Honey-patterns: Recognizing Pattern based Attacks on Websites

Prathamesh P. Churi
Assistant Professor,
Computer Engineering
Department SVKM's
NMIMS Mukesh
Patel School of Technology
Management and Engineering
Mumbai, India

Shreya Bondre
Brand & Project Manager
Marketing Department
Connoisseur Group
Shanghai, China

Neha Gavankar
Information Security
Principal consultant
NTT Data Americas
Halifax , Canada

ABSTRACT
The exact definition of a honeypot is contentious. However, the concept can be defined as "A server that is configured to detect an intruder by mirroring a real production system. It appears as an ordinary server functioning normally, but all the data and transactions are phony. Located either in or outside the firewall, the honeypot is used to learn about an intruder's techniques as well as determine the vulnerabilities of the system" Honeypot is a closely monitored decoy that is employed in networks to study the trail of hackers and to alert network administrators of a possible intrusion. Honey net is a method for detection of abnormal activity in the network. Honey net is an additional layer of security. Even though it is not a panacea for security breaches, it is useful as a tool for network forensics and intrusion detection. Data Capture and Data Control are properties of honeyed, used extensively by the research community to study issues in network security, such as Internet worms, Spam control, DoS attacks, etc. In this paper, we will be focusing on the attack part.

General Terms
Your general terms must be any term which can be used for general classification of the submitted material such as Pattern Recognition, Security, Algorithms et. al.

Keywords
Security, tools, installation problem, framework, designing, projects, Web application

1. INTRODUCTION
We Global communication is getting more and more important day by day but at the same time face the rising threat of computer crimes. Counter measures have been developed to detect or prevent these attacks and most of these measures are based on pre-known facts and attack patterns. As in the military, it is important to know, who your enemy is, what kind of strategy he uses, what tools he utilizes and what he is aiming for, gathering this kind of information is not easy but extremely important. By knowing attack strategies, counter measures can be developed and vulnerabilities can be fixed. To gather as much information as possible is one of the main goals of honeypot [1] .

Honeypot is primarily an instrument for information gathering and learning. The primary purpose of honeypot is not to be ambush for the black hat community, to catch them in action and to press charges against them that lies on silent collection of as much information as possible about their attack patterns, used programs, purpose of attack and black hat community itself. All this information is used to learn more about the black hat proceedings and motives as well as their technical knowledge, abilities and strategies used. This is just primary purpose if honeypot. There are a lot of other possibilities for a honeypot-divert hackers form productive systems for catching a hacker while conducting an attack these are just two possible examples [6].

Honeypots are not the perfect solution for solving or preventing computer crimes. Honeypots are hard to maintain and they need the good knowledge about the operating systems and network security [1] .

In the right hands, honeypot is an effective tool for the information gathering. In the wrong and inexperienced hands, a honeypot can become another infiltrated machine and an instrument for the black hat community.

2. RESEARCH MOTIVATION
All The idea of honeypots began in 1991 with two publications, "The Cuckoos Egg" and "An Evening with Breford". "The Cuckoos Egg" by Clifford Stoll was about his experience catching a computer hacker who was in his corporation searching for secrets. The second publication, "An Evening with Berferd" by Bill Chewick is about a computer hacker's moves through traps that he and his colleagues used to catch the hacker. In both of these writings were the beginnings of what became honeypots [1] .

The first type of honeypot was released in 1997 called the Deceptive Toolkit. The point of this kit was to use deception to attack back. In 1998 the first commercial honeypot came out. This was called Cybercop Sting. In 2002 the honeypot could be shared and used all over the world. Since then honeypot technology has improved greatly and many honeypot users feel that this is only the beginning. In the year, 2005, The Philippine Honeypot Project was started to promote computer safety in the Philippines [2].

A honeypot is a resource whose value is being in attacked and compromised. This means, that a honeypot is expected to get probed, attacked and potentially exploited. Honeypot does not fix anything. They serve as additional, valuable sources of information. A honeypot is a resource, which pretends to be a real target. A honeypot is expected to be attacked or compromised acting as the main goal or the distraction for an attacker and then used to obtain information about the attack and the attacker.

Honeypots can be classified based on their deployment and on their level of involvement. Based on deployment, honeypots may be classified as:

1. Production honeypots [6]
2. Research honeypots [6]

Production honeypots are easy to use and capture only limited information. Production honeypots are placed inside the production network with other production servers by an organization to improve their overall state of security. Normally, production honeypots are low-interaction honeypots, which are easy to deploy. They give less information about the attacks or attackers than research honeypots do.

