Journal of Engineering and Sustainable Development
Vol. 30, No. 05, August 2026
https://jeasd.uomustansiriyah.edu.iq/index.php/jeasd
ISSN 2520-0917
Research Article
https://doi.org/10.31272/jeasd.2706

A Lightweight 2-factor Authentication Scheme for Smart Homes

Ali T. Al-Hachami[1*], Mohammed F. Al-Gailani[2]

[1]Department of Information and Communications Engineering, College of Information Engineering, Al-Nahrain University, Baghdad, Iraq
[2]Department of Computer Networks Engineering, College of Information Engineering. Al-Nahrain University, Baghdad, Iraq

[*]Email: ali.thaer@coie-nahrain.edu.iq

Article Info

Received 22/05/2024
Revised 02/08/2026
Accepted 23/08/2026

Abstract

Smart home technology provides the ability for homeowners to remotely monitor and control home appliances by connecting them to the internet. Despite its benefits, many people around the world are reluctant to adopt smart devices into their home for security reasons. Namely, connecting smart homes to the internet opens an attack surface for hackers to gain access and possibly control the smart home. Also, most smart home devices have low computational power and therefore cannot provide complex security protocols. In that case, to achieve the goal of remote user authentication, this paper proposes a 2-factor authentication scheme that consists of local biometric authentication and remote authentication with user identity and a time-based one-time password (TOTP). The security of the scheme is validated with formal and informal security checks. The proposed system outperforms similar systems since its computation overhead is (0.02 ms) and its communication overhead is (416 bits). A possible avenue of future work may involve using this authentication scheme in wider IoT applications such as wireless sensor networks.

Keywords: Authentication, Home gateway, Scyther tool, Smart homes, Time-based one-time password

1. Introduction

One of the fastest-growing sectors in computing and information and communication technology (ICT) is smart homes, where everyday home devices (like doorbells, fridges, lights, etc.) are connected to the internet, allowing homeowners to control them remotely even when they are outside the home. This technology offers numerous benefits for the end user, increasing productivity and convenience [1]. A typical smart home design is shown in Fig. 1, consisting of smart appliances and a home gateway device that connects smart devices to external networks and maintains access control for the smart home's users [2]. Despite that, internet-connected homes still have a detrimental flaw: smart home devices have numerous vulnerabilities. Maintaining the security of smart homes is a complex task because smart devices are heterogeneous and resource-constrained; they cannot implement complex security protocols without a high cost to performance. Therefore, any security scheme that will be implemented in smart homes needs to be lightweight and fast while maintaining adequate security of users' information[3]. In this paper, the problem of authenticating remote users of smart homes is addressed. When a user tries to connect to their home over a network connection, the home gateway validates their identity while maintaining the user's privacy. The proposed scheme for achieving authentication of remote smart home users is discussed in detail. In addition to that, the scheme is compared with previous schemes to ensure proper security and performance.

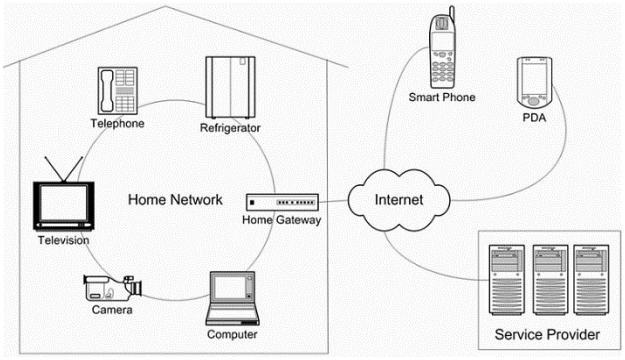

Figure 1. Typical smart home architecture.

In 2021, Oh et al[4] proposed an improved version of a smart grid-enabled smart home authentication scheme presented a year earlier for a situation-aware smart home. The improved scheme by Oh et al utilized a mobile device and home gateway where these devices needed to register with a registration

authority (RA). The RA would help the two devices exchange authentication information. After that, the mobile device and gateway can perform mutual authentication without the need for RA.

In the same year, Kaur and Kumar [5] proposed an improved version of a 2-factor authentication scheme originally proposed by Shuai et al. [6]. The improved scheme protected the user against password guessing attacks, replay attacks, and insecure session key agreement. The scheme uses elliptic curves for establishing session keys, identity, and password for the user. However, the scheme does not improve the performance of the previous one when it comes to resource consumption.

