ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 5, Issue 4, April 2018

Honeypot Approach for Web Security

[1] Mayank Soni, [2]Aman Prakash, [3]Harshit Mittal, [4]Mohit Tiwari
[1][2][3] B. Tech Student,[4] Assistant Professor,
[1][2][3][4]Department of Computer Science, Bharati Vidyapeeth's College of Engineering, New Delhi, India

Abstract: Today, the Internet has been a boon in the modern era but, along with it the expanding advancements towards Internet and Network security exploitations' have been an issue of great concern. To overcome this hurdle, this paper introduces with an elderly practical and pre-emptive security technology – Honeypot to explicitly lure, ploy and stratagem anyone who attempts to penetrate anyone's computer or network systems. Honeypot's work on deluding the mugger by tempting, thus delaying and distracting the target and ploying to track, record and analyze the attacker's deeds comprehensively. This paper details the concept of honeypots applied to websites, their possible placements, interaction levels and potential threats surrounding it.

Keywords— honeypot, security ,SQL injection.

I. INTRODUCTION

As the boon of Internet expands its branches over the world, the corresponding bane which comes along with it are the dereliction of duty, usually intentional and seldom fortuitous which can cause blunders like Information theft, espionage, sabotaging etc.Various methods have been devised to overcome these conducts, one of which is honeypot.

Talking in computer parlance, Honeypot is a computer security technique whose aim is to perceive, hurdle and countervail any of such attacks.. The formatter will need to create these components, incorporating the applicable criteria that follow.

It would be since the Medieval Period when people have been calling deception as an "Art of War". So, the whole idea of honeypot has been procured from the honeypots that prevail in the real life. The real honeypot development was initially released by Clifford Stoll and Bill Cheswick in 1991. The first implementation was performed by Fred Cohen in 1997 and it was first commercialized into Cybercop Sting in 1998. The first honeypot for windows i.e. Back Officer Friendly was introduced in 1998 immediately followed by a huge gain in the popularity of honeypots and its proliferated growth in applications.

Orthodoxically, an information security has a shielding firewall. Instead of taking the defensive route, honeypots follow an antagonistic path to sustain network security. They unearth the various malicious tactics used by intruders to gain robustness for the security systems. [3]

Traditionally, a Honeypot comprises of imitated data (in various forms. For example, a webpage) that is hoaxed to be a statutory fragment of an original website which is an exhibitionist and is used to lure the attacker to attempt to use malicious ways to access encrypted data while the page is secluded from the information of essential significance and is monitored for gaining access to acquiring knowledge of the compromising and unauthorized ways used to attackers to exploit data resources.[1][4][5]

1.1 TYPES OF HONEYPOTS:

1)Low Interaction Honeypots - Low Interaction Honeypots work in a way by only authorizing a very constrained data access to the database. These types of honeypots simulate just the particular methods which are very often accessed by attackers. This type of honeypots grants the attacker emulated functions with a small and usually insignificant portion of the functionality they would expect from a server, with the intent of detecting sources of unauthorized activity. For example, the HTTP service in this case would only provide the functions required to detect that data exploit/misuse is being attempted by an attacker.

2) High Interaction Honeypots - High Interaction Honeypots allow access to the real-time vulnerable function/ software/data. Here, emulation is minimized and the real server resources are compromised by the attacker. It permits the attacker to interact with the system resources same as they would be accessing from any regular operating system, with the goal of capturing the maximum amount of information on the attacker's techniques. But these types of honeypots majority ascend the danger as attackers can use these real honeypot operating systems to attack and compromise the contents of the real production systems.

II. RECENT DEVELOPMENTS

2.1 HONEYNETS:

A honeynet can be defined as a network initialize while creating intentional and known vulnerabilities for the attacker

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 5, Issue 4, April 2018

to exploit. The major purpose of a honeynet is to lure the attacker into performing an attack, so that the attacker's logs and methods can be examined and the subsequently gained data is utilized to improve the network security. A honeynet compromises of one or more individual honey pots, which are basically just individual computer systems on the Internet purposely set up to lure and "trap" hackers who illegally attempt to gain access to other people's private network resources.

