Multifactor authentication enabled JSON Web Tokens for Authentication, Authorization usingRBAC in Web Applications

Pratheep V	Dhanush S R	Jayanth Kumar S
21BCE1093	21BCE1204	21BCE1218

INFORMATION SECURITY MANAGEMENT (BCSE354E)

TB1 & L33+L34 – SENDHIL R

WINTER SEMESTER 23-24

Multifactor authentication enabled JSON Web Tokens for Authentication, Authorization using RBAC in Web Applications

Pratheep V
School of Computer Science and Engineering
Vellore Institue of Technology, Chennai
Chennai, India
pratheep.v2021@vitstudent.ac.in

Dhanush S R
School of Computer Science and Engineering
Vellore Institue of Technology, Chennai
Chennai, India
dhanush.sr2021@vitstudent.ac.in

Jayanth Kumar S
School of Computer Science and Engineering
Vellore Institue of Technology, Chennai
Bengaluru, India
jayanthkumar.s2021@vitstudent.ac.in

Abstract - This study proposes the integration of multifactor authentication (MFA) within a JWT-based model to enhance user authenticity verification in blockchain applications. By combining these technologies, a robust and secure system for verifying user identities is established. MFA adds an extra layer of security with multiple verification methods. The MFA-enabled JWT model, implemented with Spring Boot, Spring Security, MySQL, ReactJS, and Java Mail Sender offers protection against CSRF, brute force attacks, credential stuffing, and man-in-the-middle attacks. Leveraging JWT for authentication and authorization minimizes performance overhead while ensuring secure user access. Rigorous testing using Postman demonstrates the model's reliability and effectiveness in safeguarding web applications, providing a strong defense against cyber threats and enhancing overall system security and user trust.

Keywords - Multifactor authentication (MFA), JWT (JSON Web Token), User authenticity verification, Security enhancement, Tamper-resistant storage, Web application security, CSRF protection, Man-in-the-middle attack, RBAC

I.INTRODUCTION

In the ever-evolving landscape of Web 3.0 applications, security stands as a paramount concern. With the proliferation of sensitive data such as users' wallet information, passwords, and cryptocurrencies, the stakes for robust authentication and authorization mechanisms have never been higher. Traditional methods often fall short, grappling with security vulnerabilities and performance limitations.

Traditional authentication techniques like username and password suffer from several limitations:

1. Password Weakness: Users often create weak passwords that are easy to guess or crack, leaving accounts vulnerable to brute force attacks.

2. Phishing Vulnerability: Users may fall victim to phishing attacks where they unknowingly provide their credentials to malicious actors, compromising their accounts.

3. Credential Reuse: Users tend to reuse passwords across multiple accounts, increasing the risk of a single breach leading to multiple compromised accounts.

These limitations highlight the need for additional authentication measures, such as multi factor authentication, to enhance the security of online accounts and systems.

While passwords suffer from weaknesses like susceptibility to phishing and credential reuse, stateful authentication maintains session data on the server, relying on session identifiers or cookies for subsequent requests. On the other hand, stateless authentication generates tokens containing user information, eliminating the need for server-side session storage. These approaches offer different trade-offs in terms of scalability and security

Traditional authentication and authorization techniques face several security and performance limitations:

1. Security Limitations:

- Vulnerability to Password Attacks: Traditional username and password authentication is susceptible to various attacks such as brute force, dictionary attacks, and credential stuffing.

- Risk of Phishing: Users can fall victim to phishing attacks, where malicious actors trick them into revealing their credentials.

- Limited Granularity: Role-based access control (RBAC) used in traditional authorization lacks granularity, making it challenging to implement fine-grained access control policies.

- Session Fixation: In stateful authentication, session fixation attacks can occur, where an attacker forces a user's session ID to a known value, enabling unauthorized access.

2. Performance Limitations:

- Session Management Overhead: Stateful authentication requires server-side session management, leading to increased server load and potential scalability issues.

- Latency: Traditional authentication methods often involve multiple round-trips between the client and server, resulting in increased latency, especially in distributed systems.

- Token Size: Tokens used for authentication and authorization in traditional methods can become large, increasing network overhead, especially in high-traffic scenarios.

Stateful Authentication: In stateful authentication, the server keeps track of the user's session after successful login. This session information is typically stored on the server's memory or database. The client is usually assigned a session identifier, often stored in a cookie, which is sent with each subsequent request to identify the session. The server uses this session identifier to retrieve the corresponding session data and authenticate the user.

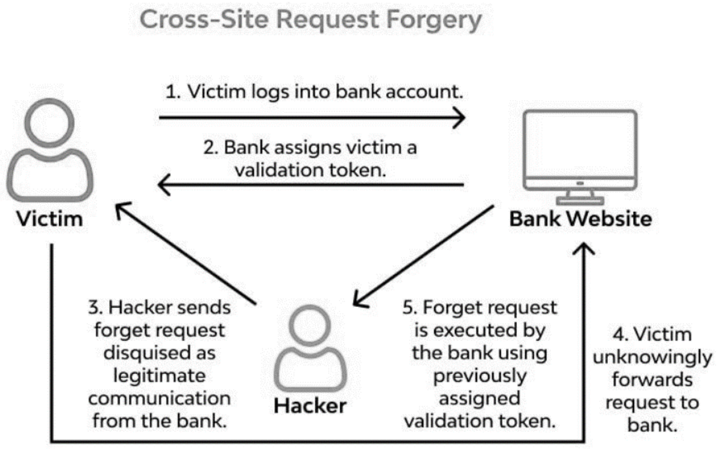
Cross-Site Request Forgery
1. Victim logs into bank account.
2. Bank assigns victim a validation token.
Victim
Bank Website
3. Hacker sends forget request disquised as legitimate communication from the bank.
Hacker
5. Forget request is executed by the bank using previously assigned validation token.
4. Victim unknowingly forwards request to bank.

