Honey Encryption Techniques: Strengths, Limitations, and Challenges

Nidaa Flaih Hassan
Computer Science Collage, University of Technology

Hala Bahjat AbdulWahab
Computer Science Collage, University of Technology

Ayad al-Adhami
Computer Science Collage, University of Technology
Ayad.h.ibrahim@uotechnology.edu.iq

Rehib F. Hassan
Computer Science Collage, University of Technology

Abstract: Currently, information security is a difficult issue due to growing information capacity and transmission procedures. One encryption method that can provide security above and beyond the brute-force limit is honey encryption (HE). Ultimately, HE becomes a part of everyday systems. Most researchers use improved encryption and decryption for HE schemes to improve information security domains since traditional HE has several drawbacks. An overview of current developments in the Honey Encryption scheme will be provided in this article. HE's drawbacks and difficulties are outlined and examined. Specific of the solutions suggested have been introduced.

Keywords: Cryptography, Honey Encryption, Brute Force Attacks, Honey Passwords, Password-based Encryption.

1. Introduction

Password-Based Encryption (PBE) is used in the majority of encrypted systems. Brute force guessing attacks can be used against these systems. The encryption of honey by limiting the amount of information that attackers may learn via password guessing, honey encryption seeks to mitigate this issue. The system generates a decrypted message that seems legitimate for every potential key [1]. Therefore, it is challenging to determine which password is accurate [1].

An adversary may launch an offline brute-force assault if they get to access the encrypted file, and given the information we know about user password choosing,

there's a significant probability this would be successful. The issue of weak passwords cannot be solely attributed to users; it seems sense that individuals would select passwords that are simple to remember and use on several websites. Although a lot of services are starting to implement strong password restrictions, it's crucial to consider alternative strategies to address the issue [2]. As it is known, the cornerstone of the secret world is the secret key [3]. The lack of a trustworthy technique for retrieving misplaced keys and the absence of a secure and efficient way to save users' private keys are the two main problems with key management [4].

Use an indexing database to create an efficient intelligent session (mask keys) that reduces the size of multi-session secret mask keys and raises security by transmitting unique codes between the sender and the recipient over a secure channel [5].

One encryption technique that has been shown to increase security above the brute-force limit is Honey Encryption (HE). By creating decoy passwords that closely resemble actual passwords, honey encryption is a potential method for improving password manager security. It makes it harder for attackers to guess the right password by making it more complicated. Juels and Ristenpart first proposed honey encryption in 2014, proving how successful it is in preventing unwanted access to private information. According to studies, honey encryption makes it computationally impossible to figure out the real password and creates a wide search field, which dramatically lowers the success probability of offline brute force assaults. Adding honey encryption to password managers strengthens their defenses against offline brute force assaults and provides an additional degree of protection [6].

In this paper, a comprehensive background of study for Honey Encryption (HE) is presented, Recent Honey Encryption related works are examined and briefly discussed. Additionally, in order to facilitate comparison, the structure of Honey Encryption is provided alongside that of standard encryption. After reviewing and analyzing previous research, we identified key concerns that center on strength and limitation for Honey Encryption.

2. Related works

Honey encryption (HE) is one of the numbers approaches established by computer scientists to reduce the probability that a hacker would effectively crack passwords and get vital information. Many recent works are proposed for applying HE, the following section is a brief to this work:

In [7], Edwin Mok et al. expanded the Honey Encryption (HE) technique by giving the encrypted data an extra layer of security. Instead of denying access, the HE

algorithm creates indistinguishable false data when the attacker tries to access these encrypted data by inputting the wrong password, making it impossible for the attacker to tell if the assumed password is accurate. Consequently, password guessing and cracking attempts become more sophisticated.

Santhi Baskaran et al. [8], adopted new cryptographic architecture for IoT devices which is based on honey encryption scheme. Honey encryption constructs a cipher text that decrypts under any password to a plausible-looking message. They used the property of distributed transforming encoder property for distribution of message seeds. By their honey encryption scheme, the attacker succeeds in decrypting the cipher text he doesn‘t knows which is correct one.

Wei Yin et al. [9] designed and put into use honey encryption techniques, which were then used to protect three different kinds of sensitive information: debit card passwords, mobile phone numbers, and Chinese identity numbers. They assessed how well their process and solution to the overhead problem worked. They suggested additional security methods for future studies that can handle various attack and vulnerability kinds. Additionally, the efficacy and efficiency of their mechanism may be assessed by testing and evaluation in various cloud settings.

