International Journal of Computer (IJC)
ISSN 2307-4523 (Print & Online)
http://ijcjournal.org/

Enhanced DNA Encoding Scheme in Honey Encryption

Nwe Ni Khin[a]*, Thanda Win[b]

[a,b]*Yangon Technological University, Computer Engineering and Information Technology, Republic of the Union of Myanmar*

[a]*Email: dawnwenikhin89@gmail.com,* [b]*Email: dr.thandawin@ytu.edu.mm*

Abstract

Nowadays, Security plays a vital role in protecting sensitive data from attackers in many organizations. Many researchers have developed security research to prevent attacks. Password-based encryption (PBE) is used to prevent an attacker from attempting to break into the password file. However, the current PBE is vulnerable because attackers can easily access keys by attempting again and again. The use of weak passwords in PBE is an ongoing problem. At present, Honey Encryption (HE) is an encryption method that overcomes (PBE) vulnerabilities. It is resistant to brute force attacks and allows encryption of data using minimal keys. HE generates a plausible message that looks real when the attacker decrypts with an incorrect key. Deoxyribo Nucleic Acid (DNA) is a new way of computing used in medical research. In this paper, DNA sequences are generated as the key distribution of Honey Encryption. The main idea of the paper is five random data lookup tables in the DNA encoding scheme in order to be more secure. It will be shown as the experimental results the same message encryption with the different passwords and the encryption of the different messages with the same password. In this system, diagnosis symptoms such as Influenza, Toothpaste, etc., will be used as the input messages of the DNA scheme. Compared to the results of only one data lookup table, it can be seen that the result of five data lookup tables in the key generation of DNA encoding sequence is more secure and less execution time. According to the experimental results, the proposed method is more secure than the existing method.

Keywords: DNA Algorithm; Honeyword Generation; Data Lookup Table; Brute Force Attack; Security.

1. Introduction

DNA scheme has encrypted a message into a DNA sequence and converted it into ciphertext. DNA was invented in 1994 by Leonard Max Adleman. Biological or molecular computation is also often defined as the concealment of information in a DNA program [1]. In the DNA code, the letter A (adenine); T (thymine); C (cytosine) and G**.** (guanine) are the four bases that represent the DNA sequence [2].

---

* Corresponding author.

DNA encryption is used early in data storage, proof of authenticity and digital certificates. DNA can also be used to produce registration cards and certificates. Honey Encryption (HE) has become a challenging technique in the security area and it can strongly against various attacks such as brute force attack and many other attacks. HE produces plausible messages if the brute force attacks attempt to enter the system using the honeywords (incorrect passwords). Otherwise, HE produces the correct message. Since DNA integration into the AES algorithm does not protect against brute force attacks in the existing system. In this research, we use DNA encode scheme in the key generation of HE. The main idea of the proposed method is to protect the brute force attack. In addition, we proposed a less complex DNA algorithm for storage security passwords. The organization of the paper is presented in six sections. In section 2, the previous researches concerned with DNA and Honey Encryption are discussed briefly. The methods of the proposed system will be described in section 3. In section 4, the flowchart of the proposed system is discussed in detail. In section 5, the experimental results are discussed, and then the conclusion and discussion are described in section 6.

2. Related Works

Bhavani, Y., Puppala, S, S., Krishna, B, J., Madarapu, S presented the DNA was modified by using AES in 2019 [3]. There are several stages in the system: the user input data (plaintext) was converted to Hexa decimal format, and then converted to binary data which was converted to DNA code. This system used several steps to prevent differential and linear cryptanalysis attacks, but not brute force attacks protection.

Bonny B. Ra, Vijay, J. F. and Mahalakshmi, T. discussed about the data was transferred securely using DNA [4]. Firstly, the plaintext was converted to ASCII value, ASCII value was converted to binary value and then the binary value was converted to DNA code. Secondly, a random key was generated in the range of 1 to 256 which corresponds to the permutation of four characters, A, T, G, and C were produced the ciphertext. Thirdly, decryption was taken place. That system provided data confidentiality and data security rather than data transmission. However, the fake messages from ASCII encryption algorithm produced the meaningless fake messages. Therefore, the attackers would know that they are the fake messages and he would try to get the original plaintext messages.

