SUDAN UNIVERSITY OF SCIENCE AND TECHNOLOGY

COLLEGE OF GRADUATE STUDIES

INFORMATION TECHNOLOGY

Preservation of Data Confidentiality Using Honey Encryption

الحفاظ على سرية البيانات باستخدام التشفير العسلي

A thesis submitted in partial fulfillment of requirements for the master degree of Information Technology

Preparation:

Nosiba Altoom Adam Abdalla

Supervision:

Dr **. Faisal Mohammed Abdalla Ali**

December 2018

الآية

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ (1) الْحَمْدُ لِلَّهِ رَبِّ الْعَالَمِينَ (2) الرَّحْمَنِ الرَّحِيمِ (3)

مَالِكِ يَوْمِ الدِّينِ (4) إِيَّاكَ نَعْبُدُ وَإِيَّاكَ نَسْتَعِينُ (5) اهْدِنَا الصِّرَاطَ الْمُسْتَقِيمَ (6)

(7)صِرَاطَ الَّذِينَ أَنْعَمْتَ عَلَيْهِمْ غَيْرِ الْمَغْضُوبِ عَلَيْهِمْ وَلَا الضَّالِّينَ

سورة الفاتحة

ACKNOWLEDMENTS

The best thanks to the God Almighty for tender which is completed , and which we have succeeded in our Exalted what was this search for the vision of light without the Almighty and his generosity .

I dedication to my mother and father may God have mercy on him , whose taught me that the best kind of knowledge to have is that which is learned for its own sake.

Thanks to Dr. **Faisal Mohammed Abdalla Ali** , the research supervisor , who did not spend days giving us information, advice and everything that could benefit him in producing this research.

I am also pleased to thank all those who contributed to bringing this work, especially the beloved friends Doaa Mohammed Osman, Omran Eltegani, Mutwakil Mohammed, Musap Mohammed and Mahmoud Norain Basi .

I ask Allah to reward us with his precious reward and to keep them a champion of knowledge .

I dedicate this research to all the people in my life near my heart .

Abstracts

Password-based encryption methods are used to protect private data that is vulnerable to brute force attacks by giving a message that the key has been decrypted or the key is guessed is incorrect, which causes the attacker to re-decrypt until the original message is reached.

Honey Encryption is a general and simple way to encrypt messages and produce encrypted text using the lowest value keys for entropy. When decoding with any number of incorrect keys, it produces a reasonable or correct message , but it is false called honey messages.

In this search the application was applied to the password because it is more susceptible to attack for , its ease and simplicity. Therefore, hashing and salting method use in this search, is a way to defend password theft, used to store and classify the password entered by the user into a real password and honey words, and store them in the database in hexadecimal format if the user is new.

If the user is already logged on, the system checks the password entered in the database , if it exists, does the second check, the server matches the password with the user name. If the password does not exist or the password does not match the user name, the system sends a message to the admin containing the name of the attacker and the password that he tried to enter and the transfer of the attacker to the imaginary system.

The result of use this algorithm is better password security in a faster time, helped to reduce the existing congestion in password-based encryption methods, password storage problems, provide security beyond conventional brute force limits, and provide high protection against partial disclosure of Min-entropy keys.

This research can be applied to protect all types of private data such as security messages and can be complicated by the complexity of salting and the choice of a better encryption algorithm. Over time, the security of honey encryption reduces the security of password-based encryption methods that with efficiency and computing power.

المستخلص

يتم إستخدام أساليب التشفير القائمة على كلمة المرور لحماية البيانات الخاصة التي أصبحت عرضة لهجمات القوة الغاشمة وذلك بإعطاء رسالة أن المفتاح الذي تم فك التشفير به أو المفتاح الذي خمن غير صحيح , وذلك مما يجعل المهاجم القيام بإعادة فك التشفير حتى يتم الوصول الي الرسالة الأصلية .