Abiodun et al. suggested a state-of-the-art method for modifying the Honey Encryption (HE) technique to enable the encoding and decoding of human messages, including emails and lengthy written documents. This method is implemented using Princeton's Wordnet and Stanford Dependency Parser [10]. The outcome demonstrated that the suggested plan successfully fooled the attacker with convincing decoy messages. Furthermore, the original message's content, length, and structure are hidden. Lastly, they checked the entropy of decoy messages from the plaintext to confirm the efficacy of the suggested method.

Nahri Syeda Noorunnisa and Khan Rahat Afreen, established a safe foundation by putting the suggested Honey Encryption idea into practice and utilizing a secure password manager that assists in creating strong passwords that are less vulnerable to brute force hacking [11]. For convenience, the user may safely store them all in one location and won't need to remember them all. Additionally, integrating the Vernam cipher with Honey Encryption adds an extra layer of protection. When HE is combined with AES, Blowfish, and OTP, it is shown that OTP takes almost as long as Blowfish.

In [12], Sathoshini Sahu et al. employed a brute force attack against the login systems using the FDGA (Fake Data Generating method) security method. His approach has a drawback in that their field vision is constrained. Some servers cannot accommodate the enormous size required for honey pots. However, this technique may be used in conjunction with host-based intrusion defense and

networks. The creation of honey messages utilizing effective DTEs for all kinds of issues organically is a major hurdle when employing this approach.

Navneet Singh Khurana [13] offered Cloud Computing Honey Encryption Data Security, which offers superior security over competing products. The main drawbacks of Elliptic Curve Cryptography and Homomorphic Encryption are their complexity and high computing and storage costs. AES and Honey Encryption integration may be assessed in the future to stop known-plaintext attacks for cloud security.

The research provided by [14] offered an effective and safe authentication method that enables MPs to access patient data over a Cloud IoT network. Light Weight Self-Adaptive Honey Encryption (LWSHE) was incorporated into the suggested protocol to protect medical data stored on a cloud server via the Internet of Things (IoT). A discrete distribution function in the Distribution Transforming Encoder (DTE), which converts the plain text strings into the seed strings, solves the message size restriction issue with the conventional Honey encryption technique. The honey word produced approach removed typo safety issues and storage overhead from their honey encryption scheme. According to the findings of the security and performance comparison study, the suggested protocol accomplished more sophisticated security operations and effective user authentication in addition to thwarting the execution of the susceptible assaults.

Nwe Ni Khin et al .[15] enhanced honey encryption algorithm by fusing the advantages of the enhanced honey encryption method with the DNA-based encoding technique. Input messages are encrypted using the honey encryption procedure, and the passwords are created as the keys utilizing the DNA techniques based on five distinct lookup tables. By using five distinct lookup tables, this hybrid approach can lessen the storage overhead issue with the DNA technique and simplify the current honey encryption process in terms of time complexity.

V Vasanthi, and J Logeshwaran suggested a Honey Encryption scheme to safeguard data stored on public cloud servers. The password-based encryption used for the current file protection was susceptible to brute-force, dictionary, and rainbow table attacks, among other password guessing techniques [16]. The proposed scheme strengthened the security of files that were already encrypted. When the attacker attempted to access the encrypted data using his guessing password, the extended HE method produced a phony file that was nearly identical to the real file, rather than preventing them access to the data as would be the case with a regular file encryption technique. It is clear that the message space of the proposed technique is pre-fixed, and that the size and complexity of inverse sampling tables increase with the length of file names and their extensions.

Yusuf Musa et al., suggested a model that combined the power of the Honey Encryption technique with the capacity to create a randomized message encoding known as a Distribution-Transforming Encoder (DTE) that was able to both detect and identify the attacker [17]. Security personnel might use the suggested model as a guide to find and apprehend the suspected criminal. An analysis of the model revealed that it was 62% successful at identifying attackers. A closer look at the model revealed that 21% of the attackers were able to obtain access by developing a strong bond with the users who were logged in. By increasing the model's detection rate, the suggested model may be extended.

In [18], Anas et al. provided a method for password manager security that combines the Honeypot technique and the Honey Encryption algorithm. We may successfully direct attackers to a honeypot, which is a collection of phonies or honeyword passwords, by adding Honey Encryption to the password manager's authorization procedure. Because of the honeypot's high interaction design, we can obtain important details about the attacker, like their MAC and IP addresses. For additional research and the implementation of suitable measures to lessen the security breach, this information may be essential. Password managers are strengthened against unwanted access attempts by the proposal system, which adds an extra layer of protection.

In 2024, Maria Antoniate Martin, and Sharon Dominick, offered each consumer a safe online banking experience. Through client-side encryption, the system offered a robust defense against online fraud [19]. Fraud detection was guaranteed on the server side of data mining technology. One may conclude that the picture meets all requirements for quality and efficiency based on all of the tests done on the image-generated honey encryption. The data mining module was tested using two distinct datasets to make sure the method can handle both regular and severe instances. Delivering a secure and secured online banking experience was the goal of the study, and based on the analysis and testing done, it appears to have been successfully accomplished.