2.2 HONEYFARMS:

Honeyfarms (also known as honeypot farms) are the solution to the implementation of deploying honeypots at a given environment. A honeyfarm is a centralized implementation of honeypots. It is differentiated from a honeynet by the fact that in a honeyfarm, several honeypots can co-exist in the same place without being networked. These are particularly very productive for cloud hosting providers who have several clients on various computer systems and they want to keep all them separate from each other.

3.SQL INJECTION FUNDAMENTALS

3.1 SQL injection attack:

SQL injection attack is a hacking technique in which the hacker uses the addition of SQL statements at the input fields of a web application to get access of the secured and hidden resources. The negligent approach towards the input authentication makes the hacker successful in his intent.[11]

3.2 MAIN CAUSE OF SQL INJECTION ATTACK:

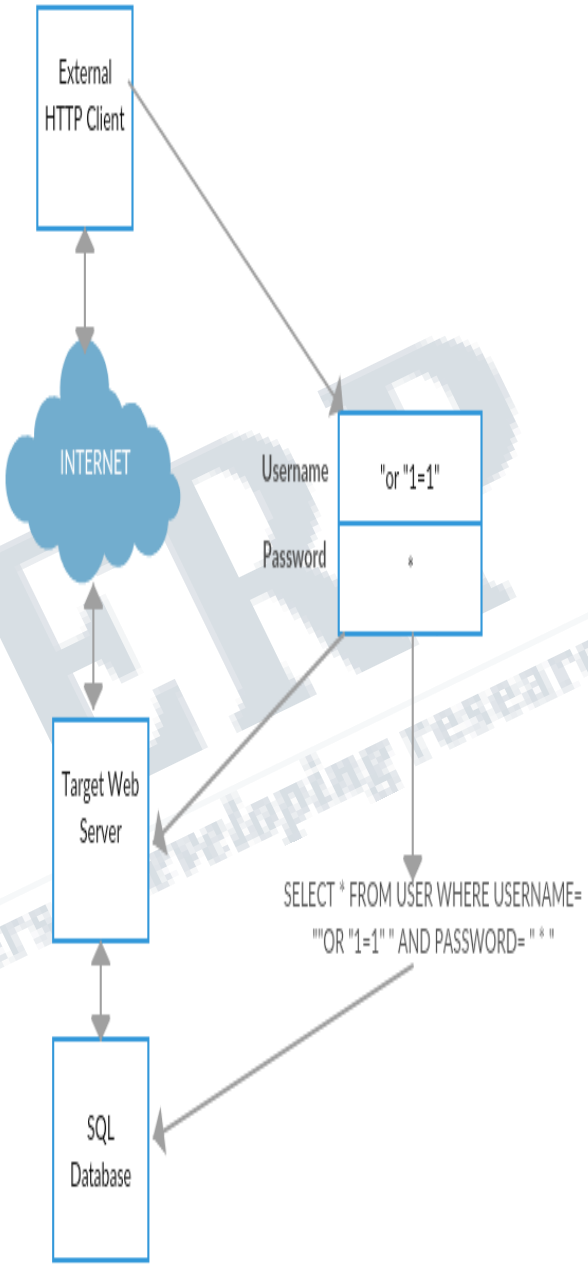
Web applications are generally insecure and susceptible which is the principal cause for any type of attack. In the following section, such susceptibilities which may be implicitly or explicitly present in a web application and which can be exploited by SQL injection are presented and discussed:[11]

Invalidated Input:

This is the frequent most routine susceptibility in performing a SQL injection attack. A web application consists of some definitions which are used in SQL queries. If there is lack of secured validity or authentication of these queries, then these queries can be exploited in SQL Injection attacks.

Dynamic SQL:

When application is designed in such a way that it works on the combination of two or more inputs like name and password using SQL statements using WHERE (clause) then generation of Dynamic SQL takes place. Apparently. Trespasser or invader can exploit the data by making new queries in runtime.



INJECTING SQL IN WEB APPLICATION

Stored procedures:

The database's store a lot of information and statements, the invader may exploit or deplenish the quality of the database by abusing its data.