Fig 1. CSRF

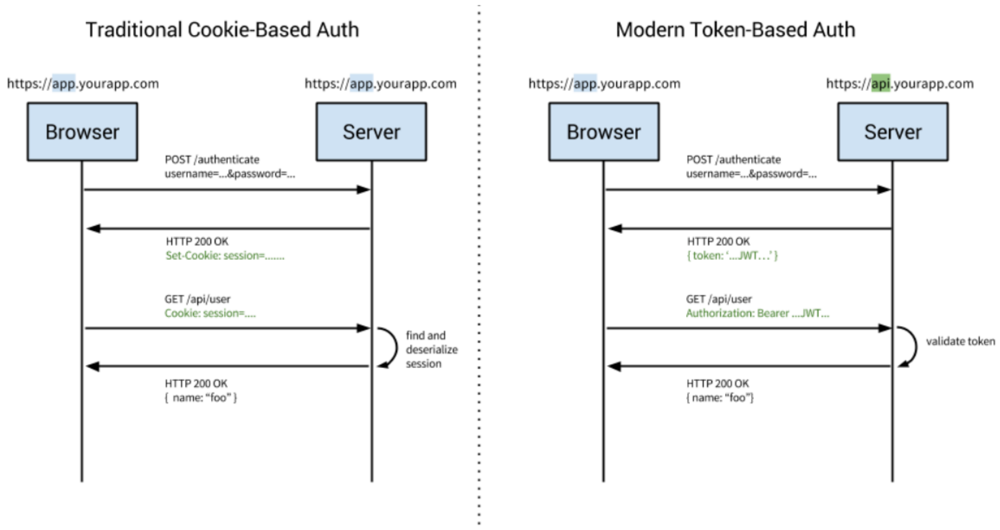

Fig 2. Stateful vs Stateless Authentication

Stateless Authentication: With stateless authentication, the server does not store any session information. Instead, upon successful authentication, the server generates a token containing user information (such as user ID, roles, etc.). This token is then sent to the client, usually in the form of a JSON Web Token (JWT). The client includes this token in the header of each subsequent request to the server. The server verifies the authenticity and integrity of the token to authenticate the user. Since there is no session maintained on the server, stateless authentication is often used in distributed systems and APIs.

Cross-Site Request Forgery (CSRF) is a type of security vulnerability that occurs when an attacker tricks a user into unintentionally executing actions on a web application in which the user is authenticated. The attacker typically crafts a malicious request and lures the victim into submitting it through various means, such as phishing emails or compromised websites. Since the user is already authenticated with the targeted web application, the malicious request appears legitimate, and the application processes it, potentially resulting in unauthorized actions being performed on behalf of the victim.

However, amidst these challenges emerges a beacon of promise: JSON Web Tokens (JWTs). JWTs offer a compelling alternative, addressing many of the security issues prevalent in Web 3 applications. Renowned for their compactness, statelessness, and security, JWTs provide a streamlined mechanism for transmitting authentication data efficiently and securely. By leveraging JWTs, applications can not only enhance security but also achieve greater scalability and efficiency in managing user authentication and authorization. Our research endeavors to delve deeper into this realm, proposing a stateless multifactor-enabled JWT solution tailored specifically for Web 3.0 applications. With an emphasis on securing endpoints using Role-Based Access Control (RBAC), our research aims to pave the way for a safer and more resilient future in the realm of Web 3.0.

Timing Diagram for JWT based Authentication in web applications

Components of a JWT and their roles:

Header: The header typically consists of two parts: the type of the token, which is JWT, and the signing algorithm being used. It is base64url encoded and provides information about how the token should be processed.

Payload: The payload contains the claims, which are statements about the user and any additional data. These claims can be divided into three types: reserved claims, public claims, and private claims. The payload is also base64url encoded. We can implement RBAC in JWT token using claims in playlod.

Signature: The signature is created by combining the encoded header, encoded payload, and a secret key using the specified algorithm. It ensures the integrity of the JWT and verifies that it hasn't been tampered with.

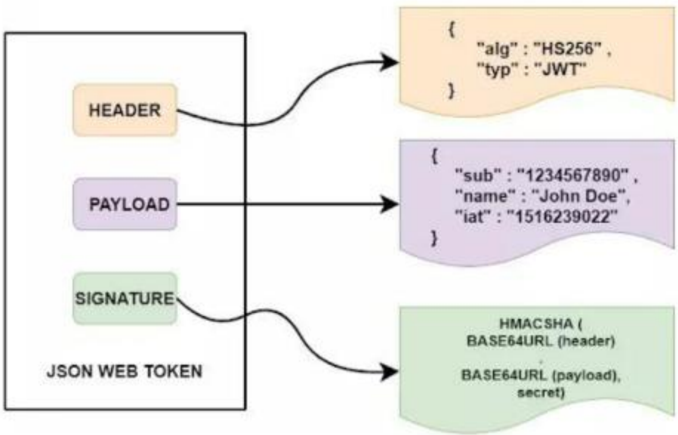
HEADER
PAYLOAD
SIGNATURE
JSON WEB TOKEN
{ "alg" : "HS256" , "typ" : "JWT" }
{ "sub" : "1234567890" , "name" : "John Doe", "iat" : "1516239022" }
HMACSHA (BASE64URL (header) BASE64URL (payload), secret)

