National College of Ireland

HoneyBlow - An enhanced hybrid encryption method for Messages.

MSc Research Project
Master of Science in CyberSecurity

Arpit Dharod
Student ID: X22186964

School of Computing
National College of Ireland

Supervisor: Eugene McLaughlin

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name: Arpit Mukesh Dharod

Student ID: X22186964

Programme: MSc in Cybersecurity **Year:** 2024-25

Module: Research Project

Supervisor: Eugene McLaughlin

Submission Due Date: 12th August 2024

Project Title: HoneyBlow - An enhanced hybrid encryption method for messages

Word Count: **5886** **Page Count 20**

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: Arpit Mukesh Dharod

Date: 12th August 2024

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

HoneyBlow - An enhanced hybrid encryption method for messages.

Arpit Dharod
X22186964

Abstract

In today's world, there are more cyber-attacks going on due to increased utilization of web usage. The most important thing is to focus on protecting the user accounts through secured passwords/credentials. Despite of various research on strong passwords /credentials, yet confidentiality of credentials is not possible and is not reliable till now. The concept of cryptography has been widely accepted, there can be possibility of breaking unless the attacker somehow intercepts the communication and finds the key. The suggestion of model is to integrate Honey encryption (HE) with Blowfish encryption. As honey encryption prevents from common password-based attacks. The proposed model tries to build combination of Blowfish with Honey encryption. The approach of two layers of encryption applied ensures the safety of passwords and data security. As it makes honeyword utilize for prediction of fake password and confuse the attacker for the password or credential breach. This paper also classifies the creation of honeyword as well as how honey encryption typos errors can be avoided. One of the main objectives is to study about honey word creation which can be implemented for communication.

1 Introduction

In today's rapid increase of technologies cybersecurity is crucial due to its role in safeguarding communication and business activities. The increase in use of digital platforms for storing personal data, financial transactions and confidential information such as government information, data privacy & security is major concern. Encrypting data is one method of enhancing security from multiple methods. Using concepts of cryptography, data is converted into an unreadable form which is a ciphertext. Cryptography can be divided into many categories of procedures. The two majors methods are symmetric encryption and asymmetric encryption. The most used encryption method by many organizations in this digital era to secure data, passwords and other confidential information is password-based encryption also known as PBE. The passwords created by users are easily guessed or can be cracked with tools. Repeated passwords by user at multiple websites can result in losing confidential information. One of the most commonly binary models used is ASCII (American Standard Code for Information Interchange). The technique made an advantage of a statistical coding strategy to create a fake plain text using the ASCII, however since the plain text was not encrypted, the attacker can easily determine which password is valid and invalid. As a result, using same passwords at multiple websites can cause common outcome of Brute force attack using PBE. This is critical because the brute force attacks are significantly dangerous in leaking confidential information. For this year 2024, it has been estimated that brute force attack breaches would be 35%(*The State of Identity Security for 2024*, 2024). The use of hybrid

encryption model started the plain text not encrypted making it easily to attack and know password of the user. Which is the main reason of two encryption techniques used these days. The flow of hybrid encryption model works is shown below in figure 1.

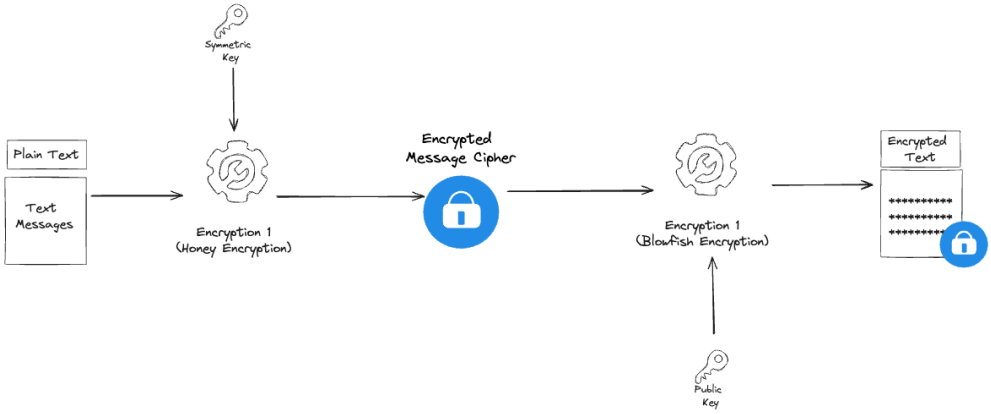

Figure 1: Hybrid Encryption Flow.

The (Erguler, 2016) paper stated that Honey Encryption works as a secondary protective layer. There are many previous systems designed with a single layer safeguarding technique that allows encoding the credentials. To prevent against such attack, honey encryption is developed to offer as a protective layer. This technique of Honey encryption is possibly a strategy that makes an attacker mislead with correct credentials. Honey encryption is beneficial as it has the capability to safeguard during brute-force attack without causing harm to server.

The usage of honey words will fool the attacker by unwanted words that are processed during encryption making attacker mislead with the correct one. Major advantage of HE is it can't be accessed like Password Based Encryption. Utilization of honey encryption can be made in various factor like authentications, personal information, etc., The pre-existing models of Distribution Transforming encode was created to improve the effect of Honey encryption. In recent combination of AES and HE, the resources required with high performance of GPUs and CPUs can help to accelerate process. The benefits of honey encryption can be used in various scenarios for personal data and to protect against any type of malicious attack. The paper from (Dibas and Sabri, 2021), as AES is widely used and considered as secure but still shows slower performance. However, Blowfish can perform more complex key schedule and encryption making it faster and resistant with many attacks. The papers that were used also classifies the utilization of RC2 model, provides disadvantage in cycles in which it results similar outcome to the AES model (Vivek Raj, H. Ankitha, N.G. Ankitha and L. S. Kanthi, 2020).

1.1 Motivation

Various flaws across verification and authencity of security practices and multiple caused on systems making it a threat and leaking crucial information without authorization. The attacks now days can be easily cracked regardless of any strong hashing technique or any data encryption technique applied as a security measure to protect against it. This research of hybrid encryption is to secure the transmission of message encryption without any eavesdropping or any attack. As honey encryption technique improves the security level against various kind of attack. The research on various new techniques is untested. The use of honey model encryption

is a new technique of research that can prevent attack and loss of data. In this study, will be going through the new hybrid encryption model of Blow fish encryption and honey encryption model.

