Proceedings of the Third International Conference on Electronics and Sustainable Communication Systems (ICESC 2022)
IEEE Xplore Part Number: CFP22V66-ART; ISBN: 978-1-6654-7971-4

2022 3rd International Conference on Electronics and Sustainable Communication Systems (ICESC) | 978-1-6654-7971-4/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICESC54411.2022.9885691

OWASP Attack Prevention

Ms. B. Kiruba, Assistant Professor,
Department of Computer Science and Engineering,
Sri Krishna College of Engineering and Technology,
Coimbatore, Tamil Nadu.
kirubab@skcet.ac.in

Mr.V. Saravanan,
Department of Computer Science and Engineering,
Sri Krishna College of Engineering and Technology,
Coimbatore, Tamil Nadu.
18eucs100@skcet.ac.in

Mr.T. Vasanth,
Department of Computer Science and Engineering,
Sri Krishna College of Engineering and Technology,
Coimbatore, Tamil Nadu.
18eucs125@skcet.ac.in

Mr.B.K.Yogeshwar,
Department of Computer Science and Engineering,
Sri Krishna College of Engineering and Technology,
Coimbatore, Tamil Nadu.
18eucs131@skcet.ac.in

Abstract— **The advancements in technology can be seen in recent years, and people have been adopting the emerging technologies. Though people rely upon these advancements, many loopholes can be seen if you take a particular field, and attackers are thirsty to steal personal data. There has been an increasing number of cyber threats and breaches happening worldwide, primarily for fun or for ransoms. Web servers and sites of the users are being compromised, and they are unaware of the vulnerabilities. Vulnerabilities include OWASP's top vulnerabilities like SQL injection, Cross-site scripting, and so on. To overcome the vulnerabilities and protect the site from getting down, the proposed work includes the implementation of a Web Application Firewall focused on the Application layer of the OSI Model; the product protects the target web applications from the Common OWASP security vulnerabilities. The Application starts analyzing the incoming and outgoing requests generated from the traffic through the pre-built Application Programming Interface. It compares the request and parameter with the algorithm, which has a set of pre-built regex patterns. The outcome of the product is to detect and reject general OWASP security vulnerabilities, helping to secure the user's business and prevent unauthorized access to sensitive data, respectively.**

Keywords—OWASP, Security Vulnerabilities, Web Application Firewall

I. INTRODUCTION

A) CYBER-ATTACKS

Cyber-attacks are the new generation of war targeting the technologies and people. Earlier, many disasters happened, including Bio war and nuclear attacks. These attacks are poisonous, affecting both the harmony of a nation and its economy. The economy is directly proportional to the threats and breaches. Many attackers aim for money. Cyber-attacks happen in government and corporate organizations with traditional or old security infrastructure. In the mid-1990s, the first Cyber-attack happened, "The Morris Worm," a virus that led to a DDOS attack (Distributed Denial of Service) created by a student. Though technology did not rise, cyber threats correspond to global destruction. Many dangerous attacks had happened in the early '20s where hackers used to steal stolen credit card credentials from the darknet [4][9].

B) OWASP

Open Web Application Security Project (OWASP) is a non-profit organization established in 2001 to provide security to website users and prevent their sites from malicious viruses and attacks. OWASP has listed the top 10 vulnerabilities for website users since 2003, and the list changes year to year. The latest top 10 vulnerabilities include Injection, Broken Authentication, sensitive data disclosure, XSS, XSE, Broken Access Control, Insecure Deserialization, Insufficient logging and monitoring, Security misconfiguration. OWASP is an open-source community, and it helps security analysts, pen testers, developers, and enthusiasts with provided materials, documentation, and tools to deal with the vulnerabilities. It also acts as a checklist for verifying and finding the bugs in a website, and it helps fix them. Many web applications have vulnerabilities that are relatively easy to exploit. Most successful attacks happen on the application layer (Layer 7 of OSI model). About 49% of web applications contain vulnerabilities of high-risk levels detected during automatic scanning [11]. Open Web Application Security Project (OWASP) is a non-profit organization established in 2001 to provide security to website users and prevent their sites from malicious viruses and attacks. OWASP has listed the top 10 vulnerabilities for website users since 2003, and the list changes year to year. The latest top 10 vulnerabilities include Injection, Broken Authentication, sensitive data disclosure, XSS, XSE, Broken Access Control, Insecure Deserialization,

Insufficient logging and monitoring, Security misconfiguration.

