Scientific Research Publishing

Journal of Computer and Communications, 2023, 11, 159-166
https://www.scirp.org/journal/jcc
ISSN Online: 2327-5227
ISSN Print: 2327-5219

Captcha-Based Honey Net Model against Malicious Codes

Adeniyi Akanni[1], Williams Akanni[2], Oluwafunmilasyo Helen Daso[3]

[1]Department of Computer Science, Caleb University Lagos, Lagos, Nigeria
[2]Akademia Humanistyczno-Ekonomiczna w Łodzi, Łódź, Poland
[3]National Population Commission, Headquarters, Abuja, Nigeria
Email: adeniyiakanni@gmail.com

How to cite this paper: Akanni, A., Akanni, W. and Daso, O.H. (2023) Captcha-Based Honey Net Model against Malicious Codes. *Journal of Computer and Communications*, **11**, 159-166.
https://doi.org/10.4236/jcc.2023.113012

Received: July 16, 2022
Accepted: March 28, 2023
Published: March 31, 2023

Abstract

The rate of passive and active attacks has been on the increase lately affecting both individuals and institutions. Even when internal control procedures are in place, malicious codes from intruders into the network have left so much to be desired. As a result, many Chief Information Security Officers have grown grey hair because of their inability to effectively handle attacks from various ends. Various attempts and technologies have been made in the time past with a measure of success. Intrusion Detection Software (IDS), Intrusion Prevention Software, firewall, honey pots and honey nets have been deployed and with great respite from losses arising from cyber-attacks. Cyber security is the duty of everyone and all must see it as such. As tiers of government and law enforcement agents are doing their best, everybody must be seen to play their parts. Fraudsters have also not seemed to be tired of seeking vulnerabilities to exploit. Then, cyber security experts should not let off their guards but make efforts to harden their security. A way of doing is to intelligently provide a solution that has the capability of detecting and proactively hardening security. This paper proposes a honey net model that is captcha-based and capable of extracting details from hackers with a view to building a robust defense against black hat attackers. This research was able to prevent the botnet with the use of captcha and also redirect suspected traffic to the honeynet which was then captured for the purpose of improving the security of the network. The result showed that any bandwidth greater than the set threshold was not allowed to go into the network but redirected to honeynet where details were logged. Also, with a threshold of 100 mbs, inbound traffic of higher bandwidth such as 110 mbs and 150 mbs was denied access thereby giving 100% detection rate.

Keywords

Passive Attack, Active Attack, Honey Net, Malware, Internet of Things (IoT),

Log

1. Introduction

Honey has been known to have many benefits [1]. Apart from the microbial effects that can bring healing to mankind from honey, the attraction it gives insects can also be a useful lead to preparing decoys. Increase in deployment and usage of Information and communication Technology have brought the world a relief. The trend is not likely going to reverse soon especially with the global adverse effect of COVID-19 pandemic that brought about working from home, virtual learning and more transactions on eBusiness platforms. Intruders are bound to launch more attacks through organizations. Ability to detect botnet that can cause denial of service (DOS) is major headache on one hand. This gave the impetus to this work. Equally important is the ability to extract details with a view to building a robust network against Malware. Honey pots have proven to be quite helpful by serving as a decoy to lure attackers to another purposely built server which will in turn capture details of attacker's activities. Such details include IP address and possible location of the attacking codes. As it is typical of Intrusion Prevention Systems (IPS) and Intrusion Detection Systems (IDS), identified pattern or signatures used to be robust security against intruders. However, known patterns may not be available until they are captured to determine which is human or machine driven activities. This research is proposing a model that helps to fortify network security against malicious codes using honey net that is based on Completely Automated Public Turning test to tell Computers and Human Apart (CAPTCHA). As a form of building up to security, honey pot or better still, honey net is placed within a network to mimic the production server but lures invaders to interact with it and by so doing captures useful information that can be utilised to secure the network and block malicious codes from harming the live server.