1.2 Research Problem

If combined with Blowfish encryption, how can honey encryption protect the password/ credential to secure the data.

1.3 Structure of the Report

In chapter 2, literature review on research for honey encryption is provided. The related work focuses on the progressive steps that were required to solve. This research is done with consideration of Distribution-Transforming Encoder (DTE), hybrid models which already exist, honey word production strategies, and explanations of the constraints of these models.
The subsequent section 3 gives an overview of the research methodology about simple honeyword creation methods and how an easily implementable of DTE can be developed. Later, the implementation is provided regarding the screenshot of the results, including the honey word formation, method of message retrieval, and blow fish encryption.
Then, the assessment of the proposed model is carried out to determine its effectiveness. The evaluation is performed by calculating the change in the encryption and decryption time to see with the changing password size.
In end, AES and Blowfish algorithm performance has been described and followed with conclusion and learnings from proposed model with required references.

2 Related Work

In these days, there are many various types of attacks increasing rapidly making it more complex for securing personal information. There are more than 193 million passwords that are cracked every hour (Antonov, 2024). It has possible to describe plenty of encryption models and hashing techniques which are used to protect data in idle and In-transit data. To protect password storage, one method that can be implemented is honey encryption or sweet words. This is also referred as strategy of deceit, where a defender tries to make the attacker believe what they are attacking is not worth defending. This use of hybrid models, making two or more encryption model integrate to get sustained output. Further, I will discuss on honey encryption, its applications and in detailed explanation for Blowfish encryption.

2.1 Honey Encryption

Honey is a term which is frequently used for presenting fake information in security. Honeypots are best option for honey. As fake server attracts intruders, to target user that can be caught using fake websites. The security of credential can be suggested by honey words also knows as sweet words. The user tries several numbers of fake passwords that results in adding layers of attempts to the real password. This technique creates an obstacle for the attacker to figure out the real password. The use of honey words was in 2013 by (Juels and Rivest, 2013). The method used was to add one credential on other credential making the

original credential saved in history and causing storage issue. The below showed example in diagram:

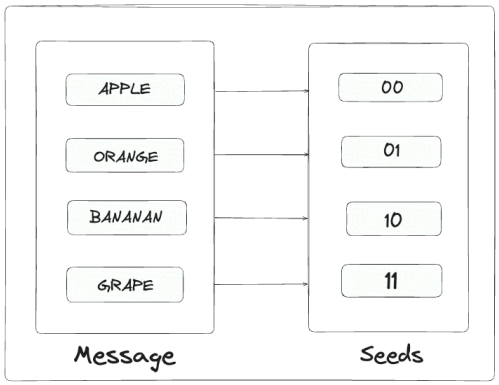

Figure 2 : Mapping of Words in Honey Encryption

As a result, the honey encryption impacts the performance of time and restoring of credential. Other solution for the credential was to make a list of certain credentials which can be used for the place of the model for credential storage known as Circular list of honey words (Chakraborty and Mondal, 2017). In the list of credentials are stored in order that goes through circular direction puzzling the attacker to undetermined the real password. However the drawback is that it still create a risk as there's no security on encryption that can easily lead to processing the model (Lindholm, 2019).

2.2 Honeyword Creations

This technique has many different methods of creating honey words as mentioned by (Pagar and Pise, 2017) which as contains tweaking, toughnut and many more. As (Erguler, 2016) in this paper the author creates, manipulates and alters some alphabets for using same credentials. The following technique used in paper has not been that effective, but credential is stored in the same file. The paper used MCA algorithm which shows accuracy of 0.707389 for creating sample of honey word passwords. Below table 1 shows the different methods of honey word generation (A.Ahmed, M. 2023).

Table 1: Different model for generating honeywords(Juels and Rivest, 2013).

Technique	Description	Example
Tweaking Digit	Small alteration made to real password to create honeyword.	Real: pass1234 , Honeywords: pass1235
Password Model	Using probabilistic model to generate fake password	Real: qwerty , Honeywords: qwjrtx
Hybrid Model	Combination of Tweaking and password model	Real: admin123 , Honeywords: admin124, qweasd
Transforming Function	Transforming Function to honeywords.	Real: pass2023, Honeywords: pass2024

2.3 DTE (Distribution Transforming Encoder)

The concept of DTE uses a function which is used to separate and combine with the different security concept. The research conducted by (Shen, C., Yu, T., Xu, H., Yang, G., Guan, X., 2016), a lot of users that tends to know the login details of all users, logged in using their prior

login details. As the technique used before used to created fresh honeywords every time, that resulted in ambiguity of the different user's systems which had transferred user's same credentials. Honeyword-based authorization model technique (HBAT) is a security mechanism which enhance security by using honeywords with real credentials. The author, (Chakraborty and Mondal, 2017) developed a new UI based honeyword generation technique which is called as PDP(Paired Distance Protocol). The purpose of the author is to reduce the storage issue for the reason PDP was introduced which helps the storage issue solve by reusing, and using robust methods which will not further require space to produce honeywords.

There was other technique which was introduced by (AlMuhanna, AlFaadhel and Ara, 2022), where the credential manager which used honeywords encryption that worked as if any user had the same honeywords detected. However, if the user attempts, the user is blocked with a notification to admin where if the owner enters incorrect credential, they can lose their account. But the following consequence as of the owner himself attempt honeyword, they will be flagged and blocked getting caused in a serious problem.

The other solution was against most common and harmful attack, (mohammed, KURNAZ and Mohammed, 2020) created a honey encryption validation technique model through java library. The user who setups this model can create the approach of credentials input by attacker, if the attackers cross the input he/she will be blocked with a unresponsive webpage. If the user forgets their credential and tries multiple entries, it will become inaccessible to webpage.

These days data is saved in shared server which are available online and can be easily destroyed by multiple tools. So, (Nirmalraj and Jebathangam, 2022) created a method of altering keypad dynamically making it tough for the attacker to access. But if we considered this method, it would take a lot of time to analyse the credential. Therefore, the attacker won't be able to access and for user it would excusive take a lot of time to analyse. This can also be solved in future by machine learning algorithm.

