

# Man in the Middle Attack Prevention using Token Generation Technique

V.Gokula Krishnan, B.Ajith Kumar, R.Mohan,V.Mohanasunder

Dept.of Computer Science and Engineering  
Panimalar Institute of Technology  
(Anna university)

gokul\_kris143@yahoo.com, mhnsunder@gmail.com,ajithpgb1998@gmail.com,abeethmohan@gmail.com

**Abstract** – Man-in-the-middle attack is a type of cyberattack where a malicious actor inserts him/herself into a conversation between two parties, impersonates both parties and gains access to information that the two parties were trying to send to each other. A man-in-the-middle attack allows a malicious actor to intercept, send and receive data meant for someone else, or not meant to be sent at all, without either outside party knowing until it is too late. When we create a new session, the session is can be polluted by a third party The use of token generation both in the server and in the web page of the client is used for matching the session if the web page token pattern dose not match the server token pattern then it is identified to be a attacker .This method prevents a fake user to accessing the data in the database. The goal of the project is to deduct and prevent the attackers who uses duplicate session and tries to access the web site. A random number in both server and website is generated by this way session attack can be prevented.

**Keywords** – component, formatting, style, styling, insert (key words)

## I. INTRODUCTION

A man in the middle (MITM) attack is a general term for when a perpetrator positions himself in a conversation between a user and an application—either to eavesdrop or to impersonate one of the parties, making it appear as if a normal exchange of information is underway. The goal of an attack is to steal personal information, such as login credentials, account details and credit card numbers. Targets are typically the users of financial applications, SaaS businesses, e-commerce sites and other websites where logging in is required. Information obtained during an attack could be used for many purposes, including identity theft, unapproved fund transfers or an illicit password change. Additionally, it can be used to gain a foothold inside a secured perimeter during the infiltration stage of an advanced persistent threat (APT) assault. Broadly speaking, a MITM attack is the equivalent of a mailman opening your bank statement, writing down your account details and then resealing the envelope and delivering it to your door.

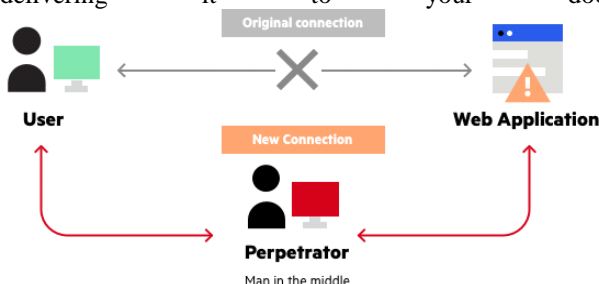


Fig . 1.1 Man in the Middle Attack Overview

## II. MITM ATTACK PROGRESSION

Successful MITM execution has two distinct phases: interception and decryption.

### 1. Interception

The first step intercepts user traffic through the attacker’s network before it reaches its intended destination. The most common (and simplest) way of doing this is a passive attack in which an attacker makes free, malicious WiFi hotspots available to the public. Typically named in a way that corresponds to their location, they aren’t password protected. Once a victim connects to such a hotspot, the attacker gains full visibility to any online data exchange.

Attackers wishing to take a more active approach to interception may launch one of the following attacks:

IP spoofing involves an attacker disguising himself as an application by altering packet headers in an IP address. As a result, users attempting to access a URL connected to the application are sent to the attacker’s website.

ARP spoofing is the process of linking an attacker’s MAC address with the IP address of a legitimate user on a local area network using fake ARP messages. As a result, data sent by the user to the host IP address is instead transmitted to the attacker.

DNS spoofing, also known as DNS cache poisoning, involves infiltrating a DNS server and altering a website’s address record. As a result, users attempting to access the site are sent by the altered DNS record to the attacker’s site.

## 2. Decryption

After interception, any two-way SSL traffic needs to be decrypted without alerting the user or application. A number of methods exist to achieve this:

HTTPS spoofing sends a phony certificate to the victim's browser once the initial connection request to a secure site is made. It holds a digital thumbprint associated with the compromised application, which the browser verifies according to an existing list of trusted sites. The attacker is then able to access any data entered by the victim before it's passed to the application.

SSL BEAST (browser exploit against SSL/TLS) targets a TLS version 1.0 vulnerability in SSL. Here, the victim's computer is infected with malicious JavaScript that intercepts encrypted cookies sent by a web application. Then the app's cipher block chaining (CBC) is compromised so as to decrypt its cookies and authentication tokens.