To ensure that the solution to the smart home authentication problem is future-proof, Tanveer et al. [7] discussed a remote user authentication system for smart homes that supports IPv6, known as 6LoWPAN (low-power wireless personal area network). The authors attempt to create an authenticated key exchange (AKE) solution using the SHA-256 algorithm and XOR operations. However, this solution seems to incur a high communication/computational cost relative to other smart home authentication schemes.

In a more non-traditional approach, Nimmy et al[8] proposed an authentication scheme that relies on using photo response non-uniformity (PRNU). In this scheme, the user sends an image of their face to the smart home gateway, which recognizes two things: the user's face and the mobile phone that sent the image. If both conditions are satisfied, then the user is authenticated. This authentication scheme, however, comes at the cost of user privacy and may be vulnerable to user impersonation attacks.

A common issue with remote user authentication methods is using a smart home gateway node to authenticate users and offload some of the computational cost of computing the session keys after authentication. However, this method suffers from two problems: first, the gateway is accessible from the public internet, which is undesired since leaving any device open to the internet means an attacker can try to access the gateway as well, which leads to the second problem. The gateway node is used to compute session keys between remote users and every smart home device, which is inefficient and increases the computational load on the home gateway.

The aim of this paper is to create a smart home authentication scheme where the user can authenticate himself once and access all smart home devices.

User authentication can be achieved using two steps: firstly, the user authenticates himself locally on the smartphone using a biometric fingerprint. Secondly, the user connects to a smart home gateway using a virtual private network VPN); the gateway authenticates the user's identity using a username and TOTP. Then, the user can communicate securely with the smart devices over the encrypted home Wi-Fi without having to compute additional session keys for each smart device and user. This scheme should be secure against well-known IoT vulnerabilities.

2. Threat Model

This paper will follow the Dolev and Yao threat model[9], which is a threat model that has the following assumptions about security:

- The user U_i and smart device SD_i can communicate over an insecure channel (the internet).
- Attackers can affect any communication over a public channel by capturing, modifying, deleting, or inserting any message.
- The attacker can either capture the smart device or the user identity and password but not both.
- The home gateway HG is assumed to be a trusted device since it only connects to the private smart home network.

3. Method of the research

Two entities are involved in the authentication process: the user mobile device and the smart home gateway. The authentication scheme involves two parts. The first is biometric fingerprint authentication[10]. The second is remote authentication using a randomly generated TOTP [11], which is a random sequence of 6-digit numbers that change every 30 seconds. This is achieved using a shared secret number and accurate time. The shared secret is combined with the current time, hashed, and then the hash is truncated to generate the random passwords[12]. After that, the user is fully authenticated and can send commands to the smart home gateway to control and monitor the smart home devices. Fig.2 shows the proposed system model.

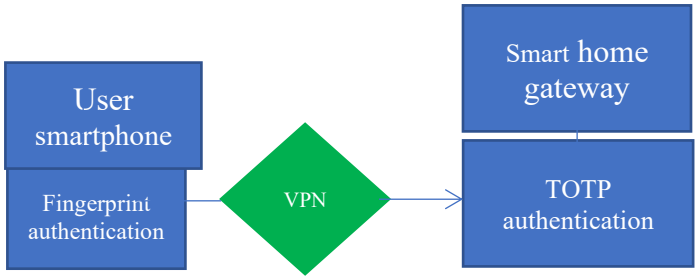

Figure 2. The Proposed System Model.

3.1 System Initialization Phase

The home gateway HG is first initialized and connected to the home network. After that, a secure communication channel (VPN) is established between HG and the user mobile smartphone US of the user U_i.heThe tworties (US and HG) exchange a symmetric encryption key using the Internet Key Exchange (IKE) protocol, which uses a Diffie-Hellman exchange to agree on a shared symmetric key. The two communication parties now have a secure connection over the Internet, and Alice (U_i) can communicate with Bob (H_G) remotely. Fig.3 describes the Diffie-Hellman exchange to create the session key of the encrypted VPN tunnel.

