International Journal of Engineering & Technology, 7 (3.15) (2018) 124-129

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

A Preliminary Review on Web Session Hijacking

Nor Naematul Saadah Ismail[1]*, MNM Warip[2], Shamsul J. Elias[3],R Badlishah Ahmad[4]

[1] *Department of Information Technology & Communication, Politeknik Tuanku Syed Sirajuddin, Perlis, Malaysia*
[2]*School of Computer and Communication Engineering University Malaysia Perlis, Malaysia*
[3]*Faculty of Computer and Mathematical Sciences,Universiti Teknologi Mara Kedah, Malaysia*
[4] *Faculty of Informatics and Computing, Universiti Sultan Zainal Abidin (UniSZA), Malaysia*
**Corresponding author E-mail: naematulsaadah@gmail.com,*

Abstract

A fast internet-based technology such as web-driven and mobile application is the most popular usage of people nowadays. With the blink of the eye, the number of users is increasing tremendously every single second. So does the number of web-driven and mobile application. Thus, the number of a hacked website also increases daily. As the end user, we are likely to concern about the security breach that might occur. One of the threats that can happen at application layer users is web session hijacking. Generally, session hijacking is the attack occurs to impersonate the real resource to serve the pretended as the legitimate user. One of the ways for this threat to happen is by stealing the session identification number. This paper presents the session hijacking definition, the taxonomy and the attack that might occur also a various possible method to detect or prevent the attack in a various situation by previous researchers. In addition, this review hopes to bring some awareness to the end users about the threat that is likely to occur while accessing any resources over the Internet using website.

Keywords: *Information Security; Session Hijacking; Web Attack; Web Security*

1. Introduction

Our daily activities nowadays completed by relying on the Internet. From online banking and shopping, e-governance, medical recording, education system, security surveillance as well as manufacturing and productions, we are witnessing a cyber-revolution which increased every year. Figure 1 shows the number of Internet users in the world is nearly four billion, the number of websites is more than one billion and more than sixty thousand websites being hacked daily. The increased amount of information spread over the Internet, the potential for information abused by unlawful entities grows rapidly. The increased amount of information spread over the Internet, the potential for information abused by unlawful entities grows rapidly as shown in Figure 1. Because of the



Fig. 1: Potential of information abused by unlawful entities

number of information, resources and services are provided over the Internet, which they are increasingly being used in our daily life.

The Internet also offers limitless communication, which hackers can use to establish a connection with cohorts or with their victims and network-based attacks disrupt information on the Internet. Besides, the Internet has become extremely popular and is used by people from all different backgrounds. The development of web applications has increased recently. Thus, the impact and amount of security vulnerabilities in web application are also increasing. [1]

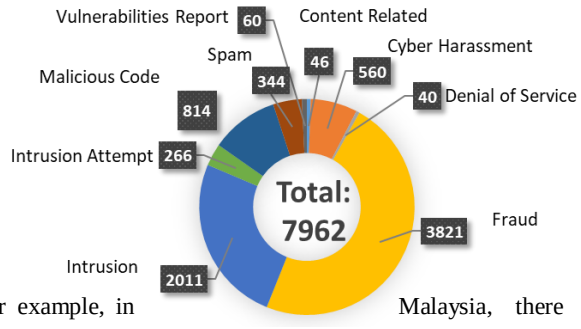


Fig. 2: Cyber incidents occurred in Malaysia for the year 2017

For example, in Malaysia, there are various reports on security for the year 2017 were reviewed in order to have a better understanding about attack's evolution and pattern of the attack that can be related to an intrusion attempt. According to Malaysia CyberSecurity general incident statistic, nine types of the cyber incident have occurred throughout the year as illustrated in Figure 2. Malaysia affected with 33% intrusion

Copyright © 2018 Authors. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

attack, which is a combination of both intrusion attempt attack and intrusion attack. These showed, attacking that masquerading someone's identity to perform a legitimate action still possible to happen.