(Chen, H.-C., Wijayanto, H., Chang, C.-H., Leu, F.-Y., Yim, K., 2016) used MIMS which helps in maintaining a cheap and powerful portable tool. It uses the Diffie-Hellman Key exchange for one time pad encryption that can be receive on user mobile. The utilization of resources that makes the security level at highest using Crypt21, and RTC-MTT. There were some recommendations to protect session keys that could be helpful in securing and being responsive.

Protecting data from attacker which is important by generating fake data generation model created by (Sahu, 2020). (Omolara, A.E., Jantan, A., Abiodun, O.I., Poston, H.E., 2018) research used Natural language toolkit from Stanford and Wordnet from Princeton was utilized making honey encryption to decrypt the messages possible by creating honey messages on fixed data. But, due to the growing technology, it become impossible to provide an efficient solution on the technique.

2.4 Hybrid Encryption Model

Researchers are utilizing multiple hybrid models which consist of encryption algorithm or hashing algorithm with honey encryption model. (Moe and Win, 2018) incorporated salting and hashing technique to enhance security and time complexity of the algorithm which proved to be much better before. There are multiple servers used which are the data management to store personal data. When other servers use honeywords, it sends reports to admin if entered

with the same honeywords. To enhance security (Burgess, J. 2017) it was recommended to employ the RSA algorithm to enhance encryption, considerable enhancements in terms of encryption and bits related to brute force attacks.

Honey Encryption with Blowfish encryption and AES are combined making it to protect the system against any type of brute force or multiple attacks (Sahu and Ansari, 2017). This makes a clear view of these two processes to honey encryption which are connected with each other. The researcher (Shamini, P.B., Dhivya, E., Jayasree, S., Lakshmi, M.P., 2017) creates a server that make a false request from separate server to user. He implemented that on a website to check whether it works or not and it was tested by user with multiple login attempt trying of DoS attack. When honey checker identifies the user is blocked and it the web request received from the user. Therefore, the owner of website would be able to monitor website and user trying to login would be blocked to visit the website.

The proposed paper from (Sahu and Ansari, 2017) that explains that the combination of HE and Blowfish has been efficient than in combination with AES for encryption and decryption where Blowfish had taken 248 milliseconds and AES had taken 250 milliseconds. It was tested by the (Dibas and Sabri, 2021) that Twofish does not outperforms AES in terms of big data file with size more than 1MB making it more utilization of memory and additional resources.

The performance analysis was carried out (Verma and Singh, 2012; Patel and Kamboj, 2016) which discussed the three algorithm. If Blowfish is faster in performance and is inversely proportional to key size, it will decrease performance while increases its key size increases and vice versa. Also, to increase security for transmission of data RC5 was used by (Vivek Raj, K., Ankitha, H., Ankitha, N.G., Kanthi Hegde, L.S., 2020) that provides stronger rotation of rounds of randomness of 42% in 12 rounds. These are the techniques and model of latest research for hybrid encryption model.

3 Research Methodology

The overall analysis of the related work provides a base foundation for the research methodology. In fact, the major research on hybrid model is combination of Honey encryption and Blow fish encryption. The procedure to achieve the secured model, the following steps were followed: Plan, Design, Code, Test. The working of the Blowfish encryption and different methods that can be secured, efficient to achieve accurate results are discussed in this report.

3.1 Random Honey Word Creation Model

The main use of Honey encryption is to make the attacker try multiple tools and techniques which cannot be broken easily. The main objective is to generate different honey words that can be used in Honey encryption. The bogus credential from honey words used by attacker can notify if any assault has been caused. It also makes difficult for the hacker to find the correct credential(Jordan, 2021).

There are multiple ways to save credential in a server or it can be created by generating honeyword algorithm. The credentials are validated by utilization honey checker. The checker can notify user with sound and report information with destruction that was analysed for future purpose (Jain, Muntean and Verma, 2023).

The message and credentials are taken as input data that pick random seeds of dictionary credentials which are seeds to message. After that, the word are created by changing the alphabets with technique of tweaking and tailing that works as bogus credential.

3.2 Blowfish Encryption

After generation of honey words, the credential is secured by Blow fish encryption. It is type of block encryption which makes text to 64 bits. There are some issues which needs to solve and can be solved easily by this method. So, the test length is 64bits that is divided into two-halves each of 32 bits using bits of encryption model (C and K, 2023). Both of section is used for encryption not like Twofish also, blowfish does not use whitening process it straight away uses Feistel network. The other function it uses four S-boxes for non-linear substation and one p-Array for complex key transformation and in end this all will come together forming a group making a block of 64 block (Verma and Singh, 2012).

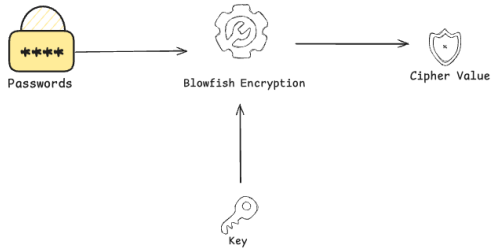

Figure 3 : Blowfish Encryption Technique

3.3 Decryption and Retrieval

Once the data is encrypted its credential is decrypted making it check with original credential and honey words. The decryption of encrypted data can cause 3 different scenarios that can lead to successful decryption, or loss of the data. The following are 3 scenarios:

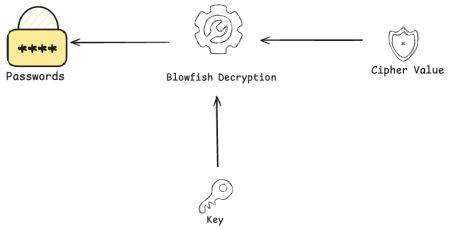

Figure 4: Blowfish Decryption Technique

Scenario1: The credential Matches with the Original Credential.

A notification will appear making it was decrypted.

Scenario 2: The credential Matches with list of honey words.

If it matches it will notify with words to user and data would be easily accessed to the screen.

Scenario 3: The credential doesn't Match.

The attacker will continue this in loop making a wrong credential without any response.

4 Design

The following is design for the research with aim to execute a simple, feasible approach for honey word generation, and user-friendly approach. It uses Blowfish encryption to secure credential and followed with honey encryption to protect from attacks.

