CTICTR
CURRENT TRENDS IN INFORMATION COMMUNICATION TECHNOLOGY RESEARCH

BABCOCK UNIVERSITY ®
KNOWLEDGE TRUTH SERVICE
ILISHAN - REMO, NIGERIA.

CTICTR 3(1): 59 - 71 (June 2024)

ISSN: 2814-1709

Received: 06-03-2024

Accepted: 02-06-2024

https://doi.org/10.61867/pcub.v3i1a.192

DESIGN AND IMPLEMENTATION OF AN ENHANCED TWO FACTOR AUTHENTICATION MODEL

Adegbola Adesoji

Bassey Etido A.

Akande Oyebola

Fatade Oluwayemisi B.

Adeoti Babajide

Corresponding Author Email: adegbolaa@babcock.edu.ng

School of Computing and Engineering Sciences, Babcock University, Ilishan-Remo, Ogun State, Nigeria

Design and Implementation of an Enhanced Two Factor Authentication Model

Adegbola Adesoji[1], Bassey Etido Aniefiok[2], Akande Oyebola[3], Fatade Oluwayemisi Boye[4], Adeoti Babajide[5]

[1,5]*Department of Software Engineering, Babcock University, Ilisan-remo, Ogun State*
[2,3,4]*Department of Computer Science, Babcock University, Ilisan-remo, Ogun State*

Email: adegbolaa@babcock.edu.ng[1], bassey0559@pg.babcock.edu.ng[2], akandeo@babcock.edu.ng[3], fatadao@babcock.edu.ng[4], adeotib@babcock.edu.ng[5]

Abstract

Software security is paramount in today's digital landscape, especially where sensitive information and functionalities are increasingly accessed online. Traditional single-factor authentication, relying solely on passwords, is susceptible to breaches due to weak password practices and phishing attacks. This paper explores the design and implementation of a two-factor authentication (2FA) model as a robust security measure to strengthen software applications. It emphasizes the enhanced security provided by 2FA, requiring possession of both a "something you know" (password) and a "something you have" (secondary factor) for successful login. This study delves into the design aspects of the 2FA model, considering various secondary factor options. Common implementations like SMS-based one-time passwords (OTPs), authenticator apps generating time-based OTPs, and hardware tokens are discussed. The selection of an appropriate secondary factor depends on factors like usability, cost-effectiveness, and desired security level. The paper acknowledges potential challenges associated with 2FA, such as user adoption, potential inconvenience, dependence on external factors like phone connectivity, Man-in-The-Middle Attack (MTMA). By adding an extra layer of verification, it significantly reduces the risk of unauthorized access, protects sensitive data, and fosters trust among users. The paper emphasizes the importance of user education on strong password practices and secure handling of secondary factors to maximize the effectiveness of the 2FA system., and concludes by emphasizing the importance of 2FA as a critical security measure for software applications and the positive impact it has on user trust and data protection.

Keywords: *Security, Authentication, Two-factor Authentication, Cyber-attack.*

1. INTRODUCTION

The evolution of digital technologies and the increasing dependence on online platforms for both personal and professional activities have led to an escalating need for robust security measures. Among the various methods employed to enhance security, authentication plays a crucial role in ensuring that individuals accessing sensitive information or conducting transactions are indeed who they claim to be. Traditional authentication methods, primarily reliant on single-factor mechanisms such as passwords, have proven increasingly vulnerable to various forms of cyberattacks, including phishing, brute force attacks, and credential theft. The inadequacy of these methods has necessitated the development of more sophisticated approaches to secure user identities.

Two-factor authentication (2FA) emerged as a pivotal solution to address the shortcomings of single-factor authentication. By requiring users to present two different forms of verification, typically something they know (like a password) and something they possess (such as a mobile device), 2FA significantly enhances security by adding an additional layer of defense against unauthorized access. However, despite its effectiveness, the adoption of 2FA has faced several challenges. User experience issues, such as the inconvenience of additional steps in the authentication process, and concerns about usability have hindered widespread acceptance. Furthermore, existing 2FA systems are not immune to vulnerabilities, as demonstrated by the increasing sophistication of attacks targeting authentication processes.