OWASP is an open-source community, and it helps security analysts, pen testers, developers, and enthusiasts with provided materials, documentation, and tools to deal with the vulnerabilities. It also acts as a checklist for verifying and finding the bugs in a website, and it helps fix them. Many web applications have vulnerabilities that are relatively easy to exploit. Most successful attacks happen on the application layer (Layer 7 of OSI model). About 49% of web applications to exploit. Most successful attacks happen on the application layer (Layer 7 of OSI model). About 49% of web applications contain vulnerabilities of high-risk levels detected during automatic scanning [11].

C) WEBSITES - A INTERNET RULER

Websites are the ultimate ruler in the World Wide Web, where their uses are unimaginable. Every person in the world depends upon websites, and half of the network traffic is caused by websites' usage. Websites help fulfil people's day-to-day needs, starting from Food to their work in their offices. Site pages consistently provide various information to the users and help them do their work instantly. Though websites' usage is extensive, most people rely upon the sites rather than an IOS or an Android application. Due to this usage, fraud and data theft are common, and prying eyes like attackers steal data easily. Websites are moderately easy to penetrate, allowing hackers to control the server to crash them or steal confidential data. Large companies use cloud servers, and due to these types of attacks, they also face many economic issues. Recovering from an attack can be more complex, and it also consumes much money in remodelling the servers. 60% of the attacks are happening on the websites, and the technology has emerged, leading to new vulnerabilities and loopholes. As far as I'm concerned, it cannot be entirely stopped, but a highly complicated network infrastructure might help prevent these types of threats and breaches.

D) FUTURE IN CYBER SECURITY FIELD

Since the attacks and threats happened, the demand for security researchers and professionals has proliferated in millions. As the world has only 20% of researchers and developers in the security field, there are fewer developers to prevent the attack or analysis. By the end of 2025, there will be a 35% uprising in the cyber security jobs in various organizations and private companies. International data corporations also estimated that the funding used for the security infrastructure rose to a value of $174.7 billion shortly. The National Association of Software and Service Companies (NASSCOM) had predicted that there would be a need for 1 million Cyber security professionals to ensure people's data security and privacy. In recent times, many organizations have separate posts for security engineers with high pay. Most of the Cyber related research is done by the Military wing Cyber Space of India to analyze the threat and prevent it before the attack happens. They also give awareness to people not to share highly confidential information with strangers.

II. EXISTING SECURITY SYSTEM

Our idea got even more vital when we saw an existing system with alternative features. "Detectify" is an application that provides deep layered security to its users. It is a cloud-based attack surface that maintains asset discovery and vulnerability assessments. Detectify uses ethical hackers to the source to encounter a payload-based testing system in less time and gives reliable security to the user domain. They also have a mobile-based application published in the google play store to increase its availability. The thing that separates us from "Detectify" is that we provide ultra-layer security to the domain users by tracking down the attackers and having an option to block the attacker's IP (Internet Protocol). We also have a rate limiter function and our custom-built API to prevent the attacker from making more API requests to the user's site, usually a hit.