SSL hijacking occurs when an attacker passes forged authentication keys to both the user and application during a TCP handshake. This sets up what appears to be a secure connection when, in fact, the man in the middle controls the entire session.

SSL stripping downgrades a HTTPS connection to HTTP by intercepting the TLS authentication sent from the application to the user. The attacker sends an unencrypted version of the application's site to the user while maintaining the secured session with the application. Meanwhile, the user's entire session is visible to the attacker.

Random numbers are useful for a variety of purposes, such as generating data encryption keys, simulating and modeling complex phenomena and for selecting random samples from larger data sets. They have also been used aesthetically, for example in literature and music, and are of course ever popular for games and gambling. When discussing single numbers, a random number is one that is drawn from a set of possible values, each of which is equally probable, i.e., a uniform distribution. When discussing a sequence of random numbers, each number drawn must be statistically independent of the others.

With the advent of computers, programmers recognized the need for a means of introducing randomness into a computer program. However, surprising as it may seem, it is difficult to get a computer to do something by chance. A computer follows its instructions blindly and is therefore completely predictable. (A computer that doesn't follow its instructions in this manner is broken.) There are two main approaches to generating random numbers using a computer: Pseudo-Random Number Generators (PRNGs) and True Random Number Generators (TRNGs). The approaches have quite different characteristics and each has its pros and cons.

## III. PSEUDO-RANDOM NUMBER GENERATORS (PRNGS)

PRNGs are efficient, meaning they can produce many numbers in a short time, and deterministic, meaning that a given sequence of numbers can be reproduced at a later date if the starting point in the sequence is known. Efficiency is a nice characteristic if your application needs many numbers, and determinism is handy if you need to replay the same sequence of numbers again at a later stage. PRNGs are typically also periodic, which means that the sequence will eventually repeat itself. While periodicity is hardly ever a desirable characteristic, modern PRNGs have a period that is so long that it can be ignored for most practical purposes.

These characteristics make PRNGs suitable for applications where many numbers are required and where it is useful that the same sequence can be replayed easily. Popular examples of such applications are simulation and modeling applications. PRNGs are not suitable for applications where it is important that the numbers are really unpredictable, such as data encryption and gambling.

### True Random Number Generators (TRNGS)

TRNGs extract randomness from physical phenomena and introduce it into a computer. You can imagine this as a die connected to a computer, but typically people use a physical phenomenon that is easier to connect to a computer than a die is. The characteristics of TRNGs are quite different from PRNGs. First, TRNGs are generally rather inefficient compared to PRNGs, taking considerably longer time to produce numbers. They are also nondeterministic, meaning that a given sequence of numbers cannot be reproduced, although the same sequence may of course occur several times by chance. TRNGs have no period.

## IV. SYSTEM DESIGN

The primary goal of the attacker is to eavesdrop on your private conversations, they can also target all the information inside your devices. The goal of the project is to deduct and prevent the attackers who uses duplicate session and tries to access the web site. A random number in both server and website is generated by this way session attack can be prevented

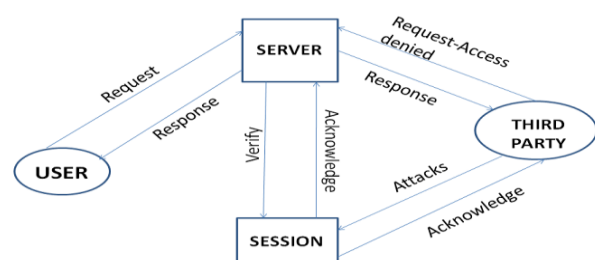


Fig. 4.1 .System Architecture Diagram.

The above diagram describes the complete architecture of the system.

User- User logins and access the data.

Server- Server generates a random pattern on both server and web browser.

Session- session stores the details of the user.

Third party- Attacks the session to get the duplicate session data to access the data illegally.

#### List of Modules

- User login
- Attacks the session
- Server generates random pattern

## V. MODULES

The proposed system contains the following modules such as User login, Attacks the session, Server generates random pattern.

### 5.5.1 User Login

User will have their user id and password. If the user doesn't have id they have to register first before login to access the data. To register user has to enter the details like user name, user password, mail id, user id. The server checks if the details are already exist in the database if it exist or if the id is not unique it displays a warning saying the data already exist, if the data doesn't exist the new registration will be added to the database and moves to login page. In login page, the user enters the id and password if it matches with the database stored data it logs in and moves to the home page. If the user id or the password doesn't matches with the database it displays a warning saying you to first register and then login to open their home page. User will have their own login id and password to login to the website and access the data and to manipulate the data.