Research highlights a variety of approaches to improving two-factor authentication systems. For instance, [1] emphasize the significance of evolving from Single-Factor Authentication (SFA) to Multi-Factor Authentication (MFA), noting how emerging technologies are enhancing user authentication methods to be more user-friendly and reliable. Similarly, [2] present formal models for mobile applications incorporating 2FA, focusing on the security goals and threat models that help clarify the properties of 2FA systems.

Moreover, innovative frameworks such as [3] introduce a three-factor authentication model, demonstrating the potential for enhanced security by integrating additional factors while preserving user privacy. The increasing use of biometrics, QR codes, and blockchain technology further illustrates the diverse methodologies being explored to fortify authentication processes [4],[5].

Despite these advancements, there remains a notable gap in the integration of usability and security in existing two-factor authentication implementations. For instance, studies indicate that many users prioritize convenience over security, which can lead to the rejection of secure authentication systems, as highlighted by [6]. This user-centric perspective is crucial for the design and implementation of an enhanced two-factor authentication model that effectively balances security needs with user experience.

The proposed research into the design and implementation of an enhanced two-factor authentication model aims to address these challenges by leveraging the latest technological advancements and user-centered design principles. By systematically analyzing existing 2FA systems, identifying their vulnerabilities, and integrating innovative solutions, this study aspires to develop a more robust, user-friendly authentication framework that not only improves security measures but also fosters greater user acceptance and adoption.

2. STATEMENT OF THE PROBLEM

Different application implements Two-Factor Authentication (2FA) model to enhance security through the use of the 2FA software-as-a-Service providers deployed in the cloud [7]. Third party applications are able to interface with the Two-Factor Authentication (2FA) service through dedicated provided endpoint, with this endpoint, applications request a valid token generation from 2FA service and use the originated token to authenticate users.

The problem with Two-Factor Authentication (2FA) design is that an attacker can intercepts the token generated, use a replay attack to compromise the application for which the Two-Factor

Authentication (2FA) service is meant to protect. Hence, [7] the 2FA still remains weak to Man-in-The-Middle (MITM) attack using reverse proxy.

In order to solve the problem and rectify this gap, we introduced an enhanced 2FA service and added two functions;

i. A function that retrieves the login users' system hostname location and authentication token receiving device location. These two locations are matched by the 2FA service, If the user's login location matches that of the phone meant to receive the authentication token, a successful generation of the token will occur. Otherwise, no token will be generated.

ii. Another function that checks that the generated token is successfully delivered to the device/phone meant to receive the authentication token, else authentication token is invalidated.

3. LITERATURE REVIEW

The increasing frequency and sophistication of cyber-attacks have amplified the importance of robust security measures in the digital landscape. Among these measures, Two-Factor Authentication (2FA) has gained prominence as an effective strategy to enhance security by requiring users to provide two distinct forms of verification before gaining access to sensitive information or systems. This literature review synthesizes current research on the design, implementation, and usability of 2FA models, highlighting key challenges, innovative approaches, and user-centered strategies that contribute to the effectiveness of these authentication mechanisms.

3.1 The Evolution of Two-Factor Authentication

The transition from traditional single-factor authentication (SFA), which primarily relies on passwords, to 2FA reflects a growing recognition of the limitations inherent in password-only systems. Passwords have been shown to be susceptible to various forms of cyber threats, including phishing, credential stuffing, and brute force attacks. As highlighted by [8], [19], reliance on passwords alone is increasingly deemed inadequate, leading to the adoption of more robust authentication methods that combine something the user knows (e.g., a password) with something the user possesses (e.g., a security token or smartphone) [9].

The conceptual framework for 2FA has evolved significantly, as evidenced by the works of [9], who emphasize the critical role of user convenience and security in shaping the adoption of various 2FA methods, such as one-time passwords (OTPs), security tokens, and biometric verification [10]. This evolution is also characterized by a shift towards Multi-Factor Authentication (MFA), which encompasses 2FA as a subset, enhancing security by incorporating multiple verification methods [8].