Honey pot is known with espionage; where a law enforcement agent pretends to be a victim whereas he is equipped with information-gathering gadgets for the purpose of investigating and getting to the root of the matter. In much the same way, a honey pot is subjected to being made to interact with a system or code or bots so that attentions are shifted away from the live servers to the decoy. It is has become so much useful in cybersecurity. A major challenge in the world today is that black hat hackers are also perfecting their strategies to infiltrate systems to carry out nefarious activities. They sometimes build intelligent codes that can by-pass or beat network of security. This gave rise to honey net, having a number of honey pots. This is why this research goes employed honey net.

Honey pots can be categorized by so many ways such as by roles or level of interaction. This research is based on the latter, low or high interaction honey pot. Where the few resources are involved in obtaining vital information from

attackers, this is known as low interaction honey pot. This category does not interact for so long with the attacker. Hence, it is economical to deploy. On the other hand, high interaction honey pot engages attacker for so long to get as much details as possible. Virtual machines are used for this category and compromised machines are isolated. With this category, deep details are extracted such as vulnerabilities exploited and process of escalating privileges among others. It should be noted this type of honey pot is expensive and consume more bandwidth. In all, honey pots would capture IP address, base of the attacker, how serious the attack is, methodologies used by hackers and shows the strength or otherwise in combating attacks which invariably help to improve on security of the network. Honey nets have been defined as an extension of a network of honey pots with three basic elements involving log, control and storing of details [2].

Malicious codes are ubiquitous. The effect of managing them can be damning. Stress in terms of emotion, financial and time involved in treating infested system or software may so difficult to quantify. They could come in through diverse ways and means. Sometimes, they can come through direct active attack on targeted system in form of brute force, trapdoor among others. They may even come through download of free software. Unsuspecting staff can download applications (apps) to their devices which may have harmful effects not only on that device but as many as are in that network. The best approach to this is to avoid free items: WiFi or apps. In practical terms, this is not feasible. However, mission critical servers can be secured such that even when other systems within the network are compromised. The compromised system can be traced and isolated for necessary attention. This paper proposes a way to secure a network through a decoy to the extent that attackers are lured away from the production servers into the honey nets that will be capable of interacting with the attacking malicious codes and log essential details obtained in that process. The extracted information would then be used to beef up the security on the network.

Honey pots or honey net can prove useful because one may not need new systems to set them up. Chances of raising false alarms are slim. It is should however be noted that honey net is not a sufficient prove of security. Honey pots in a honey net can serve better in complimenting one another. In order to build appropriate controls, details obtained from honey pots would need to be properly reviewed [3]. Honey net was used here to overcome the challenge of bypassing a honey pot by hackers such if one is passed, others would be able to capture its details and redirect it away from the live server.

2. Malicious Code

On a daily basis, we work with files. We create new ones and update existing ones. There are files that readily came with our systems and computing devices. Files are thus, main targets to be attacked by appending themselves to such authentic files. They came as file but are not the ones needed. These unsolicited

files or programs or codes are called malicious codes. They can cause havoc to computers as well as files stored on them. Malicious codes can be viruses, worms, Trojan horses or malicious data files [4]. A malicious code or software is also called a malware. Most times, attackers would send malware from systems like robot that may not be able to do some rational thinking. This is otherwise known as bots. Since they are run on severally devices that are interconnected, then we think of botnets signifying robot joined with network [5]. Several attacks such as spam mails, Distributed Denial of Service, can result from botnets. Generally, malware can be seen to exist in three main ways viruses, worms and Trojan horses.

3. Honey Pots and Honey Nets

Honey pots are known security mechanisms capable of luring attackers in order to extract information that can in turn be used to build security around a network. They are broadly categorized as Production and Research Honey pots. Both of them are set up to extract details. The Production honey pots are used to obtain details from an intruder whose attention is shifted away from actual or servers. Honey pots give rise to honey nets. The deployment of honey pots and honey nets are practically of two types: the Low Interaction Honey pots (LIH) and High Interaction Honey pots (HIH).

LIH usually maintains limited interactions with intruders and may not be fully compromised. This may not be unconnected with the restricted access to system resources. For this reason too, the cost is reduced. It is not quite difficult to configure. On the other hand, HIH has more interactions.