4.1 Blowfish Algorithm

Encryption

1. Initialize
2. Take input message
3. Prepare for honey encryption
4. Select random seed value
5. Mapping of seeds to passwords and vice versa.
6. Generate honeyword by changing the password using tweaking and tailing technique.
7. Validate block bit size of 8bytes. If more than in multiple of 8 add padding.
8. Encrypting the password with Blowfish algorithm.
9. Provided would be encrypted blocks as cipher text.
10. Stop

Decryption

1. Input Password
2. If the user matches with password its original password, follows step 5
3. If the user matches with list of honey word, follow step 6
4. If the user doesn't match password, follow step 7
5. Print of the correct data will appear.
6. Print of random data will appear making the owner notify about the attack.
7. Print the wrong password.

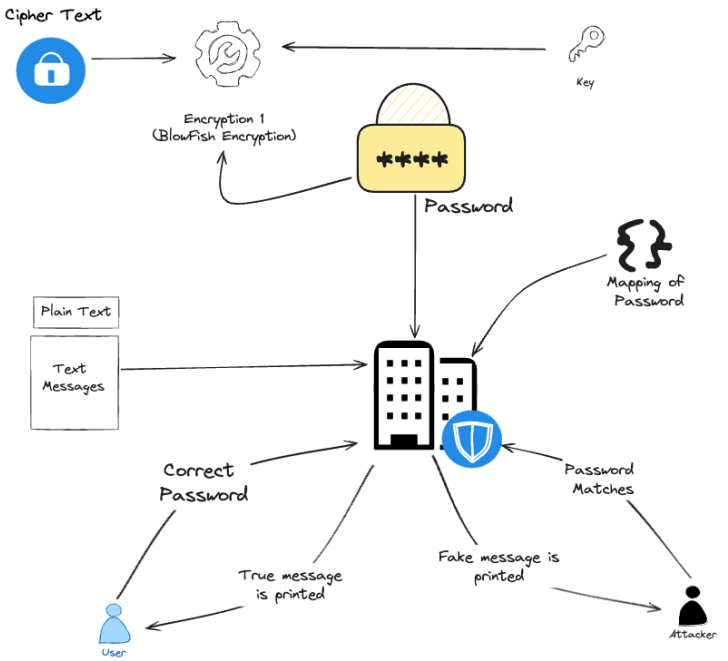

Figure 5: HoneyBlow model

5 Implementation

In this section, will discuss about the implementation of the research with Mac os operating system.

Model: MacBook Air M2 @ macOS Sonoma Version 14.5 (23F79)
Processor: Apple M2 chip (8 core CPU)
Memory: 8 gigabytes

Also, Visual studio code was used for code editor for the python3 programming language. There are multiple python files created which are Aes, blowfish, honey encryption, bits_by_bits(avalanche effect), and the main(HoneyBlow model).
The process of encryption was through an input text of message is sent, that gets honey encryption with random seeds, and a coded dictionary is created with list of states from US. Same other dictionary has original passwords and honey word is generated to protect original password.

Honey Encryption: sk ⊕ sm
As using the US list of honey words dictionary and mapping passwords to seed and seed to message, encrypting with honey encryption(Jordan, 2021).

Pseudo Code:

1. Gather User input.
2. Initialise dictionaries (Passwords to seeds & seeds to Messages)
3. Encode message
4. True seed (Original message, Users password)
5. Cipher ← ms ⊕trueseed
6. Output Cipher text

This is output for Honey Encryption

```
arpitdharod@Arpits-MacBook-Air Honey_encryption-main % python3 honeyencryption.py
Honey Encryption Initialisation...
Please enter a password: Thesis@Semester3
Please enter a secret message to store (one word): NY
Your password is Thesis@Semester3, your seed value is 11, and your secret message is NY
================================================================================
passwordsToSeeds:
{'THESIS@SEMESTER3': 16,
 'THESIS@SEMESTER3135': 17,
 'Thesis@Semester3': 11,
 'Thesis@Semester310': 12,
 'Thesis@Semester391': 13,
 'thesis@semester3': 14,
 'thesis@semester3123': 15}
seedsToMessages:
{11: 'NY',
 12: 'Alabama',
 13: 'California',
 14: 'Florida',
 15: 'Texas',
 16: 'Tennessee',
 17: 'Washington'}

Honey Encryption Done...
================================================================================
```

Figure 6 : Honey Encryption output.

Blowfish Encryption:
It is symmetric key encryption designed for effective and user friendly making it easier to handle cryptographic operations. As the passwords are set as a block of bytes , known as cipher. It is encrypted in equal size of blocks making a cipher text (O, 2024).

```
arpitdharod@Arpits-MacBook-Air Honey_encryption-main % python3 test.py
Blowfish encryption started...
Encrypted text: b'\xea<\xe1\xf3\x87\x86\xf1\x95F\xf6g\xf3\xa9V\xed\xfa'
Encryption time in milliseconds: 0.234
```

Figure 7 : Blowfish Encryption Output

AES Encryption:
The encryption is initialized by AES 128 bit which is selected random. Passwords and key are encrypted to AES technique which is then encrypted with CBC technique.

```
AES Encryption Started.....
Encrypted: b'j}c4\xc9f\x80\xde4KI\xbd9L \x1b\t\x14*KA\xe98S\x03\x1c\xfe\xe5\xd5\xc0\xcb\xec'
Decrypted: mysecretpassword
```

Figure 8: AES Encryption Output

Decryption and retrieval of honey words: sk ⊕ Cipher → Data
The data is decrypted in form of AES and Blowfish, which depends on the decryption process. There after the honey decryption process takes place. Once the password is decrypted the message/data is decoded with multiple scenarios:

Pseudo code:

1. Cipher Input text.
2. Trueseed ⊕ Cipher → encoded data
3. Decoding to encoded data to the message.
4. Output Message/Data.

Scenario 1: The Password is Right. The message is decrypted with the user as NY and the dictionary used was US with the Secret message as New York.