Moe, K. S. M., and Win, T. proposed improved hashing and Honey-based stronger password prevention against brute force attack [5] solved the typo error and used the other users' passwords as honeywords in the database instead of creating honeywords, reduced storage space and used a unique hashing algorithm that saved a lot of time. That paper has more execution time than the method using Honey Encryption and blowfish.

Noorunnisa, N. S and Dr. Afree, K. R. discussed Honey Encryption combined with OTP (Vernam Cipher) encrypted the original message. The security level has been increased and the time consuming was similar to a Honey Encryption combined with blowfish [6].

The system of A New Technique for Data Encryption using DNA Sequence was proposed by Pushpa, B. R [7]. The system was more powerful against certain attacks. But ASCII has a longer key length which caused memory overload. Honey Encryption is now being used instead of ASCII to prevent memory overload.

Mavanai, S., Pal, A., Pandey, R., and Asst Prof. Nadar, D. proposed Message Transmission Using DNA Crypto-System was performed transposition and folding operations to increase security and prevent brute force attack but increased complexity [8].

The proposed system was introduced file encryption and decryption using DNA technology by Kumar, B.R., Sri, S., Katamaraju, G.M.S.A., Rani, P., Harinadh, N. and Saibabu, Ch [9]. That paper contained several intermediate steps to conceal the data from attackers and hijackers. And then, securely disseminated their information.

According to the above related works, the combination of DNA and some encryption methods improve the security, protect against many attacks. It also prevents brute force attack but increase complexity. Honey Encryption is now being used instead of ASCII to reduce the memory overload and to prevent the brute force attack strongly. The method of using Honey Encryption with other encryption methods is more complicated than the method using Honey Encryption with Blowfish. Therefore, our proposed method mainly deals with two processes. Honey Encryption and encryption keys from the user's passwords are produced using a DNA encoding scheme. In the key or password distribution section, the resulting DNA code is randomly mapped to seed space using the DTE process. In addition, we propose enhanced DNA encoding using five data lookup tables in the password encoding process. The method improves security, against the brute force attack, and reduces the time complexity.

3. Proposed Method

The purpose of the proposed method is to overcome the security level and DNA password management problems and to reduce the time complexity and prevent the brute force attack. The enhanced honeyword generation, honeychecker and enhanced DNA encryption are discussed in the following sections.

3.1. Enhanced Honeyword Generation

The users are often chosen poor and repeated passwords. Most of the attackers are interested to attack the passwords file. The attackers can easily find the user's password when the password is weak. In the honeyword generation, sweetwords (the actual passwords and the plausible passwords) are stored in the main server and honeychecker. Honeywords, the fake passwords and the attackers are used to decrypt with the honeyword, the attackers received the meaningful message and is not the truth. Sugarword is the actual password. In the previous method, messages are encrypted by hashing algorithm in honeyword generation. In the proposed system, the DNA sequences are used in the key distribution of honeyword generation. DNA code sequences are generated by choosing five different data lookup tables randomly. The purpose of enhanced honeywords generation methods is to issue an appeal to remove the attackers and to detect the brute force attack on a database.

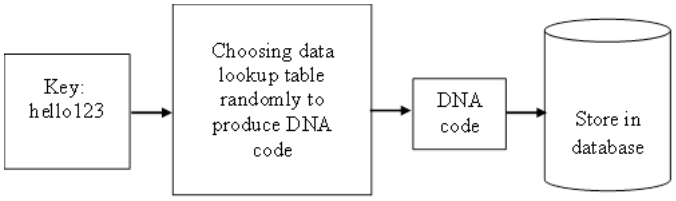

Figure 1: enhanced honeywords generation using DNA code.