Fig 3. Composition of JWT

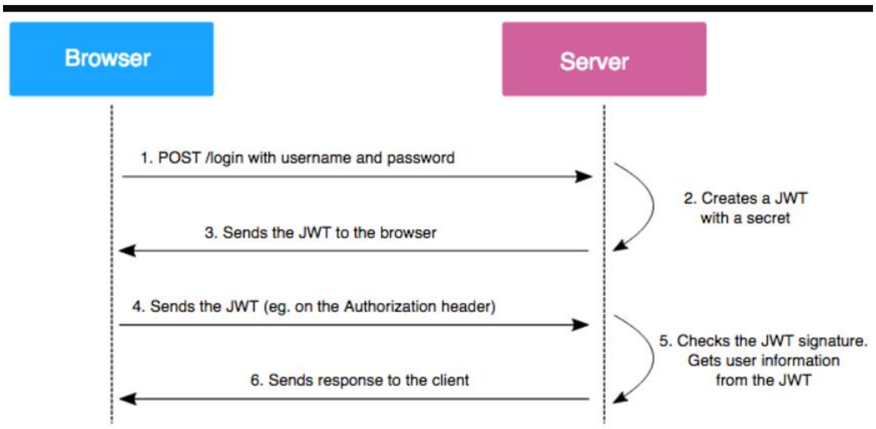
Browser
Server
1. POST /login with username and password
2. Creates a JWT with a secret
3. Sends the JWT to the browser
4. Sends the JWT (eg. on the Authorization header)
5. Checks the JWT signature. Gets user information from the JWT
6. Sends response to the client

Fig 4. Basic JWT Timing

II.LITERATURE REVIEW

Raghu Vamsi and Agrah Jain (2021) [1] explores the practical security testing of electronic commerce web applications, with a specific emphasis on vulnerability assessment and penetration testing. The authors highlight the importance of security in e-commerce platforms, especially in light of the increasing cyber attacks and vulnerabilities faced by these websites. The paper discusses various tools and methods used for security testing, as well as the identification of common vulnerabilities such as authentication issues, HTTP parameter pollution, XSS, CSRF, and IDOR. Additionally, the authors propose countermeasures to address these vulnerabilities and emphasize the need for manual testing in modern applications. Overall, the paper provides a comprehensive overview of the challenges and strategies related to security testing in the context of e-commerce web applications.

Subramanian [2] delineates a security token architecture tailored for smart contract platforms like Ethereum. Contrasting conventional centralized control mechanisms, the paper highlights security tokens' advantages, including minimal transaction time and cost, enhanced security via smart contracts, and transparent information flow. Challenges such as non-standardization in layered stack components and scalability concerns on platforms like Ethereum are acknowledged. Legal and regulatory uncertainties surrounding crypto-tokens may hinder adoption, though initiatives in jurisdictions like Zug and Malta aim to establish conducive regulatory frameworks. Overall, the paper contributes insights into the potential of security tokens while addressing challenges in their adoption within the evolving landscape of blockchain-based financial instruments.

Syabdan Dalimunthe, Joeharsyah Reza and Asep Marzuki (2022) [3] provided a literature review on various studies related to reducing security threats on web services. It discusses the application of JSON Web Token (JWT) for authentication, the use of HMAC algorithm, and the comparison of token-based authentication performance. Additionally, the paper explores the implementation of RESTful web services as a technology to realize interoperability and the security feature model of JWT on web service in mode system authentication. The literature review also covers the development of web service data management systems using REST API with access tokens and the comparison of SOAP and REST based web services.

Hagui, Msolli, Helali, and Hassen [4] introduce a novel authentication method comprising two phases. Initially, they employ a merge algorithm to create a secure pattern using image and password features. Subsequently, this pattern is applied to develop an authentication system ensuring information security and user location authenticity. The proposed approach accommodates both centralized and decentralized architectures across various applications. Security recommendations for biometric authentication methods are also provided, aiding developers and designers in crafting reliable systems. Users can leverage the image authentication technique in conjunction with blockchain for system access, meeting established criteria through blockchain and encryption technologies. This structure addresses data pattern loss concerns while enhancing security standards in identity verification processes. The study delves into additional aspects of this multifaceted issue.

Vijay and Indumathi [5] present a novel approach for multimodal biometric recognition utilizing the DBN-based CEWA model. Their proposed method combines CSO and EWA to design CEWA. Initially, ear, iris, and finger vein images undergo preprocessing. Feature extraction is then performed for each modality. The iris image is processed using the HT and Daughman's rubber sheet model for segmentation, while the ear image undergoes adaptive thresholding, and the finger-vein image is preprocessed using radon transform. Texture features are extracted from preprocessed images, and bicomp features are obtained based on the BiComp mask. Recognition scores are calculated using the Multi-SVNN classifier for each trait, and these scores are input to the DBN for final recognition.

Panguluri, Lakshmy, and Srinivasan [6] tackle data management challenges in the cloud by proposing a multi-factor authentication and verification scheme for key-aggregate searchable encryption. Their research integrates time-based one-time passwords and Inter Planetary File System (IPFS) to enhance data security. By addressing concerns surrounding data management in the cloud and introducing innovative security measures, their work contributes to the ongoing discourse on safeguarding sensitive information in cloud-based environments.

H Mehraj et al, [7] explores the use of the Protective Motivation Theory (PMT) and multi-factor authentication to enhance online safety and security in social networking sites, with the inclusion of new components improving the explanatory power of the model.