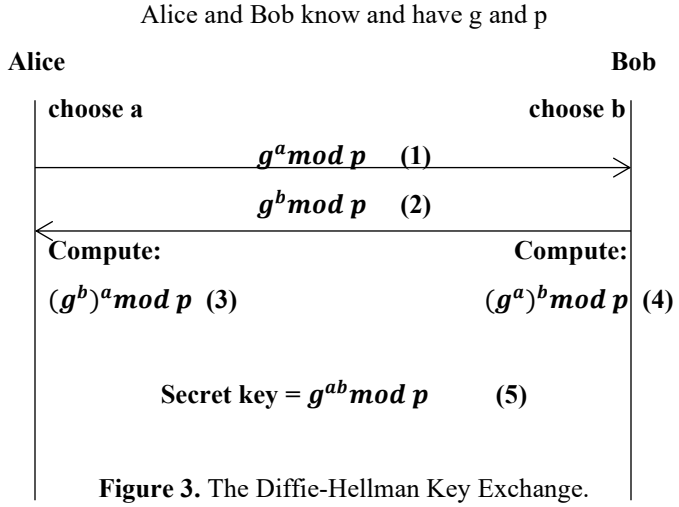

Figure 3. The Diffie-Hellman Key Exchange.

3.2 System Registration Phase

This process can take place either in the private home network or remotely through a secure VPN connection. The user chooses a unique identity UID_i for the home gateway to recognize that user. Then the HG generates a random secret S and shares that secret with the US. The shared secret will be used to generate the TOTP every 30 seconds by using S and the current time (in seconds) as input[13], both S and the current time are concatenated and hashed with SHA-1 cryptographic hashing algorithm[14][15], the 160-bit hash will be truncated into 24-bits to produce a random 6-digit password. This hashing process repeats every 30 seconds to generate new passwords.

3.3 System Authentication Phase

The user locally authenticates himself using the fingerprint biometric data B_i [16][17]. Then, the remote user connects to the smart home network via a secure VPN connection. Once in the home network, the user connects to the private IP address of the smart home gateway and enters the following: $E(UID_i||TOTP_U||T_1||N_1||C)$ where $T_1$is the time stamp, $N_1$is the nonce and C is the cryptographic checksum. And finally, the smart home gateway calculates the one-time password in the gateway $TOTP_{gwn}$ and checks if it equals the one-time password sent by the user $TOTP_U$, if $TOTP_U = TOTP_{gwn}$. the two passwords are equal and the user is fully authenticated by the gateway and have the ability to send commands to the gateway to control the smart home devices SD_i[18][19].

As mentioned in, the TOTP is generated as follows:

TOTP = $H(S \oplus TC)$ where S is the secret value shared between the user and gateway in the registration phase, and TC is the time counter given by:

$TC = (TIME - TO)/TS$, where TO refers to epoch start and TS refers to the time step amount in seconds (30 seconds). The values of the time counter and secret value are hashed using the SHA-1 algorithm, and the message digest is truncated into 24 bits (i.e., a 6-digit password).

4. Security Analysis

To ensure that the proposed scheme is protected against attacks on smart homes, an extensive security analysis is conducted. This analysis involves two separate steps: an informal analysis, which examines the most common attacks smart homes encounter and discusses how the proposed scheme protects against each attack individually. The formal security check relies on a tool called scyther[20]. This Python-based security analysis tool tests the security of Internet protocols and discovers potential vulnerabilities.

4.1 Informal Security Analysis

This section discusses how the proposed system can protect against common smart home vulnerabilities.

4.1.1. Impersonation Attacks

If we assume that an attacker has already stolen valid credentials to log in as a legitimate user. TOTP is changed every 30 seconds, which makes the stolen credentials invalid.

4.1.2. Replay Attacks

Assuming the adversary tries to replay the encrypted authentication message, the messages are protected against replay attacks since each message contains a nonce. Thus, the gateway would drop any replayed messages from the attacker.

4.1.3. Man-in-the-Middle Attack

If an attacker successfully intercepts and modifies message content, the gateway would detect that the message integrity has been violated since the modified message would not generate the same checksum that was sent with the message.

4.1.4. Insider Attack

If the attacker is physically inside the home, he would not gain any advantage to access the gateway since he still needs the TOTP. Assuming the attacker wants to access the secret key used to generate the TOTP, he would need to log in to view that secret key.

4.1.5. Stolen Smartphone Attack

Stealing the user's smartphone gives no valuable information to the attacker since the smartphone is protected using biometric fingerprint authentication.

4.1.6. Denial Of Service Attack

Since the home gateway is on the private home network and can only be accessed through a VPN, no denial of service attacks can be performed since the gateway does not have any open ports to the internet.