Research honeypots are run to gather information about the motives and tactics of the Black hat community targeting different networks. These honeypots do not add direct value to a specific organization; but instead, they are used to research the threats organizations face and to learn how to better prevent those threats. Research honeypots are complex to deploy and maintain, capture extensive information, and are used primarily by research, military, or government organizations.

Based on design criteria, honeypots can be classified as

1. Pure honeypots [2]
2. High-interaction honeypots [2]
3. Low-interaction honeypots [2]

Pure honeypots are full-fledged production systems. The activities of the attacker are monitored using a casual tap that has been installed on the honeypot's link to the network. No other software needs to be installed. Even though a pure honeypot is useful, stealthiest of the defense mechanisms can be ensured by a more controlled mechanism [6] .

High-interaction honeypots imitate the activities of the real systems that host a variety of services and, therefore, an attacker may be allowed a lot of services to waste his time. According to recent researches in high interaction honeypot technology, by employing virtual machines, multiple honeypots can be hosted on a single physical machine. Therefore, even if the honeypot is compromised, it can be restored quickly. In general, high interaction honeypots provide more security by being difficult to detect, but they are highly expensive to maintain. If virtual machines are not available, one honeypot must be maintained for each physical computer, which can be exorbitantly expensive. For example: Honeynet.

Low-interaction honeypots simulate only the services frequently requested by attackers. Since they consume relatively few resources, multiple virtual machines can easily be hosted on one physical system. The virtual systems have a shorter response time, and less coding is required, reducing the complexity of the security of the virtual systems. Example: Honeyd.

3. PROBLEM STATEMENT

Please There are different kinds of attacks on the websites or portals, which introduce new vulnerabilities into it. Cross-site scripting (XSS) is a type of computer security vulnerability typically found in web applications that enables malicious attackers to inject client-side script into web pages viewed by other users. An exploited cross-site scripting vulnerability can be used by attackers to bypass access controls such as the same origin policy. Their impact may range from a petty nuisance to a significant security risk, depending on the sensitivity of the data handled by the vulnerable site, and the nature of any security mitigations implemented by the site's owner. SQL injection is a code injection technique that exploits a security vulnerability occurring in the database layer of an application. The vulnerability is present when user input is either incorrectly filtered for a string of literal escape characters embedded in SQL statements or the user input is not strongly typed and thereby unexpectedly executed. It is an instance of a more general class of vulnerabilities that can occur whenever one programming or scripting language is embedded inside another. SQL injection attacks are also known as SQL insertion attacks [1] .

Remote File Inclusion (RFI) is a type of vulnerability most often found on websites, it allows an attacker to include a remote file usually through a script on the web server. The vulnerability occurs due to the use of user supplied input without proper validation. This can lead to minimal as outputting on the contents of the file, but depending on the severity, to list a few it can lead to:

- Code execution on the web server
- Code execution on the client-side such as Javascript which can lead to other attacks such as cross site scripting (XSS).
- Denial of Service (DoS)
- Data Theft/Manipulation

Directory Change Attack (or path traversal) consists in exploiting insufficient security validation / sanitization of user-supplied input file names, so that characters representing "traverse to parent directory" are passed through to the file the APIs. The goal of this attack is to order an application to access a computer file that is not intended to be accessible. This attack exploits the lack of security (the software is acts exactly as it is supposed to) as opposed to exploiting a bug in ‘the code. Directory traversal is also known as the “../” (dot dot slash) attack, directory climbing, and backtracking. Some forms of this attack are also canonicalization attacks [1] .

CURL is a Client URL, a library created by Daniel Stenberg. It is a predominantly command line based tool, which can be used to force parameters into a web request. The URL library was imported to PHP as an optional module and can be useful when International Journal of attempts to gain reconnaissance information or unauthorized access to a designated URL. PHP supports libcurl which currently supports the http, https, ftp, gopher, telnet, dict, file, and ldap protocols. Libcurl also supports HTTPS certificates, HTTP POST, HTTP PUT, FTP uploading (this can also be done with PHP's ftp extension), HTTP form based upload, proxies, cookies, and user + password authentication. CURL can be used in conjunction with PHP scripts for brute force attacks (including SQL injection table brute forcing), reconnaissance attacks, spoofing, and data theft.