3.2 Current Challenges in Two-Factor Authentication

Despite its advantages, the implementation of 2FA is not without challenges. A prominent concern is usability, as many users find the additional steps required for authentication to be cumbersome. This sentiment is echoed in the work of [11], which identifies a trade-off between enhanced security and user convenience, noting that while users perceive 2FA as more secure, they often regard it as inconvenient compared to SFA systems.

[1] further underscore this point by revealing that the perceived inconvenience of 2FA leads to low adoption rates among users, who often prioritize usability over security. In addition to usability issues, existing 2FA methods are vulnerable to specific attacks, such as SIM swapping and man-in-the-middle attacks. [12] provide insights into the security gaps present in current 2FA implementations, particularly the reliance on PINs and subscriber identity modules (SIMs), which can be exploited by attackers. According to [20] and [21], the key challenges appear to be around device compatibility, usability, organizational adoption, security limitations, and the need for more customizable authentication solutions beyond just 2FA. Addressing these challenges could help improve the overall effectiveness and adoption of multi-factor authentication approaches.

3.3 Innovative Approaches to Enhance Two-Factor Authentication

Recent advancements in technology have facilitated the development of innovative approaches to 2FA that aim to mitigate existing challenges. For instance, decentralized authentication frameworks that leverage blockchain technology have been proposed as a solution to the vulnerabilities associated with centralized systems. Alharbi and [13] introduce a 2FA framework based on blockchain, which enhances security through encrypted OTPs and smart contracts, thereby addressing concerns related to man-in-the-middle attacks.

Moreover, the integration of biometric authentication methods has garnered attention as a means to improve both security and usability. [14] and [21] discuss the potential of biometrics, such as fingerprint and facial recognition, as effective second factors in 2FA systems, providing additional security while reducing the cognitive burden on users. The inclusion of user-friendly biometric solutions is vital, as highlighted by [15], who found that usability and acceptability are significant factors influencing the adoption of hardware-based 2FA methods.

Additionally, innovative models such as the Transparent Two-Factor Authentication (T2FA) proposed by [16] have emerged. This model utilizes physical unclonable functions (PUFs) and voiceprints to provide secure authentication with minimal user interaction, thereby addressing usability concerns while maintaining robust security.

3.4 User-Centric Design for Enhanced Adoption

The literature emphasizes that user experience is a critical determinant of the successful adoption of 2FA models. [17] propose a user-centric 2FA system that incorporates personalized image verification, enhancing user engagement and satisfaction in the authentication process. This approach not only improves security but also addresses usability issues by allowing users to select images that hold personal significance, fostering a more intuitive authentication experience.

Furthermore, the importance of consistent user experiences across platforms is highlighted by [18], who argue that inconsistencies in 2FA implementations can lead to cognitive friction and deter user adoption. Their advocacy for establishing general UX guidelines for the design of 2FA systems is crucial, as improved consistency can significantly enhance user satisfaction and acceptance.

3.5 Technological Integration and Future Directions

The integration of emerging technologies, such as artificial intelligence (AI) and machine learning, presents promising opportunities for enhancing the security and efficiency of 2FA systems. [5] illustrate how AI can be employed to develop adaptive authentication systems that respond to user behavior patterns, thereby enhancing security while minimizing user inconvenience.

Moreover, the continued evolution of Multi-Factor Authentication (MFA) frameworks suggests a trend towards increasingly sophisticated and user-friendly authentication solutions. [10] discuss the modeling of MFA using Petri Nets, providing a systematic approach to analyze and design authentication processes that can be effectively applied to 2FA.

In conclusion, the design and implementation of Two-Factor Authentication models necessitate a comprehensive approach that addresses existing vulnerabilities while prioritizing user experience. The literature highlights the importance of integrating innovative technologies, adopting user-centered design principles, and establishing consistent and reliable authentication processes. Future research should continue to explore the intersection of security and usability, aiming to develop robust 2FA frameworks that meet the evolving demands of digital security while fostering user acceptance and engagement. The synthesis of existing studies underscores the need for ongoing innovation in the field of authentication, as the landscape of cyber threats continues to evolve.

4. METHODOLOGY

Developing an enhanced Two-Factor Authentication (2FA) system requires a cautiously planned approach that includes several phases: analyzing requirements, designing, implementing, and evaluating. Each phase plays a crucial role in creating a secure and user-friendly model, addressing known weaknesses and usability concerns highlighted in current research.