Reshma Siyal et al. , suggested a hybrid key-generation method for safe data exchange in a multi-party blockchain-based system dubbed the Identity and Attribute-Based Honey Encryption (IABHE) Algorithm combined with Deep Spiking Neural Network (DSNN), or IABHE+DSNN [20]. To guarantee data security, this method combines several entities and security features. Initialization, authentication, initial registration, data protection, validation, and data sharing are the phases in the data-sharing process. The MapReduce framework is used to carry out data protection; IABHE is used for data encryption, and DSNN is used for key creation. According to experimental data, the suggested IABHE+DSNN strategy

outperforms current techniques with a key complexity of 0.887, an encryption time of 15.765 s, and a decryption time of 10.786 s.

In 2025, Reshma Siyal et al., adopted Adaptive Attribute-Based Honey Encryption (AABHE), a cutting-edge method that enhances data security by combining honey encryption with Ciphertext-Policy Attribute-Based Encryption (CP-ABE) [21]. AABHE ensures that sensitive information cannot be accessed by unauthorized individuals, even in the event that it is intercepted. Focusing on safeguarding large files in HDFS, the proposed method outperformed other encryption techniques such as Advanced Encryption Standard combined with Attribute-Based Encryption (AES+ABE), Ciphertext-Policy Attribute-Based Encryption (CP-ABE), and Key-Policy Attribute-Based Encryption (KB-ABE), achieving 98% security robustness and 95% encryption efficiency. AABHE strengthens its defenses against data breaches and improves Hadoop's reliability as a platform for processing and storing enormous volumes of data by addressing Hadoop's security vulnerabilities.

3. Key Generation techniques

In cryptography, key creation is the process of creating keys, thus whatever data is being encrypted or decrypted a key is required for both processes. A key generator, often known as a keygen, is a software or device that generates keys. Both public-key methods like RSA and symmetric-key algorithms like DES and AES are part of contemporary cryptography systems. The term "key range" describes the collection of all potential keys that may be utilized in cryptography. The range impacts the encryption's security and is dictated by the underlying algorithm and key size.

The effectiveness and security of improving the ongoing images encryption technique are directly determined by the secret key generator [23]. When the key is strong, it can withstand cryptanalysis even when the attacker finds all system data connected to the encryption key's production or validation [24]

The key needs to be created, shared, and kept safely [25]. The length of the key is troublesome, particularly when the key itself needs to be sent via an unsecure channel, since any safe way to send it would unquestionably be a way to deliver the message itself. The other is that we depend on the creation of random elements [26]. With the use of the chaotic map, keys with random sequences may be created; these keys pass the NIST test of randomness since chaotic can produce unexpected and random numerical sequences [27].

The message and the key are required inputs for the HE encryption algorithm. The key creation procedure was omitted from the original method for simplicity's sake, but it must be described in depth here to serve as a model for implementation. The

encryption key K should have a sufficient length, at least 128 bits, but ideally 256 or 512 bits [28].
There are other key-generation-based honey encryption techniques available, such as a hybrid encryption algorithm that combines a complex Deoxyribonucleic acid (DNA) encoding technique with the honey encryption process. DNA offers the highest level of protection and strong security due to its enormous capacity and low rate of mutation; it is now being researched as a possible information security carrier [29]. Another hybrid key-generation method for safe data exchange in a multi-party blockchain system is the Identity and Attribute-Based Honey Encryption (IABHE) Algorithm in conjunction with Deep Spiking Neural Network (DSNN). To guarantee data security, this method combines a number of entities and security features. [30]. Even with a wide range of tools available for creating encryption keys, there are still several pieces of information to consider thus any secure way to transmit the key will be unquestionable.
A way to send the message itself depends on the using of the key length, particularly when the key itself needs to be sent over an insecure channel.
Additionally, the creation of random numbers needs to be dependent due to third party attempts to copy or steal the key during transmission. Thus, an attacker can decipher any ciphertext that was encrypted using the key. The requirement for several key pairs between communication parties is another issue. The more there have resulted in harder to be controlled.

3.1 Brute Force Attack

Brute force attack is required for a better understanding before the study offers a remedy. Since an attacker's primary concern is message recovery, if they have an encryption cipher $C = enc(K, M)$ of a message M and key K (where K and M are selected from known distributions), their objective is to recover M. The collection of messages $M_1, M_2, M_3, \ldots M_q$ is the result of the attacker trying many keys to decode C. With a winning probability equal to the attacker's capacity to select message M from the q candidate, access to message M is ensured if the attacker is successful with key K [22].