4. Related Work

Several researches have been carried out which also show that there is need for further work in this field of cybersecurity. These are review in this section.

In their work, [6] set up an experiment to monitor levels of attack on various applications. Their result showed that honey pot can be a useful tool to foresee attackers' target. The research revealed that phpadmin was attacked the most and the brute force ranked highest. Most of the attacks came from USA and China.

According to [7], not all attacks can be sufficiently tracked by using signatures. They used decoy ports in their research to lure attackers such that details of their activities were captured which could be used to build a robust secure network.

In a survey on dynamic honeypots where researchers did a comparative analysis of dynamic honeypots and recommended that their novel plug and play honeypot would serve better. They also x-ray merits and demerits of where to position the honey pot whether in front or behind the firewall. Their work also revealed greater bandwidth consumption when dynamic honey pot is deployed [8] [9]. In their study of honey pot in Internet of things, they created a fake envi-

ronment via honeypot. This was then used to interact with intruders to extract essential details which could be used to fortify the production environment.

[10] employed CAPTCHA-based honey pot. It was able effectively track IP address and location of intruder and blocked. Their underline principle was that human be separated from spyware during authorization just as other applications have successfully leveraged on this technology.

[11] monitored and predicted the likelihood of an attack using network traffic flow. They discovered that authentic user did not experience any performance degradation whereas honey pot was able to detect the distributed denial of service.

This work aim at solving some of the challenges noticed in the previous solutions of bypassing the honey pot by deploying honey net. Since previous identified captcha as effective technology to separate botnet from human-related activities, the work relied and deployed this in the research. Been able to set a threshold beyond which traffic are redirected away from the production server, performance degradation was overcome in this research.

5. Proposed Solution

This work is proposing a model that uses honey net to detect denial of service attempt through buffer overflow. CAPTCHA is used to detect non-human attempts. A review of regular activities is done and a limit is set such that any other traffic flows above that threshold is redirected to the decoy server where interaction is done to obtain relevant details.

The process flow has two main decisions to make: to decide whether the access requests to the server is human or bot and also check the traffic for the envisaged or planned bandwidth threshold. Where it is human, it goes further to check against a preset level of bandwidth. Either of the checks could come first. Where it fails the captcha check, the honey net interacts with such requests.

Similarly, the second check which is done against bandwidth limit is to further guard against intruders from accessing the production environment. Before getting to this level, the network must have been survey and monitor for peak and lowest periods before capping. When the threshold is exceeded honey net picks the requests for necessary interactions.

It should be noted that the honey net is placed behind the firewall so as not allow the intruders from suspecting anything. HIH model is also adopted. Although it consumes more resources, it is preferred so that essential details can be logged for review and eventual hardening of the network.

The flow chart shows that two CAPTCHAS are set up in the honey net to the extent that the first expects no writing and an alternate one that would require reproducing the displayed letter (**Figure 1**), where at least one of these fails, the intruder is redirected to the honey net for further interaction and extraction of essential details with a view to building a robust secure production environment.