For the past five years, fraud, intrusion, and spam have been the top three-cyber threats in the country. The threats are getting more serious as the attacks have become more sophisticated. The summarization of the incident shows in Table 1. Session hijacking or website hijacking can be classified under intrusion incident. If the attacker can intercept communication between user and website server by stealing the session identification, the attacker can pretend as the legitimate user. Hence, the attacker can intrude into the legitimate user's session.

Table 1: Reported Incidents based on General Incident Classification Statistics

Reported Incident	**2013**	**2014**	**2015**	**2016**	**2017***	**Total**
Content Related	54	35	33	50	43	126
Cyber Harassment	512	550	442	529	530	1501
Denial of Service	19	29	38	66	40	144
Fraud	**4485**	**4477**	**3257**	**3921**	**3580**	**10758**
Intrusion	**2770**	**1125**	**1714**	**2476**	**1900**	**6090**
Intrusion Attempt	76	1302	303	277	257	837
Malicious Codes	1751	716	567	435	740	1742
Spam	**950**	**3650**	**3539**	**545**	**319**	**4403**
Vulnerabilities Report	19	34	22	35	57	114
	10636	11918	9915	8334	7466	25715

2. Session Hijacking

Session hijacking generally can be described as a process of accessing the session by stealing session ID or cookies. An unauthorized person can impersonate one of the sessions of a victim and takes control over it like a legitimate user. Session hijacking gives great advantages to the attacker as they do not need to waste time to steal the confidential information as the user already being authenticated in the active session. There are three different types of session hijacking attack which are; active, passive and hybrid. (prevention of session hijacking and IP spoofing with sensor node). Session hijacking attacks will exploit the session management process as the attacker wants to bypass the authorization and verification process that performed by a website application [2]. Session hijacking has become one of the major threats to web applications and make them vulnerable to loss of significant data like banking and login credentials. [1]

There are two types of session in the website application. Firstly, the client session which resides on the website browser and the second one is server session. A server session is also known as Hypertext Transfer Protocol(HTTP) session. A browser session is managed by the browser that is used at the user's side. A session begins when the browser is opened and ends when the browser is closed. The HTTP session that managed by the server provides a way to link the browser session to the server session. Since the HTTP does not support session, the website application must devise a way to implement and manage the session. There are three areas within the session management process involved session token creation and deletion, session token transmission and session token verification [2].

The HTTP is designed for stateless transactions, but many Web applications require a session to be maintained between a Web browser and a server creating a stateful environment. Each Web application decides how its session is managed and needs to be able to trust the session identifier. However, it is possible for sessions to be hijacked, and an intruder can gain unauthorized access to the hijacked session. [2]. As mobile and embedded devices are widely used for essential security purposes, such as paying and personal sensing, security protection on the wireless communication between these devices are significantly important. Thus, the conventional cryptographic scheme is not preferable anymore as this traditional scheme is time-consuming. They also do not utilize the characteristic of the wireless environment. [3]

3. Session Hijacking Attack

Session hijacking attack takes over the session by compromising the session token by stealing or predicting a valid session to gain unauthorized access to the Web Server. The messages being passed back and forth while a session is in effect contain a key that provides access to stored information about a user (like, conceivably, authentication status and even credit card number), anyone who intercepts the messages can use that key to impersonate the legitimate user, in effect hijacking that user's identity. So empowered, the abuser can do anything the legitimate user could do. There are several vulnerabilities to the current web application as listed in Table 2 below:

Table 2: Vulnerabilities in website application [4]

No	Vulnerability	Description
1	Session sniffing	An unauthorized way of viewing the session's data during data transmission.
2	HTTP sniffing	Sniffing the HTTP packet of a web application session established between a client and a server.
3	Session prediction	Predicting the session ID of a web session by using a brute force attack.
4	Session fixing	The session ID is fixed by the attacker before the client establishes the session with the server.
5	Session Hijacking	Session Hijacking Session ID is sniffed and the session is hijacked after the client has established the session with the server

4. Taxonomy of Web Attack

The browser at the client-side shows the Internet service content supported by Hypertext Markup Language (HTML). Then the user operates basic steps like reading, create, update and delete that involving the process of entering username and password. In Hypertext Transfer Protocol (HTTP) method is for a server to respond to the client requests. When there is some leakage during the transmission process, all the stated attack might happen including session hijacking. This attack can be performed in various platforms using any type of programming language that is resides on the server-side. This kind of attacks can bring several damages to the users. [5] Describes an overview relation of attacks in Figure 3 below.

