Asian Journal of Electrical Sciences (AJES)
Vol.3.No.1 2015 pp 46-49
available at: www.goniv.com
Paper Received : 08-03-2015
Paper Accepted: 20-03-2015
Paper Reviewed by: 1. R. Venkatakrishnan 2. R. Marimuthu
Editor : Prof. P.Muthukumar

SESSION HIJACKING PREVENTION USING MAGIC COOKIE WITH MAC

[1]P.MANJU BALA, [2]D.SHANMUGAPRIYA
[1]Senior Associate Professor/CSE , [2]Final year/CSE
IFET College of Engineering
[1]pkmanju26@gmail.com , [2]dshanmugapiya11@gmail.com

ABSTRACT:
The Session Hijacking attack compromises the session token by stealing or predicting a valid session token to gain unauthorized access to the Web Server. Session hijacking is nothing but attacker uses a sniffer to capture a valid token session called "Session ID", then he uses the valid token session to gain unauthorized access to the Web Site. Preventing this type of session hijacking using MAGIC Cookie. To prevent session hijacking, a special technique is proposed under which, using magic Cookie with MAC Address to prevent this Session hijacking attack. Magic cookie is not like a normal cookie which gets the MAC address of the machine and it convert the MAC address into some encrypted format and with enables the session cookie.
Keywords: session hijacking,MAC,magic cookie

I . INTRODUCTION

Session hijacking refers to the exploitation of a valid computer session—sometimes also called a session key—to gain unauthorized access to information or services in a computer system. In particular, it is used to refer to the theft of a magic cookie used to authenticate a user to a remote server. It has particular relevance to web developers, as the HTTP cookies used to maintain a session on many web sites can be easily stolen by an attacker using an intermediary computer or with access to the saved cookies on the victim's computer. If an attacker is able to steal the session cookie, he can pretend to be the same user, or hijack the session during its lifetime. An attacker can therefore send requests (or issue transactions) in the name of the user until either the session times out or the user manually terminates the session by clicking some logout button. Airlock mitigates this threat by using a separate session cookie for HTTPS requests, using SSL to transport the token, clearing the token when the session ends, and causing the token to expire after a period of client inactivity.

1.1 TYPES OF SESSION HIJACKING

1.Active Session Hijacking
2.Passive Session Hijacking

1.1.1 ACTIVE SESSION HIJACKING

Active Session Hijacking means that original user has logged in his account or profile and then attacker steal the cookies to hijack the active session and then disconnect the original user from the server. Why we call it active session hijacking because attackers need to interact and need some actions to be performed by the victim to steal the session successfully which can raise the suspicion level.

1.1.2 PASSIVE SESSION HIJACKING:

In passive session hijacking attackers does not hijack active session instead they capture the login credentials while the original user is trying to establish a new connection with the server, and attacker is sitting silently on the same network and recording the login credentials.

2. PROPOSED SYSTEM

Session cookies allow users to be recognized within a website so any page changes or item or data selection you do is remembered from page to page. The most common example of this functionality is the shopping cart feature of any e-commerce site. When you visit one page of a catalog and select some items, the session cookie remembers your selection so your shopping cart will have the items you selected when you are ready to check out. Without session cookies, if you click CHECKOUT, the new page does not recognize your past activities on prior pages and your shopping cart will always be empty.

Without cookies, websites and their servers have no memory. A cookie, like a key, enables swift passage from one place to the next. Without a cookie every time you open a new web page the server where that page is stored will treat you like a completely new visitor. To meet this, have to go for magic cookies. A magic cookie, or just cookie for short, is a token or short packet of data passed between communicating programs, where the data is typically not meaningful to the recipient program. The contents are opaque and not usually interpreted until the recipient passes the cookie data back to the sender. To prevent session hijacking, a special technique is proposed under which, using magic Cookie to prevent this Session hijacking attack. Magic cookie is not like a normal cookie which gets the MAC address of the machine and it convert the MAC address into some encrypted format and with enables the session cookie. Any attacker or intruder may steal the cookie/session as like normally but in this case even when attacker steal the cookie/session he/she not able to access the webpage without user credential.

2.1 Proposed System Advantage:

1.Using a MAGIC cookie will prevent this session hijacking attack by encrypting the Cookie with MAC address so that attacker cannot any type of session hijacking
2.This cookie/Session will be changing frequently so that attacker will not all be guess Session details even this can prevent Session brute forcing attack also.

2.3 SYSTEM ARCHITECTURE:

3. PREVENTION MODULES

1. Cookie Management
2. Magic cookie
3. Idle Timeout

3.1 Module Description:

3.1.1 Cookie Management:-

Where the attacker uses packet sniffing to read network traffic between two parties to steal the session cookie. Many web sites use SSL encryption for login pages to prevent attackers from seeing the password, but do not use encryption for the rest of the site once authenticated. This allows attackers that can read the network traffic to intercept all the data that is submitted to the server or web pages viewed by the client. Since this data includes the session cookie, it allows him to impersonate the victim, even if the password itself is not compromised. Unsecured Wi-Fi hotspots are particularly vulnerable, as anyone sharing the network will generally be able to read most of the web traffic between other nodes and the access point.