3.1 Objective

The system generated after implementing this paper acts as a service provider for Honeypot Security for various websites. It will be used as a framework to implement honeypot which can be used by any organization to test their website applications /portals.

We plan to trace mainly the following characteristics of hackers:

- The browser they use.
- Their IP address from the IP header.
- The files accessed.
- The loopholes they discover.
- Various inputs that are used for various input fields
- Script Injection.
- Time and date when the operation occurred.

4. IMPLEMENTATION

Honeypots are closely monitored decoys that are employed in a network to study the trail of hackers and to alert network administrators of a possible intrusion. Theoretically, a Honeypot should see no traffic because it has no legitimate activity. This means any interaction with a Honeypot is most likely an unauthorized or malicious activity.

The exact implementation of this project will be done using the following steps.

- IP tracing & HTTP packet analysis
- Honeytokens
- Honeypages
- Browser Defect Tracking
- Attacks Tracing [SQL Injection, Cross Side Scripting, etc].

4.1 IP tracing & HTTP packet analysis

We plan to inject certain scripts into the code of the web pages which will act as our Honeypot sniffer. These scripts could be JavaScript or SQL injections. These sniffers will then acquire the information and store it in our database. All unauthorized activities would then be tracked and stored in an administrative website for future analysis.

4.2 HoneyTokens

Honeytokens are fake records that are inserted in the database. These fake records are not expected to be used by normal users. If any of these honey tokens are used, they alert us of the database being compromised. An example of honeytokens is fake username/passwords in the user database.

These users do not exist in the real world, and hence are not expected to be logging in to the application. If the application sees these credentials being used, it immediately recognizes that the user database has been compromised.

4.3 HoneyPages

These are obscure web pages distributed over the web site. They have no legitimate purpose and they are not even linked from any valid active page. So, normal users would never reach these pages.

However, we drop hints about these pages by embedding their url as comments or hidden fields in valid pages. While normal users would never see this, an attacker who analyzes the source code, or a vulnerability scanner that spiders the site would see these and follow the link. When the page is accessed, it points us to the intruder.

4.4 Browser Defect tracking

All browsers have various configurations and accessibility. Hackers usually attack a website through loopholes in the browser. We intend to track the loopholes used by the hacker and change the settings of the website.

4.5 Attacks Tracing- SQL injection

SQL injection is a code injection technique that exploits a security vulnerability occurring in the database layer of an application. The vulnerability is present when user input is either incorrectly filtered for string literal escape characters embedded in SQL statements or user input is not strongly typed and thereby unexpectedly executed. It is an instance of a more general class of vulnerabilities that can occur whenever one programming or scripting language is embedded inside another. SQL injection attacks are also known as SQL insertion attacks.

4.6 Tracking Code [4] :

Tracking code is responsible for Tracking user activities within the network. Tracking code is not visible to the normal end-user. The tracking code is done using PHP MySQL Framework.

5. FEATURES AND WORKING

The Paper will help to make a system which will do the following activities.

- Automatically scan for known attacks.
- Detect SLQ-Injections, (Remote) File-Inclusions, Cross-Site Scripting (XSS), Download attempts for malicious files e.g. with WGET or CURL, Command-Injections, etc [5] .
- Provide an overview mode which allows you to look and scan for new incidents quickly (semiautomatic mode).
- Supports detailed information about all data correlated with every access to the honeypot. This includes but is not limited to HTTP-GET, HTTP, POST and COOKIE data [3] .
- Saves copies of malicious tools in a secured place for later analysis.
- Provides a geographical, IP-based mapping about the attacking sources. The generated map shows the origin of the attacks and offers additional details for each location.
- Generates numerous statistics about all traffic recognized by the system.
- Working through both Android as well as IOS technologies.