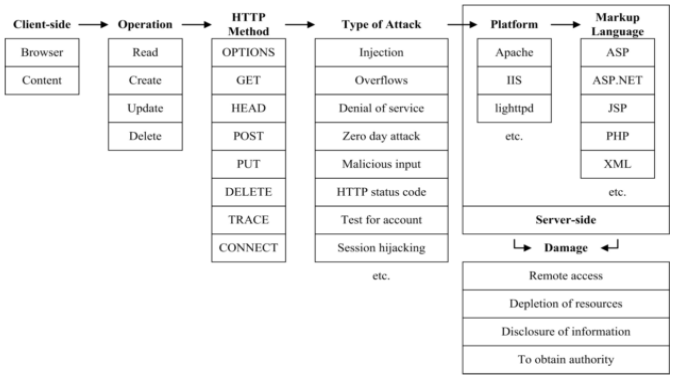


Fig. 3: Relation of attacks

5. Taxonomy of Session Hijacking Attack

Session hijacking attack is a part of website attack as it is involved a browser attack as well. There are varieties of session hijacking attack techniques. In essence, there are three categories of session hijacking attack, active, passive and hybrid attack [6]. The details are as follows:

A. Session Prediction

The purpose of session prediction for attacker is attacker will predict a valid session token. Thus, exploits the first session management area of session token creation. An attacker can be analyzed the website application's cookies if the session token generated in a non-random way.

B. Session Fixation

This threat happens when a vulnerable web application does not assign a new session token while authenticating user. Thus, the use of existing session token is possible. An attacker can have the session token by stealing it without prior notice to the legitimate user.

C. Session Sniffing Attack

In session sniffing attack, the attacker uses a sniffer to capture a valid token session called "Session ID", the attacker uses the valid token session to gain unauthorized access to the Web Server without the knowledge of victim and Web Server.

D. Cross Script Attack (XSS)

The attacker can compromise the session token by using malicious code or programs running at the client-side. If an attacker sends a crafted link to the victim with the malicious JavaScript, when the victim clicks on the link, the JavaScript will run and complete the instructions made by the attacker. Using the same technique it is possible to create a specific JavaScript code that will send the cookie to the attacker.

E. Man-in-the-Middle-Attack (MITM)

The man-in-the-middle attack intercepts a communication between two systems. For example, in HTTP transaction, the target is the Transmission Control Protocol (TCP) connection between client and server. Using different techniques, the attacker splits the original TCP connection into two new connections, one between the client and the attacker and the other between the attacker and the server. Once the TCP connection was intercepted, the attacker acts as a proxy, being able to read, insert and modify the data in the intercepted communication.

There are three types of session hijacking attack, which are; active session hijacking, passive session hijacking, and hybrid session hijacking attack. The attacks can occur at the two layers of Open System Interconnection (OSI) model, which are network and application layer. In network layer, the session can be hijacked by intercepting the packet during transmission and at the application layer, the session hijacking occurred by gaining the access to the application. At the network level, there are two types of attacks; TCP attack and User Datagram Protocol (UDP) attack. The method involved in this attack is likely IP spoofing, blind hijack, and packet sniffing. For application layer, Hypertext Transfer Protocol (HTTP) session hijacking can happen by obtaining the session ID, observation or sniffing and brute force which is trial and error. Another overview of session hijacking attack is simplified in the taxonomy below.