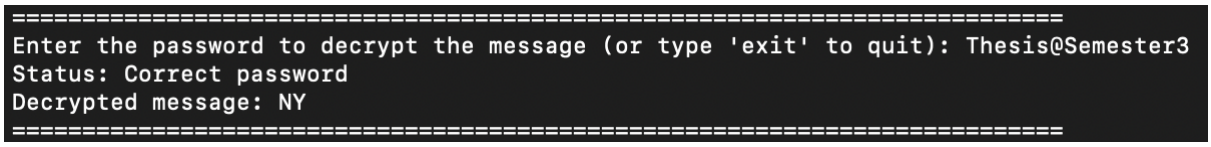

Figure 9: Correct Password with Decrypted Message

Scenario 2: Matches honey word and notification alerted. The password used was of another list of Alabama.

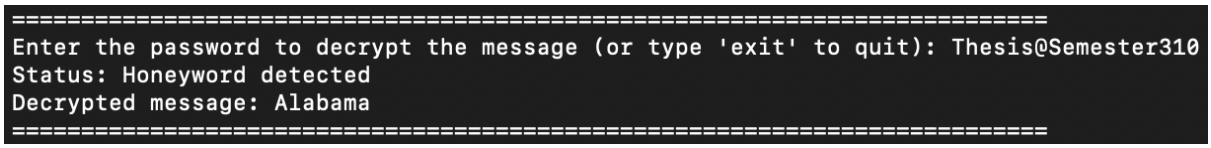

Figure 10: Password matches with honeyword.

Scenario 3: Password was incorrect. The password it was wrong password.

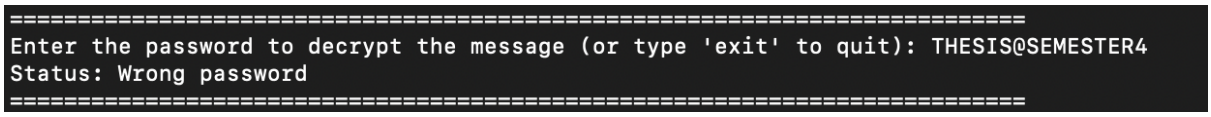

Figure 11 : Incorrect Password no message shown.

The overall scenarios can be identified that if the password matches you get an appropriate result. Making an attacker confuse about the fake message been printed that will make real message/data hide also the attacker would be unknown about the message/data.

6 Evaluation

The following section describe the evaluation for research model. This model is evaluated based on execution, encryption and decryption time. This analysis was done by calculating the performance of this research. This performance of security is calculated by avalanche effect that is compared with AES model.

6.1 Performance of encryption and decryption on various data size.

As encryption and decryption of file changes with the input data making it a variation of multiple data size. Blowfish algorithm been one of hybrid mechanism is used to which can calculate large data size faster and small. The below graphs show the when the packet size of data is increasing the time decreases.

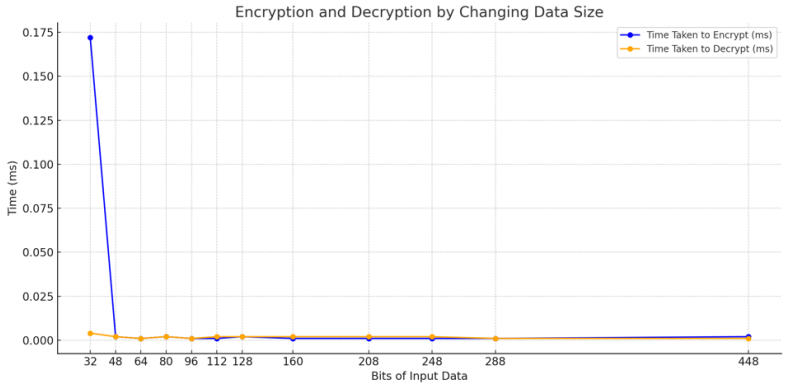

Figure 12 : Performance of Encryption and Decryption time.

The analysis of Honey Blow model is calculated by total execution time for encryption of honey encryption, blowfish encryption, blowfish decryption and data retrieval.

$$\text{Honey Blow Execution Time} = (\text{Honey Encryption time} + \text{Blowfish encryption time} + \text{Blowfish Decryption} + \text{Retrieval of text message})$$

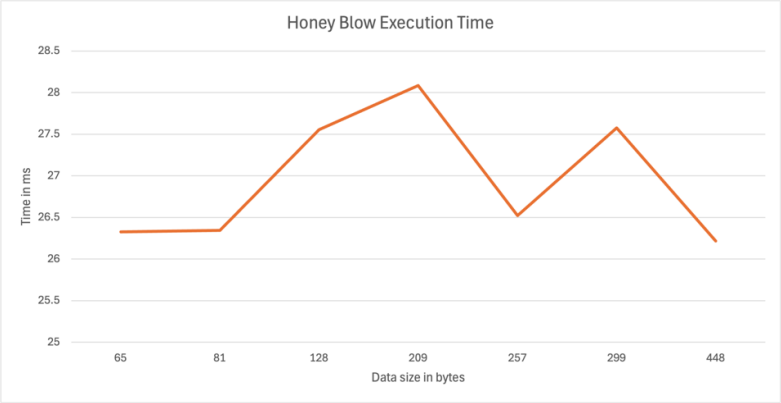

Figure 13: Honey Blow model execution time.

The performance of Hybrid Encryption would be on Throughput analysis which is depended on length of password getting changed. The Throughput is calculated through the size of data in (MB) upon the total time taken for encryption and decryption.

Table 2 : Throughput value with different data sizes.

Size (in bytes)	Size (in MB)	Total time(in ms)	Total Time(in s)	Throughput(MB/s)
65	0.000062	32.35	0.03235	0.00191
81	0.000077	28.75	0.02875	0.00268
128	0.000122	28.72	0.02872	0.00424
209	0.000199	29.52	0.02952	0.00675
257	0.000245	29.43	0.02943	0.00832
299	0.000285	26.46	0.02646	0.01077
448	0.000427	28.44	0.02844	0.01502

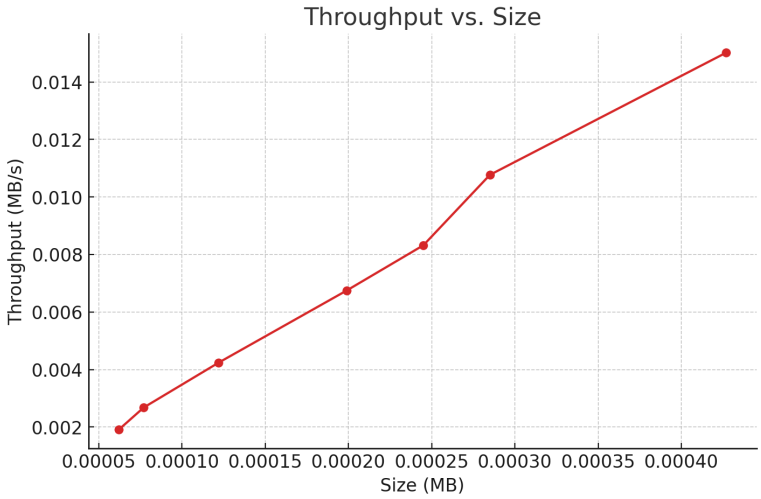

Figure 14 : Graph of the Throughput vs Size.