N. Alomar et al, [8] presents a detailed analysis and social authentication schemes, including their features, attacks, defenses, and opportunities,to guide research in leveraging users' social interactions for authentication purposes.

P. Varalakshmi et al, [9] proposed novel approaches to address scalability and efficiency challenges in JSON Web Token (JWT) authentication within Software-Defined Networking (SDN) environments. By implementing dynamic keys and removing manual blacklist management, the system enhances security and simplifies key management processes. These innovations promise to bolster authentication mechanisms in SDN architectures, ensuring robustness and adaptability to evolving security requirements.

Kannan et al. (2018) [10] present "Secure External Login Based on Authorization Code Flow using JWT," outlining the implementation of JWT for secure external login. However, Rawat et al. (2020) highlight potential vulnerabilities in web applications, including CSRF attacks leveraging JWT storage security. They stress the importance of meticulous system planning to address challenges such as misconfigurations in "set-httponly" attributes, emphasizing the need for secure cookie practices and broader CSRF mitigation strategies in JWT-based authentication systems.

Bonneau et al. (2012) [11] lay the groundwork for evaluating web authentication schemes, setting the stage for advancements like the streamlined JWT implementation described in the present paper. This implementation focuses on optimizing user authentication through reduced server overhead, minimized database calls, and efficient token management. By addressing challenges in technology integration and token management complexities, the system achieves improved server efficiency, secure token storage, and effective invalidation methods, thereby enhancing authentication and authorization processes. Ongoing refinement is necessary to ensure practical applicability in real-world scenarios.

Nugraha et al. (2023) [12] conduct a comparative research study on the performance and security aspects of JSON Web Tokens (JWT) and Platform Agnostic Security Tokens (PASETO) in RESTful APIs. Their findings reveal that while PASETO exhibits longer token generation and transfer times compared to JWT, it offers higher security levels by defending against the top three OWASP 2019 API attacks. However, acquiring diverse and representative data for both JWT and PASETO poses a challenge, necessitating a comprehensive range of samples to ensure broad applicability.

Mansur et al. (2023) [13] employ a mixed methodology to evaluate token-based authentication for enhancing Single Sign-On (SSO) security across systems. Through tests involving token changes and scanning for sniffing attacks, they demonstrate the potential of JSON Web Tokens (JWT) in improving system security. However, they suggest further research to explore alternative algorithms for JWT signatures, aiming to identify the most optimal algorithm from various choices. Additionally, they propose the use of multiple JWTs in SSO to potentially enhance security further. These findings underscore the importance of ongoing research and development efforts in advancing authentication mechanisms for improved system security.

Osman Salem, Khalid Alsubhi, Aymen Shaafi, Mostafa Gheryani, Ahmed Mehaoua, and Raouf

Boutaba (2021)[14] explored the ways of preventing Man-in-the-Middle (MitM) attacks in healthcare applications using wireless sensor networks. They provided a method to reduce energy consumption for data transmission, prevent unauthorized access to health data, and mitigate modification and replay attacks. They also conducted a comprehensive analysis and comparative study to analyze the performance of the proposed approach. Additionally, they investigated the impact of MitM attacks and evaluated the effectiveness of the proposed security measures.

Compagna et al.[15] explored the efficacy of SameSite cookies in thwarting Cross-site Request Forgery (CSRF) attacks. It delves into the evolution of SameSite cookies, their adoption by major browsers, and their default enforcement from 2020 onwards. Findings suggest SameSite cookies are effective in preventing CSRF in certain scenarios, notably for same-site state-changing actions post-authentication. However, they may not cover all CSRF attack variants. The survey underscores the need for further research to bolster CSRF defense strategies and enhance the effectiveness of SameSite cookies.

R Xu. [16] focused on bolstering transaction security amidst the vulnerability of SMS verification codes in mobile payment scenarios. The abstract highlights the risk of SMS code exposure due to hacker intrusion or device loss, proposing a multi-digit security code solution leveraging cryptographic technologies. This enhancement aims to safeguard financial transactions even if the SMS code is compromised. The conclusion emphasizes the simplicity and effectiveness of the 3-digit security code addition, ensuring user privacy and minimizing fund loss. The solution is lauded for its practicality, cost-effectiveness, and seamless integration with existing systems, promising heightened security and user-friendly mobile payment experiences.

Fan et al. [17] introduced a novel approach to enhance user access control in power marketing systems. By utilizing the Role-Based Access Control (RBAC) model and integrating trust attributes, the method improves access strategy expression and assigns user rights based on access levels. The results indicate that this approach reduces access control time, lowers the error rate of authority allocation, and enhances user satisfaction. This method offers a promising solution for efficient and effective user authority distribution in power marketing systems, addressing the challenges of traditional approach.

III.PROPOSED SYSTEM

Our proposed solution offers a robust blend of performance and security enhancements tailored for Web 3.0 applications. Central to our approach is the integration of OTP (One-Time Password) authentication within JWT (JSON Web Token) tokens, ensuring each login session's uniqueness. By embedding the OTP within the JWT, we fortify the token's validity to the specific login session, bolstering security measures. We shorten the token's lifespan to a mere 10 post-login, minimizing the window of vulnerability.

Complementing this strategy is the implementation of RBAC (Role-Based Access Control), which restricts unauthorized users from accessing critical endpoints housing sensitive resources. Upon user logout, our system promptly revokes the token, regardless of its remaining lifespan, imbuing it with session-like behavior while retaining statelessness.