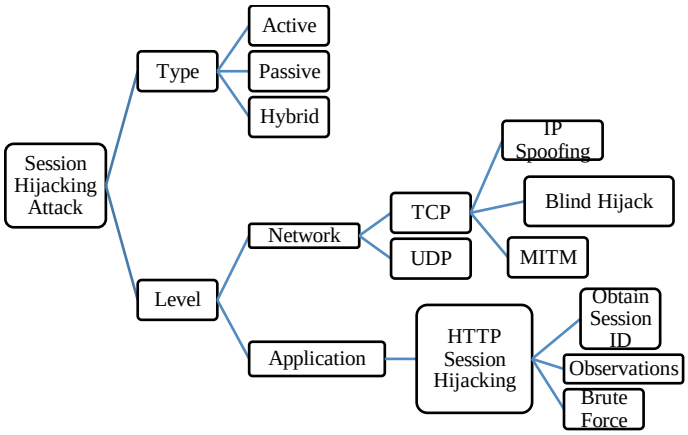


Fig. 4: Taxonomy of session hijacking

6. Session Hijacking Detection

Various researchers have done research on the session hijacking detection. This section will briefly explain the existing works. At the end of the section, a table of summarization will be included.

i. Physical Layer Key Agreement

Researchers in [1] concluded that there are two criteria for using physical layer key agreement schemes that used to generate a shared key between pervasive and resource-constrained devices (Physical properties of channel measurement of two users). There also wireless channel characteristic: time-dependent and location-dependent, thus it is hard for eavesdroppers to get the key. They derived session hijacking attack with an active and passive attack. The purpose of active attacks is for blocking, sabotaging and key manipulation. They designed physical layer key generation scheme with a user can be introduced randomly. Thus, it breaks the security of previous protection scheme.

ii. Enhanced Integrated Central Manager (EICM)

Researchers in [7] said that Denial of Service (DOS) is the most significant attack in Wireless Local Area Network (WLAN). Wired Equivalent Privacy (WEP) and Wi-Fi Protected Access (WPA) security protocols are used to protect wireless network infrastructure against intruders. They proposed an algorithm called EICM to enhance DOS detection and prevention time. This EICM is proposed to overcome network overhead and latency in previous ICM as the algorithm needs to maintain five tables for observing activities in the network. When a client requests for a connection, all five tables need to be processed and response. They obtained results that demonstrate the proposed algorithm outperforms its predecessor in terms of DOS detection and prevention time by decreasing network overhead.

iii. Risk Based Access Control

Authors in [8] discussed cloud computing technology. Most of the resources that are used on an everyday basis are stored online as cloud storage. Therefore, files that have sensitive information thus accessible on valid authentication. The attack can be launch from outside and inside organization. They proposed a model that can calculate the risk. The proposed module is placed between the users and the organization's resources placed in the cloud. A user can access the resources only after being assessed by the module. The steps involved module is simplified as follow; a user will need to be authenticated and once the permission for the files is chosen, the risk values are calculated. There are four steps involved which are; user registration and login, grant of session and event associated, provision of permission and session termination.

Therefore, any possibility of insider threat such as buffer overflow and session hijacking attack is tackled before it oc-

curs. The model allows a maximum risk of 70% which means that even in the worst case scenario, 30% of the data is still secure.

iv. Pseudo Random Alteration of Sequence Numbers (PRAS)

In [9], mentioned that existing methodologies are not comprehensive. Some serious attack in WLAN: MITM, DOS, present of rogue AP, wormhole attack and session hijacking. The attacker performs DOS attack by sending ping flood. Session hijacking is not only happening to a website but any protocol in which state is maintained using a key passed between two parties is vulnerable. Thus they proposed mobile agent based mechanism. This architecture addresses methodologies of cookie encryption, token generation, and session key generation. The idea about the mobile agent is Mobile Agent Server also a Dynamic Host Configuration Protocol (DHCP) server that will allocate IP address. The general overview of the model is in Figure 5.