4.1 Email Based 2FA Authentication Model Design

Figure1 below represents the general email-based/phone-based two-factor authentication (2FA) model design that requires a user to provide two forms of authentication in order to access an online account or an application. In addition to the standard username and password, email or phone based 2FA requires users to enter a unique code that is sent to their email address or phone. The user enters their username and password on the login page of a Spring Security application, The application checks the username and password against its database to confirm that they match. If the username and password are correct, the application generates a unique code and sends it to the email address/phone associated with the account. The user retrieves the code from their email/phone and enters it into the login page. If the code is correct, the user is granted access to their account.

In this general model, problem arises user who cannot access email/phone immediately due to network downtime, email gateway can also encounter message broker error where the needed authentication token is not sent as at when expected, hence creating delay in application access. In an extreme case, a hacker who successfully brute force user password together with having unauthorize access to user’s mail/phone has already compromise the application security.

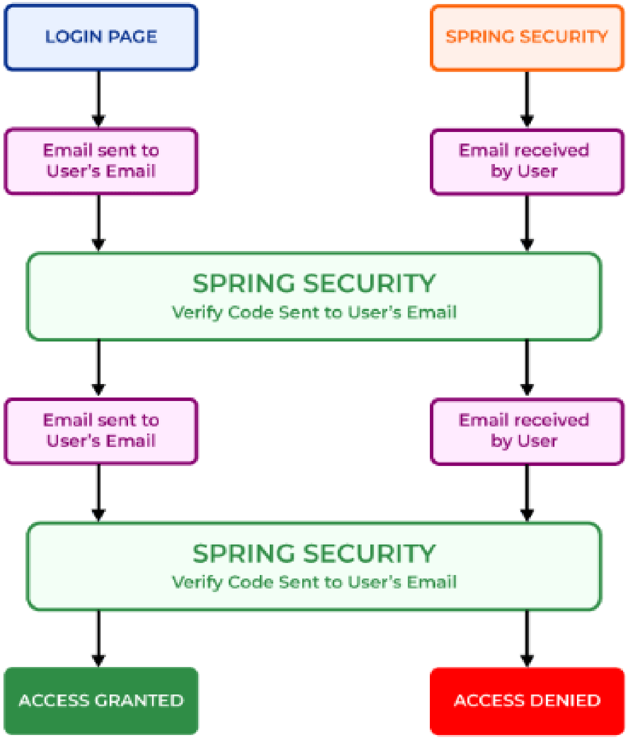

Figure 1: Email based 2FA authentication model

4.2 Proposed 2FA Authentication Model Design

To strengthen this solution and rectify this gap, we added a function to the authentication services that retrieves the system hostname/IP address and user location during login, then cross-references this information with the device location where the multi-factor authentication service transmits a secret key. If the user's login location matches that of the phone generating the token, then a successful generation of the token will occur; otherwise, there will be no token generated.

The geo-location enhancement of the 2FA model implementation leverages on GeoIP2 API web service. The GeoIP2 system processes the IP address of and application user, process the IP address of the phone that receives authentication code and returns the corresponding geolocation data to the 2FA model class in the source code which compares and two location and make informed decision of either generating the 2FA code, validate and sending to the SMS receiving device/phone or fail to generate and send code if locations are not same.

The idea behind this enhancement is that attackers always intercept secret keys from a location different from the actual account owner. Therefore, the proposed enhancement introduces a check mechanism to ensure that the location of the attempted login is equal to the location of the phone that receives the secret authentication key. Login is only possible when the two locations correspond.

The enhanced architecture below provides a step-by-step explanation of the proposed implementation. A user tries to log in, the login process triggers a two-factor authentication service that requests the hostname of the user's device attempting to log in. Using the hostname, the authentication service keeps track of the location of the system used for login. The authentication service sends a secret key to the device that is required to generate the actual code and also retrieves the device's location. If the location of the attempted login user is equal to the location of the phone displaying the valid token, a 2FA code is generated. Otherwise, the 2FA code fails to generate. This helps to prevent and invalidate any attempts to hijack the secret key or session by a middleman.