(**Figure 2**) shows the proposed model where an intruder is allowed to access

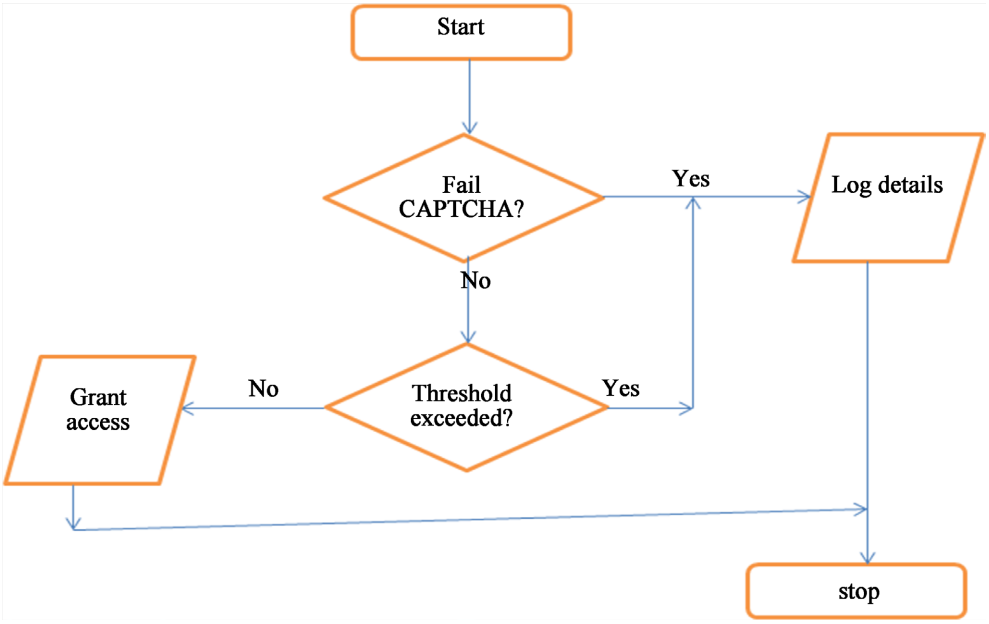

Figure 1. Flow chart of CAPTCHA based Honey net.

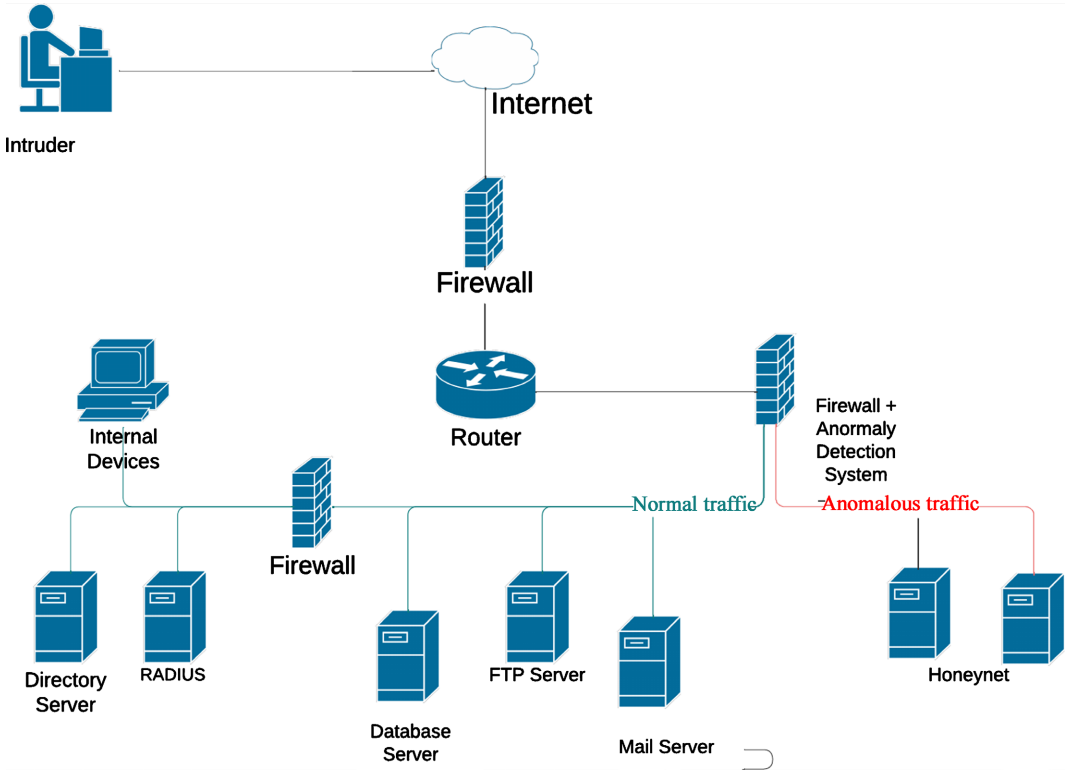

Figure 2. Proposed Honey net Model.

the site via internet. However, firewalls were placed before and behind the router. The first was set in promiscuous mode so as to allow more interaction and hence capture more details to build with. The second filters in order to redirect suspicious activities away from the live server, by way of threshold and captcha, to the honey net.