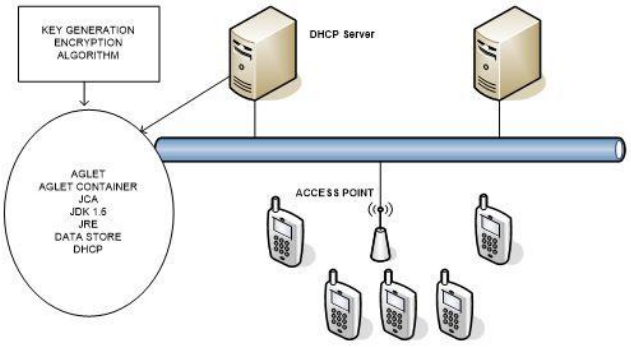


Fig. 5: Mobile agent model

v. One Way Hash Chain

Typically session hijacking targets HTTP based communication is used to keep a user/browser status. Before a session is established, many websites use the HTTPS secure connection to replace login credentials with session identifiers such as session cookies. A threat is more prominent when the website usually offers the "remember me" option. Session cookies typically stored in clients' machine and usually transmitted over an unsecured wireless network. The authors proposed a cache supported hybrid two-dimensional one-way hash construction to handle social networks' user sessions authentication in collaborative applications efficiently.[10]

vi. Real-time Detection

Authors in [11] proposed a real-time detection of Medium Access Control (MAC) layer attacks in wireless networks is proposed. There can be different kinds of Denial of Service (DoS) attacks observed at the MAC layer such as misbehavior and selfish attacks. The idea of the proposed method is about; the malicious nodes manipulate the MAC protocol parameters such as back-off time, network allocation's vector value and short interframe space, or flood the network with a huge volume of dummy packets. Attacker nodes will capture entire network bandwidth causing legitimate nodes do not communicate with other nodes, consequently it will decrease the throughput of the nodes. This real-time detection provides minimal delay and proved using simulation.

vii. Self-Configuring Repeatable Hash Chains (SCRHC)

Researchers in [12] present a security authentication protocol for mitigating the risk of hijacking social networking and collaboration sites. The protocol is based on the recognition that users of social and collaborative media connect to their websites using a variety of platforms and connection speeds. To appeal to both mobile devices such as smartphones or tablets using Wi-Fi connections and high-end workstations such as PC's using high-speed connections, a novel Self-Configuring Repeatable Hash Chains (SCRHC) protocol was developed to prevent the hijacking of session cookies. The protocol supports three different levels of caching, giving the user the ability to forfeit storage space for increased performance and reduced workload. Performance evaluation tests are presented to show the effectiveness and flexibility of the SCRHC protocol. The hash chain is proven to offer a better performance and enough flexibility and can run on various platforms. Dynamic caching also reduced the workloads while requiring trivial amounts of storage.

viii. Runtime Detection

Researchers in [13] work on mobile based computing which are vastly used nowadays. The concept of honey-pot in the detection of these two authentication hijacking problems is used in order to counter the detection of two fatal malware attacks which are intent based hijacking and authenticated session hijacking.

Various applications have been tested and their interaction with the honey-pot maintained by a device or an emulator. The Benign app has been designed as a honey framed app. The approach, Anti-Hijack, which has provided the detection accuracy as high as 96%. This has been highly accurate to detect the unwanted interaction between hijacking malware and designed benign app. The approach has been tested on a strong data-set of Android apps for experiment and identifying vulnerable points.

ix. Pseudo Random Alteration of Sequence Numbers (PRAS)

In [14] introduced a method for defending session hijacking attack in Mobile AdHoc Network (MANET) called Pseudo Random Alteration of Sequence Numbers (PRAS). Based on the comprehensive evaluation based on simulations, it showed that PRAS could successfully defend session hijacking under a diverse mobile multi-hop network environment with low communication, computation, and memory overhead. A major advantage of PRAS is lower communication and computational complexity associated with avoiding the attack. These facts proven with the comprehensive evaluation based on simulations. From the analysis, the PRAS algorithm is prone to session hijacking attack as the time evolves since the attacker may be possible to identify the logical function from the analysis of a large number of packets. This problem can easily overcome by changing the logical functions used for modifying the sequence number pseudo-randomly that also known only for the communicating nodes.