4. Methodology of Honey Encryption

Honey encryption (HE) offers security above and beyond the brute-force barrier after being inspired by such decoy systems. During brute-force assaults, these strategies produce candidate messages that are identical to legitimate ones [23].
To guarantee a fundamental comprehension of the evolution of Honey encryption, a summary of traditional encryption is provided in the subsequent section. When an

opponent uses a brute-force assault to get the key used to encrypt a message in a traditional password-based encryption system, the anticipated return is either an error signal or nonsense (a non-uniform distribution). This output serves as a reminder that the key he is attempting is wrong, and he keeps searching until he finds a message that seems convincing and may be the plaintext. When the distribution is not uniform, he promptly discards the message during his attack. He will have more time to look further, and he has a good chance of finding the message or plain text. Figure 1 shows a description of how a password-based encryption system reacts to a brute force assault.

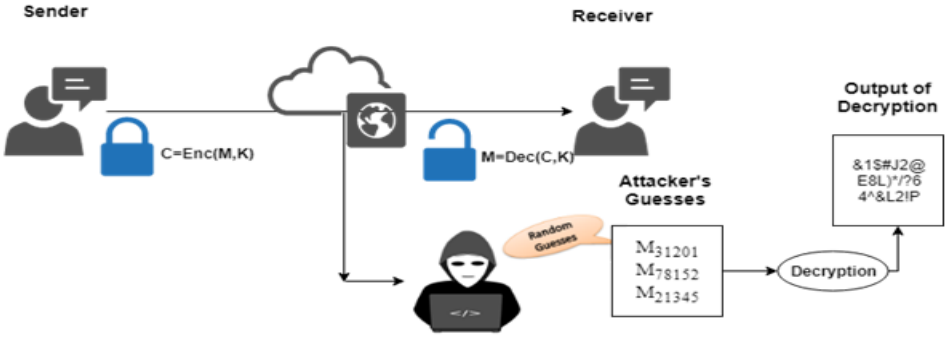

Figure 1: Diagram for a brute-force attack using the traditional encryption technique

The sender uses a cipher and a key to encrypt his message. He transmits the encrypted message to the recipient. Using the same key that was used by the sender and a decryption technique, the recipient decrypts the encrypted text. By guessing the key at random, an attacker who intercepts the ciphertext may attempt to decrypt the communication. Because of the non-uniform distribution of the plaintext, the attacker may quickly determine from the assumed plaintext in the traditional setup scenario that the key he provided is inaccurate. We provide a brief description of an attack model for the HE system.

For a message M encrypted using $C = enc(M, K)$. if a known distribution is used to determine K and M. An adversary's goal is to retrieve the message M. He attempts to use several keys to decode C. He receives $M_1, \ldots, M_n$ for each key he attempts. M is certain to be on his list for a minimal entropy distribution such as passwords. This is made feasible by people selecting readily guessed, basic passwords. Additionally, attackers know how users select their passwords because of previously published information on password leaks on the internet.

As a result, the security in this case is based on the likelihood that, if one of the keys the opponent tested was true, he could select message M out of all n potential

messages. Even if an opponent guesses the key correctly, he is still trapped with fake data and is unable to determine whether message is the correct one, particularly if he is unaware of the target message. Identifying the message from the list of messages he collected during his attack is the only way he can win [24]. A description of how the honey encryption system reacts to a brute force assault is provided in Figure 2.

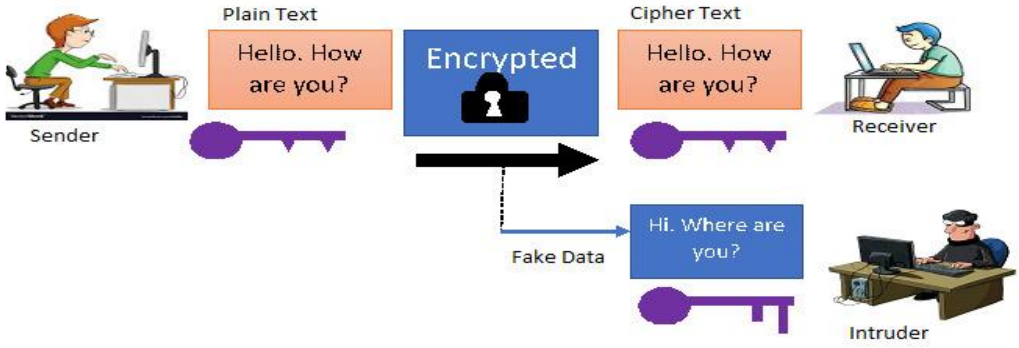

Figure 2: The Model of Honey Encryption

5. Honey Words

In information security, the term "honey word" is used to identify a fake resource. A honeypot is a phony server that typically holds fictitious data. False information placed in databases is called "honey word." Based on honey encryption, the encryption algorithm fools the invader by providing fictitious data that looks like the actual data whenever it senses an incursion. Hackers that steal user login and password databases do so in order to guess one valid password and get access to the data. It is really challenging for the hacker to figure out how they know they have the right password [25].