6.2 Analysis of Blowfish and AES.

The variation of different password size is varied in comparison of Blowfish and AES encryption shown below in the graph.

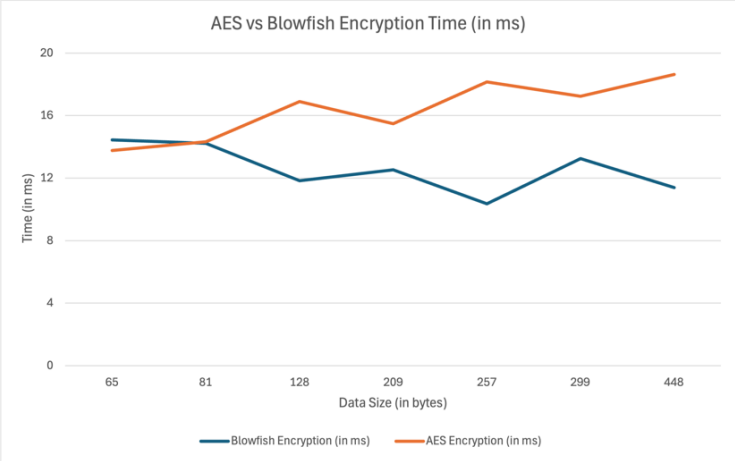

Figure 15 : AES and Blowfish encryption time

Similarly, the variation of different password size is varied in comparison of Blowfish and AES decryption in below graph.

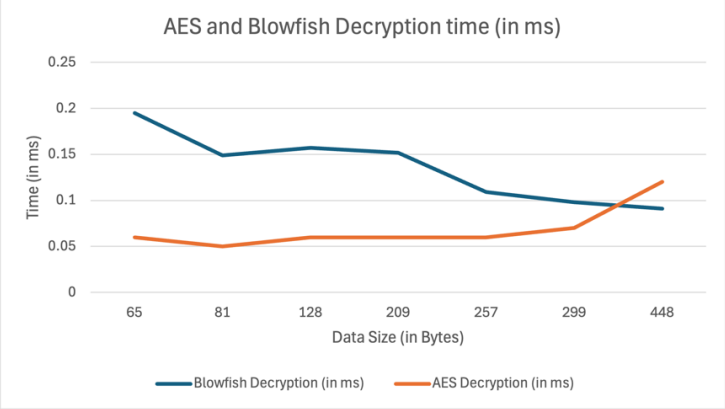

Figure 16 : AES and Blowfish decryption time

6.3 Avalanche Effect

Avalanche Effect is used to analyse the security of the encryption technique. The benefits of having the avalanche effect to any cryptosystem is benefit for its desirable system. The smallest change in the input of data could make a huge change in value difference for the output data.

$$Avalanche\ Effect = \left(\frac{Number\ of\ bits\ changed}{Total\ Number\ of\ bits\ output}\right) \times 100\%$$

The avalanche effect is calculated as the data is encoded and the converting a bit which changes the encrypted data that changes the whole number by changing the bits. The below table shows the sample data for avalanche effect.

Table 3: Avalanche Effect for data.

Original Text	Size in Bytes	Cipher Text (Original Text)	Cipher Text (Changed Text)	Avalanche Effect
Hello_World!	12	b'\x18?\xcb)\xd0g\xda<\xa1\x0b\xa89\x83\xd8\x00{'	b'\x03\x89\x85\x12\xdaK\xa8g~`\x83sXF|\xe5'	56.25%
Thesis@ Semester 3	15	b'\xf8\x07\x8d\xb3,\xdbk7\x9a\xa6\xb7\xb8\x82:ko'	b'\x1f\x03!+0\xb1p\xed'\x03\xc3\xfa\x19\xc6\x07\xfb'	50.00%
MangoBerry	10	b'\xd8\xf1\xfb\x9ecr?\xc8\t\x99L\xda\xa1\xea\xd2_.rvK\x83\xb5t\xde$\xd1\xf8\x8c<K\xf4\xa5S\xaa_\xfegKhMYE\xef{\xbe\x03\xc6\xd3\x93)M_\x93O#{\x15\x9e\xed\x10Y\xc7\xdd{'	b'\x1d\x8dg\x15\xf7\xb6\xa5\x1b\x97~Z,\x08\xc8z\'4\xc1\xc7\x93\xc4\xd9\x19\xafdD\\\x9b5P\x91\xcc\r\x9cH\x97HK\x92\xe3\x10\\\xae|m\xe1\xfd\x0e\x8b\xbd\x93\xc9\xcd"c\xd7\x9b9s3\xa7\r\xa5\x98'	49.80%
Passout	7	b'Ab\xd2_\xd3s\xa0\xce\xaf\x00\x94\xd8\x8b:\x95\x8e\xecuA ~\xd3\xc0D\xee\xbd\x88\x15\xbd\x8d'\xa4dz\xfc\xe53\xa03\xdcB\xeb\x94\xfb\xb0\xb9Uy\x1d\x10)\xecxI\x9aA\xaf\xab\xa61,@\x08\xea'	b'P{Q\xaf\xe6S2\xf1D\xcby\xc8\xa3\x9d<\xf6Z\xe5j\xc6E\xf6\x8d>\x82\xd4OH]\x03\xbfY3w\xde\x15\xcd7\xae-\x10WW\x9c\xaf\xe5*H\xf7\xb1L\x8c3\x9d,}\x92\xb9\x8es\xa5LM['	49.41%
MSc_Cybersecurity	16	b'\xc8\xb3<\xf0\x96\xd5\xf4DB\xf3\xf3\x8a\x91\xc0;\xd7\x8c\xc2x\x14P\xc5\xb3\x95C\x06\xf0K\x8c\xe1\xa1B\x9c4U\xaacP\xb8\xfd\x1b4\x01\x1d5ZP(P\xc8\x82\xdd\xf9<0n\xde<\xc2"g\x16\xff\xca'	b'>P\xdf<80g\x90\xd4\xb6\xf0\x02\x18:\xc1\xd06\x81\x9f\x07\xa97\x8c\xd9\x93\x9b\x81\xd5\x8a\xc8\xa6\x83\x1f\xf4\xb3\x1e\xc9\xba\xdeC\\>\x19[\x06\x0c\xb6\x86\xa0\x15\x9dq\xbc\xbb\x90\x1c\xce\x92\xd4|,\xe2\x9a\xbe'	50.20%