x. Sum Chain

In [15], the authors focused on the Multiple Path Transmission Control Protocol(MPTCP) which session hijacking is considered as the possible attack. Instead of normal hash chain function, sum chain was introduced to overcome the complex process of computation the hash chain value. Sum chain is a simple modular addition. They categorized the sum chain into two; random prime and fixed prime. From the result, the execution time using both type of sum chain decreased. To add more security to the random prime number, it is better to be encrypted.

xi. Automatic Detection Algorithm

Algorithm that proposed by authors in [16] utilizes probabilistic approach since the users' input on the web application hopes to be normal but sometimes it has turned to malicious. The algorithm can detect the cross site scripting (XSS) by mathematical formula. There is one problem with this algorithm, which is instability of the algorithm.

xii. Java Script Validation

The authors in [17] performed an assessment to detect the security strength of a web application using java script validation for the user input. Then java script signature used to identify the validity of java script. A unique token will be assigned for client-server request during communication. Lastly sanitization method is used to clean-up HTML text.

xiii. Headless Browser for Hybrid XSS Detection(HXD)

The author in [18] proposed a hybrid XSS detection using headless browser. They approached this technique using string analysis and dynamic browser rendering. HXD does not need to crawl for Uniform Resource Locator(URL) inputs. HXD used PhantomJS, a headless browser to execute javascript. It also used a heuristic to decide analysis approach; static or dynamic. HXD brings three advantages; can be applied in static and dynamic, no source code needs and automatically extract targeted URLs form web logs and filtered the duplications.

xiv. Automated Detection

Authors in [19] proposed a technique; Amberate that checked various security holes in web applications like SQL injection and Cross-Site Scripting (XSS). Amberate provides a set of Application Programming Interfaces (APIs). This technique brings two advantages; it can check any kind of vulnerabilities even the test operator is not familiar and the environment is not hard to set up. However, this technique only tested in simulation software and did not tested on the real web application and real Internet traffics. This is because the confidentiality of real network. It also cannot detect if there is any false positive event occurred.

7. Comparison of related works

Comparison of related work for session hijacking detection technique is summarized as in Table 3 below:

Table 3: Comparison of related works

No	Authors/Technique	Contribution	Advantage	Disadvantage
1	(Hancke, 2017) Key Generation Agreement	Designed physical layer key generation scheme	Key generated randomly	Issues with background noise or manufacturing errors because rely on the device
2	(Manivannan & Sathiyamoorthy, 2014) Encrypted Session ID	Strong and encrypted session ID	Session ID encrypted using secret key sharing	Tested on 3 different sizes of session ID only.
3	(Abdallah, Abd Razak, & Yahaya, 2015) Enhanced Integrated Central Manager (EICM)	Algorithm; ICM enhanced	to overcome network overhead and latency	
4	Hi, Namitha, Gopalan, Sanjay, & Chandrashekaran, 2015) Risk Base Access Control	proposed a model to calculate the risk in the cloud environment	allows a maximum risk of 70% which means that even in the worst case scenario, 30% of the data is still secure	
5	(Jha & Ali, 2015) Mobile agent based architecture to prevent session hijacking attacks in IEEE 802.11 WLAN.	mobile agent based mechanism	Session hijacking detection in the mobile device	
6	(Alabrah & Bassiouni, 2014). hybrid cache-supported one-way hash chains	cache supported hybrid two-dimensional one-way hash construction	Concern in social networking site session hijacking	Computational overhead
7	(Dasari, 2017) Real time detection	Real time MAC layer detection	provides minimal delay	
8	(Cashion & Bassiouni, 2011) Protocol for Mitigating the Risk of Hijacking Social Networking Site	Hash-Chain protocol is introduced.	Support 3 level of cache	Cache memory dependant.
9	(V. G. Shankar and G. Somani, 2016) Anti-Hijack: Runtime Detection of Malware Initiated Hijacking in Android	Honeypot	The accuracy of detection 96% using simulation	
10	(N. Nishanth, J. Zareena, Dr. S. Suresh Babu, 2013) Pseudo Random Alteration of Sequence Numbers (PRAS)	New method; Pseudo Random Alteration of Sequence Numbers (PRAS).	lower communication and computational complexity associated with avoiding the attack	
11	(V. Akash Krishnan, P.P Amritha and M. Sethumadhavan, 2017) Sum Chain Based Approach against Session Hijacking in MPTCP	Sum hash chain	Computation execution time reduced	Prime number did not encrypted
12	(D. Koizumi, T. Matsuda and M. Sonoda, 2012) Automatic Detection Algorithm with Non-Stationary Bernoulli Distribution	Automatic detection algorithm using Bernoulli Distribution	Using probabilistic approach	Instability of algorithm
13	(A. Shrivastava, S. Choudhary, and A. Kumar, 2016) Java Script Validation	Java script validation		Not suitable to all platform which is not using HTML as the language
14	(H. Choi and Y. Kim, 2017) Headless Browser for HXD	Proposed a hybrid XSS detection (HXD)	String analysis and browser rendering. Static and dynamic, no	Experiment tested using only one headless browser.