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 5, Issue 4, April 2018

Multiple Statements:
Mostly DB's uses statements and clauses like UNIONS which makes it more vulnerable to exploitation and threats because of more no. of exploitation methods

Generous Privileges:
Basic SQL clauses and statements like SELECT, DROP, UPDATE, DELETE, INSERT should be made executable for certain objects. There is a great risk of susceptibility of data when there are more number of privileges' in the system. Privileges are defined as the set of directives to concern with DB accessing what object and what operations are associated with it.[11]

4. HONEYPOT AIDES IN DETECTION OF SQL INJECTION

Application Susceptibilities: Honeypot's helps in the recognition of system susceptibilities, helping in preventing attacker's intent.

Attack methodologies: congregated info residing in honeypot helps recognize what methodologies have been used by attacker to abduct data.

Attack's Intent: Different intruder's may have different intents for attacking like manipulation, deletion, violation of the application data or redirecting the client on some other site for injecting Trojans and viruses.

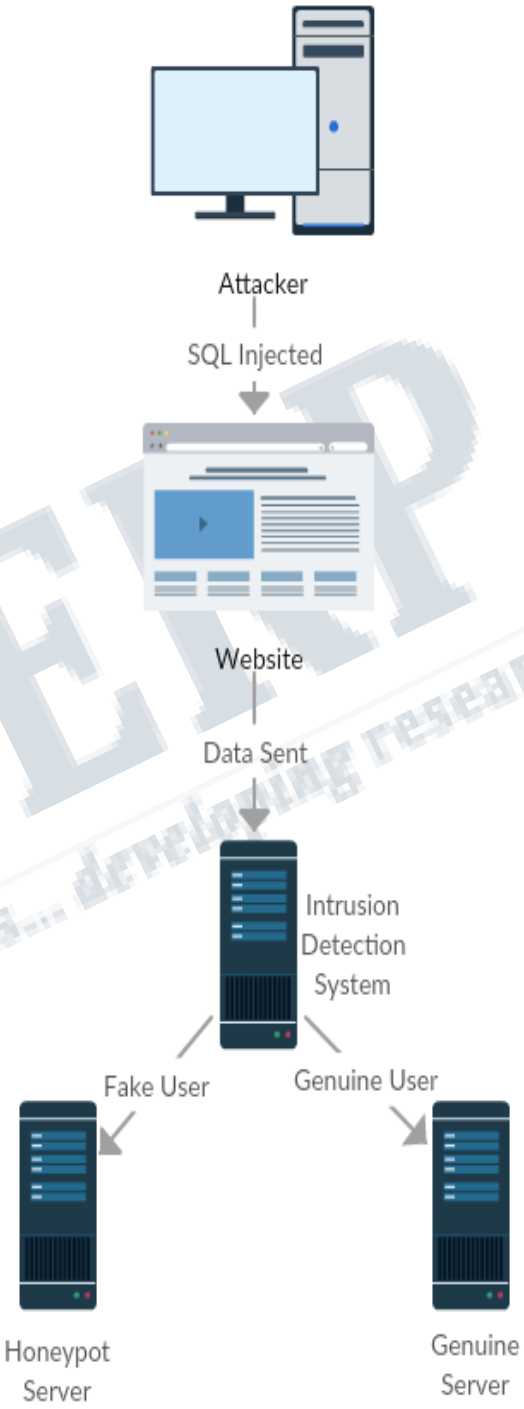
Attack Frequency: The analysis of the congregated data provides us with figure of which attack has been the most frequent.

Source and origin of attack: The unique IP address of the hacker can be pictured as the source or origin of attack.

Attack patterns: The Successful attack can be classified different from the other attack by the help of honeypot logs which keeps logs of all the attacks.

Tools and techniques used in Attacks: The analysis of congregated data helps honeypot to conclude which tools and techniques have been used by the intruders.

Aversion of prospective attacks: All the congregation of data and it's analysis about frequency, source, patterns, tools, techniques of attacks aides as a preventive measure for future attacks.



ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 5, Issue 4, April 2018

5. IMPLEMENTATION DETAILS

IMPLEMENTATION OF HONEYPOTS IN A WEB APPLICATION

We are using a simple website which contains a login form and this login form can provide the administrative access to the precious resources of the company. The login form contains no captcha so we can also test our techniques with the bots.