Additionally, our solution incorporates measures to prevent simultaneous logins, ensuring that no two individuals can access the same account concurrently. This proactive defense mechanism complicates attackers' efforts to orchestrate malicious activities, as users can swiftly reclaim control over their accounts without resorting to password resets or external support.

While our implementation currently utilizes Email OTP for its practicality, we acknowledge SMS OTP as a more secure alternative. Leveraging technologies such as ReactJS for the frontend, SpringBoot for the backend, MySQL for database management, and Metamask for blockchain-wallet integration, our solution offers a comprehensive security framework poised to fortify Web applications against emerging threats. Refer Fig 5

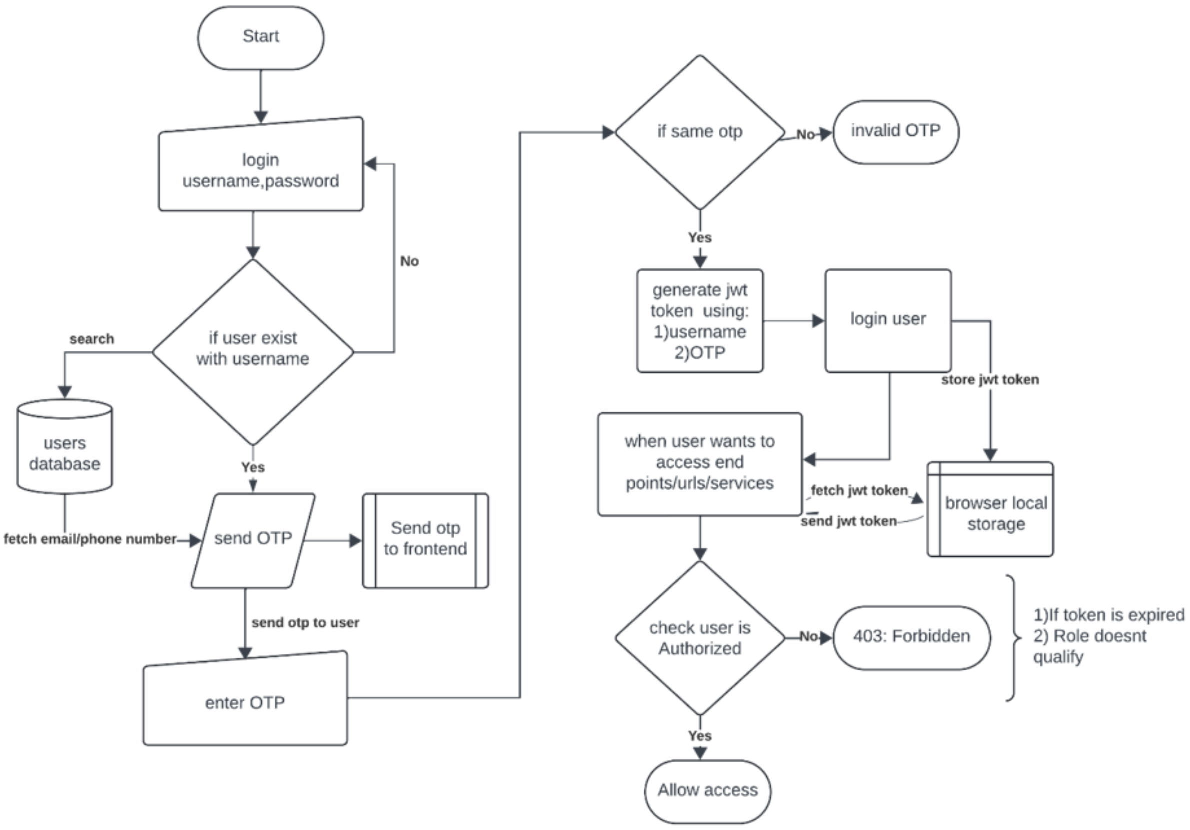

Fig 5. MFA using OPT and JWT token flowchart

IV.RESULTS

1. Enhanced Security: Integration of multifactor authentication (MFA) with JWT significantly improves security by adding an extra layer of verification beyond traditional username/password authentication.

2. Reduced Vulnerabilities: Implementing JWT with MFA mitigates common security risks such as phishing attacks, credential theft, and session fixation, enhancing overall system resilience against unauthorized access.

3. Improved User Experience: The use of JWT tokens and MFA mechanisms provides a seamless and user-friendly authentication experience, minimizing friction while ensuring robust security measures are in place.

4. Scalability: Stateless nature of JWT along with MFA support allows for scalability in distributed systems, facilitating efficient authentication across multiple services without the need for server-side session storage.

5. Streamlined Development: Integration of JWT and MFA functionalities into the application architecture streamlines development processes, leveraging existing libraries and frameworks to implement secure authentication mechanisms efficiently.

Overall, the implementation of JWT yields a more secure, scalable, and user-friendly authentication solution for web applications, bolstering confidence in data protection and access control.