No	Authors/Technique	Contribution	Advantage	Disadvantage
			source code needs and automatically detect the URLs from web logs.	
15	(Y. Takamatsu, Y. Kosuga, and K. Kono, 2012) Automated Detection of Session Management Vulnerabilities in Web Applications	Automatically detect the vulnerabilities of session management in web applications.	Executing real attacks in simulated environment	Cannot check on false positives event.

8. Conclusion

Many researchers conducted research before. Some of the expert said that it is hard to detect the session hijacking. If detection is hard to perform, it is better to prevent from the attack to happen. That is why most of the current research is concern about the web application vulnerabilities. If the vulnerabilities possibly to prevent, the threat will be less to occur. However, there are several element to secure the web application at the application layer. Software and hardware element can be tight together so that the application will be more secure. For example, the developer needs to concern on the application. How the server manages the session requested by the users.

For example, among the previous related works that mentioned previously, several disadvantages like cache memory dependent most likely require the high capabilities of random access memory (ram) to store the cache or the cache level of the processor architecture. This situation will be varies based on the architecture itself. Another concern is about the computational overhead. This overhead will be varies based the speed of the processor itself. Thus, the related concern is about the hardware issue. The detection result will be varies and will depend on the programming that will be run as well. If the website server manages in proper way and if it is, secure, the website session hijacking can be prevented. The website servers' security issues will rely upon with the website developer as well. Despite the hardware used at the application level that resides on the same network with the web server. Maybe, a firewall can be placed together with the web server as mentioned by one of the previous researcher.

To summarize session hijacking, it is a small part in web application threat as from the well-known attack can make it to occur. A small threat can bring a huge damage to unauthorized information disclosure because it can be either active attack or passive attack. Usually the victims did not realized that they have victimized.

For future works, it is suggests to do more study in website development security on how the web server handles the session granting access and does the certificates authorities should take part in ensuring the session handling access grant permission.

Acknowledgment

This paper published with research collaboration between Politeknik Tuanku Syed Sirajuddin Perlis, Universiti Malaysia Perlis, Universiti Teknologi Mara Kedah and Universiti Sultan Zainal Abidin, Terengganu in Malaysia.

References

[1] D. R. S. & D. S. T. Vineeta Jain, "Session Hijacking : Threat Analysis and Countermeasures," no. May, 2015.

[2] S. Wedman, A. Tetmeyer, and H. Saiedian, "An Analytical Study of Web Application Session Management Mechanisms and HTTP Session Hijacking Attacks," Inf. Secur. J. A Glob. Perspect., vol. 22, no. 2, pp. 55–67, 2013.