We are experimenting mainly with the example of the SQL injection technique to gain a false access to the admin page of the website. The SQL injection is implemented by giving a SQL statement in place of user id and user password that will run on our database without your knowledge. By injecting an SQL statement that always hold true like "1=1" will be read by the system like:

SELECT * FROM Users WHERE Name="admin" AND Pass ="abc" or ""1=1"". [6][7]

When the database run this statement, it will always authenticate the hacker as genuine giving administrative access to the hacker. This makes the SQL injection a serious issue and we need a system to detect when a SQL statement is injected in a field of the login form. We can easily check for the SQL injection at both server side and client side. This will prevent the SQL injection attacks. But, if a DDOS attack continuously sends this much queries to the server and the server keeps on denying it, it will consume most of our resources. This will lead to unexpected failures and server breakdowns.

Now in order to stop these attacks, we are going to create a system for the website which will be able to detect the SQL injection in the login form. But, instead of denying the service we are going to redirect the attack to our honeypot page which can easily be corrupted, modified or edited like an original administrator page.

The attacker may think he/she has the administrator rights but we can easily log their details and IP address to alert the system about the attack or to block certain rights. Using this approach, the honeypot constructed will consume the attacker's resources and lure him/her into a trap that will monitor their activities.

This will also reduce the possibilities of DDOS attacks as it does not deny the services. Logging the details and action of the attack helps in study the website security flaws and helps in improving them overtime. As this approach completely transfer the attacker away from the website original resources our website will always remain safe and can function properly.

6.CONCLUSION

In this paper, we described a method which aims to redirect the attacks of an intruder to another server containing fake data when the injection of SQL statement in the login form has been intended. This makes the genuine servers secure and isolated from the attack. The system also gives attacker a space for different malpractices which can be monitored by the security experts. This deludes the attacker into thinking the successful implementation of the attack i.e. SQL statement is successfully injected and the control is given to the attacker. It provides security experts a platform to trick and trap the attacker.

REFERENCES

[1] Honeynet Project. "Know Your Enemy: Honeynets." 24 March 2008. http:/ /old. honeynet. org/papers /honeynet/ An overview of honeypots with a good discussion of their advantages and disadvantages.

[2] Mukherjee, B., L. Heberlein and K. Levitt. "Network Intrusion Detection." IEEE Network May/Jun 1994: 26-41.

[3] honeypot Definition - PC Magazine. pcmag.com. 24 March 2009. http://www.pcmag.com/encyclopedia_term/0,2542,t=honeypot&i=44335,00.asp PC Magazine's encyclopedia entry for honeypotK. Elissa, "Title of paper if known," unpublished.

[4] Talabis, Ryan. "Honeypots 101: A Brief History of Honeypots." 2007. A fairly recent history of honeypots. The article does not provide in-depth information about events and implementations, but is a good starting point for a thorough survey of honeypot history.

[5] Talabis, Ryan. "Honeypots 101: A Honeypot By Any Other Name." 2007. A non-technical introduction to honeypots. M. Young, The Technical Writer's Handbook. Mill Valley, CA: University Science, 1989.

[6] Verwoerd, Theuns and Ray Hunt. "Intrusion detection techniques and approaches." Computer Communications 15 September 2002: 1356-1365.

[7] Wagner, David and Paolo Soto. "Mimicry Attacks on Host-Based Intrusion Detection Systems." Proceedings of the 9th ACM conference on Computer and communications security. ACM, 2002. 255 – 264.

[8] Nathalie Weiler. " Honeypots for Distributed Denial of Service Attacks" Proceedings of the Eleventh IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE'02).

[9] Eric Peter and Todd Schiller " A Practical Guide to Honeypots" https://www.cse.wustl.edu/~jain/cse571-09/ftp/honey/index.html.

[10] Kostas G. Anagnostakis,1 Stelios Sidiroglou,2 Periklis Akritidis,1,3 Michalis Polychronakis,4 Angelos D. Keromytis,4 Evangelos P. Markatos5 "Shadow Honeypots" (IJCNS) International Journal of Computer and Network Security, Vol. 2, No. 9, September 2010

[11] Sayyed Mohammad Sadegh Sajjadi and Bahare Tajalli Pour "Study of SQL Injection Attacks and Countermeasures" International Journal of Computer and Communication Engineering, Vol. 2, No. 5, September 2013