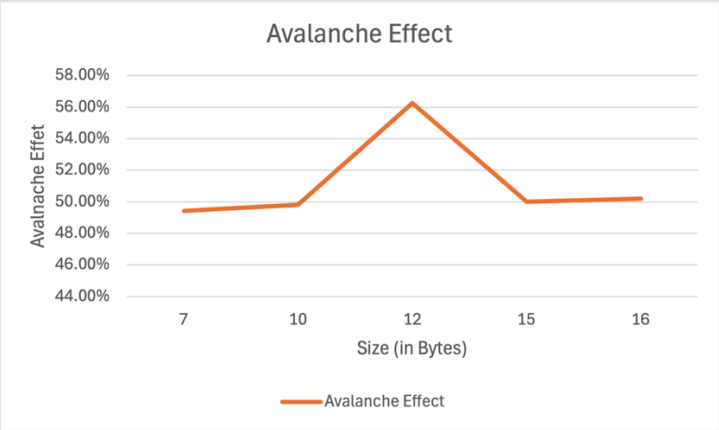

Figure 17 : Avalanche Effect when size of the data is changed.

6.4 Discussion

The model of creating honeyword was easily implemented which was used by (Sahu and Ansari, 2017) Blowfish algorithm resulted in total time taken for execution with RC2. The different password takes different execution time which results in size difference of file. In blowfish algorithm the encryption and decryption were depended upon the size of password which could be seen in evolution section of this paper. The results for the honey blow were as size of password increase and time decreases. Password Based encryption was used to safety of personal information which are weak for brute force attack as it can be easily guess or can be examine by the cipher text. Honey encryption being one of the techniques which is to protect the system from an attacker by attacking as it is defence for the weakness.

Avalanche effect is seen to have slightly improved results with the proposed method compared to prior model. Since the result of avalanche effect is secure it sure that the hybrid model is better and secure. The throughput is also commutated and compared with various password lengths. The proposed model also provides complex solution of DTE and using concept of avalanche effect you can check security of the password. The comparison which allows the Blowfish more robust and versatile than AES (Nadeem and Javed, 2005).

7 Conclusion and Future Work

Personal information needs to be kept secured to prevent from loss and leaks of data. The honey encryption is combines both honey word creation and distribution transformation encoder. To implement model, I have used honeyword creation technique which can be executed easily. Above that to enhance level of security blowfish encryption was being implemented making it more secure. The process of securing the message is implemented by honey encryption which is then secured by blowfish encryption with the help of password/credential. So, from the above execution of research evaluates that blowfish is good for large file encryption and performance as compared to AES. Also, compared to Blowfish is more secure than Rivest Cipher-2. The performance of encryption may change as the system performance matters (Elminaam, Kader and Hadhoud, 2010).

These types of encryptions can further setup in real world examples like secured messaging apps, banking sector, fraud detections, etc. many more. It can also be further implemented with different symmetric encryptions algorithm as well to increase security; hashing can also be implemented.

References

A.Ahmed, M. (2023) ‘Honeywords Generation Technique based on Meerkat Clan Algorithm and WordNet’, *Wasit Journal for Pure sciences*, 2(4), pp. 106–115. Available at: https://doi.org/10.31185/wjps.269.

AlMuhanna, A., AlFaadhel, A. and Ara, A. (2022) ‘Enhanced System for Securing Password Manager Using Honey Encryption’, in *2022 Fifth International Conference of Women in Data Science at Prince Sultan University (WiDS PSU)*. *2022 Fifth International Conference of Women in Data Science at Prince Sultan University (WiDS PSU)*, pp. 150–154. Available at: https://doi.org/10.1109/WiDS-PSU54548.2022.00042.

Antonov, A. (2024) *Hackers can crack 59% of passwords in an hour*. Available at: https://www.kaspersky.co.uk/blog/password-can-be-hacked-in-one-hour/27738/ (Accessed: 11 August 2024).

Burgess, J. (2017) ‘Honey Encryption’.

C, N.R. and K, M.S. (2023) ‘An In-Depth Review of Blowfish Encryption Algorithm: Security, Performance, and Application’, in *2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT)*. *2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, pp. 1–5. Available at: https://doi.org/10.1109/ICCCNT56998.2023.10307323.

Chakraborty, N. and Mondal, S. (2017) ‘On designing a modified-UI based honeyword generation approach for overcoming the existing limitations’, *Computers & Security*, 66, pp. 155–168. Available at: https://doi.org/10.1016/j.cose.2017.01.011.

Chen, H.-C., Wijayanto, H., Chang, C.-H., Leu, F.-Y., Yim, K. (2016) ‘Secure Mobile Instant Messaging key exchanging protocol with One-Time-Pad substitution transposition cryptosystem’, in *2016 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. *2016 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pp. 980–984. Available at: https://doi.org/10.1109/INFCOMW.2016.7562224.

Dibas, H. and Sabri, K.E. (2021) ‘A comprehensive performance empirical study of the symmetric algorithms:AES, 3DES, Blowfish and Twofish’, in *2021 International Conference on Information Technology (ICIT)*. *2021 International Conference on Information Technology (ICIT)*, pp. 344–349. Available at: https://doi.org/10.1109/ICIT52682.2021.9491644.