أما التشفير العسل هو منهج عام وبسيط لتشفير الرسائل وإنتاج نص مشفر بإستخدام أدنى قيمة مفاتيح للأنتروبيا , وعند فك التشفير بأي عدد من مفاتيح غير صحيحة , تنتج رسالة نصية معقولة أو صحيحة , لكنها زائفة تسمى برسائل العسل .

في هذا البحث تم التطبيق علي كلمة المرور لأنها أكثر شي عرضة للهجوم لسهولتها وبساطتها . لذلك إستخدمت طريقة التجزئة والتمليح وهي طريقة للدفاع عن سرقة كلمة المرور يتمثل الإبتكار الأساسي لها في تخزين وتصنيف كلمة المرور المدخلة من قبل المستخدم إلى كلمة مرور حقيقة وكلمات عسل , وتخزينهما في قاعدة البيانات على شكل سداسي إذا كان المستخدم جديد .

أما إذا كان المستخدم مسجل مسبقا , عند الدخول يقوم النظام بفحص كلمة المرور المدخلة هل موجودة في قاعدة البيانات أما لا , إذا كانت موجودة يقوم بالفحص الثاني وهو تطابق كلمة المرور مع إسم المستخدم . أما إذا كانت غير موجودة أو لم تتطابق كلمة المرور مع إسم المستخدم يقوم النظام بإرسال رسالة إلي المدير تحتوى على أسم المهاجم وكلمة المرور التي حاول الدخول بها وتحويل المهاجم إلى النظام الوهمي .

إستخدام هذه الخوارزمية ادى إلي توفير أمان أفضل لكلمة المرور في وقت أسرع , وساعد على تقليل الثقرة الموجودة في أساليب التشفير القائمة على كلمة المرور, ومشاكل تخزين كلمة المرور , وتوفير الأمن خارج حدود القوة الغاشمة التقليدية وتوفير حماية عالية ضد الإفصاح الجزئي لمفاتيح ال Min-entropy .

هذا البحث يمكن تطبيقه في حماية جميع أنواع البيانات الخاصة مثل الرسائل الأمنية ويمكن تعقيدها بواسطة تعقيد التمليح وإختيار خوارزمية تشفير أفضل , فمع مرور الزمن ينخفض أمن التشفير العسل إلي أمن أساليب التشفير القائمة على كلمة المرور وذلك بوجود الكفاءة وقوة الحوسبة .

List of Contents

List of Figure

List of Tables

List of Abbreviations

Abbreviations	Stand For
HE	Honey Encryption
DTE	Distribution Transforming Encoder
PBE	Password Based Encryption
RSA	Rivest Shamir and Adleman
PAP	Password Authentication Protocols
PPP	Point to Point Protocol
SDK	Software Development Kit
FPE	Format Preserving Encryption
FTE	Format Transforming Encryption
WORA	Write Once Run Anywhere
JVM	Java Virtual Machine
TCP	Transmission Control Protocol
IP	Internet Protocol
HTTP	Hyper Text Transfer Protocol
FTP	File Transfer Protocol
IDE	Integrated Development Environment
SDK	Software Development Kit
S	Seed
C	Cipher text

Chapter I

Introduction

1.1 Background

Most of the users select the passwords that are very simple and therefore easy to remember. Thus the problem is that, if it is easy to remember it is also easy to be predicted by the attacker. Most users would like to pick one password and use it for all their accounts, never change it and write it down for future references.

About 32 million clear-text passwords were exposed when an attacker was able to break into the database of RockYou.com which provides services and applications for social networking sites, through SQL vulnerability. This search provided a proof that consumers repeatedly use easy to guess login credentials. After analyzing the data it was found that the top ten common password that were used are 123456, 12345, 123456789, password, iloveyou, princess, rockyou, 12345678, abc123. The trivial nature of the top ten exposed passwords is bad enough, but more worse is that nearly 50 percent of passwords which were exposed from RockYou breach used name, slang words or dictionary words. If these login names and passwords become easy to guess then it becomes more likely that the attackers or hackers will be able to break into accounts using various attacking techniques such as Brute force, dictionary attacks and readily available password cracking tools.[1]

For that most people in any country are annoyed by junk text messages. The Internet users can also be affected by identity theft when criminals it's used. This can occur because some sensitive private data was not well protected and was then maliciously used by other parties causing damage to finances and reputation of the data owner.