3.2. Honeychecker

Honeychecker is a backup server that can be used to categorize sweetwords. The purpose of a honeychecker is to store confidential information, such as authentic passwords and DNA sequences [10]. When the user enters the system with his username and password, the main server checks the login passwords with the honeychecker. When entering a plausible password or honeyword to the system, that can be immediately identified by a honeychecker. There are two main processes of honeychecker: the first step is to distinguish which password is the plausible password and which password is the actual password when entering into the system. The next step is to send the alarm message is displayed when entering honeywords [11]. On the other hand, you must be able to login into the system.

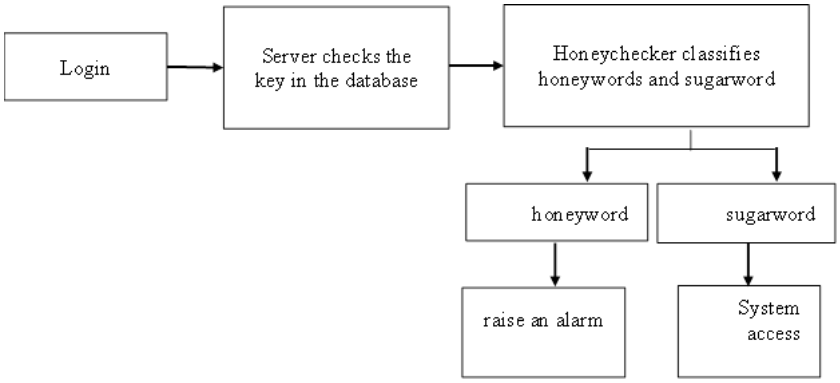

Figure 2: process of server and honeychecker.

3.3. Enhanced DNA Encryption

DNA Encryption is the process of converting text containing numbers and special characters into a DNA code sequence [7]. It transfers the characters to the DNA-based entities and send an encrypted message to the recipient. This encrypted text will be the input of the decryption process. The second approach generates the DNA code that can be used as the key of the HE processes. Therefore, encryption and decryption can be done using data lookup tables. In the proposed system, the user types the user's name and password in the user login process and then the system runs the random process. If the random result is 1, the DNA code comes from data lookup table-1. The attackers cannot know easily which DNA table is used. In the proposed system, the symptoms of the disease will be encrypted by combining HE and DNA to increase the security level. Enhanced

DNA encoded scheme steps are as followed: Step 1: Create the five DNA lookup tables for the passwords such as alphabets, numbers, special characters, and symbols as 64*5=320 characters are randomly encoded (64 means the characters mapped in Table 2 to Table 6 and “5” means the different distribution tables). Step 2: The passwords are converted into the three DNA -based sequences using a random DNA lookup table. For example, the password is "239*4026!Thanda", if the random lookup table is 1, the DNA code of the password will be ATT GAT CGA CAC GCT CTA ATT GTT TAA ATG GTA AAA GCC ATA AAA as shown in Figure 3.

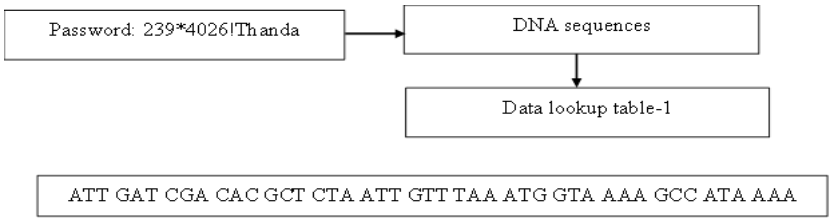

Figure 3: DNA sequences generation using data lookup table-1.

Data lookup table is created by producing DNA sequences based on (A, C, G, and T) which is DNA-based code. In the proposed method, the characters are assigned the table of 8 columns and 8 rows using triplet DNA-based code 4^3=64 as shown in Table-1. Letters A to Z, numbers 0 to 9, the special characters and symbols on the keyboard are can be used as passwords, can be shown in Table 2 to Table 6 and they are distributed from the different mappings. The encryption and decryption processes, these data lookup tables are randomly used. Table 2 distributes the characters A to H from up to down so that character “M" is assigned by column 5 and 2 row (5,2) will be the triplet DNA-based “GAC” as shown in data lookup table 1. And the other from down to up in Table 3, character “M" is assigned column 5 and 7 row (5,7) will be “TAG”. And then, Table 4 distributes from left to right, character “M" is assigned by column 2 and 5 row (2,5) will be “CCA” and right to left distribution is used in Table 5, character “M" is assigned column 7 and 5 row (7,5) will be “TGA”. Finally, the characters are diagonally distributed in Table 6, character “M" is assigned column 5 and 6 row (5,6) will be “TAC”. The results of DNA sequences are not the same in similar character “M” because of different table distributions are shown in the following tables.