III. PROPOSED SYSTEM

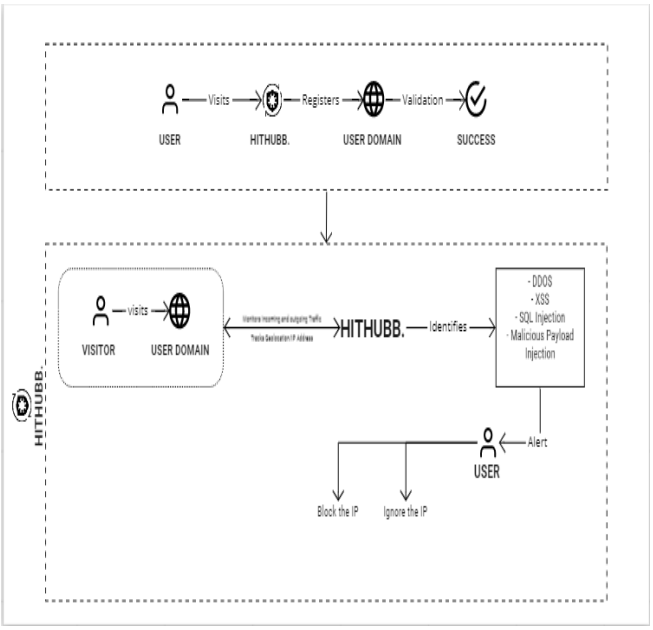

Figure: 3.1 Proposed System Architecture

The main goal of this system is to reduce vulnerable attacks from cyber attackers. There are three important steps involved to achieve this prevention from vulnerable attacks. The first and foremost step is monitoring the particular domain requests received. From that, it categorize the vulnerable attacks. And then tracing is the second foremost process. It can trace them down by using IP addresses. After confirmation from the domain's owner, "whether it is a vulnerable attack or not?". If it is, then we followed the third step by blocking the IP address from that particular domain to send requests. Otherwise, whitelist the IP address and then use that website. This blocking step is based on acknowledgement from the

domain holder's side. Let us see what the technologies used to develop this system are. It use webpack (HTML, CSS, Bootstrap) for front-end development. PHP is used for back-end development which is used to develop custom API. MongoDB is used to develop a Database for this system to store user data. The architecture of proposed work shown in above diagram 3.1

Furthermore, some more third-party APIs are used in a project like Twilio Google API, which is exclusively used to make a very efficient user authentication process. This is the base idea of this project and their technological stack used. Implementation and workflow model will be given in the below figure 3.2. The proposed module consists of three modules mainly—authentication module, user module, and then admin module. Detailed description of each module will be explained below.

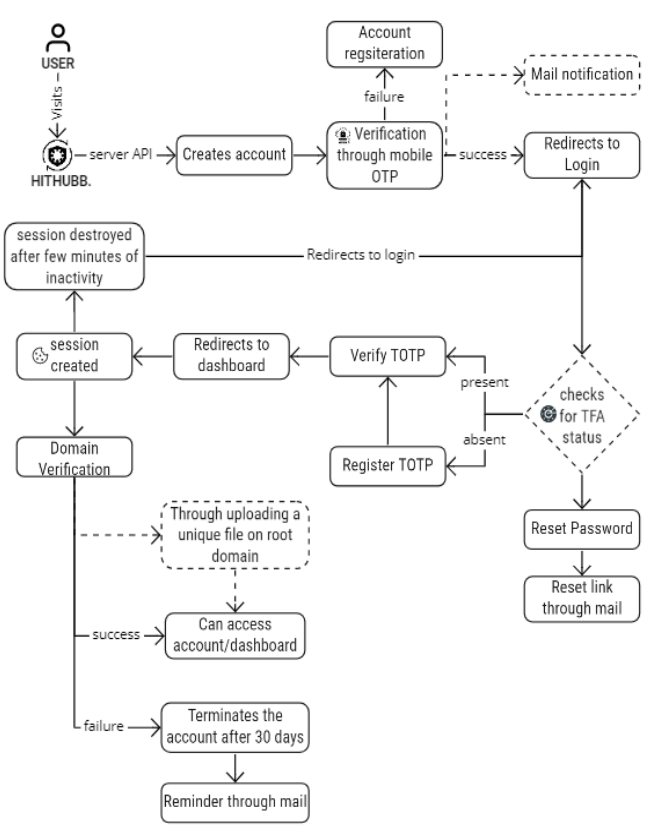

Fig 3.2 Workflow of Proposed Work

Authentication Module

The authentication part starts with the end-user registering with their valid domain name and email address. The proposed system verifies the user's information with two-factor authentication (TFA) [3] using time-based one-time-password (TOTP) [10]. Simultaneously, an email will be triggered to notify the user. Our system also has a feature of verifying email addresses to provide an extra layer of security to the user [2]. Only the domain name is verified using the in-built APIs, but the OTP generation and authentication, 'Twilio' third party API is used. To ensure a smooth onboarding and registering of the end-user, Google API also plays a significant role. Our proposed system verifies whether the domain is present or not with the help of our build APIs. If there are any mis-concerns in the domain name, the end-user will be given approximately 30 business days to provide a valid domain name for verification. If the user fails, the credentials will be erased from the server. Fig 3.3 represents the registration page and Fig 3.4 sample for Hithub login credentials.