When purchasing a product online, we are asked to provide our mobile phone number for the delivery purpose. When buying a train ticket, we need to fill in the identification card number. The commercial parties gather such sensitive private data. Some store them in a plaintext format. Some employ password-based encryption (PBE) . However, the robustness of encryption depends on the key length.

Although current encryption algorithms are considered secure, given enough time and computing power, they will be vulnerable to brute-force attacks. Also, the existing encryption mechanisms have vulnerability; that is, when decrypting with a wrongly guessed key, they yield an invalid-looking plaintext message, while when decrypting with the right key, they output a valid-looking plaintext message, confirming that the cipher text message is.[1]

The common selection of passwords by the users which are easy to remember is the main reason behind the development of Honey Encryption (HE). The honey term in the information security terminology describes a false resource. For example, honeypot [2] is a false server that attracts attackers to probe and penetrate. Honeyword [3] is a false username and password in the database. Once used for login, an intrusion is detected. Honey encryption can also address the previously mentioned vulnerability. Even when a wrong key is used for decryption, the system can yield a valid-looking plaintext message; therefore, the attacker cannot tell whether the guessed key is correct or not. correctly decrypted.

Juels and Ristenpart [4] proposed the honey encryption concept to address this vulnerability and make the PBE more difficult to break by brute-force. The contribution of their paper is three fold. First, they design and implement the honey encryption system and apply the concept to three applications including Chinese identification numbers, mobile numbers, and passwords. These applications are based on uniformly distributed message spaces and the symmetric encryption mechanism. They also extend honey encryption to applications with non-uniformly distributed message spaces and an asymmetric encryption mechanism (RSA).

1.2 Problem statement

The existing password-based encryption (PBE) methods that are used to protect private data are vulnerable to brute-force attacks. The reason is that, for a wrongly guessed key, the decryption process yields an invalid-looking plaintext message, confirming the invalidity of the key, which make attacker keep trying guessing password which make system vulnerable to brute force attack.

Users are experiencing "password fatigue" because there are a growing number of websites that require a user name and password, and this has led to some Internet behavior on the part of users, making them vulnerable to attack .

1.3 Objectives

The main objectives are:

(a) To show a new way to secure password and messages and find a way to mitigate attacks.

(b) To implement a good and strong encryption algorithm to attack or need as long as possible to break the password or decrypt the message according to the proposed plan for a secure password .

1.4 Methodologies

The innovation of honey encryption is the design of the distribution-transforming encoder (DTE). According to the probabilities of a message or password in the (message , password) space, it maps the message to a seed range , then it randomly selects a seed from the range and encrypt it with honeyword generation algorithm to get the ciphertext. For decryption, the ciphertext is XORed with the key and the seed is obtained. Then DTE uses the seed location to map it back to the original plaintext message. Even if the key is incorrect, the decryption process outputs a message from the message space and thus confuse the attacker.

1.5 Research layout

This thesis has five chapters, after the introductory chapter, chapter two talked about background , literature review and related works . the background covered database security , security services , authentication , cryptography techniques used in the Password protection and Honey Encryption techniques. chapter three include Methodology it illustrates the structure of system and design decision made in order to achieve the goals . Chapter four include implementation and contain analysis of results obtained and snapshots of implantation .The results of the implementation will be shown and the success of the project will be evaluated in terms of time , space overheads and security .Last chapter is conclusion and recommendation gives a summary and discuss the future work .

Chapter II

Background and Literature Review

2.1 Introduction

This chapter provides basic concepts of protection data security, as well as background information on the proposed technology. Moreover, a brief summary of the prior technology of data encryption in historical order and its advantages and disadvantages.