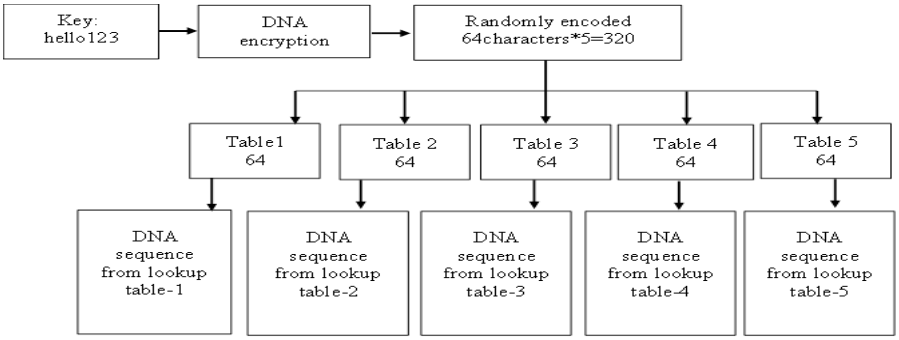

Figure 4: enhanced DNA encryption.

Table 1: Triplet of DNA-based distribution.

AAA	ACA	AGA	ATA	GAA	GCA	GGA	GTA
AAC	ACC	AGC	ATC	GAC	GCC	GGC	GTC
AAG	ACG	AGG	ATG	GAG	GCG	GGG	GTG
AAT	ACT	AGT	ATT	GAT	GCT	GGT	GTT
CAA	CCA	CGA	CTA	TAA	TCA	TGA	TTA
CAC	CCC	CGC	CTC	TAC	TCC	TGC	TTC
CAG	CCG	CGG	CTG	TAG	TCG	TGG	TTG
CAT	CCT	CGT	CTT	TAT	TCT	TGT	TTT

Table 2: Mapping the characters for table-1.

A	B	C	D	E	F	G	H
I	J	K	L	M	N	O	P
Q	R	S	T	U	V	W	X
Y	Z	1	2	3	4	5	6
7	8	9	0	!	@	#	$
*	?	/	>	<	~	Space	|
\	_	=	+	-	,	.	:
;	%	&	^	(	)	[	]

Table 3: Mapping the characters for table-2.

;	%	&	^	(	)	[	]
\	_	=	+	-	,	.	:
*	?	/	>	<	~	Space	|
7	8	9	0	!	@	#	$
Y	Z	1	2	3	4	5	6
Q	R	S	T	U	V	W	X
I	J	K	L	M	N	O	P
A	B	C	D	E	F	G	H

Table 4: Mapping the characters for table-3.

A	I	Q	Y	7	*	\	;
B	J	R	Z	8	?	_	%
C	K	S	1	9	/	=	&
D	L	T	2	0	>	+	^
E	M	U	3	!	<	-	(
F	N	V	4	@	~	,	)
G	O	W	5	#	Space	.	[
H	P	X	6	$	|	:	]

Table 5: Mapping the characters for table-4.

;	\	*	7	Y	Q	I	A
%	_	?	8	Z	R	J	B
&	=	/	9	1	S	K	C
^	+	>	0	2	T	L	D
(	-	<	!	3	U	M	E
)	,	~	@	4	V	N	F
[	.	Space	#	5	W	O	G
]	:	|	$	6	X	P	H

Table 6: Mapping the characters for table-5.