4.1.7. Dictionary Attack

Assuming the attacker can access the gateway and use a dictionary attack to find the current password, it is not possible since the TOTP changes after 30 seconds. However, assuming the attacker has a hardware device powerful enough to check every combination of 6-digit passwords in less than 30 second, it would still not be feasible since the gateway rate limits the number of attempts for each password.

4.2 Automated Security Verification

The scyther tool will be used to check if the proposed scheme has any vulnerabilities that an attacker could exploit to break the cryptographic algorithms of the scheme. In order to ensure the security of the scheme, a test was run on the Scyther tool for the authentication process, and the results in Fig.4 show that the scheme proposed in this paper does not have any vulnerabilities

Figure 4. Verifying Security Using Scyther.

5. Results and Discussion

5.1 Resource Consumption

The computational and communication costs of the smart home system are compared with those of previous smart home systems to assess whether the proposed scheme can provide better performance while incurring lower hardware costs. The most similar schemes used for the comparison process are: Oh et al [4], Kaur and Kumar [5], Tanveer et al [7], and Nimmy et al [8]. The hardware used includes: a Huawei Y9 Prime for mobile users, which has an octa-core processor, 4 GB of RAM, and a fingerprint sensor. For the smart home gateway, a Raspberry Pi 3B+ is used, which has a Broadcom BCM2837B0 1.4 GHz processor and 1 GB of RAM.

5.2 Communications Cost Comparison

The communications overhead cost of the data being sent from the user smartphone to the gateway, based on assumptions of the cost, consists of: UID is 128 bits, T is 32 bits, N is 128 bits, the plaintext/ciphertext block size (which uses the AES-GCM algorithm) is 128 bits, and the cryptographic checksum (SHA-256) is 256 bits. Table 1. Demonstrates the comparison of the communication overhead to similar schemes, in which the proposed scheme achieves a more efficient cost.

Note that in Table 1. Similar to previous works, the communication cost comparison includes the overhead cost of the authentication messages only, since we are only interested in the messages that authenticate the user.

Table1. Communication Cost Comparison.

Scheme	Total messages	Total cost (bits)
Oh et al [4]	5	2080
Kaur and Kumar [5]	3	1856
Tanveer et al. [7]	3	1440
Nimmy et al. [8]	1	544
Proposed scheme	3	416

5.3 Security Feature Comparison

The security of the proposed scheme is compared with the similar schemes mentioned previously. The comparison displays that the new smart home system is not vulnerable to any of the listed attacks, while previous schemes are vulnerable to at least one of the following attacks mentioned in Table 2.

Table 2. Comparison of vulnerability to attacks

Security feature	Oh [4]	Kaur and Kumar [5]	Tanveer [7]	Nimmy [8]	Proposed scheme
Impersonation attack	No	No	No	Yes	No
Replay attack	No	No	No	No	No
Man-in-the-middle	No	Yes	No	No	No
Insider attack	No	No	No	No	No
Stolen smartphone attack	No	No	No	Yes	No
DDOS attack	Yes	Yes	Yes	Yes	No
Dictionary attack	No	No	Yes	No	No

5.4 Cryptographic Overhead Cost

Measuring the cryptographic overhead consists of recording CPU cycles and time that is required for executing the cryptographic operation of that scheme. In this instance, the cryptographic algorithms used are found in the TLS protocol, which is utilized for sending sensitive information on the internet. TLS uses an AES-128-bit cipher in GCM mode for encrypting data in transit and uses SHA-256 as a checksum. In

addition to that, generating the TOTP every 30 seconds, using SHA-1, also needs to be accounted for. However, the cost of generating the initial secret key used in the TOTP will be ignored since this operation only occurs once in the registration phase. The software used for calculating the computational cost is the CRYPTO++ benchmarking library. Table 3 shows the computational cost incurred by each individual algorithm used in the proposed scheme.

Table 3. Time Needed for Each Cryptographic Algorithm.

Notation	Cryptographic operation	Time (ms)
T_h/T_{mac}	Hashing (SHA-256)	0.002
T_{ed}	Symmetrical encryption/decryption (AES 128 bits)	0.001
T_{otp}	OTP (SHA-1)	0.001

The computational cost will be as follows: $6T_{ed} + 6T_{mac}$ is the cost of authentication between the end user and the gateway, which costs 0.018 ms. The TOTP will be generated simultaneously in the mobile phone and home gateway with a cost of 0.002 ms. The final cost to compute this authentication is 0.02 ms. Table 4 shows that the proposed scheme achieves the desired authentication with a lower computational cost to the smartphone and home gateway device. Note that the computational cost of the proposed scheme is negligibly higher than [6] (only 0.004 ms); the proposed scheme achieves better security features. Thus, the proposed scheme manages to outperform all related schemes and is practical for real smart home environments.