If the user doesn't have a login id they have to register. To register fill the required details and create a login id and password to login to the system.

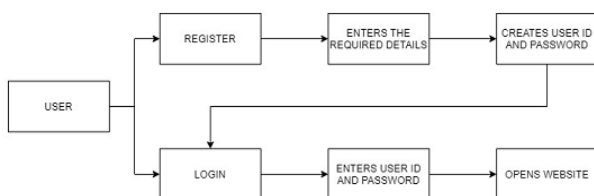


Fig 5.5.1 User login.

### 5.5.2 Attacks the session

The attack involves sniffing data packets to steal session cookies and hijack a user's session. These cookies can contain unencrypted login information, even if the site was secure.

The Attackers attacks the session and gets the duplicate user's session id and password using the id and password the attacker tries to enter the system and access the data.



Fig 5.5.2 Attacks the session.

### 5.5.3 Server generates random pattern

A Random Number Generator is a technology designed to generate a sequence that does not have any pattern, therefore appear to be random. ... However, it is extremely difficult to come up with a completely random seed value, since most such operations only provide seeds with a small range of values.

The server generates tokens on both server side and browser side. The generated token is displayed on browser.

The user enters the token displayed on the browser.

Server then compares the entered token with the generated token if the tokens are same then the access is granted to the user or else the access is denied.

## VI. IMPLEMENTATION

### 6.1 Techniques used in The Project

MITM attacks can be executed in a number of different ways that exploit communications between other parties. Whether by passive or active means, an MITM attack finds a way between a user and an entity and attempts to conceal the breach and information theft. Below are common ways Man-in-the-Middle Attacks manipulate communication systems.

#### Types of Man-IN-The-Middle Attacks

Email Hijacking – attackers gain access to a user's email account and watch transactions to and from the account. When the time is right, for instance the user is exchanging funds with another party, the attacker takes advantage of the situation by attempting to intercept the funds by spoofing one or all members of the conversation.

Wi-Fi Eavesdropping – a passive way to deploy MITM attacks, Wi-Fi eavesdropping involves cyber hackers setting up public Wi-Fi connections, typically with an unsuspecting name, and gain access to their victims as soon as they connect to the malicious Wi-Fi.

Session Hijacking – session hijacking is when an attacker gains access to an online session via a stolen session key or stolen browser cookies.

DNS Spoofing – an attacker engages in DNS spoofing by altering a website's address record within a DNS (domain name server) server. A victim unknowingly visits the fake site and the attacker will attempt to steal their information.

IP Spoofing – similar to DNS spoofing, IP Spoofing sees an attacker attempt to divert traffic to a fraudulent website with malicious intent. Instead of spoofing the website's

address record, the attacker disguises an IP (internet protocol) address.

- **Session Hijacking**

The http communication uses many TCP connections and so that the server needs a method to recognize every user's connections. The most used method is the authentication process and then the server sends a token to the client browser. This token is composed of a set of variable width and it could be used in different ways, like in the URL, in the header of http requisition as a cookie, in other part of the header of the http request or in the body of the http requisition. The attack compromises the session token by stealing or predicting a valid session token to gain unauthorized access to the web server. This compromising of session token can occur in different ways.

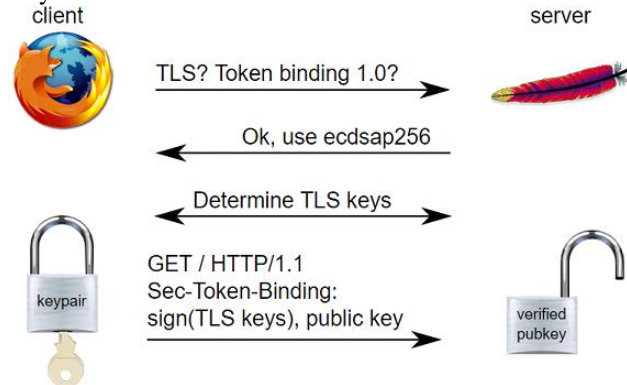


Fig . 6.1 Session Hijacking

- **Session Sniffing**

There is a string called tokens. This is the session id of a valid session. The first step by the attacker is getting this session id. The attacker uses a sniffer to get the session id. When the session id is captured, the attacker uses this session id to gain unauthorized access to the web server.