A	!	>	_	.	%	)	]
I	B	@	<	=	:	^	]
P	J	C	#	~	+	;	(
V	Q	K	D	$	Space	-	%
1	W	R	L	E	*	|	,
5	2	X	S	M	F	?	\
8	6	3	Y	T	N	G	/
0	9	7	4	Z	U	O	H

4. Flow of Proposed System

This flowchart of the proposed system includes the following two portions as shown in Figure 5. These are honeyword generation and DNA code sequence generation. In the first part, a new user needs to be registered. A new user can only login using the username and password after the registration process. If you are not a registered user, the password does not exist in the database. The server is retrieving the message that the login failed. If the password is in the database, the server generates the passwords to honeychecker to distinguish which is the actual password and which are the honeywords. The honeychecker allows this user to access when the password is correct. Otherwise, the system displays the alert message.

The second portion is DNA code sequences production using a DNA encoding scheme. The first step is to convert the key or passwords into DNA code and the next step is to map the resulting DNA key into seed space

using the Distribution Transforming Encode (DTE) process. We use the five different data lookup tables to be secured the proposed method to convert the DNA code sequence. This DNA code sequence is used as input of the Honey Encryption because DNA code generation is very fast.

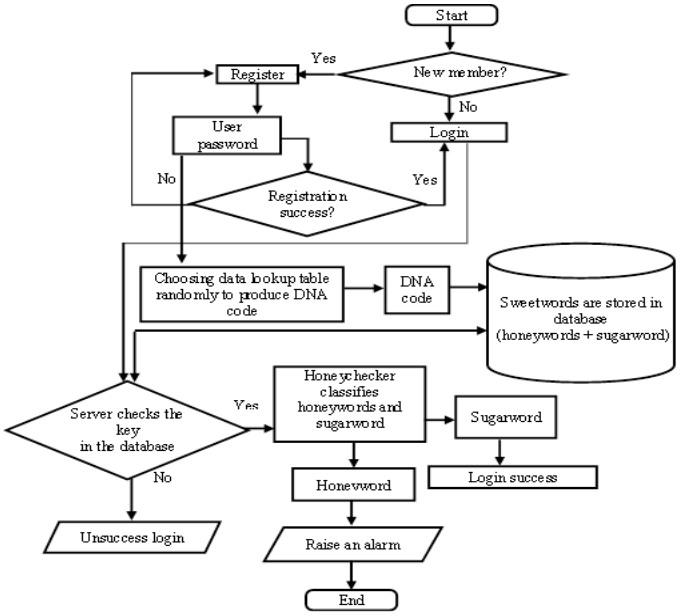

Figure 5: flowchart of proposed system.

5. Results and Discussions

The experiments are carried out the processor AMD Ryzen 5 3500U and memory 8 GB and the python programming language is used to test the experiments. Creating the different random tables generate different random DNA codes, making it more difficult for attackers, and sending meaningful messages to deceive the attacker and prevent brute force attack according to Honey Encryption. The experiment results are discussed in the following sections.

5.1. DNA Code Sequences Generation from Same password

The different DNA code sequences are generated from the same password "3@Wint" by using data lookup Table 1 to Table 5 as shown in Table 7. The results of DNA sequences are not the same because the outputs are depended on the characters distribution of the tables.

Table 7: Different DNA code sequences from “3@Wint”.

Table No.	Password	DNA Code Sequences
1	3@Wint	GATTCAAGTGGGAACGCCATG
2	3@Wint	TAAGCTCGATGCCAGTCGCTC
3	3@Wint	CATTACATGCGGACACCCAGT
4	3@Wint	TAACTCGAGTCGGGATGCGCT
5	3@Wint	CGGAGCCAACCAAACTCGTAG

5.2. Same Messages Encryption with Different Passwords

Firstly, the experimental results of the same message "Influenza" are encrypted with the different number of password lengths "7,10,15,25, and 30" characters using five different data lookup tables and only one data lookup table is shown in Table 8 and Table 9.

Table 8: Same messages encryption with different passwords using five different data lookup tables

Same Message Length	Different Password Length (characters)	No. of Data Lookup Tables	Execution Time (ms)
7	7	5	2.07
7	10	5	2.18
7	15	5	2.23
7	25	5	2.28
7	30	5	2.29

Table 9: Same messages encryption with different passwords using only one data lookup table.