2.2 Cryptography

Cryptography is the ability to send information between participants, in a mangled format, that prevents others from reading it. now encompasses much more than secret communication . and it's Scientific study of techniques for securing digital information, transactions, and distributed computations .

Deals with the problems of message authentication, digital signatures. It have protocols for exchanging secret keys , authentication protocols , electronic auctions and elections modern cryptography can be said to be concerned with problems that may arise in any distributed computation .

In cryptography it have Basic terminology like:

Plaintext: original message to be encrypted

Ciphertext: the encrypted message

Enciphering or encryption: the process of converting plaintext into ciphertext

Encryption algorithm: performs encryption (Two inputs: a plaintext and a secret key).

Deciphering or decryption: recovering plaintext from ciphertext

Decryption algorithm: performs decryption (Two inputs: ciphertext and secret key) .

Secret key: same key used for encryption and decryption (Also referred to as a symmetric key) .

Cipher or cryptographic system **:** a scheme for encryption and decryption

Cryptography**:** science of studying ciphers

Cryptanalysis**:** science of studying attacks against cryptographic systems

Cryptology**:** cryptography + cryptanalysis .

2.3 Classical encryption techniques

Encryption is a challenging field which has attracted a lot of researchers. Therefore, we can find a variety of techniques of implementing encryption in the database. Firstly encryption is divided into two major types depending on which various encryption algorithms are implemented. They are symmetric encryption and asymmetric encryption.

2.3.1 Symmetric algorithms:

Sometimes called conventional algorithms **,** are algorithms where the encryption key can be calculated from the decryption key and vice versa . In most Symmetric algorithms , the encryption key and decryption key are the same key These algorithms also called secret key algorithms , single-key algorithms , or one key algorithms , require that the sender and receiver agree on a key before they can communication securely .The security of a Symmetric cipher rests in the key ; divulging the key means that anyone could encrypt and decrypt message . As long as the communication needs to remain secret , the key must remain secret .[5]

Encryption and decryption with asymmetric algorithms are denoted by:

EK (M) = C

DK (C) = M

Symmetric algorithms can be divided into two categories . some operate on the plaintext a single bit (or sometime byte) at a time ; these are called **stream algorithms** or **stream ciphers** . Other operate on the plaintext in groups of bits are

called **bloke algorithms** or **bloke ciphers** . For modern computer algorithms , a typical bloke size is 64 bits – large enough to preclude analysis and small enough to be workable . (Before computer algorithms generally operated on plaintext one character at a time of characters.) [5] .

2.3.1.1 Symmetric Strengths

a. Much faster than asymmetric systems .
b. Hard to break if using a large key size .

2.3.1.2 Symmetric Weaknesses

a. Key distribution it requires a secure mechanism to deliver keys properly .
b. Scalability each pair of users need a unique pair of keys , so he number of keys grow exponentially .
c. Limited security it can provide confidentiality , but not authenticity or non-repudiation .

2.3.2 Asymmetric algorithms:

Called public-key algorithms are designed so that the key used for encryption different from the keys used for decryption . Furthermore , the decryption key cannot (at least in any reasonable amount of time) be calculated from the encryption key . The algorithms are called ' public-key ' because the encryption key can be made public : A complete stranger can use the encryption key to encrypt a message , but only a specific person with the corresponding decryption key can decrypt the message . In these systems , the encryption key is often called the **Public key** , and the decryption key is often called the **Privet key** . The Privet key is also called the secret key , but to avoid confusion with symmetric algorithms , that tag won't be used here .

Encryption using Public key K is denoted by :

EK1 (M) = C

Even though the Public key and Privet key are different .

Decryption with the corresponding Privet key is denoted by :

DK2 (C) =M

Sometime , message will be encrypted with the private key and decrypted with the public key , this is used in digital signatures . Despite the possible confusion ,

These operations are denoted by , respectively :

EK2 (M) = C

DK1 (M) = M

2.3.2 .1 Asymmetric Strengths

a. Better key distribution than symmetric system .
b. Better scalability than symmetric system .
c. Can provide confidentiality , authentication and non repudiation .