3.1.2 Magic cookie:

A magic cookie, or just cookie for short, is a token or short packet of data passed between communicating programs, where the data is typically not meaningful to the frecipient program. The contents are opaque and not usually interpreted until the recipient passes the cookie data back to the sender or perhaps another program at a later time. The cookie is often used like a ticket – to identify a particular event or transaction

3.1.3 Idle Timeout:-

The other type of session attack is session fixation. Here, instead of stealing/hijacking the victim's session, the attacker fixes the user's session ID before the user even logs into the target server (that is, before authentication), thereby eliminating the need to obtain the user's session ID afterwards. Before going into detail of session fixation attacks, we must classify two types of sessions managed on Web servers:
1. Permissive sessions allow the client's browser to propose any session ID, and create a new session with that ID if one does not exist. After that, the server continues to authenticate the client with the given ID.
2.Strict sessions allow only server-side-generated session ID values.
A successful session fixation attack is generally carried out in three phases:
1.Phase I or session set-up: In this phase, the attackers set up a legitimate session with the Web application, and obtain their session ID. However, in some cases the established trap session needs to be maintained (kept

alive) by repeatedly sending requests referencing it, to avoid idle session time-out.
2.Phase II or fixation phase: Here, attackers need to introduce their session ID to the victim's browser, thereby fixing the session.
3.Phase III or entrance phase: Finally, the attacker waits until the victim logs into the Web server, using the previous session ID.

4. IMPLEMENTATION RESULTS

User login to the web page by using the username and password to the system. It is shown in figure below

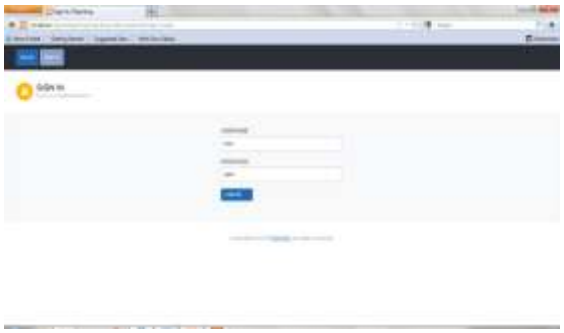

Fig4.1. User login

Once entered the request from the client to server , the system MAC id is automatically gets from the client request system. Then it stored and the server provide the access permission for open profile .

Fig4.2.Admin side information

After the user login to open web page for authentication to view the profile information's.

Fig4.3.login to web page

5. CONCLUSION

The session to be safe by using the magic cookie, it is not hacked by any other unauthorized persons. So the web login information's are protected and avoid the illegal activities from the hacker in a web page. To use the MAC id of the system for generate the magic cookie to client server activities and provide security to the web page.

6. FUTURE ENHANCEMENT

To prevent session hijacking attack against attacker by implementing the software like RSA ID generator that helps communication between server and client machine will be safe attacker not able to perform any sort of attack.

REFERENCES

1. Mark Lin "An Overview of Session Hijacking at the Network and Application Levels," SANS institute 2005.
2. Paul Jess, "Session Hijacking in Windows Networks" Richard Wanner, SANS Institute , 2006.
3. Laxman Vishnoi and Monika Agrwal, "Session hijacking and its countermeasure" 2013.
4. Dinesh Yadav and Anjali Sardana," Enhanced 3-Way Handshake Protocol for Key Exchange in IEEE 802.11i"
5. Bo Li and Shen-juan LV "The Application Research of Cookies in Network Security"
6. Faheem Fayyaz and Hamza Rasheed "Using JPCAP to prevent man-in-the-middle attacks in a local area network environment"
7. Joon S. Park and Ravi Sandhu "Secure Cookies on the Web" George Mason University
8. Hulusi Onder "Session Hijacking Attacks in Wireless Local Area Networks" Monterey, California , March 2004

9. Italo Dacosta, Saurabh Chakradeo, Mustaque Ahamad and Patrick Traynor "One-Time Cookies: Preventing Session Hijacking Attacks with Stateless Authentication Tokens"
10. Huyam AL-Amro and Eyas El-Qawasmeh "Discovering Security Vulnerabilities And Leaks In ASP.NET Websites"
11. Preecha Noiumkar "Top 10 Free Web-Mail Security Test Using Session Hijacking"
12. Sheng Pang, Changjia Chen, Jinkang jia" Session Hijack in the Great Firewall of China"
13. Kevin Lam, David LeBlanc, and Ben Smith (2005). Prevent Session Hijacking [Online]. Available: http://technet.microsoft.com/en-us/magazine/2005.01.sessionhijacking.aspx
14. Definition of Session Hijacking [Online]. Available: http://hitachi-id.com/concepts/session_hijacking.html
15. Session Hijacking [Online]. Available: http://en.wikipedia.org/wiki/Session_hijacking
16. Anim Saxena (Jan 23, 2013) Session Hijacking and Web based Attacks [Online]. Available: https://supportforums.cisco.com/community/netpro/security/web/blog/2013/01/23/session-hicjacking-and-some-web-based-attacks
17. Luke Millanta (Friday 23 August 2013). How to: Understanding session hijacking [Online]. Available: http://www.pcauthority.com.au/Feature/354468, how-to-understanding-session-hijacking.asp