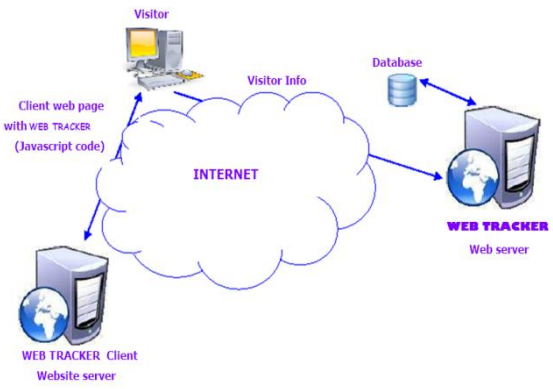

Fig 1: Block Diagram of Honeypot System

Fig 2: Screenshot of Honeypot System – Attack detecting page

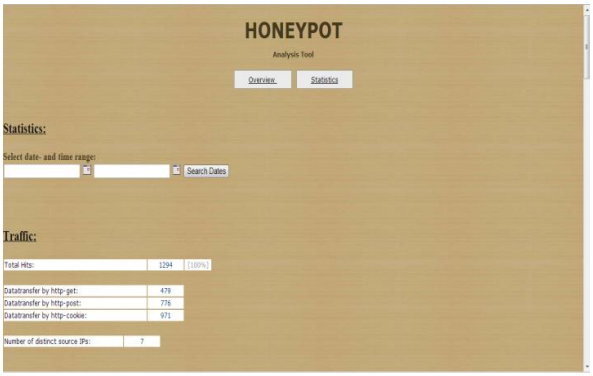

Fig 3: Screenshot of Honeypot System – Analysis page 1

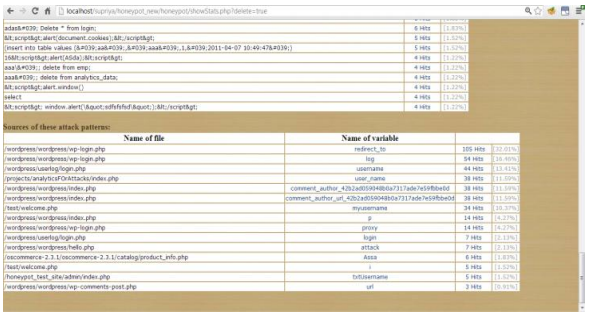

Fig 4: Screenshot of Honeypot System- Analysis page 2

6. CONSEQUENCES

ADVANTAGES

Honeypots only collect attacks or unauthorized activities, dramatically reducing the amount of data they collect. Organizations that may log thousands of alerts a day may only log a hundred alerts wit honeypots. This makes the data collected by honeypots much easier to manage and analyze.

- Honeypots dramatically reduce false alerts, as they only capture unauthorized activity.
- Honeypots can easily identify and capture new attacks never seen before.
- Honeypots require minimal resources, even on the largest of networks, which makes them an extremely cost effective solution.
- Honeypots can capture encrypted attacks.

Limitations

- Honeypots can introduce risk to your environment. As we discussed earlier, different honeypots have different levels of risk. Some introduce very little risk, while others give the attacker entire platforms to launch new attacks. The risk is variable, depending on how one builds and deploys the honeypot.
- Implementing this system on an existing website could cause legality issues. Hence we intend to make dummy websites to demonstrate how our application functions.

7. CONCLUSION

Honeypot is a highly flexible technology that can be applied to a variety of situations. As security tools, honeypots have specific advantages. Mainly, honeypots collect small amounts of data, but most of this is information of high importance. They have the ability to effectively work in resource intensive environment, and conceptually they are very simple devices. Also, they quickly demonstrate their value by detecting and capturing unauthorized activity. A honeypot is just a tool. There are a variety of honeypot options, each having different solutions and value based on the needs of the organizations. Production honeypots help reduce risk in an organization. Research honeypots are different in the sense that they are not used to protect a specific organization. Instead, they are used as a research tool to study and identify the threats in the Internet community. Regardless of what type of honeypot you use, keep in mind the 'level of interaction'. This means that the more your honeypot can do and the more you can learn from it, the riskier it is. You will have to determine what is the best relationship of risk to capabilities that exist for your problems. However, honeypots may be a tool to help contribute to those best practices.

8. ACKNOWLEDGEMENT

I (Prof. Prathamesh Churi) want to sincerely thank to Mr. Rohan Chaudhari for motivating me for doing research task right from the age of 22. His motivation leads me to write many research papers at the youngest age.

9. REFERENCES

[1] Honey net Projects. Website : http://www.honeynet.org/ Last Accessed : 3rd Feb 2017

[2] Spitzner, L. *Honey-pots Tracking Hackers*, MA: Addison-Wesley, 2002

[3] Sams. Teach Yourself My-SQL in 21 Days

[4] Software Engineering by TATA mc-graw hill publications

[5] Learning PHP and MySQL by O'REILLY publications

[6] For tracking Honey-pots and related information , http://www.tracking-hackers.com