Table 4. Computational Cost Comparison.

Scheme	Total time (ms)
Oh et al [4]	0.0168
Kaur and Kumar [5]	1.366
Tanveer et al [7]	7.6
Nimmy et al. [8]	1.28
proposed scheme	0.02

6. Conclusion

Smart homes are fundamentally insecure due to their reliance on devices with limited computational power; maintaining a secure environment in a smart home while still being user-friendly is a challenging problem. This paper proposes a scheme for reliably verifying users with multifactor authentication. This is achieved by a smart gateway that acts as a middleman between users and the smart environment, which handles the cryptographic verification and the networking required to establish a secure connection, achieving a processing overhead of (0.02 ms) while maintaining a low communication cost of (416 bits in total) .the system has received a vulnerability assessment against many vulnerabilities that infect smart devices and was found to be resistant to all of them. In terms of performance of the system, the proposed scheme has been compared with many similar schemes for authenticating smart home users and the results show that the proposed scheme outperforms previous schemes by a large margin. This concludes that the new proposed system creates a safer environment for smart home users while isolating smart home devices from accessing the internet directly. Making the scheme suitable for real world application and can be implemented using simple hardware devices such as a raspberry pi 3. a direction that future work may follow is applying this security scheme on a wider range of applications of IoT, such as using it in wireless sensor networks, for instance, sensor nodes are deployed in remote locations (power plants, gas pipelines and jungles) where maintaining security can prove difficult and expensive. Creating a gateway node to authenticate any user trying to connect to the sensor network can prove very useful.

List of Abbreviations.

Abbreviation	Meaning
VPN	Virtual private network
TOTP	One-time password (time based)
$TOTP_U$	TOTP of the user
$TOTP_{gwn}$	TOTP of the gateway node
U_i	the ith user
SD_i	The ith smart device
HG	Home gateway
US	User smartphone
IKE	Internet key exchange
UID_i	The ith user identity
S	Random secret value
E	Encryption
T_i	The ith time stamp
N_i	The ith nonce
C	Cryptographic checksum
H	Hashing
TC	Time counter
TO	Start of an epoch
TS	Time step
T_h	Time needed for hashing
T_{mac}	Time needed for generating message authentication code
T_{ed}	Time needed for encryption/decryption
T_{otp}	Time needed for generating TOTP

Conflict of interest

The authors declare that there are no conflicts of interest regarding the publication of this manuscript.

Author Contributions

Ali T. Al-Hachami developed the theory and performed the computation and analysis of the results.

Mohammed F. Al-Gailani proposed the research problem and provided general guidelines for the research.

Both authors discussed the results and agreed on the final script.

References

[1] H. Touqeer, S. Zaman, R. Amin, M. Hussain, F. Al-Turjman, and M. Bilal, "Smart home security: challenges, issues and solutions at different IoT layers," *Journal of Supercomputing*, vol. 77, no. 12, pp. 14053–14089, Dec. 2021, doi: https://doi.org/10.1007/s11227-021-03825-1

[2] Y. Meng, W. Zhang, H. Zhu, and X. S. Shen, "Securing Consumer IoT in the Smart Home: Architecture, Challenges, and Countermeasures," *IEEE Wirel Commun*, vol. 25, no. 6, pp. 53–59, Dec. 2018, doi: https://doi.org/10.1109/MWC.2017.1800100

[3] R. Yu, X. Zhang, and M. Zhang, "Smart Home Security Analysis System Based on the Internet of Things," in *2021 IEEE 2nd International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering, ICBAIE 2021*, Institute of Electrical and Electronics Engineers Inc., Mar. 2021, pp. 596–599. doi: https://doi.org/10.1109/ICBAIE52039.2021.9389849

[4] J. Oh, S. Yu, J. Lee, S. Son, M. Kim, and Y. Park, "A secure and lightweight authentication protocol for IoT-based smart homes," *Sensors*, vol. 21, no. 4, pp. 1–24, Feb. 2021, doi: https://doi.org/10.3390/s21041488

[5] D. Kaur and D. Kumar, "Cryptanalysis and improvement of a two-factor user authentication scheme for smart home," *Journal of Information Security and Applications*, vol. 58, May 2021, doi: https://doi.org/10.1016/j.jisa.2021.102787