The Figure3 below depict the high-level representation of this 2FA process flow. This model is developed based on the study and understanding of calypso application and the different authentication methods that come out of box.

From the diagram above, an application user calls the login page of calypso application to gain authentication access. The user provides username and password and calypso application checks the validity of the details in the database. If the username and password are wrong, user gets incorrect password error message, else the application request a 2FA pin from the user. The application checks the token by calling the second factor authentication method which internally does some computations, look through the logic implementation in the model and decide if token provided by user matches the one generated internally by application. If taken is not valid, login fails else, user is granted access to the application.

The 2FA authentication model class implements core calypso authentication class which requires additional layer of authentication request from the user before access is granted.

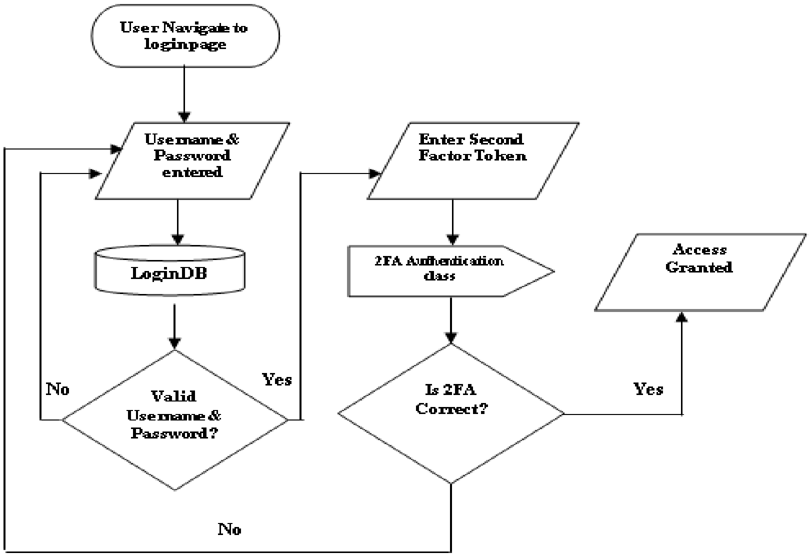
User Navigate to loginpage
Username & Password entered
LoginDB
No
Valid Username & Password?
Yes
Enter Second Factor Token
2FA Authentication class
Is 2FA Correct?
Yes
Access Granted
No

Figure 2: Authentication Process Flow

4.3 System Development Tools

Java Programming Language: Java Programming Language is used for the development of this software because it is a crucial language for developing enterprise applications due to its ability to connect to databases, communicate with other systems, and handle input and output operations.

- **Gradle Builder**: Gradle allow users to build applications with enormous demands for computational resources, it is a popular open-source build automation tool widely used in Java projects. Gradle supports the build deploy and customization of source code. It is used to build source code into Java Archive (.jar) file.

- **Spring Boot Framework:** Spring Boot is used for this project because it is an open-source framework built on top of the popular Spring framework and it is designed to simplify the creation of projects using Java language, especially microservices. It is a powerful and flexible framework for building enterprise applications, and it is widely known for offering autoconfiguration of application components based on various criteria and development tools and needs. It is particularly well-suited for web applications, with Spring MVC being the most popular Java web framework
- **Integrated Development Environment**: While there are many Integrated Development Environments (IDEs) available for Java development, Visual Studio Code (VS Code) has become a popular choice because of its unique blend of features and flexibility. Visual studio Editor was used for this project because like other IDEs, VS Code is a lightweight code editor that doesn't bog down the system with unnecessary resources. This makes it ideal for developing fast and responsive application.
- **Application Server:** The application server provisioned for this 2FA application deployment and evaluation is the Red Hart Linux which is a powerful and versatile operating system. Unlike its commercial counterparts, Linux boasts a unique open-source nature, granting users the freedom to modify and distribute its code. At the heart of Linux lies the kernel, responsible for managing hardware resources like memory and processors. It acts as the bridge between software applications and the underlying hardware. I interacted with the Linux primarily through a command-line interface (CLI) called the shell.

