JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 3, 2022, p. 4004-4009
https://publishoa.com
ISSN: 1309-3452

A Light Weight Self-Adaptive Honey Encryption for User Authentication Scheme in Cloud-IoT Health Services

P. Renuka[1], Dr. B. Booba[2]

Research Scholar, VISTAS, Vels University, Chennai

renuka.paramasivam@gmail.com

Professor, Department of CSE, VISTAS, Vels University, Chennai

boobarajashekar@gmail.com

Received 2022 March 15; **Revised** 2022 April 20; **Accepted** 2022 May 10.

Abstract

The recent advancement in cloud computing and Internet of Things (IoT) paradigms, paves a way to support the patients through remote monitoring and they can enjoy their health services at home. The medical professional (MP) needs to analyze IoT health data or Electronic Health Records (EHR) from a cloud server using their authorized login credentials. However, the Cloud-IoT network is vulnerable to many malicious attacks, it is necessary to make strong user authentication. Therefore, strong user authentication is a prerequisite for a successful global deployment of centralized healthcare systems. In this paper, we present a secure, efficient authentication protocol to allow MPs to access patient data through a Cloud IoT network. The proposed protocol contains Light Weight Self-Adaptive Honey Encryption (LWSHE) for securing the Internet of Things (IoT) medical data in a cloud server. The traditional Honey encryption algorithm has a message space limitation problem which is overcome through a discrete distribution function in Distribution Transforming Encoder (DTE) which maps the plain text strings into the seed strings. In the honey encryption algorithm, our proposed honeyword generation method can eliminate storage overhead and typo safety problems. Based on the result of the security and performance comparison analysis, the proposed protocol does not only prevents the vulnerable attacks from being performed but also achieves more complex security operations and efficient user authentication.

Keywords: cloud computing, Internet of Things (IoT), Electronic Health Records (EHR), Light Weight Self-Adaptive Honey Encryption (LWSHE), Distribution Transforming Encoder (DTE).

1. Introduction

Internet of Things (IoT) involves multiple physical sensors/devices/virtual objects communicating over a public network. The physical objects may include sensors, smart devices, and the virtual objects may include health records, wallets, etc. In essence, IoT aims to improve environmental and social systems accuracy by utilizing a computer-based system. The connected objects or things are intelligent enough to take ingenious decisions without human interference. In healthcare applications, the security and privacy of patients' data are among the biggest concerns. To overcome these issues, medical professional authentication has to be protected strongly with algorithms. To secure the medical data Password-Based Encryption (PBE) algorithm is utilized but it considers a weak password that can be easily attackable with brute force. Honey Encryption (HE) is a user data protection algorithm that can generate valid-looking plaintext if an attacker tries to decrypt it with the wrong key or honeyword and it can deceive unauthorized users.

In the honey encryption process, a set of messages with common features (like credit card numbers, cloud passwords, or other credentials) are protected. It is important to determine the type of message set or message space before encrypting the message. It is necessary to sort the messages before encryption. There are then two measures of probability to determine: the probability of each message in the Probability Distribution Function (PDF) space as well as the Cumulative Distribution Function (CDF) probability. When hackers try to get the ciphertext by guessing one of several incorrect passwords, the HE process produces a honey message. Otherwise, the HE process produces the correct ciphertext. HE redirects hackers with each incorrect guess into a confusing dead end. A key component in HE is the Distribution Transforming Encoder (DTE). The DTE consists of a set of encoding and decoding processes, where encode takes as its input space of plaintext messages M, and returns a value in the seed space S of n-bit strings. During decoding,

JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 3, 2022, p. 4004-4009
https://publishoa.com
ISSN: 1309-3452

n-bit strings' values in seed space S are converted to plaintext. The DTE process assigns a proportion of messages according to the probability distribution theory. Using the DTE process in a seed space, it maps a message according to its probabilities in the message space to a seed range. After that, the resulting seed space is XOR'd with the key to produce the ciphertext. To decrypt, ciphertext and key are XORed and a seed is calculated. It is then mapped back to the original plaintext message using the seed location. Due to limitations in this existing HE algorithm, only four messages can fit into the message space, which will be overcome through the proposed Light Weight Self-Adaptive Honey Encryption (LWSHE). The main contribution of the proposed system is mentioned below,

- A lightweight self-adaptive authentication scheme based on honey pattern encryption is implemented for user authentication.
- The existing Honey encryption message limitation is overcome through Distribution Transforming Encoder (DTE) using a self-adaptive process (discrete distribution function).