2.3.2 .2 Asymmetric Weaknesses

a. Works much slower than symmetric system .

Now we will see the basic difference between the symmetric encryption and asymmetric encryption by the following diagram.

2.3.3 Symmetric VS Asymmetric

Table (2.1) show the comparison between symmetric and asymmetric encryption .

Characteristic	**Symmetric Key Cryptography**	**Symmetric Key Cryptography**
Key used for encryption and decryption	Same key is used for encryption and decryption	One key is used for encryption and another different key is used for decryption
Speed of encryption and decryption	Very fast	Slower
Size of resulting encryption text	Usually same as or less than the original clear text size	More than the original clear text size
Key agreement /exchange	A big problem	No problem at all
Number of keys required as compared to the number of participants in the message exchange	Equals about the square of the number of participants so scalability is an issue	Same as the number of participants so scales up quite well
Usage	Mainly for encryption and decryption (confidentiality) .Cannot be used for digital signatures (integrity and non-repudiation checks)	Can be used for encryption and decryption(confidentiality) as well as for digital signatures (integrity and non- repudiation checks)

2.4 Cipher

2.4.1 Block Cipher & Stream Cipher

Block cipher is Symmetric encryption algorithm in which the plaintext is processed by dividing it into equal blocks (typically 64 or 128) then converts each block to ciphertext . **Stream cipher** is also symmetric encryption algorithm in which ciphertext output is produced bit-by-bit or byte-by-byte from a stream of plaintext input [2]

2.4.2 Mode of Operations

A mode of operations is a technique for enhancing the effect of a cryptographic algorithm or adapting the algorithm for an application , such as applying a block cipher to a sequence of data block or data stream[2] .

2.4.3 Block Cipher Modes of Operations

There are five modes of operation for use with symmetric block ciphers electronic codebook mode , cipher block chaining mode , cipher feedback mode , output feedback mode and counter mode [2] .

2.5 Cryptanalysis :

The whole point of cryptography is to keep the plaintext and key secret from eavesdroppers also called adversaries , attackers , interceptors , interlopers , intruders , opponents , or simply the enemy) .

Eavesdroppers are assumed to have complete access to the communications between the sender and receiver .

Cryptanalysis is the science of recovering the plaintext of a message without access to the key . Successful cryptanalysis may recover the plaintext or the key . It also may find weaknesses in a cryptosystem that eventually lead to the previous results . (the loss of key through non cryptanalytic means is called a **compromise .**

An attempted cryptanalysis is called an attack .A fundamental assumption in cryptanalysis , first enunciated by the Dutchman A . kerchhoffs in the nineteenth century , is that the secrecy must reside entirely in the key .

kerchhoffs assumes that the cryptanalyst has complete details of the cryptographic algorithm and implementation while real-world cryptanalyst don't always have such detailed information , it's a good assumption to make . If others can't break an algorithm , even with knowledge of how it work , then they certainly won't be able to break it without that knowledge.

2.5.1 There are general types of cryptanalytic attacks.

Each of them assumes that the cryptanalyst has complete knowledge of the encryption algorithm used :

2.5.1.1 Ciphertext-only attack : The cryptanalyst has the ciphertext of several message , here use same encryption algorithm in encryption and decryption . The cryptanalyst's job is recover the plaintext of as many message as possible , or better yet to deduce the key or keys used to encrypt messages , in order to decrypt other message encrypted with the same keys .

Given : C1=Ek(P1) , C2=Ek(P2) ,…... Ci=Ek(Pi)

Deduce : Either p1, p2 ,…..pi ; k ; or an algorithm to infer pi+1 from Ci+1=Ek(Pi)

2.5.1.2. Known-plaintext attack: The cryptanalyst has access not only to the ciphertext of several message , but also to the plaintext of those message . His job is to deduced he key or keys used to encrypt the message or an algorithm to decrypt any new message encrypted with the same key or keys .