4.4 Integration 2FA Model with Existing Software System

Calypso components are contained in various jars that are scanned for Spring annotation to start the servers. Therefore, client custom components must become part of jar files when starting the server. To facilitate this, Calypso provides the Custom-Extensions directory located within the installation directory. Custom-Extensions contains the open-source tools used by Calypso to create, package and deploy the Calypso binaries, as well as to maintain a changes audit trail.

Using a naming convention for custom code (2FA model) prevents clashes in name space and makes the process smoother by making it clear in stack traces, thread dumps, and when the customization is visible to the user that we are dealing with custom code as opposed to core code.

The Custom-Extensions directory of calypso application (/custom-extensions) is intended to capture all local modifications made to an instance of Calypso. Such modifications may range from a simple configuration file change, the addition of new Java classes to extend platform functionality, or the deployment of third-party libraries required by Calypso or new Java classes. Some application like Calypso also included tools to assist in developing, compiling, packaging, and deploying custom components into the Calypso binaries. The use of the Using Custom-Extensions directory helps ensure that any modifications made to core Calypso are clearly separated to simplify the change management process for clients. The provided tools ensure that a proper audit trail is made available for each change and that potential incompatibilities are made visible during development rather than in production.

Calypso requires all modifications to the platform to be made via the Custom-Extensions project. Build scripts contained in the Custom-Extensions directory provide support for VSCode by preparing a series of projects (organized into subdirectories) with the required class path for compiling extensions. These projects can also simply contain modified configuration files or new classes and libraries. The projects (and directories) are separated by the Calypso deployment artifacts including the dataserver, engineServer, client and common shared library. This separation ensures that artifacts are not distributed to components unless required.

To deploy the 2FA model in calypso, I first leveraging on the Gradle tool library and run the project to ensure the source code is free from compiled time errors, this also compile the project from java source code to byte code (binary) as seen in the screenshot below.

Upon success, the Gradle created the project and added the class files as well as structure the files according to package structure implemented before build process.

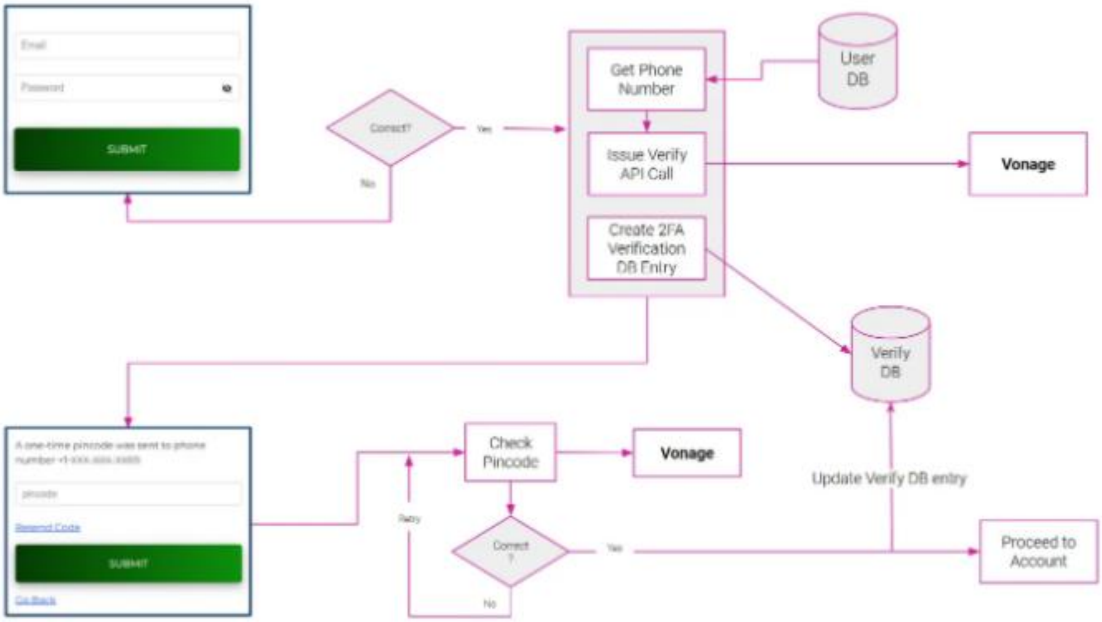

Figure 3: The Enhanced 2FA Flow to Generate Valid Token

4.5 Testing the Enhanced 2FA Model

The service verifies the location of the login user, compares that with the SMS receiving device meant to receive the validated token. Once the location of the login user and device meant to receive SMS is validated as having same location, and token can only be valid once response is received from the SMS receiving device that token is delivered, then login is successful else, login fails.