Same Message Length	Different Password Length (characters)	No. of Data Lookup Tables	Execution Time (ms)
7	7	1	2.15
7	10	1	2.15
7	15	1	2.18
7	25	1	2.30
7	30	1	2.36

The results of the same message "Influenza" is encrypted with the different number of password lengths using five different data lookup tables and only one data lookup table are shown in Figure 6.

It can be seen that password lengths 10,15 and 25 are nearly the same in execution time and the password lengths 7 and 30, the execution time of five data lookup tables is faster than on one data lookup table.

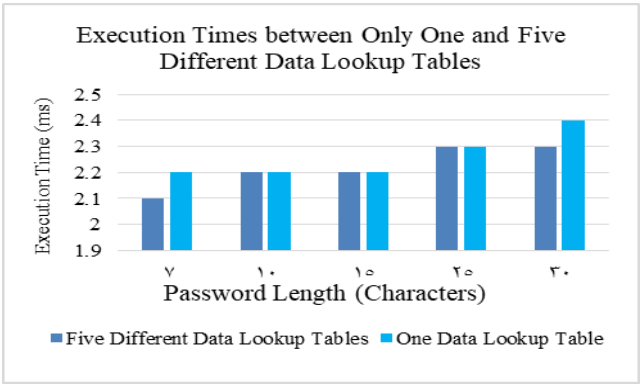

Figure 6: results chart of same messages encrypts with different password lengths using only one and five different data lookup tables.

5.3. Different Messages Encryption with Same Password Length

The experimental results of the different messages "Toothpaste" (10 characters), "Avian Influenza" (15 characters), "TB Data & Statistics" (20 characters) are encrypted with the same password lengths 7 characters using five different data lookup tables and only one data lookup table are shown in Table 10 and Table 11.

Table 10: Different messages encryption with same password using five data lookup tables.

Message Length	Password Length (characters)	No. of Data Lookup Tables	Execution Time (ms)
10	7	5	1.73
15	7	5	2.11
20	7	5	2.12

Table 11: Different messages encryption with same password using only one data lookup table.

Message Length	Password Length (characters)	No. of Data Lookup Tables	Execution Time (ms)
10	7	1	1.65
15	7	1	2.14
20	7	1	2.21

The results of the different messages encrypted with the same password lengths using five different data lookup tables and only one data lookup table are shown in Figure 7. It can be seen that message lengths 10,15 and 20 are nearly the same in execution time and the more message lengths you send, the more execution time also increased. Although, the execution time of five data lookup tables is faster than on one data lookup table.

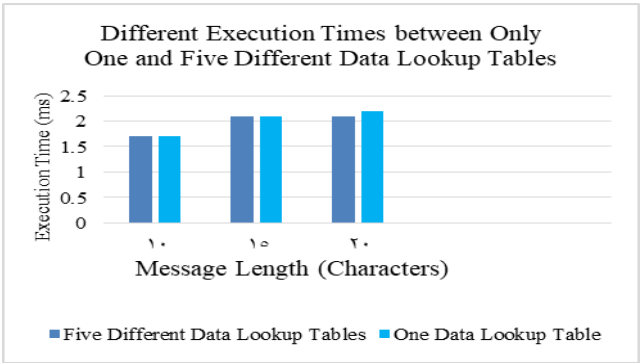

Figure 7: results chart of different messages encrypts with same password length using only one and five different data lookup tables

5.4. Same Message Encryption with Same Password Length

In this section, the proposed system is suitable for important privacy information (diagnosis symptoms) send under the same Wifi in the private hospital because the same messages send repeatedly, the less time execution time according to Table 12 and results as shown in Figure 8. It can be seen that the results of execution time were gradually reduced from the beginning.

Table 12: Same messages encryption with same password length using five data lookup table.