Given : P1 , C1 = Ek(P1) , P2 , C2 = Ek(P2) ,…... Pi , Ci = Ek(Pi)

Deduce : Either k , or an algorithm to infer pi+1 from Ci+1=Ek(Pi+1)

2.5.1.3 Chosen- plaintext attack: The cryptanalyst not only has access to the ciphertext and associated ciphertext for several message , but he also chooses the plaintext that gets encrypted . This is more powerful than a known-plaintext attack , because the cryptanalyst can chooses specific

plaintext blocks to encrypt , ones that might yield more information about the key . His job is to deduce the key or keys used to encrypt the message or an algorithm to decrypt any new message encrypted with the same key or keys .

Given : P1 , C1 = Ek(P1) , P2 , C2 = Ek(P2) ,….. Pi , Ci = Ek(Pi) , where the cryptanalyst gets to choose P1 , P2 , ….. , Pi

Deduce : Either k , or an algorithm to infer pi+1 from Ci+1= Ek(Pi+1)

2.5.1.4 Chosen- ciphertext attack: The cryptanalyst can chooses different ciphertexts to be decrypted and has access to the decrypted plaintext . for example , the cryptanalyst has access to a tamperproof box that dose automatic decryption . His job is to deduce the key .

Given : C1, P1 = Dk(C1) , C2 , P2 , = Dk(C2) ,…... , Ci , Pi = Dk(Ci)

Deduce : K

This attack is primarily applicable to public-key algorithm . A choosen-chipertext attack is sometimes effective against a symmetric algorithm as well .

*Sometimes a chooses-plaintext attack and a choosen -chipertext attack are together known as a chooses-text attack . *

2.5.2 There are at least three other type of cryptanalytic attack :

2.5.2.1 Adaptive- chosen- plaintext attack: This is special case of a chooses-plaintext attack . Not only can the cryptanalyst chooses the plaintext that is encrypted , but he can also modify his choice based on the results of previous encryption . In a choosen-plaintext attack , a cryptanalyst might just be able to choose one large block of plaintext to be encrypted , in an adaptive -choosen- plaintext attack he can choose a smaller block of plaintext and then choose another based on the results of the first and so forth .

2.5.2.2 Chosen-key attack: This attack doesn’t mean that the cryptanalyst can choose the key , it mean that he has some knowledge about the relationship between different keys . It’s strange and obscure , not very particle .

2.5.2.3 Rubber-hose cryptanalysis: The cryptanalyst threatens , blackmails , or tortures someone until they give him the key . Bribery is

sometime referred to as a **purchase-key attack .**These are all very power ful attacks and often the best way to break an algorithm .

Known-plaintext attacks and choosen-plaintext attacks are more common than you might think . It is not unheard-of for a cryptanalyst to get a plaintext message that has been encrypted or to bribe someone to encrypt a choosen message .

2.6 Database security

With increased use of the Internet, the enterprise become more vulnerable to security breaches even as they gain productivity and efficiency advantages. Though a number of techniques, such as encryption and electronic signatures, are currently available to protect data when transmitted across sites, It is well understood today that the semantics of data must be taken into account in order to specify effective access control policies. Also, techniques for data integrity and availability specifically tailored to data systems must be adopted. In this respect, over the years, the data security community has developed a number of different techniques and approaches to assure data confidentiality, integrity, and availability.

However, despite such advances, the data security area faces several new challenges. After that have introduced both new security requirements and new contexts in which to apply and possibly extend current approaches.

For that we must protection of information and its critical elements, including systems and hardware that use, store, and transmit that information. The security is the quality or state of being secure to be free from danger.

2.7 Security Services

As a service provided by a protocol layer of communicating open systems, which ensures adequate security of the systems or of data transfers . Also the defines security services as a processing or communication service that is provided by a system to give a specific kind of protection to system resources. Security Services implement security policies and are implemented by security mechanisms [6].