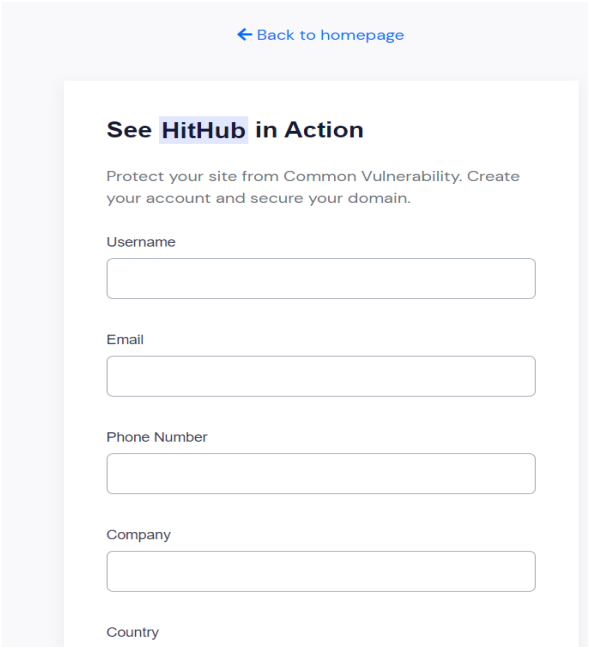

Figure 3.3 Registration page

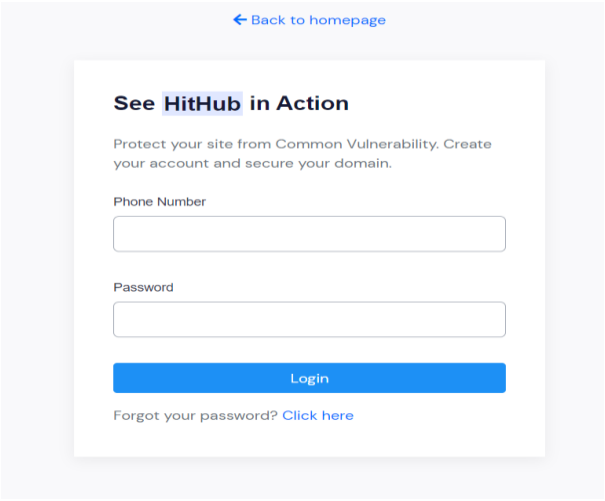

Figure 3.4 Hithub Login page

User Module

The user flow starts with authentication. A user needs to give basic information like Full Name, Address, Email, Mobile, and Password to register an account. We store the collected data in an encrypted format in a cloud database (MongoDB). The account registration is done through the successful verification through mobile OTP. In addition to that, the respective user must use Two Factor Authentication to secure their account. The user, after successful registration, has to verify the domain to access the dashboard. The domain verification process is done with the help of our inbuilt APIs; based upon the web server either of Apache or Nginx, our system will generate a new file with the encrypted hash. The respective user must upload the encrypted hash file to their root site to verify their domain. Upon successful verification of the domain, the user can access the dashboard. However, the failure of domain verification will lead to account deletion after 30 days from the date of account registration. Through the dashboard, the user can use the available security modules to protect the respective domain, and each module has a set of predefined rules to be followed by the user. A user with a base profile can access the DDOS module free of cost. In addition to that, users can select other modules after upgrading the user profile as per our pricing plan. The below figure 3. 5 shows the landing page.