[3] Q. H. ;Gerhard P. Hancke, "A Session Hijacking Attack on Physical Layer Key Generation Agreement," 2017 IEEE Int. Conf. Ind. Technol., pp. 1418–1423, 2017.

[4] S. S. Manivannan and E. Sathiyamoorthy, "A prevention model for session hijack attacks in wireless networks using strong and encrypted session ID," Cybern. Inf. Technol., vol. 14, no. 3, pp. 46–60, 2014.

[5] J. Y. Lai, J. S. Wu, S. J. Chen, C. H. Wu, and C. H. Yang, "Designing a taxonomy of web attacks," Proc. - 2008 Int. Conf. Converg. Hybrid Inf. Technol. ICHIT 2008, pp. 278–282, 2008.

A. K. Bharti and M. Chaudhary, "Prevention of Session Hijacking and IP Spoofing With Sensor Nodes and Cryptographic Approach," Int. J. Comput. Appl. (0975 – 8887), vol. 76, no. 9, pp. 22–28, 2013.

B. E. Abdallah, S. Abd Razak, and C. Yahaya, "Detection and prevention of denial of service attacks (DOS) in WLANs infrastructure," J. Theor. Appl. Inf. Technol., vol. 71, no. 3, pp. 417–423, 2015.

[6] L. Hi, S. Namitha, S. Gopalan, H. N. Sanjay, and K. Chandashekaran, "Risk Based Access Control In Cloud Computing," pp. 1502–1505, 2015.

[7] S. K. Chung et al., "Pseudo Random Alteration of Sequence Numbers (PRAS): A novel method for defending sessiion hijacking attack in mobile adhoc network," 2016 IST-Africa Conf. IST-Africa 2016, vol. 2, no. 2, pp. 1–10, 2015.

A. Alabrah and M. Bassiouni, "Preventing session hijacking in collaborative applications with hybrid cache-supported one-way hash chains," no. CollaborateCom, pp. 27–34, 2014.

[8] M. Dasari, "Real Time Detection of MAC Layer DoS Attacks in," 14th IEEE Annu. Consum. Commun.Netw. Conf. II., pp. 946–951, 2017.

[9] J. Cashion and M. Bassiouni, "Protocol for Mitigating the Risk of Hijacking Social Networking Sites," Proc. 7th Int. Conf. Collab. Comput. Networking, Appl. Work., pp. 324–331, 2011.

[10] V. G. Shankar and G. Somani, "Anti-Hijack: Runtime Detection of Malware Initiated Hijacking in Android," in Physics Procedia, 2016, vol. 78, pp. 587–594.

[11] H. A. Mangut, A. Al-Nemrat, C. Benzaïd, and A. R. H. Tawil, "ARP cache poisoning mitigation and forensics investigation," Proc. - 14th IEEE Int. Conf. Trust. Secur. Priv. Comput. Commun. Trust. 2015,vol. 1, pp. 1392–1397, 2015.

[12] V. Akash Krishnan, P. P. Amritha, and M. Sethumadhavan, "Sum Chain Based Approach against Session Hijacking in MPTCP," Procedia Comput. Sci., vol. 115, pp. 794–803, 2017.

[13] D. Koizumi, T. Matsuda, and M. Sonoda, "On the automatic detection algorithm of Cross Site Scripting (XSS) with the non-stationary Bernoulli distribution," 5th Int. Conf. Commun. Comput. Appl., no. October, pp. 131–135, 2012.

A. Shrivastava, S. Choudhary, and A. Kumar, "XSS Vulnerability Assessment and Prevention in Web Application," 2016 2nd Int. Conf. Next Gener. Comput. Technol., no. October, pp. 850–853, 2016.

[14] H. Choi and Y. Kim, "HXD : Hybrid XSS Detection by using a Headless Browser.", 2017.

[15] Y. Takamatsu, Y. Kosuga, and K. Kono, "Automated detection of session management vulnerabilities in web applications," 2012 10th Annu. Int. Conf. Privacy, Secur. Trust. PST 2012, pp. 112–119, 2012.