[6] M. Shuai, N. Yu, H. Wang, and L. Xiong, "Anonymous authentication scheme for smart home environment with provable security," *Comput Secur*, vol. 86, pp. 132–146, Sep. 2019, doi: https://doi.org/10.1016/j.cose.2019.06.002

[7] M. Tanveer, G. Abbas, Z. H. Abbas, M. Bilal, A. Mukherjee, and K. S. Kwak, "LAKE-6SH: Lightweight User Authenticated Key Exchange for 6LoWPAN-Based Smart Homes," *IEEE Internet Things J*, vol. 9, no. 4, pp. 2578–2591, Feb. 2022, doi: https://doi.org/10.1109/JIOT.2021.3085595

[8] K. Nimmy, S. Sankaran, K. Achuthan, and P. Calyam, "Lightweight and Privacy-Preserving Remote User Authentication for Smart Homes," *IEEE Access*, vol. 10, pp. 176–190, 2022, doi: https://doi.org/10.1109/ACCESS.2021.3137175

[9] D. Dolev and A. Yao, "On the security of public key protocols," *IEEE Trans Inf Theory*, vol. 29, no. 2, pp. 198–208, 1983, doi: https://doi.org/10.1109/TIT.1983.1056650

[10] S. Dargan and M. Kumar, "A comprehensive survey on the biometric recognition systems based on physiological and behavioral modalities," *Expert Syst Appl*, vol. 143, p. 113114, 2020, doi: https://doi.org/10.1016/j.eswa.2019.113114

[11] I. Gordin, A. Graur, and A. Potorac, "Two-factor authentication framework for private cloud," in *2019 23rd International Conference on System Theory, Control and Computing (ICSTCC)*, 2019, pp. 255–259. doi: https://doi.org/10.1109/ICSTCC.2019.8885460

[12] S. Babkin and A. Epishkina, "Authentication Protocols Based on One-Time Passwords," in *2019 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus)*, 2019, pp. 1794–1798. doi: https://doi.org/10.1109/EIConRus.2019.8656839

[13] C. Sudar, S. K. Arjun, and L. R. Deepthi, "Time-based one-time password for Wi-Fi authentication and security," in *2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, 2017, pp. 1212–1216. doi: https://doi.org/10.1109/ICACCI.2017.8126007

[14] D. Eastlake 3rd and P. Jones, "US Secure Hash Algorithm 1 (SHA1)," *RFC 3174*, Sep. 2001, doi: https://doi.org/10.17487/RFC3174

[15] Z. Al-Odat, A. Abbas, and S. U. Khan, "Randomness Analyses of the Secure Hash Algorithms, SHA-1, SHA-2 and Modified SHA," in *2019 International Conference on Frontiers of Information Technology (FIT)*, 2019, pp. 316–3165. doi: https://doi.org/10.1109/FIT47737.2019.00066

[16] A. J. Mohamed Abdul Cader, J. Banks, and V. Chandran, "Fingerprint Systems: Sensors, Image Acquisition, Interoperability and Challenges," *Sensors*, vol. 23, no. 14, Jul. 2023, doi: https://doi.org/10.3390/s23146591

[17] J. Priesnitz, C. Rathgeb, N. Buchmann, C. Busch, and M. Margraf, "An overview of touchless 2D fingerprint recognition," Dec. 01, 2021, *Springer Science and Business Media Deutschland GmbH*. doi: https://doi.org/10.1186/s13640-021-00548-4

[18] H. Seta, T. Wati, and I. C. Kusuma, "Implement Time-Based One-Time Password and Secure Hash Algorithm 1 for Security of Website Login Authentication," in *2019 International Conference on Informatics, Multimedia, Cyber and Information System (ICIMCIS)*, 2019, pp. 115–120. doi: https://doi.org/10.1109/ICIMCIS48181.2019.8985196

[19] D. M'Raihi, S. Machani, M. Pei, and J. Rydell, "TOTP: Time-Based One-Time Password Algorithm," *RFC 6238,* May 2011, doi: https://doi.org/10.17487/RFC6238

[20] C. J. F. Cremers, "The Scyther Tool: Verification, Falsification, and Analysis of Security Protocols," *Lecture Notes in Computer Science*, vol. 5123, pp. 414–418, 2008, doi: https://doi.org/10.1007/978-3-540-70545-1_38