FAMILY	DIGITAL SIGNATURE	KEY SIZE (Bits)	TIME (s)	STORAGE (kb)
SHA	HS256	256	0.06755	0.15527
	HS384	384	0.05841	0.17578
	HS512	512	0.06159	0.19726
ECC	ES256	256	0.071419	0.19726
	ES384	384	0.064285	0.23828
	ES512	512	0.064886	0.28515

Table 6 : SHA and ECC Signed JWT comparsion

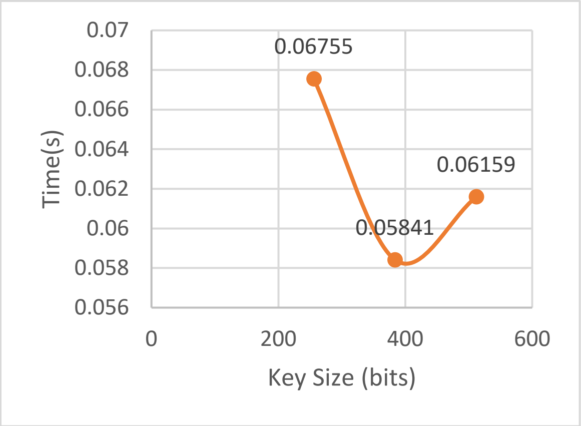

Fig 6a :Time(s) for SHA signed JWT

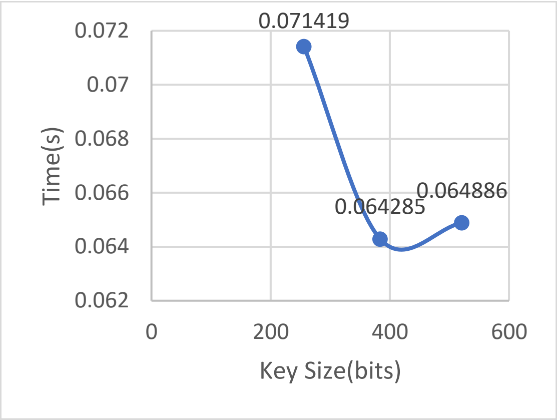

Fig 6b: Time(s) for ECC signed JWT

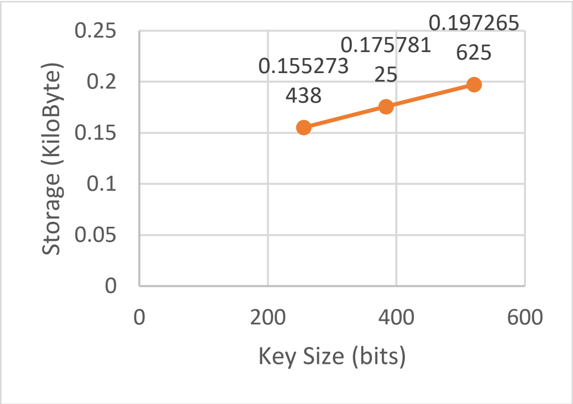

Fig 6c: Storage(kb) for SHA signed JWT

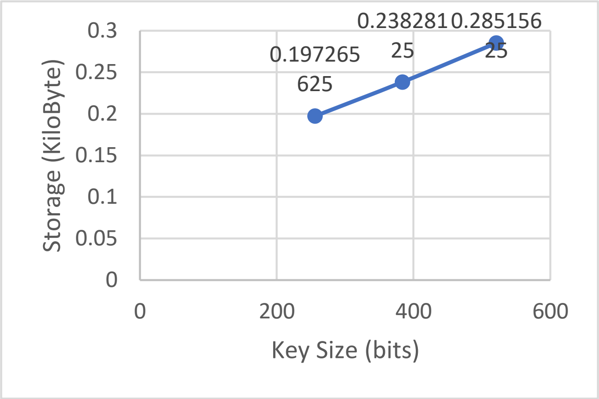

Fig 6d: Storage(s) for ECC signed JWTFig

FAMILY	DIGITAL SIGNATURE	KEY SIZE (Bits)	TIME (s)	STORAGE (kb)
RSA	RS256	2048	0.23371	0.44726
		4096	0.82741	0.78027
		8192	19.2896	1.44726
	RS384	2048	0.49365	0.47070
		4096	1.38743	0.80371
		8192	8.43310	1.47070
	RS512	2048	0.17452	0.44726
		4096	1.58504	0.78027
		8192	14.7007	1.44726