The rest of this paper is organized as follows. Section II describes the previous research works on user authentication and the Honey encryption algorithm. In Section III, we present the proposed scheme and preliminaries including an overview of Honey Encryption and Distribution-Transforming Encoders. The experimental results and evaluation is given in Section IV. Finally, Section VI concludes this work.

2. Related Work

Many Research has been carried out to solve the user authentication security problem with a different algorithm. Tan et al [1], presented a paper on enhancing the security of internet banking authentication using the Extended Honey Encryption (XHE) Scheme. To overcome the password cracking attacks, the XHE scheme generates indistinguishable bogus bank data which redirects the malicious user to the fake user account. But it needs lots of storage space for generating bogus data. Agarwal et al [2], proposed a paper on securing the network using a honey encryption algorithm. To overcome the brute-force attack which diminishes through honey encryption mechanism. The Honey encryption is concerned with the 5Ws with date, passwords, and mobile numbers. But honey encryption has a message space limitation problem. Tan et al [3], investigate the mobile health care application and its security incidents. The application depends on the One-Time Password (OTP), biometric authentication, and grid-based password which are easily cracked by malicious users. To overcome this problem, grid-based honey encryption is implemented which defeats the surfing, smudge, and replay attacks. Deepthi et al [4], analyze the data security issues in cloud computing. It is possible to hack sensitive data with brute force methods through many encryption methods in order to hide it from unauthorized users. Decrypting the secret key is achieved by using the AES and proxy re-encryption with honey encryption. But this approach is concerned about the brute force attack only. Tang et al [5], organize the EHR record using blockchain technology. The identity-based signature scheme is utilized to verify the authentication and resist collision attacks. The security parameter is measured through the Diffie-Hellman assumption. Rao et al [6], proposed a paper on data integrity and hybrid authentication where the digital signature is used to encrypt the user data. The security breaches over the user data are measured and experimented on Raspberry Pi-3 based client-server network. Badr et al [7], analyze the IoT technologies and their obstacles such as limited resources, high scalability, and irregular connectivity. Based on this analysis, the security requirement of Wireless sensors, RFID, Mobile delay-tolerant networks are identified. In prior analyses, the existing system has less secure user authentication, storage overhead, and message space constraints. In order to resolve this problem, we propose a Lightweight Self-Adaptive Honey Encryption (LWSHE) which enhances the user authentication and overcomes the message limitation in existing honey encryption. Thereby reducing the storage cost and computational time.

3. Proposed Methodology

User authentication and the LWSHE (Light Weight Self-Adaptive Honey Encryption) process were discussed in this section.

3.1 User Authentication and Login Process

Whenever a new member joins, the registration process has to be completed. Once this has been accomplished, the user uses his username and password to access the system. Using the honeyword generator, the system generates the honeywords for this member after they make the registration which is depicted in Fig 1.

JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 3, 2022, p. 4004-4009
https://publishoa.com
ISSN: 1309-3452

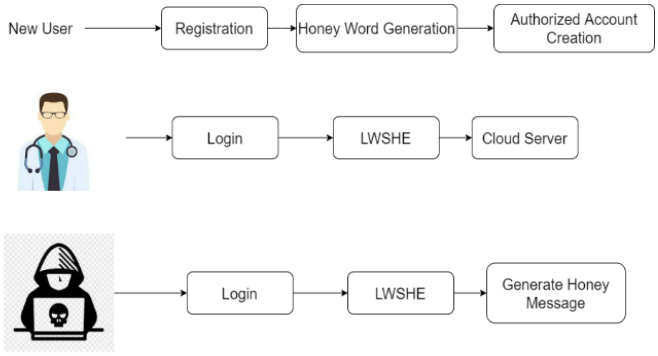

Fig 1: User Registration/Login process

Login passwords are automatically verified by the server to see if they are already stored in the database. If a password exists in the database, the server sends it to the honey checker for detection of the honeyword and the actual password. The honey checker can send a notification to the administrator if the login password doesn't match the real password stored in the database. Otherwise, the user can log in successfully.