Table 1. Bandwidth testing against set threshold.

Bandwith (Mbps)	Captcha	Honeynet	Detected
10	Pass	Pass	Yes
30	Pass	Pass	Yes
50	Pass	Pass	Yes
70	Pass	Pass	Yes
90	Pass	Pass	Yes
110	Pass	Fail	Yes
150	Pass	Fail	Yes

6. Result

Simulation method was used to mimic a real life scenario since organizations are not ready to subject their life servers to such experimental handlings. With the experimental threshold of 100 Mbps set as allowed bandwidth he result is given below (Table 1).

The result shows that any bandwidth greater than 100 Mbps were not allowed to go into the network but redirected to honeynet where details are logged.

7. Conclusion

Here, the proposed model has distinctly separated the honey net from the actual production server. The reason for this is such that any possible compromise or bypass does not infiltrate into the network. A compromise at that level can have damning effect on the entire network.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

[1] Dash, N., Panigrahi, D. and Al-Zarouni, M. (2016) Antimicrobial Effect of Honey from the Arabian Gulf Region against Bacterial Isolates from Pus and Wound Swabs. *Advances in Microbiology*, **6**, 745-752. https://doi.org/10.4236/aim.2016.610073

[2] Sokol, P., Pekarčík, P. and Tomáš Bajtoš, T. (2015) Data Collection and Data Analysis in Honeypots and Honey Nets. https://www.semanticscholar.org/

[3] Jones, J. and Romney, G. (2004) Honey Nets: An Educational Resource for IT Security. *SIGITE*'04, Salt Lake City, Utah, 28-30October 2004, 24-28.

[4] Cybersecurity & Infrastructure Security Agency (2019) Protecting against Malicious Code. Security Tip (ST18-004).

[5] Thakur, M., Khilnani, D., Gupta, K., Jain, S. and Agarwal, V. (2013) Detection and Prevention of Botnets and Malware in an Enterprise Network. *International Journal of Wireless and Mobile Computing*, **5**, 144-153.

https://doi.org/10.1504/IJWMC.2012.046776

[6] Nunes, S. and Correia, M. (2010) From Risk Awareness to Security Controls: Benefits of Honeypots to Companies. 2*nd OWASP Ibero-American Web Applications Security Conference* 2010 (*IBWAS*'10), Lisboa, December 2010, 72-83.

[7] Kim, I. and Kim, M. (2012) Agent-Based Honey Net Framework for Protecting Servers in Campus Networks. *IET Information Security*, **6**, 202-211. https://doi.org/10.1049/iet-ifs.2011.0154

[8] Gandhi, U., Priyan Kumar, P., Varatharajan, R., Manogaran, G., Sundarasekar, R. and Kadu, S. (2018) HIoTPOT: Surveillance on IoT Devices against Recent Threats. *Wireless Personal Communications*, **103**, 1179-1194. https://doi.org/10.1007/s11277-018-5307-3

[9] Mohammadzadeh, H., Honarbakhsh, R. and Zakaria, O. (2012) A Survey on Dynamic Honeypots. *International Journal of Information and Electronics Engineering*, **2**, 233-237. https://doi.org/10.7763/IJIEE.2012.V2.89

[10] Souley, B. and Abubakar, H. (2018). A Captcha-Based Intrusion Detection Model. *International Journal of Software Engineering & Applications* (*IJSEA*), **9**, 29-40. https://doi.org/10.5121/ijsea.2018.9103

[11] Rahman, M., Roy, S. and Yousuf, M. (2019) DDoS Mitigation and Intrusion Prevention in Content Delivery Networks Using Distributed Virtual Honeypots. 2019 1*st International Conference on Advances in Science*, *Engineering and Robotics Technology* (*ICASERT*), Dhaka, 3-5 May 2019, 1-6. https://doi.org/10.1109/ICASERT.2019.8934572