5. CONCLUSION AND RECOMMENDATION

Two-factor authentication (2FA), also known as multi-factor authentication (MFA), is an extra layer of security for application access. It requires two different verification methods to access an account, making it significantly harder for attackers to gain unauthorized access. Notwithstanding the security level as an added advantage, user inconvenience is still a concern with respect to remembering multiple factors or carrying additional devices still perceived as cumbersome for users, Security token dependence like lost or stolen token, SMS-based verification codes susceptible to SIM swapping attacks etc.

Looking at the basic functionality of 2FA authentication implementation in general, another kind of authentication design was introduced in this paper outlining all the different kinds of attacks that could compromise the system or even logical flaws in the implementation of current authentication models and systems that could result in potential breaches or attacks. Although 2FA services are more reliable than traditional username and password model, there are many security risks and threats related to 2FA services on cloud, and such architectures must be carefully examined to assess the risk. By way of hardening the security architecture of financial applications, implementing TLS, application access rights, and authentication and authorization methods can further reduce the security associated with 2FA implementations.

End-to-end packet traversal when using large-scale networks, like WAN is a critical component for real-time applications that introduces risk of attack to APIs endpoints, this should be further examined before implementing 2FA with a service provider in the cloud. To avoid this risk, the 2FA model introduced in this paper becomes a solution to consider while further study and research is carried out to position this model as a global solution.

Based on the findings of this study, it is recommended that further research be conducted on this 2FA model to improve on the algorithm that generates the authentication token within the system. One of the pain points of the general 2FA authentication services is that tokens can still be stolen over the network or from authenticator devices, further study can be carried out to strengthen this proposed design and introduces a method through which generated token can be communicated to user without sending it over the network tunnels.

REFERENCES

[1] A. Ometov, S. Bezzateev, N. Mäkitalo, S. Andreev, T. Mikkonen, Y. Koucheryavy. Multi-factor Authentication: A Survey. Cryptography, Volume 2 Number 1 pp. 1 – 31, 2018. https://doi.org/10.3390/cryptography201000.

[2] G. Sciarretta, R. Carbone, S. Ranise, L. Vigano. Formal Analysis of Mobile Multi-Factor Authentication with Single Sign-On Login. ACM Transactions on Privacy and Security, Volume 23, Number 3, pg.1–37, 2020. doi:10.1145/3386685

[3] J. Yu, G. Wang, Y. Mu, W. Gao. An Efficient Generic Framework for Three-Factor Authentication With Provably Secure Instantiation. IEEE Transactions on Information Forensics and Security, Volume 9 Number 12, pp. 2302–2313, 2024. doi:10.1109/tifs.2014.2362979

[4] B. Rodrigues, A. Chaudhari, S. More. Two factor verification using QR-code: A unique authentication system for Android smartphone users. 2016 2nd International Conference on Contemporary Computing and Informatics (IC3I). doi:10.1109/ic3i.2016.7918008

[5] C. McCabe, AIC Mohideen, R.A. Singh. Blockchain-Based Authentication Mechanism for Enhanced Security. Sensors. Volume 24, Number 17, pp. 5830-5842, 2024. https://doi.org/10.3390/s24175830

[6] F. Chavez, A. Fernandez-Reyes, M.D. Byrne. Context Contributes to Two-Factor Authentication Choices. Proceedings of the Human Factors and Ergonomics Society Annual Meeting, Aug 2024. doi: 10.1177/10711813241261680

[7] A. Murray, G. Begna, E. Nwafor, J. Blackstone and W. Patterson. Cloud Service Security & application vulnerability. SoutheastCon 2015, Fort Lauderdale, FL, USA, 2015, pp. 1-8, doi: 10.1109/SECON.2015.7132979.