Table 2: RSA family signed JWT

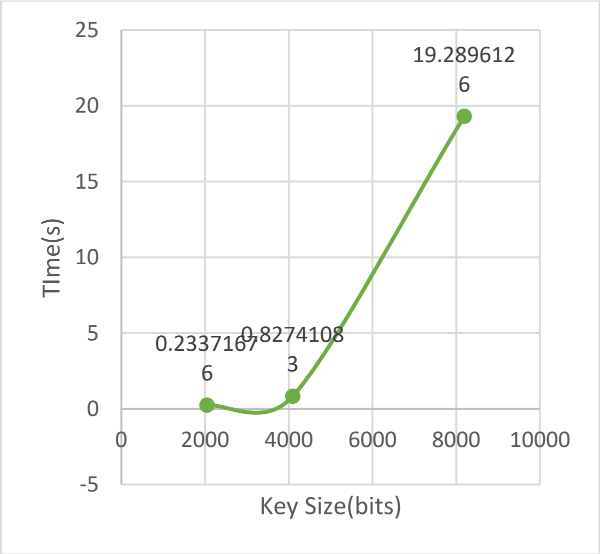

Fig 7c: Time(s) RS256 based JWT

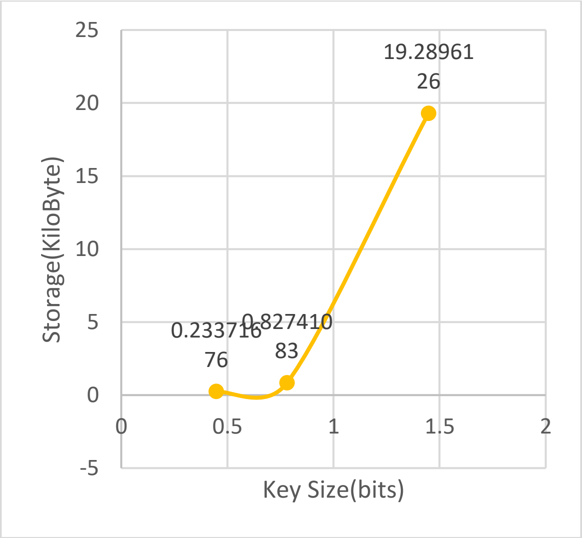

Fig 7d: Storage(kb) for R256 signed JWT

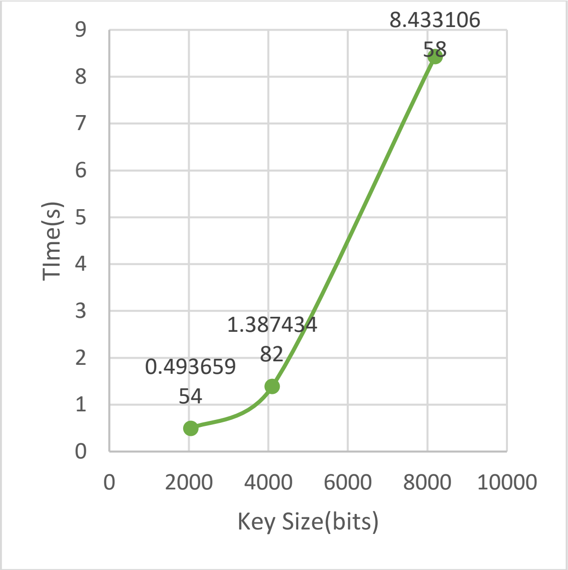

Fig 7c: Time(s) RS384 based JWT

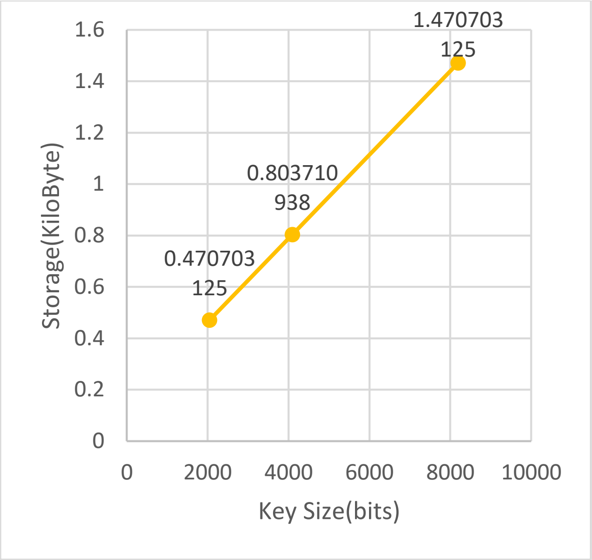

Fig 7d: Storage(kb) for RS384 signed JWT

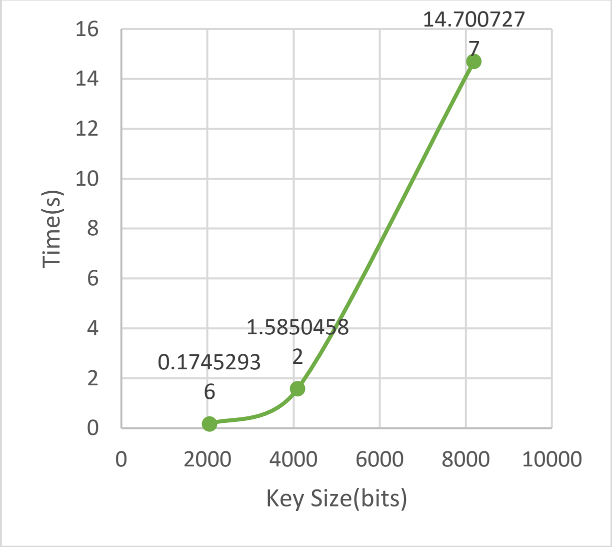

Fig 7c: Time(s) RS512 based JWT

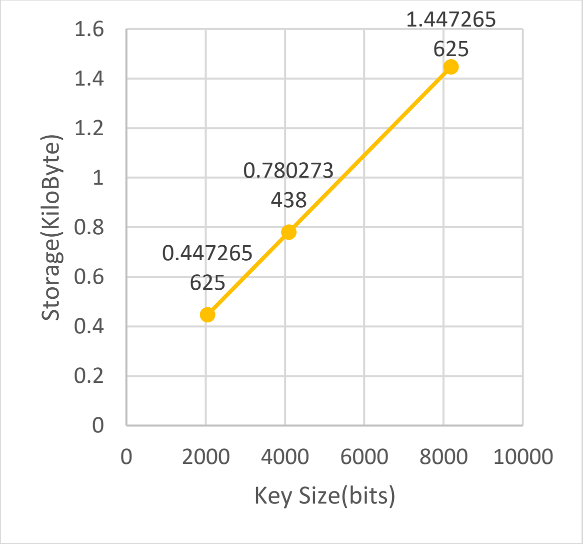

Fig 7d: Storage(kb) for RS512 signed JWT

From the Table 1 and Table 2 the best digital Signature Algorithm for signing JWT is HS384 as it has the lowest time and size of the token is also low. But the size token and the time complexity of the Algorithm will increase as we add more and more claims to the token. For this research we have added only one claims that is for OTP.

Note: the Time calucated for each digital Signature Algorithm is an average of 5 values.