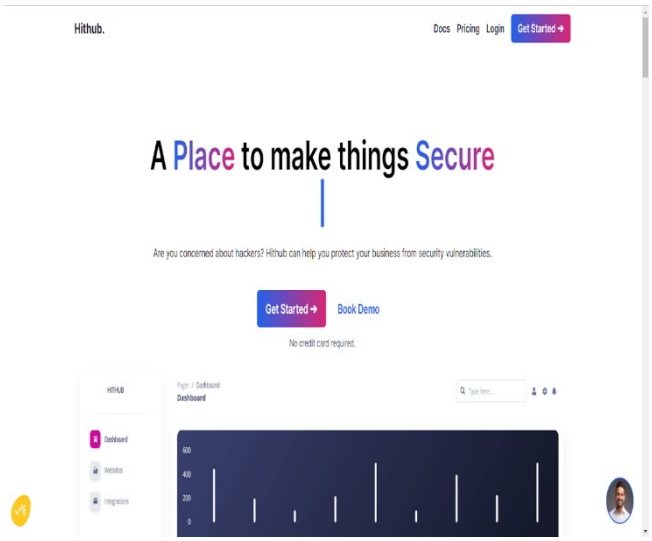

Figure 3.5 Landing page

Denial of Service Module

DOS attacks a networking structure to disable a server from servicing its clients. The host sends hundreds of ping requests (ICMP Echo Requests) with a large or illegal packet size to another host to knock it offline or keep it so busy responding with ICMP Echo replies that it cannot service its clients. The particular module helps users secure their domain from Denial-of-Service attacks. Bash file is generated based on the user's web server, either Apache or Nginx. The bash file has predefined code to monitor incoming and outgoing requests to and from the domain. The requests are noted down in a log file in the user's web server, and the file gets cleaned up after several requests using the feature of Nodemon. The data are shown to the user dashboard with the help of pub/sub. Our APIs will filter the malicious request through a set of predefined patterns. The IP address of the malicious request is traced down and notified to the domain owner to either block or ignore the IP address [6]. Figure 3.6 Show the flow diagram of Deniel of Service flow diagram.

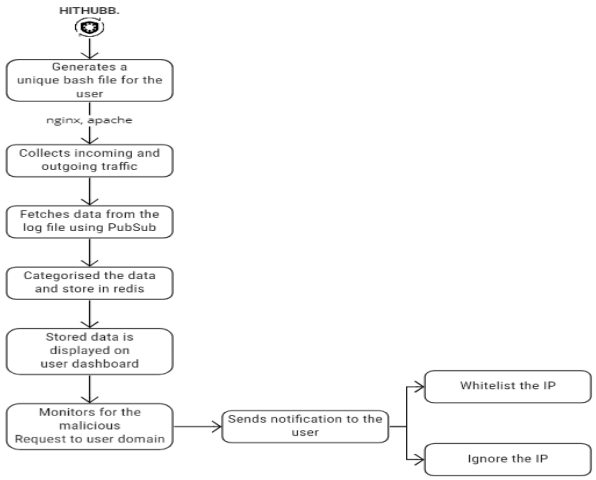

Figure: 3.6 Denial of Service flow diagram

Malicious Code Injection Module

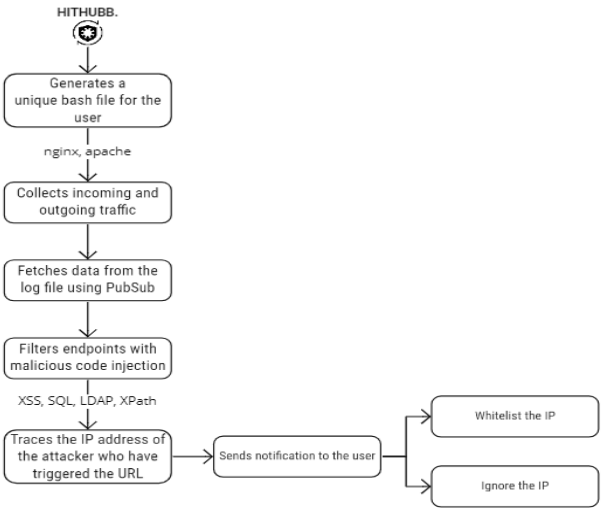

Figure: 3.7 Malicious code injection flow diagram

The most familiar type of this vulnerability class is Cross-site Scripting (XSS), a String-based code injection. Still, numerous vulnerabilities are harmful to the Application and data, notably

LDAP injection, XPath injection, and SQL injection. After successfully verifying the domain, the user can use this particular module to protect the site from malicious code injection in URL and URL parameters. Bash file is generated with predefined scripts and patterns of regex combined with machine learning algorithms based upon the user's web server, either of Apache or Nginx. A log file with URL endpoints is created in the user's root domain folder to monitor code injections. With the help of the script, the malicious code injection is traced and prevented from getting triggered on the user's site. In addition to that, the attacker's IP address who has triggered the code injection. The traced IP address is notified to the domain owner to block or ignore the IP address [7]. The above figure 3.7 shows Malicious code injection flow diagram

Domain-Based Message Authentication, Reporting & Conformance (DMARC) and Sender Policy Framework (SPF) Module