Same Message Length	Same Password Length (characters)	No. of Data Lookup Tables	Execution Time (ms)
7	30	5	1.05
7	30	5	0.98
7	30	5	0.81
7	30	5	0.77
7	30	5	0.72
7	30	5	0.66
7	30	5	0.63
7	30	5	0.62
7	30	5	0.60
7	30	5	0.59

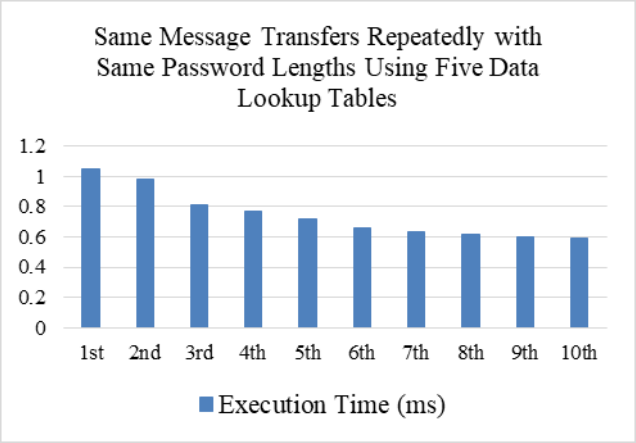

Figure 8: execution time of same message transfers repeatedly with same password length using five data lookup tables.

5.5. Different Messages Encryption with Different Password Length

Finally, the different messages encrypted with different password lengths compared to the other generations of honeywords using a hashing algorithm and our proposed models using a DNA encoding scheme. The time complexity of the proposed method is faster than the existing method [5] is shown in Table 13 and the results are shown in Figure 9. It can be seen that the proposed method which encrypts the seven characters that are the most commonly used passwords by users, is 1.46 ms faster than the existing method. And the others, about 0.8 ms faster than the previous method.

Table 13: Time comparison results of existing method and the proposed method.

Password Length (characters)	Time Complexity of Existing Method (ms)	Time Complexity of Proposed Method (ms)
7	3.050004	1.59
8	3.100004	2.29
9	3.140004	2.34
10	3.18005	2.37

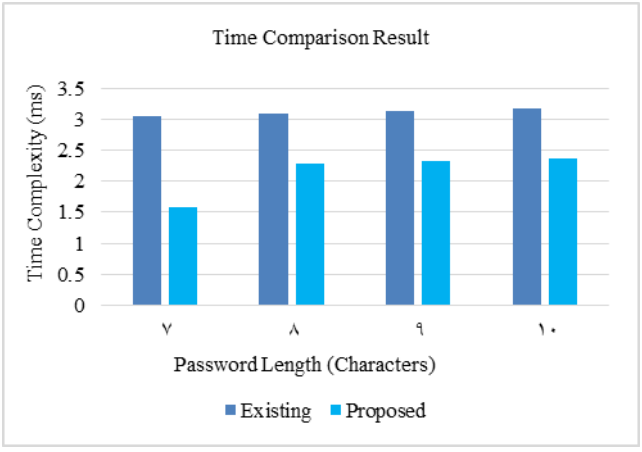

Figure 9: results chart of time complexity difference between existing and proposed methods.

In Figure 6, the same messages encrypted with different password lengths using five different data lookup tables results are faster than the only one data lookup table in execution time. On the other hand, the more passwords are used, the more secure and also the execution time is increased. Then, different messages encrypted with the same password length using the five different data lookup tables are faster than only one as shown in Figure 7. When the message lengths are increased, the execution time also increased. However, the current system using five different data lookup tables not only makes it more secure but also safe in time consumption. The current system is suitable for important privacy data sent under the same Wifi in the private hospital because the same messages send repeatedly, the less time execution time as shown in Figure 8. Finally, the time comparison

results of the proposed method are faster than the existing method in time complexity as shown in Figure 9. So, the proposed system supports data security and faster transmission time over the data transferring. And the honeywords are produced from the registered user passwords. Therefore, the proposed system also reduced storage overhead.