[8] L. F. B Soares, D.A.B Fernandes, M.M. Freire, P.R.M. Inacio. Secure user authentication in cloud computing management interfaces. 2013 IEEE 32nd International Performance Computing and Communications Conference (IPCCC). doi:10.1109/pccc.2013.6742763

[9] D. Bhanderi, M. Kavathiya, T. Bhut, H. Kaur and M. Mehta. Impact of Two-Factor Authentication on User Convenience and Security. 2023 10th International Conference on Computing for Sustainable Global Development (INDIACom), New Delhi, India, 2023, pp. 617-622.

[10] M. W. AlawadhiWasan, S. Awad. Multi-Factor Authentication Modeling using Petri Nets: Review. 2023 International Conference on IT Innovation and Knowledge Discovery (ITIKD), Bahrain, 2023.

[11] N. Gunson, D. Marshall, H. Morton, M. Jack. User perceptions of security and usability of single-factor and two-factor authentication in automated telephone banking. Computers & Security, Volume 30, Number 4, 208–220. 2011. doi:10.1016/j.cose.2010.12.001.

[12] G. Ali, D.M. Ally, S. A. Elikana. Two-Factor Authentication Scheme for Mobile Money: A Review of Threat Models and Countermeasures. Future Internet, Volume 12 Number 10, pg. 160, September 2020. doi:10.3390/fi12100160

[13] E.T. AlharbiDaniyal & A. Alghazzawi. "Two Factor Authentication Framework Using OTP-SMS Based on Blockchain" Transactions on Machine Learning and Artificial Intelligence, Volume 7, June 2019. DOI: 10.14738/tmlai.73.6524

[14] C. Rathgeb, A. Uhl. Two-Factor Authentication or How to Potentially Counterfeit Experimental Results in Biometric Systems. In: Campilho, A., Kamel, M. (eds) Image Analysis and Recognition. ICIAR 2010. Lecture Notes in Computer Science, vol 6112. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-13775-4_30

[15] S. Das, G. Russo, A. C. Dingman, J. Dev, O. Kenny, L. J. Camp. A qualitative study on usability and acceptability of Yubico security key. Proceedings of the 7th Workshop on Socio-Technical Aspects in Security and Trust - STAST '17. 2018. doi:10.1145/3167996.3167997

[16] j. Zhang, X. Tan, X. Wang, A. Yan, Z. Qin. T2FA: Transparent Two-Factor Authentication. IEEE Access, 6, 32677–32686. 2018. doi:10.1109/access.2018.2844548

[17] E. Djeki, J. Dégila and M. H. Alhassan, "Reimagining Authentication: A User-Centric Two-Factor Authentication with Personalized Image Verification," 2024 ASU International Conference in Emerging Technologies for Sustainability and Intelligent Systems (ICETSIS), Manama, Bahrain, 2024, pp. 281-285, doi: 10.1109/ICETSIS61505.2024.10459708.

[18] G. L. Sanam, B. Michael, B. Sven. A Systematic Study of the Consistency of Two-Factor Authentication User Journeys on Top-Ranked Websites. Network and Distributed System

Security (NDSS) Symposium 2023 27 February - 3 March 2023, San Diego, CA, USA ISBN 1-891562-83-5, https://dx.doi.org/10.14722/ndss.2023.23362

[19] A. Laube and R. Koenig. Secure Two-Factor Authentication with SwissPass Crypto Card: A Case Study. Future Technologies Conference (FTC) 2017 29-30 November 2017| Vancouver, Canada Pg. 837-844

[20] F. C. Chazanga, J. Phiri, S. Namukolo. Development of a Two Factor Authentication for Vehicle Parking Space Control based on Automatic Number Plate Recognition and Radio Frequency Identification. International Journal of Advanced Computer Science and Applications, Volume. 10, Number 1, Pg. 588-597, 2019

[21] M. Syahreen, N. Hafizah, N. Maarop, M. Maslinan . "A Systematic Review on Multi-Factor Authentication Framework", A Systematic Review on Multi-Factor Authentication Framework 2024, vol. 15 No. 5, Pg.1043