The honey words technique is used to create companion sugar words (real passwords) and honey words (false passwords). All of these sweet words are then entered into the username and password database and hashed [26].

The hacker should be able to correctly guess the actual password from some of the catchy terms if they are able to extract plain passwords from hashed passwords. If not, an additional server known as a honey checker notifies the system administrator in silence that password cracking is feasible [27]. Since a hacker can succeed with a probability of (1/n) by randomly predicting a sugar word (n=number of sweet

words), flatness is the hacker's predicted likelihood of successfully guessing the sugar word. The hacker's chances of choosing one are at least (1-(1/n)) if the honey words are completely flat, or 1/n flat. After logging in, assuming the user's account is authentic, the password is hashed and compared to the list of "sweet words" before being forwarded to the honey checker for confirmation. . Login will be allowed if the password is the sugar word; if not, the administrator will be alerted to a possible password break [28].

6. Strength of Honey Encryption Techniques

Honey encryption (HE) provides security beyond the brute-force barrier, it's extremely valuable for protecting consumer data and information, from many studies focused on HE, many strength issues of HE is concluded in the following section: -

1. **Mimics keys**: Hackers can figure out the password in a traditional cryptographic system by verifying that it is accurate. However, in the case of honey encryption, the attacker will be given a password that looks much like the actual one when they brute force the encryption. This makes it difficult for the hacker to tell the difference between the actual and phony passwords.
2. **Generated keys:** If a hacker had access to a large encryption password vault after a data breach, he could simply break the cybercriminal using typical cryptography techniques. However, because honey encryption creates passwords that closely resemble the real ones, the hacker would be unable to crack it. The fake passwords used for honey encryption are collected from the Internet, which offers a plethora of password dumps. These password dumps are used to produce honey encryption and can imitate the actual password. The currently fake password vault generator for honey encryption may be used to secure the password managers. Information is gathered by this generator from several sources, mostly large databases of password breaches and password samples used by password crackers.
3. **Granted access:** This verification can be performed by a honey checker that stores the password index. As a result, the computer system has some phony passwords in a standard authentication process, and the honey checker has the index. When a user logs in with the correct password, the system checks the honey checker; if the indexes match, the user is granted access without an alarm being raised. Using such a setup enhances distributed security and lowers the possibility of compromising of the computer system and honey checker. When the honey checker returns online, even if it is not live, the system could compare the cache to find the breach.

4. **Impermeable Encryption**: Databases are commonly used in Honey Encryption (HE) to store passwords in order to maintain the integrity of the password file. Therefore, if someone makes any unauthorized changes to the database, we may conclude that there has been a breach. Consequently, honey encryption is used to produce a synthetic password file that includes many fake passwords. It is challenging to identify the fake password file since it looks just like the actual one and has a lot of catchy terms. Combinations of honey words with the user's original password are referred to as "sweet words." The likelihood that the honey encryption will be compromised is reduced because the password system and honey checker operate on separate operating systems.
5. **Cloud Computing Safety:** As more and more businesses and individuals go toward the cloud, security remains a difficult problem. There is definitely more work to be done to reassure users about the security of cloud computing. When compared to other options available for cloud computing data security, honey encryption provides more security.
6. **Honey Encryption with AI:** prevents brute-force assaults, adding an extra degree of protection for AI projects. Introduces phony decrypted outputs to deter hackers from trying to decode critical AI data.

7. Limitations of Honey Encryption Techniques

There are a lot of useful applications for this honey encryption scheme, but it still needs some fine-tuning before it can be used throughout the network. But like any other technique, honey encryption has drawbacks, which add up to the following: -

1. **Known-Plaintext Attack**: Honey Encryption using Password-Based Encryption has limitation for being prone to known-plaintext attack.
2. **Lost or stolen keys:** It is practically hard to recover the original data without keys, making the data essentially useless. Permanent data loss may result from this circumstance, particularly if sensitive data or long-term storage was the goal of the encryption.
3. **Creation False Information:** The difficulty will be determining how to fabricate false information that is convincing enough to deceive an attacker; they must be straightforward accounts that are simple enough to fabricate.
4. **AI for Creation Honey Datasets**: The current techniques for crafting decoy messages don't accurately replicate human language; hence they are unable to produce acceptable and persuasive decoys that might distract an attacker from the actual resource when using AI to generate honey datasets.