5.6. Limitations

Since our proposed system is based on the patient's symptoms, it can only be used up to 30 characters for the message space, but the program is limited to a maximum of 255 characters. Most users can usually use a maximum of 30 characters, so our proposed system was tested up to 30 passwords. The limitation of the study in this system is that users can securely transfer privacy information on the web to each other over a single Wi-Fi network.

6. Conclusion and Discussion

The proposed paper combines DNA encoding and honey encryption for fast execution time and a secured algorithm. An enhanced DNA scheme is developed by modifying HE and applying DNA code sequences. The encryption process is the same as HE but uses DNA encoding is used in the password distribution of the HE to provide more security from the attacks like brute force attacks. The proposed algorithm can be applied for transferring the important information (disease symptoms) securely between senders and receivers on the web in the same private Wifi. DNA encoding is used for key generation by using randomly encoded from five different data lookup tables. The information is converted to ciphertext using DNA-based encryption to protect information from attackers and that is widely used in many organizations. The recipient converts to the original text using the decryption algorithm. The proposed honeywords generation process that uses a DNA sequence can solve a new DNA algorithm to make our system more secure and faster processing time compared to the previously proposed system. In the future, DNA algorithms will be widely used to store large amounts of data securely compared to other cryptographic methods. This process supports data security and reduces the execution time over the data transmission. Therefore, the computational capacity of the proposed algorithm can be attained only in DNA-based computers which are to be developed in near future.

References

[1] S. Mona, H. Mohamed, N. Taymoor and K.E. Mohamed. "Design of DNA based Advanced Encryption Standard (AES)", 2015 IEEE Seventh International Conference on Intelligent Computing and Information Systems (ICICIS), Cairo, 2015, pp. 390-397.

[2] H. Tatiana, V. Mircea-Florin, "Alternate Cryptography Techniques", ICCC05, Miskolc-Lillafured, Hungary, 2005, Vol. 1, pp513-518.

[3] Y. Bhavani, P.S. Sai, B. J. Krishna and M. Srija. "Modified AES using Dynamic S-Box and DNA Cryptography". Proceedings of the Third International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC 2019) IEEE Xplore Part Number: CFP19OSV-ART; ISBN:978-1 7281-4365-1, 2019.

[4] R.B.Bonny, V. J. Frank and T. Mahalakshmi (2016, Jan). "Secure Data Transfer through DNA Cryptography using Symmetric Algorithm". International Journal of Computer Applications (0975 – 8887) Vol. 133 – No.2, January.

[5] M. K. S. Myat. and W. Thanda. "Improved Hashing and Honey-Based Stronger Password Prevention against Brute Force Attack". 2017 International Symposium on Electronics and Smart Devices, 978-1-5386-2778-5/17/$31.00 ©2017 IEEE, 2017.

[6] N. N. Syeda, Dr. A. K. Rahat (2019, May). "Honey Encryption based Password Manager". JETIR, Vol. 6, Issue 5, www.jetir.org (ISSN-2349-5162).

[7] B. R. Pushpa, "A New Technique for Data Encryption using DNA Sequence". International Conference on Intelligent Computing and Control (I2C2), 2017.

[8] M. **Saifali**, P. Ajay, P. Ravi. and Asst Prof. N. Deepika (2019, April). "Message Transmission Using DNA Crypto-System". International Journal of Computer Science and Mobile Computing, Vol.8 Issue.4, pg. 108-114, April.

[9] K. B. Mohan, B. R. S. Sri, G.M.S.A. Katamaraju, P.Rani, N.Harinadh Ch.Saibabu. "File Encryption and Decryption Using DNA Technology". Second International Conference on Innovative Mechanisms for Industry Applications (ICIMIA) IEEE Xplore Part Number: CFP20K58-ART; ISBN: 978-1-7281-4167-1, 2020.

[10] Moe, K. S. M. and Win, T. "Protecting Private Data using Improved Honey Encryption and Honeywords Generation Algorithm". Advances in Science, Technology and Engineering Systems Journal Vol. 3, No. 5, 311-320, 2018.

[11] J. Ari. and R. L. Revist. "Honeywords Making Password Cracking Detectable". In MIT CSAIL, 2013.