3.2 The Light Weight Self-Adaptive Honey Encryption (LWSHE)

A honeyword, also known as a decoy password, is stored in the database alongside the user's real password to prevent unauthorized access. The honeyword file is bogus and is created to deceive the attacker if they manage to get the password file. Adding honeywords to the password file prevents attackers from knowing which password is a real one. Honeywords are the storage of bogus passwords and real passwords combined. Therefore, the honey checker alerts the system administrator if an adversary tries to log in with a bogus password. Despite the fact that honeywords can make attackers confused, the database stores many passwords which is a storage overhead issue. We have developed a Light Weight Self-Adaptive Honey Encryption algorithm that stores the honeyword in needs to assign an index to two tables to this honeyword. Instead of generating honey words and storing them, we only store the index of honey words that match the other user's real password to reduce storage costs. Consequently, in a database, indexes are stored randomly. The user's real passwords are converted into hash codes, then indexed with the user's real passwords. The second table stores the usernames and indexes of honeywords. This process translates the message space M into the string of binary bits used in the honey encryption algorithm and is known as the distribution transforming encoder (DTE). In the existing DTE process, the message space is mapped into seed space using a cumulative distribution function, thus meeting the message space limit. Decoding and coding are both performed during the DTE process. To assign the plaintext messages M to the seed space S, the DTE uses the discrete distribution function. A random seed space S is assigned to the plaintext message M during the DTE process. As a result, decoding is more difficult than encoding. Decoding the seed space S into the message space M is achieved with the DTE process. A discrete distribution function is used in the DTE process for addressing the message space constraint issue.

3.3 Proposed System Implementation

In the previous encryption algorithm, data is encrypted with the ciphertext which can be decrypted using the specific cipher key to ensure user authentication and prevent medical data attacks. An attacker using the wrong cipher key might receive jumbled-up, meaningless data. This would indicate a problem with the key, so the attacker might try another one. The Light Weight Self-Adaptive Honey Encryption (LWSHE) approach will prevent this security breach. Using LWSHE, the attacker appears to receive a fake plain-text message (username and password) or a Honey message. It causes the attacker to believe that the keyword may indeed be correct.

JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 3, 2022, p. 4004-4009
https://publishoa.com
ISSN: 1309-3452

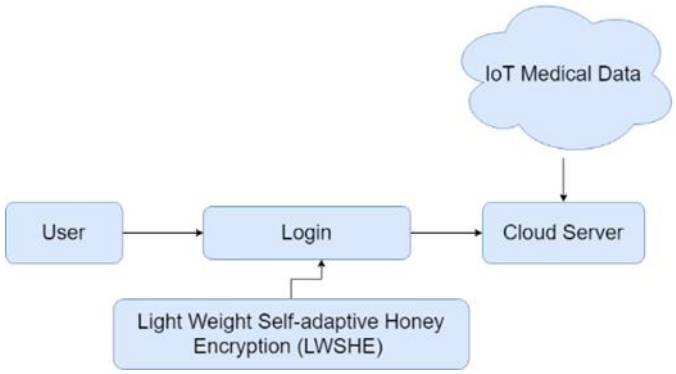

Fig 2: Block Diagram of the proposed system

During the distribution transformation, a Distribution Transforming Encoder (DTE) maps the plain text strings into the seed strings. Honey messages are created by using the probability distribution with n-bit strings which is depicted in Fig 2. The DTE overcomes the message limitation using a self-adaptive process (discrete distribution function) which is the main drawback of the traditional Honey encryption algorithm

Algorithm: 1 Light Weight Self-Adaptive Honey Encryption Algorithm (LWSHE)