V.CONCLUSION

In conclusion, our implementation of HJWT (Hybrid JSON Web Token) combining MFA (Multi-Factor Authentication) with JWT (JSON Web Token) in web 3 applications(web applications using blockchain) has successfully addressed various security threats and performance compromises. By enhancing the JWT token with MFA, we have significantly improved the security of our system. Our future enhancements will focus on improving the HMAC algorithm for signing tokens, which will not only enhance performance but also strengthen the Trap Door Function, making our system more secure. However, challenges remain in key management, as rotating keys can increase complexity. Additionally, our model is susceptible to XSS (Cross-Site Scripting) attacks, so developers must implement defenses against XSS to fully utilize the potential of HJWT. We would also recommend to use SMS based OTP over Email based OTP. Overall, our model represents a significant advancement in web application security, but continued vigilance and adaptation to emerging threats are essential for maintaining its effectiveness.

VI.REFERNCE

[1]Vamsi, P. R., & Jain, A. (2021). Practical security testing of electronic commerce web applications. International Journal of Advanced Networking and Applications, 13(1), 4861-4873.

[2] Subramanian, H. (2020), "Security tokens: architecture, smart contract applications and illustrations using SAFE", Managerial Finance, Vol. 46 No. 6, pp. 735-748.

[3] Dalimunthe, S., Reza, J., & Marzuki, A. (2022). The Model for Storing Tokens in Local Storage (Cookies) Using JSON Web Token (JWT) with HMAC (Hash-based Message Authentication Code) in

E-Learning Systems. Journal of Applied Engineering and Technological Science (JAETS), 3(2), 149-155.

[4] I. Hagui, A. Msolli, A. Helali and F. Hassen, "Based blockchain-lightweight cryptography techniques for security information: A verification secure system for user authentication," 2021 International Conference on Control, Automation and Diagnosis (ICCAD), Grenoble, France, 2021, pp. 1-5.

[5] M. Vijay, G. Indumathi,Deep belief network-based hybrid model for multimodal biometric system for futuristic security applications,Journal of Information Security and Applications,Volume 58,2021,102707,ISSN 2214-2126.

[6] Panguluri, S.D., Lakshmy, K.V., Srinivasan, C. (2022). Enabling Multi-Factor Authentication and Verification in Searchable Encryption. In: Sharma, D.K., Peng, SL., Sharma, R., Zaitsev, D.A. (eds) Micro-Electronics and Telecommunication Engineering . ICMETE 2021. Lecture Notes in Networks and Systems, vol 373.

[7] Haider Mehraj, D. Jayadevappa, Sulaima Lebbe Abdul Haleem, Rehana Parveen, Abhishek Madduri, Maruthi Rohit Ayyagari, Dharmesh Dhabliya,Protection motivation theory using multi-factor authentication for providing security over social networking sites,Pattern Recognition Letters,Volume 152,2021,Pages 218-224,ISSN 0167-8655.

[8] N. Alomar, M. Alsaleh and A. Alarifi, "Social Authentication Applications, Attacks, Defense Strategies and Future Research Directions: A Systematic Review," in IEEE Communications Surveys & Tutorials, vol. 19, no. 2, pp. 1080-1111, Secondquarter 2017, doi: 10.1109/COMST.2017.2651741.

[9] P. Varalakshmi, G. B, V. S. P, D. T and S. K, "Improvising JSON Web Token Authentication in SDN," 2022 International Conference on Communication, Computing and Internet of Things (IC3IoT), Chennai, India, 2022.

[10] R. Kannan and G. Umasankar, Secure External Login Based on Authorization Code Flow using JWT, pp. 961-965, 2018.S. Rawat, T. Bhatia and E. Chopra, "Web Application Vulnerability Exploitation using Penetration Testing scripts", Int. J. Sci. Res. Eng. Trends, vol. 6, no. 1, pp. 311-317, 2020

[11] Joseph Bonneau et al., "The quest to replace passwords: A framework for comparative evaluation of web authentication schemes", 2012 IEEE Symposium on Security and Privacy, 2012.

[12] A. F. Nugraha, H. Kabetta, I. K. S. Buana and R. B. Hadiprakoso, "Performance and Security Comparison of Json Web Tokens (JWT) and Platform Agnostic Security Tokens (PASETO) on RESTful APIs," 2023 IEEE International Conference on Cryptography, Informatics, and Cybersecurity (ICoCICs), Bogor, Indonesia, 2023

[13] E. S. Mansur, A. Rahmatulloh, R. N. Shofa and I. Darmawan, "AMAN: Token-based Authentication to Improved Single Sign-On Security Between Systems," 2023 International Conference on Advancement in Data Science, E-learning and Information System (ICADEIS), Bali, Indonesia, 2023

[14] Cite - Salem, O., Alsubhi, K., Shaafi, A., Gheryani, M., Mehaoua, A., & Boutaba, R. (2021). Man-in-the-Middle attack mitigation in internet of medical things. IEEE Transactions on Industrial Informatics, 18(3), 2053-2062.

[15] L. Compagna, H. Jonker, J. Krochewski, B. Krumnow and M. Sahin, "A preliminary study on the adoption and effectiveness of SameSite cookies as a CSRF defence," 2021 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW), Vienna, Austria, 2021, pp. 49-59.

[16] R. Xu, "Security Enhancement for SMS Verification Code in Mobile Payment," 2022 11th International Conference of Information and Communication Technology (ICTech)), Wuhan, China, 2022, pp. 3-7.

[17] L. Fan, T. Peng, R. Tian, Z. Liu, Y. Ni and L. Liu, "RBAC Model-Based User Authority Distribution Method of Power Marketing System," 2022 4th International Conference on Power and Energy Technology (ICPET), Beijing, China, 2022, pp. 969-974.