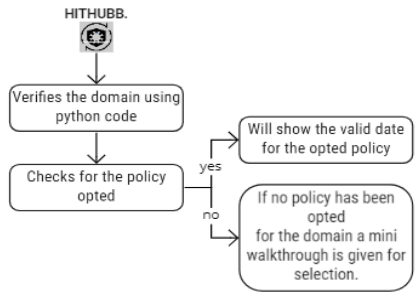

Figure: 3.8 DMARC flow diagram

DMARC (Domain-Based Message Authentication, Reporting, and Conformance) is an email authentication mechanism protecting your domain from spoofing. DMARC enables email receiver systems to detect when an email does not originate from an organization's allowed domains and instructs them how to handle these unauthorized emails. After verifying the domain, the user's site is checked for a valid DMARC and SPF record. The user's domain is checked against valid DMARC and SPF records through a predefined python code, based upon three DMARC policies: none, quarantine, and reject. In addition to that, the user can go through the policies and select the required policy for their business through the dashboard [2]. Figure 3.8 shown us the DMARC flow diagram.

IV. RESULT

User dashboard with very good user interface shown in the below diagram figure 4. In that dashboard displayed 4 things mainly such as number of users/customers of the domain, clicks, social redirect requests and banned IP addresses count. And also displayed vulnerability percentage comparison of every week with previous week vulnerability percentage.

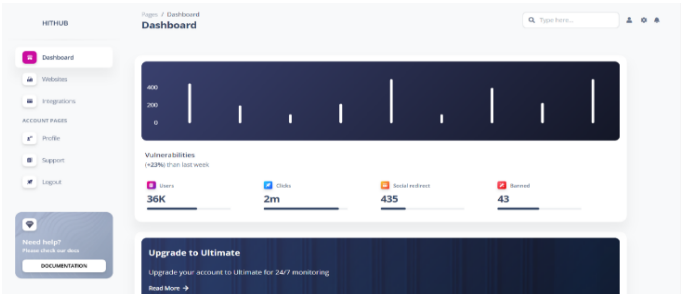

Figure 4 User Dashboard

V. CONCLUSION

This proposed system will increase the security and safety of the transaction and accessing of the information. No one can access the website without the knowledge of the domain owner. If this system implemented in a real-world application, all the data of the users are stored in the network securely. The major aim of the proposed system is that user can use their required websites whenever they want and domain owners run their websites without and data stealing by cyber attackers.

VI. REFERENCES

[1] Classification of Web Application Vulnerabilities, by Chavan.S & Meshram.B. International Journal of Engineering Science and Innovative Technology (IJESIT), 2(2), 226-234.

[2] Nightingale, J. (2017), Email Authentication Mechanisms: DMARC, SPF and DKIM, Technical Note (NIST TN), National Institute of Standards and Technology, Gaithersburg, MD, [online], https://doi.org/10.6028/NIST.TN.1945 (Accessed March 9, 2022)

[3] Gigih Forda Nama and Kurnia Muludi., Implementation of Two-Factor Authentication (2FA) to Enhance the Security of Academic Information System, Journal of Engineering and Applied science (JEASCI), 2018, Volume-13, Issue-8, Page No: 2209-2220

[4] Parthasarathi P, Shankar S, "Decision Tree Based Key Management for Secure Group Communication", Computer Systems Science and Engineering, Vol No. 42 Issue No.2, Page 561-575, January2022.

[5] ISO/ISC standard: Information technology - Open Systems Interconnection.

[6] OWASP. (2017). Owasp top 10-2017: https://owasp.org/www-pdfarchive//Cambridge_13-Mar-2018_OWASP_Top_10_2017.pdf

[7] D. Swathigavaishnave, R. Sarala, Detection of Malicious Code-Injection Attack Using Two Phase Analysis Technique- International Journal of Computer Applications© 2012 by IJCA Journal, Volume 45 - Number 18.

[8] A Learning-Based Approach to Secure Web Services from SQL/XPath Injection Attacks, IEEE, 191-198 -

[9] Saravanan Arumugam and Sathya Bama Subramanian, A Review on Cyber Security and the Fifth Generation Cyberattacks, Oriental Journal of Computer Science and Technology (OJCST), June 2019.

[10] Mariano Luis T. Uymatiao and William Emmanuel S. YU, Time-based OTP authentication via secure tunnel (TOAST): A mobile TOTP scheme using TLS seed exchange and encrypted offline keystore, 4th IEEE International Conference on Information Science and Technology, 2014.