5. **Decoy Space**: The difficult aspect of employing the Honey Encryption approach is setting up the decoy area, which has an impact on the security that HE offers. Despite its capacity to deceive attackers, HE's ability to safeguard private information differs depending on the application and message space.
6. **Sophisticated Attacks:** Does not completely defend against complex assaults that employ cutting-edge analytical techniques, which, if not used appropriately, might confuse authorized users. Application of Generative AI: Honey encryption can be used to safeguard private client information used to train AI models in a generative AI project centered on customized marketing. The integrity of AI predictions is protected because, even if an attacker manages to get access, they will be met with deceptive outputs.

8. Recommendation with Recent Technologies

1. For cloud security, the combination of AES and Honey Encryption may be tested in the future to stop known-plaintext attacks. Because of this, we are in favor of further advancements and the use of honey encryption for cloud security.
2. High interaction honeypots combined with honey encryption have shown themselves to be effective, user-friendly, and to have no influence on authorized users. We improve the overall security posture and resistance to credential theft by incorporating these strategies into current password manager solutions with ease.
3. Combine image encryption methods with the honey encryption strategy to produce digital images that are more random and more protected.
4. The novel application of the Grasp algorithm with construes phase is the random generation of keys based on the change basis point for every transmitting procedure.
5. B+ trees offer an effective indexing database for keys, and their compression functions are crucial for converting the key into a unique code that can be sent to the recipient.

9. Conclusions

Honey Encryption (HE) provides resilience to brute-force attacks. Our objective is to give investigators an overview of HE approaches, emphasizing both HE's strength and limitations. As is well known, Honey Encryption has been referenced in several studies and has been shown to provide additional security in addition to conventional encryption. . Beginning with an explanation of Brute Force Attack, this paper provides a summary of the most recent research on honey encryption. . The presentation of honey approach explains how honey encryption differs from

standard encryption. Furthermore, honey words are explained because they are the most often used password authentication method due to its simplicity and ability to be preserved as a hashed password. Strength of Honey Encryption Techniques focusing on the following factors: Mimics keys, Generated keys, Granted access, Impermeable Encryption Cloud Computing Safety, and Honey Encryption with AI. In addition, Limitations of Honey Encryption Techniques focused on the following factors: Known-Plaintext Attack, Lost or stolen keys, Creation False Information, AI for Creation Honey Datasets, Decoy Space, and Sophisticated Attacks. Main challenges cited in the difficulty of producing the honey keys and protecting them from an attack. Some of the recommendations are adopted to solve some of limitations of (HE), but there are still unresolved challenges relating to HE. AI with HE promised to solve many limitations and could move away from conventional encryption, as an example using False plaintexts are produced by GPT for incorrect keys. Decoys those are most successful at misleading attackers while reducing false alarms for authorized users might be chosen or created by a reinforcement learning agent.

10. References

[1] N. Tyagi, J. Wang, K. Wen, and D. Zuo, "Honey encryption applications," in Computer and Network Security, pp. 1–16, 2015.

[2] J. Burgess, "Honey Encryption Review ", Centre for Secure Information Technology (CSIT), Belfast, UK. 2017.

[3] Raghad Abdulaali Azeez, Abeer Salim Jamil, Ayad Al-Adhami, Nidaa Flaih Hassan, "Multibiometric System with Runs Bits Permutation for Creating Cryptographic key Generation Technique", Iraqi Journal of Science, Vol. 64, No. 1, pp: 452-468, Iraqi Journal of Science, Vol. 64, No. 1, pp: 452-46, 2023

[4] Asia Ali Salman Al-karkhi, Nidaa Flaih Hassan, and Raghad Abdulaali Azeez, "A Secure Private Key Recovery Based on DNA Bio-Cryptography for Blockchain "Iraqi Journal of Science, Volume 64, No. 2, 2023.

[5] Hala Bahjat AbdulWahab, "Image Encryption Based on Intelligent Session Mask Keys", Iraqi Journal of Science, Vol. 58, No.1B, pp: 317-326, 2017

[6] A. Danial, M. Fadzil Abdul Kadir, A. Faisal Amri Abidin, M. Afendee Mohamed, N. Abdul Hamid, and S. Dhalila Mohd Satar, " Password Manager Security using Honey Encryption Algorithm and Honeypot

Technique “, Malaysian journal of computing and applied mathematics no. 6, issue2, pp. 1-10,2023.