Elminaam, D.S.A., Kader, H.M.A. and Hadhoud, M.M. (2010) ‘Evaluating The Performance of Symmetric Encryption Algorithms’.

Erguler, I. (2016) ‘Achieving Flatness: Selecting the Honeywords from Existing User Passwords’, *IEEE Transactions on Dependable and Secure Computing*, 13(2), pp. 284–295. Available at: https://doi.org/10.1109/TDSC.2015.2406707.

Jain, S., Muntean, C.H. and Verma, R. (2023) ‘Honey2Fish - A Hybrid Encryption Approach for Improved Password and Message Security’, in *2023 IEEE 9th Intl Conference on Big Data Security on Cloud (BigDataSecurity), IEEE Intl Conference on High Performance and Smart Computing, (HPSC) and IEEE Intl Conference on Intelligent Data and Security (IDS)*. *2023 IEEE 9th Intl Conference on Big Data Security on Cloud (BigDataSecurity), IEEE Intl*

Conference on High Performance and Smart Computing, (HPSC) and IEEE Intl Conference on Intelligent Data and Security (IDS), pp. 198–203. Available at: https://doi.org/10.1109/BigDataSecurity-HPSC-IDS58521.2023.00042.

Jordan, K. (2021) 'Honey Encryption', *smucs*, 7 April. Available at: https://medium.com/smucs/honey-encryption-e56737af081c (Accessed: 31 July 2024).

Juels, A. and Rivest, R.L. (2013) 'Honeywords: making password-cracking detectable', in *Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security*. New York, NY, USA: Association for Computing Machinery (CCS '13), pp. 145–160. Available at: https://doi.org/10.1145/2508859.2516671.

Lindholm, R. (2019) 'Honey Encryption: implementation challenges and solutions'.

Moe, K.S.M. and Win, T. (2018) 'Enhanced Honey Encryption Algorithm for Increasing Message Space against Brute Force Attack', in *2018 15th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)*. *2018 15th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)*, Chiang Rai, Thailand: IEEE, pp. 86–89. Available at: https://doi.org/10.1109/ECTICon.2018.8620050.

mohammed, S., KURNAZ, S. and Mohammed, A.H. (2020) 'Secure Pin Authentication in Java Smart Card Using Honey Encryption', in *2020 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA)*. *2020 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA)*, pp. 1–4. Available at: https://doi.org/10.1109/HORA49412.2020.9152936.

Nadeem, A. and Javed, M.Y. (2005) 'A Performance Comparison of Data Encryption Algorithms', in *2005 International Conference on Information and Communication Technologies*. *2005 International Conference on Information and Communication Technologies*, pp. 84–89. Available at: https://doi.org/10.1109/ICICT.2005.1598556.

Nirmalraj, T. and Jebathangam, J. (2022) 'A Password Secure Mechanism using Reformation-based Honey Encryption and Decryption', in *2022 International Conference on Inventive Computation Technologies (ICICT)*. *2022 International Conference on Inventive Computation Technologies (ICICT)*, pp. 214–220. Available at: https://doi.org/10.1109/ICICT54344.2022.9850868.

O, K. (2024) 'Python Blowfish Encryption Example', *DevRescue*, 14 January. Available at: https://devrescue.com/python-blowfish-encryption-example/ (Accessed: 31 July 2024).

Omolara, A.E., Jantan, A., Abiodun, O.I., Poston, H.E. (2018) 'A Novel Approach for the Adaptation of Honey Encryption to Support Natural Language Message', *Hong Kong* [Preprint].

Pagar, V.R. and Pise, R.G. (2017) 'Strengthening password security through honeyword and Honeyencryption technique', in *2017 International Conference on Trends in Electronics and Informatics (ICEI)*. *2017 International Conference on Trends in Electronics and Informatics (ICEI)*, pp. 827–831. Available at: https://doi.org/10.1109/ICOEI.2017.8300819.

Patel, R. and Kamboj, P. (2016) ‘Security Enhancement of Blowfish Block Cipher’, in A. Unal et al. (eds) *Smart Trends in Information Technology and Computer Communications*. Singapore: Springer Nature, pp. 231–238. Available at: https://doi.org/10.1007/978-981-10-3433-6_28.

Sahu, R. and Ansari, M.S. (2017) ‘Securing Messages from Brute Force Attack by Combined Approach of Honey Encryption and Blowfish’, 04(09).

Sahu, S. (2020) ‘PROVIDING INFORMATION SECURITY USING HONEY ENCRYPTION’, *Advances in Mathematics: Scientific Journal*, 9(10), pp. 8249–8258. Available at: https://doi.org/10.37418/amsj.9.10.54.

Shamini, P.B., Dhivya, E., Jayasree, S., Lakshmi, M.P. (2017) ‘Detection and avoidance of attacker using honey words in purchase portal’, in *2017 Third International Conference on Science Technology Engineering & Management (ICONSTEM)*. *2017 Third International Conference on Science Technology Engineering & Management (ICONSTEM)*, pp. 260–263. Available at: https://doi.org/10.1109/ICONSTEM.2017.8261290.

Shen, C., Yu, T., Xu, H., Yang, G., Guan, X. (2016) ‘User practice in password security: An empirical study of real-life passwords in the wild’, *Computers & Security*, 61, pp. 130–141. Available at: https://doi.org/10.1016/j.cose.2016.05.007.

The State of Identity Security for 2024 (2024) *BeyondTrust*. Available at: https://www.beyondtrust.com/blog/entry/the-state-of-identity-security-identity-based-threats-breaches-security-best-practices (Accessed: 12 August 2024).

Verma, H.K. and Singh, R.K. (2012) ‘Performance Analysis of RC5, Blowfish and DES Block Cipher Algorithms’, *International Journal of Computer Applications*, 42(16), pp. 8–14.

Vivek Raj, K., Ankitha, H., Ankitha, N.G., Kanthi Hegde, L.S. (2020) ‘Honey Encryption based Hybrid Cryptographic Algorithm:A Fusion Ensuring Enhanced Security’, in *2020 5th International Conference on Communication and Electronics Systems (ICCES)*. *2020 5th International Conference on Communication and Electronics Systems (ICCES)*, pp. 490–494. Available at: https://doi.org/10.1109/ICCES48766.2020.9137849.