Inputs: Username (U_n) and Password (U_p)-> generate an index list $[U_{n1}U_{p1}, U_{n2}U_{p2}..U_{nk}U_{pk}]$

Step 1: Honey words (H_w) are assigned to the user account U_a

Step 2: For $U_a = [P_1, P_2]$ are created where $P_1 = U_n H_i$, $P_2 = U_p U_h$. The list index of honeyword and password are created by $P_1 = \{< U_1, H_1 >, < U_2, H_2 >, ... < U_n, H_i >\}$, $P_2 = \{< r_1, H(p_1) >, r_2, H(p_2) >, ... r_k, H(p_k) >\}$

Step 3: Honey checker detection

Set $r_k H(p_k)$

Set correct password index r_k for the user U_n.

The primary goal of the proposed H-LWSHE method is to enhance user authentication through honey word generation. We created two password fields P_1, P_2 in the database and the honey checker database. P_1 contains the username $U_n$and honeyword index H_i set located in the main server's database. P_2 is assigned in the honey checker database where real password index r_k and hashing password $H(p_k)$ which is represented in algorithm 1. If any user login to the system, then the r must be equal to r_k which indicates the user is an authorized one, or else the honey message will be generated.

4. Results and Discussion

We conduct the simulation experiment on the proposed LWSHE method to evaluate its performance in terms of user authentication and privacy of data in the sensitive database. Our experiment was done on a computer with a 1.80 GHz Intel Core i8-265U processor and 8 GB of RAM, running Windows 10 OS. We implemented this algorithm using Python language.

The LWSHE security analysis on user authentication is evaluated and compared with the existing encryption (password-based Encryption, Honey encryption, proposed LWSHE) methods which are represented in Table 1. The Brute Force Attack, DoS attack, Dictionary Attack are applied to the proposed LWSHE for checking the authentication. Compare to other existing encryption methods, the DTE process also overcomes message space limitation problems and enhances the privacy of the server. The time complexity of the proposed LWSHE is compared with the existing honey encryption which is represented in Table 2 and Fig 3. Based on the password length the time complexity is measured it is shown that the proposed LWSHE (5.1ms) took less time to check the 12-character password than the honey encryption algorithm (70ms).

JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 3, 2022, p. 4004-4009
https://publishoa.com
ISSN: 1309-3452

Table 1 Comparison of existing techniques with the proposed LWSHE method

Technique	Brute Force Attack	DoS Attack	Dictionary Attack	DTE process
Password-based Encryption [10]	Weak	Weak	Weak	-
Honey Encryption [2]	Strong	Strong	Strong	Message space limitation
Proposed LWSHE	Strong	Strong	Strong	Overcome the message space limitation problem

Table 2 Time complexity comparison of the existing algorithm with the proposed LWSHE method

Parameters	Complexity of Time (Milliseconds)			
Size of passwords (in character)	6	8	10	12
Honey Encryption [2]	40	48	56	70
Proposed LWSHE method	3.1	4.2	4.4	5.1