[7] Edwin Mok, Azman Samsudin, and Tan Soo Fun, “Implementing the Honey Encryption for Securing Public Cloud Data Storage” , Conference: First EAI International Conference on Computer Science and Engineering, 2017. DOI: 10.4108/eai.27-2-2017.152270

[8] Santhi Baskaran, S.V. L Sarat Chandra, P.Venkatesh , E.Silambarasan , M.Dinesh “Implementation of Enhanced Honey Encryption for IoT Security”, International Journal of New Technology and Research (IJNTR) , Volume 3, Issue-3, pp. 87-89, March 2017

[9] Wei Yin, Jadwiga Indulska , and Hongjian Zhou,” Protecting Private Data by Honey Encryption” , Hindawi Security and Communication Networks, pp. 1-9, 2017.

[10] Abiodun Esther Omolara, Aman Jantan, Oludare Isaac Abiodun and Howard Eldon Poston, “A Novel Approach for the Adaptation of Honey Encryption to Support Natural Language Message” , Proceedings of the International MultiConference of Engineers and Computer Scientists 2018 Volume I, IMECS 2018, March 14-16, 2018.

[11] Nahri Syeda Noorunnisa, and Khan Rahat Afreen , “Honey Encryption based Password Manager” , Journal of Emerging Technologies and Innovative Research (JETIR), 2019 JETIR May 2019, Volume 6, Issue 5, pp. 428- 432.

[12 SANTHOSHINI SAHU, “PROVIDING INFORMATION SECURITY USING HONEY ENCRYPTION”, Advances in Mathematics: Scientific Journal, Volume 9, No.10, pp. 8249–8258, 2020

[I3] Navneet Singh Khurana , “ Security in cloud computing using honey encryption”, International Journal of Advance Research, Ideas and Innovations in Technology, Volume 7, Issue 1, pp. 359-364. 2021 .

[14] P. Renuka, and Dr. B. Booba, “A Light Weight Self-Adaptive Honey Encryption for User Authentication Scheme in Cloud-IoT Health Services “, JOURNAL OF ALGEBRAIC STATISTICS Volume 13, No. 3, pp. 4004-4009. 2022.

[15] Nwe Ni Khin, and Thanda Win, “A Novel Hybrid Encryption Method Based on Honey Encryption and Advanced DNA Encoding Scheme in Key Generation”, Journal of Computer and Communications, Volume.10 , No.9, pp. 22-36, September 2022. DOI: 10.4236/jcc.2022.109002

[16] V Vasanthi, and J Logeshwaran, " HONEY ENCRYPTION ALGORITHMS TO PROTECT DATA", Journal of Emerging Technologies and Innovative Research (JETIR), Volume 10, Issue 3, pp.D1-D14 , March 2023.

[17] Yusuf, Musa , FAKI, Ageebee Silas, Adelaiy e, and Ishaya Oluwasegun , " Honey Algorithm for Securing and Identifying Hackers in a Pervasive Environment" , Nile Journal of Communication & Computer Science, Volume 5, Issue 1, pp. 2-9, June 2023

[18] Anas Dania, Mohd Fadzil Abdul Kadir, Ahmad Faisal Amri Abidin, Mohamad Afendee Mohamed, Nazirah Abdul Hamid,and Siti Dhalila Mohd Satar, " PASSWORD MANAGER SECURITY USING HONEY ENCRYPTION ALGORITHM AND HONEYPOT TECHNIQUE ", MALAYSIAN JOURNAL OF COMPUTING AND APPLIED MATHEMATICS (2023) 6 (2) 1-10

[19] V. Maria Antoniate Martin, and Sharon Dominick , " Enhancing Cloud Data Security with Honey Encryption", Journal of Emerging Technologies and Innovative Research (JETIR), Volume 11, Issue 1, pp. g608-g611 , January 2024.

[20] Reshma Siyal , Jun Long , Muhammad Asim , Naveed Ahmad , Hanaa Fathi, and Mohammad Alshinwan , "Blockchain-Enabled Secure Data Sharing with Honey Encryption and DSNN-Based Key Generation" , Mathematics , Vol. 12 , Issue 13 , pp. 1-25 , 2024 https:// doi.org/10.3390/math12131956

[21] Reshma Siyal, Jun Long , Muhammad Asim , Naveed Ahmad , Hanaa Fathi, and Mohammad Alshinwan , " Blockchain-Enabled Secure Data Sharing with Honey Encryption and DSNN-Based Key Generation" , Mathematics, pp. 1-25 .

[22] Reshma Siyal, Muhammad Asim, Long Jun,Mohammed Elaffendi, Sundas Iftikhar,Rana Alnashwan, and Samia AllaouaChelloug, "Adaptive Attribute-Based Honey Encryption: A Novel Solution for Cloud Data Security", Computers, Materials & Continua , Tech Science Press, pp. 1-28, January 2025

[23] Lahieb Mohammed Jawad and Ghazali Sulong, "A Novel Dynamic Secret key Generation for an Efficient Image Encryption Algorithm" , Modern Applied Science; Vol. 9, No. 13; 2015.