**The Cross-Site Script Attack**

The cross-site script attack is a way to get the session id with the helping of running malicious code or script from the client side. In this attack, the attacker executes malicious scripts, also known as malicious payloads into a legitimate website or web application. By using this attack, the attacker does not target a victim directly, but the attacker could exploit vulnerability in a website that the victim would visit and use the website to deliver malicious script to the victim's browser.

- **Session hijacking**

When you log in to a web application, you normally get a cookie with a session identifier. This random token identifies to the server that subsequent requests come from you. The server remembers you are logged in, and grants requests with that token access to your resources. Since this token is the only thing that distinguishes your requests from other requests, anyone who has this token can impersonate you. If the session identifier is

compromised, someone else can take over your session. The session identifier is known both in the browser and on the server, and is sent with every request. This is a big attack surface.

- **Using a key pair as token**

Token binding makes session hijacking harder by creating an identifier that is based on a private key. The client generates a public/private key pair for every site that it wants to use token binding on. When it connects to the server it signs something and sends this signature along with the public key to the server. The server verifies the signature against the public key. This way, the server knows that this client is in possession of the private key. After this verification step, the public key is passed to the application. This public key uniquely identifies the client, just as a session cookie would. However, it is no longer possible to simply steal the identifier and impersonate someone. The private key is kept secret and the identifier is checked against it. Without access to the private key it is not possible to reproduce a valid identifier. Even if an attacker intercepts the signature, he can't use this in another connection. The signature is over the public key and the keying material of the current TLS connection. When a new TLS connection is created, a new signature is needed. This means that if the attacker intercepts the signature, he can't reuse it in a new connection to the server..

## VII. CONCLUSION

This research has suggested security methods against the major attack Man In The Middle Attack which also known as Session hijacking that occurs when a malicious actor inserts himself as a relay/proxy into a communication session between people or systems. This provides all the information of session hijacking attack and shows how dangerous it is for the network security. Still many people are unaware from these kinds of attacks and network security expert also don't take it much seriously and lack of knowledge of session hijacking attack there is still poor session management of some of the web application and web server. In this project, we have discussed the countermeasure from the session hijacking attack which prevent the session hijacking attack.

## REFERENCES

- [1]. Bharat Bhushan ; G. Sahoo ; Amit Kumar Rai "Man-in-the-middle attack in wireless and computer networking — A review", 2019 3rd International Conference on Advances in Computing, Communication & Automation (ICACCA) (Fall).
- [2]. Mauro Conti ; Nicola Dragoni ; Viktor Lesyk "A Survey of Man In The Middle Attacks", IEEE

- Communications Surveys & Tutorials ( Volume: 18 , Issue: 3 , thirdquarter 2019 ).
- [3]. V. Radhakishan ; S. Selvakumar “Prevention of Man-in-the-Middle Attacks Using ID Based Signatures”, 2019 Second International Conference on Networking and Distributed Computing.
- [4]. Gopi Nath Nayak ; Shefalika Ghosh Samaddar “Different flavours of Man-In-The-Middle attack, consequences and feasible solutions”, 2019 3rd International Conference on Computer Science and Information Technology.
- [5]. Parth Patni ; Kartik Iyer ; Rohan Sarode ; Amit Mali ; Anant Nimkar “Man-in-the-middle attack in HTTP/2”, 2019 International Conference on Intelligent Computing and Control (I2C2).
- [6]. Shaun Stricot-Tarboton ; Sivadon Chaisiri ; Ryan K.L. Ko “Taxonomy of Man-in-the-Middle Attacks on HTTPS”, 2018 IEEE Trustcom/BigDataSE/ISPA.
- [7]. Oliver Eigner ; Philipp Kreimel ; Paul Tavolato “Detection of Man-in-the-Middle Attacks on Industrial Control Networks”, 2018 International Conference on Software Security and Assurance (ICSSA).
- [8]. Yisroel Mirsky ; Naor Kalbo ; Yuval Elovici ; Asaf Shabtai “Vesper: Using Echo Analysis to Detect Man-in-the-Middle Attacks in LANs”, IEEE Transactions on Information Forensics and Security ( Volume: 14 , Issue: 6 , June 2019 ).
- [9]. Bhargav Pingle ; Aakif Mairaj ; Ahmad Y. Javaid “Real-World Man-in-the-Middle (MITM) Attack Implementation Using Open Source Tools for Instructional Use”, 2019 IEEE International Conference on Electro/Information Technology (EIT).
- [10]. Zhe Chen ; Shize Guo ; Rong Duan ; Sheng Wang “Security Analysis on Mutual Authentication against Man-in-the-Middle Attack”, 2019 First International Conference on Information Science and Engineering