1).
18. Shui Yu, 2013. Senior Member, IEEE, Song Guo, *Senior Member, IEEE* and Ivan Stojmenovic, *Fellow, IEEE,"*Fool me if you can:Mimicking attack and anti-attacks in cyber space"IEEE transactions on computers, 2013.