[24] Zainab Khyioon Abdalrdha*, Iman Hussein AL-Qinani, Farah Neamah Abbas, "Subject Review: Key Generation in Different Cryptography Algorithm", 2019 IJSRSET | Volume 6 | Issue 5 | pp. 230-240.

[25] A. A. E. Hajomer, L. Zhang, X. Yang, and W. Hu, ‘‘284.8-Mb/s physicallayer cryptographic key generation and distribution in fiber networks,’’ J. Lightw. Technol., vol. 39, no. 6, pp. 1595–1601, Mar. 13, 2021.

[26] D. Uschner, D. Schindler, R.-D. Hilgers, and N. Heussen, ‘‘RandomizeR: An R package for the assessment and implementation of randomization in clinical trials,’’ J. Stat. Softw., vol. 85, no. 8, pp. 1–22, 2018.

[27] Abeer Al-Hyari Charlie Obimbo, Mua’ad M. Abu-Faraj , and Ismail Al-Taharwa, “Generating Powerful Encryption Keys for Image Cryptography with Chaotic Maps by Incorporating Collatz Conjecture” , IEEE Access, VOLUME 12, pp. 4825- 4844, 2024/ DOI: 10.1109/ACCESS.2024.3349470

[28] Ronja Lindholm, “Honey Encryption: implementation challenges and solutions“ , Master’s Thesis in Information Technology June 1, 2019.

[29] Nwe Ni Khin, and Thanda Win, “A Novel Hybrid Encryption Method Based on Honey Encryption and Advanced DNA Encoding Scheme in Key Generation” , Journal of Computer and Communications, Sep. 9, 2022, pp. 22-36.

[30] Y. Guo, Z. Zhang, and Y. Guo, “Superword: A honeyword system for achieving higher security goals,” Compuer Security Vol. 103, p. 101689, Apr. 2021, doi: 10.1016/j.cose.2019.101689.

[31] Z. A. Genç, G. Lenzini, P. Y. A. Ryan, and I. Vazquez Sandoval, “A Critical Security Analysis of the Password-Based Authentication Honeywords System Under Code-Corruption Attack”, Communications in Computer and Information Science, vol. 977, 2019, pp. 125–151.

[32] Yasser A. Yasser, Ahmed T. Sadiq, Wasim AlHamdani, “Generating Honeyword Based on A Proposed Bees Algorithm”, Iraqi Journal of Computers, Communications, Control & Systems Engineering (IJCCCE), Vol. 22, No. 4, December 2022.

تقنيات تشفير العسل :نقاط القوة والقيود والتحديات

تداء فليح حسن[1]
Nidaa.f.hassan@uotechnology.edu.iq

هاله بهجت عبدالوهاب[2]
hala.b.abdulwab@uotechnology.edu.iq

اياد حازم ابراهيم[3]
ayad.h.ibrahim@uotechnology.edu.iq

رحاب فليح حسن[4]
Rehab.f.hassan@uotechnology.edu.iq

المستخلص: في الوقت الحالي ، يعد أمن المعلومات مشكلة صعبة بسبب زيادة قدرة المعلومات وإجراءات النقل .إحدى طرق التشفير التي يمكن أن توفر الأمان بما يتجاوز حد القوة الغاشمة هي تشفير العسل .(HE)في النهاية ، يصبح التعليم العالي جزءا من الأنظمة اليومية .يستخدم معظم الباحثين التشفير وفك التشفير المحسنين لمخططات التعليم العالي لتحسين مجالات أمن المعلومات نظرا لأن التعليم العالي التقليدي له العديد من العيوب .سيتم توفير نظرة عامة على التطورات الحالية في مخطط تشفير العسل في هذه المقالة .يتم تحديد عيوبه وصعوباته وفحصها .تم تقديم الحلول المحددة المقترحة..

الكلمات المفتاحية: لتشفير ، تشفير العسل ، هجمات القوة الغاشمة ، كلمات مرور العسل ، التشفير المستند إلى كلمة المرور.

[1] استاذ دكتور كلية علوم الحاسوب - الجامعة التكنولوجية بغداد - العراق
[2] استاذ دكتور كلية علوم الحاسوب - الجامعة التكنولوجية بغداد - العراق
[3] استاذ مساعد دكتوركلية علوم الحاسوب - الجامعة التكنولوجية - بغداد - العراق
[4] استاذ دكتور كلية علوم الحاسوب - الجامعة التكنولوجية بغداد - العراق