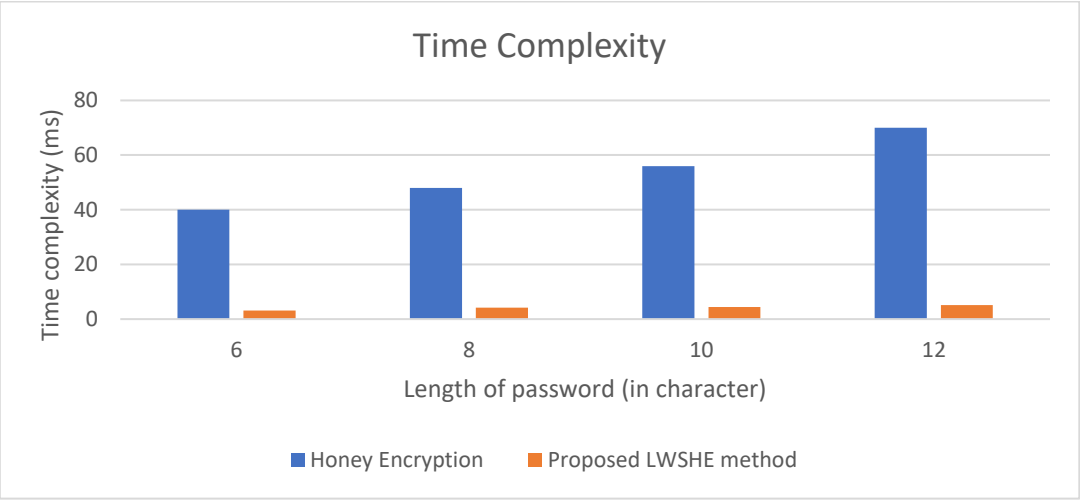

Figure 3 Comparison graph of Honey Encryption and proposed LWSHE method

Based on the various simulation result, it is shown that the proposed LWSHE enhance the user authentication for securing the medical IoT data thereby reducing the computational time and privacy issues.

5. Conclusion

In this paper, we have proposed a user authentication system for securing IoT medical data. The highlight of our proposed protocol is that it protects medical data from brute force attacks. The Light Weight Self-Adaptive Honey Encryption (LWSHE) overcomes message limitations and creates false passwords that confuse the attackers. Comparing the proposed

method to other existing methods, the experimental results show the proposed method can provide higher security and resistance to several attacks. Future work will include implementing the proposed method with real-time IoT data collected from sensors attached to the patients where we can test its stability and accuracy.

Reference

1. Tan S.F., Samsudin A. (2018) Enhanced Security of Internet Banking Authentication with EXtended Honey Encryption (XHE) Scheme. In: Zelinka I., Vasant P., Duy V., Dao T. (eds) Innovative Computing, Optimization and Its Applications. Studies in Computational Intelligence, vol 741.
2. Agarwal A.K., Rani L., Tiwari R.G., Sharma T., Sarangi P.K. (2021) Honey Encryption: Fortification Beyond the Brute-Force Impediment. In: Manik G., Kalia S., Sahoo S.K., Sharma T.K., Verma O.P. (eds) Advances in Mechanical Engineering. Lecture Notes in Mechanical Engineering. Springer.
3. S. -F. Tan, K. -M. C. Lo, Y. -B. Leau, G. -C. Chung and F. Ahmedy, "Securing mHealth Applications with Grid-Based Honey Encryption," 2021 IEEE International Conference on Artificial Intelligence in Engineering and Technology (IICAIET), 2021, pp. 1-5, doi: 10.1109/IICAIET51634.2021.9573645.
4. B. Deepthi, G. Ramani, R. Deepika and M. Shabbeer., "Hybrid Secure Cloud Storage data based on improved Encryption Scheme," 2021 International Conference on Emerging Smart Computing and Informatics (ESCI), 2021, pp. 776-779, doi: 10.1109/ESCI50559.2021.9396842.
5. F. Tang, S. Ma, Y. Xiang and C. Lin, "An Efficient Authentication Scheme for Blockchain-Based Electronic Health Records," in IEEE Access, vol. 7, pp. 41678-41689, 2019, doi: 10.1109/ACCESS.2019.2904300.
6. V. Rao and P. K. V., "Lightweight Authentication and Data Encryption Scheme for IoT Applications," 2020 IEEE International Conference on Distributed Computing, VLSI, Electrical Circuits and Robotics (DISCOVER), 2020, pp. 12-17.
7. Badr, Y., Zhu, X. & Alraja, M.N. Security and privacy in the Internet of Things: threats and challenges. SOCA (2021).
8. X. Yang, T. Li, X. Pei, L. Wen, and C. Wang,(2020) Medical Data Sharing Scheme Based on Attribute Cryptosystem and Blockchain Technology, in IEEE Access, vol. 8, pp. 45468-45476.
9. Ali R., Chandrakar P., Kumar A. (2020) On the Security Weaknesses in Password-Based Anonymous Authentication Scheme for E-Health Care. In: Das S., Samanta S., Dey N., Kumar R. (eds) Design Frameworks for Wireless Networks. Lecture Notes in Networks and Systems, vol 82. Springer.