Security services divides into five categories and fourteen specific services as

2.7.1 Data Confidentiality

The protection of data from unauthorized disclosure Or that are not authorized to read it .

a. **Connection Confidentiality:** The protection of all user data on a connection.
b. **Connectionless Confidentiality:** The protection of all user data in a single data block
c. **Selective-Field Confidentiality:** The confidentiality of selected fields within the user Data on a connection or in a single data block.
d. **Traffic Flow Confidentiality:** The protection of the information that might be derived from observation of traffic flows.

2.7.2 Access Control

The prevention of unauthorized use of a resource (i.e., this service controls who can have access to a resource, under what conditions access can occur, and what those accessing the resource are allowed to do).

2.7.3 Data Integrity

The assurance that data received are exactly as sent by an authorized entity or to make sure that data has not been changed while on Transfer, storage, etc

a. **Connection Integrity with Recovery :** Provides for the integrity of all user data on a connection and detects any modification, insertion, deletion, or replay of any data within an entire data sequence, with recovery attempted.
b. **Connection Integrity without Recovery :** As above, but provides only detection without recovery.
c. **Selective-Field Connection Integrity :** Provides for the integrity of selected fields within the user data of a data block transferred over a connection and takes the form of determination of whether the selected fields have been modified, inserted, deleted, or replayed .

d. **Connectionless Integrity :** Provides for the integrity of a single connectionless data block and may take the form of detection of data modification. Additionally, a limited form of replay detection may be provided.

e. **Selective-Field Connectionless Integrity :** Provides for the integrity of selected fields within a single connectionless data block; takes the form of determination of whether the selected fields have been modified.

f. **One Way Function** is a function that is easy to compute but computationally infeasible to inverse . Meaning it's easy to compute y=f(x) but impossible to compute x=f(^{y2})

g. **Trapdoor Function** is a one way function that can be computationally feasible to compute the inverse if secret information is given .

2.7.4 Non-repudiation

Provides protection against denial by one of the entities involved in a communication of having participated in all or part of the communication .

a. Non repudiation, Origin: Proof that the message was sent by the specified party.

b. **Non repudiation, Destination:** Proof that the message was received by the specified party.

2.7.5 Availability

To make sure that the services are always available to users .

2.7.6 Authentication

To verify the identity of the user computer . Are you who you say you are? are determine whether access is allowed or not ,or The assurance that the communicating entity is the one that it lairs to be.

a. **Peer Entity Authentication:** Used in association with a logical connection to provide confidence in the identity of the entities connected.

b. **Data Origin Authentication:** In a connectionless transfer, provides assurance that the source of received data is as claimed.

Can be based on :

(1) Something you know ' a password '

(2) Something you have ' a smartcard '

(3) Something you are ' your fingerprint '

2.8 Password

A password is a string of characters used to verify the identity of a user during the authentication process. or it is a secret string of characters is used for the authentication process in various applications . It is used to gain access to various accounts, repositories, and databases but at the same time, protects them from unauthorized access.

Passwords are typically used in conjuncture with a username; they are designed to be known only to the user and allow that user to gain access to a device, application or website. Passwords can vary in length and can contain letters, numbers and special characters.

As much as it is important to create an un decipherable password, it is important for it to be stored in human brain with the ease of its recollection. On the event of losing a password due to inability of recollection, there are certain processes through which a person may have to go to bring back or change the password. Sometimes it can be a strenuous process if it’s a bank account or an online resource of similar importance.

Other terms that can be used interchangeably are passphrase for when the password uses more than one word, and pass code and passkey for when the password uses only numbers instead of a mix of characters, such as a personal identification number.

Passwords are intended to protect sensitive information from bad guys who want to steal or exploit it. As well as keeping our identities and data safe, they’re supposed to save our employers from joining the large list of corporations that suffer the operational and reputational damage caused by hacking.

Password is more popular than something you have and something you are ? Because

a. Cost: passwords are free
b. Convenience: easier for admin to reset password than to issue a new thum
c. Space passwords are 8 characters, and 256 different characters
d. Then $256^8 = 2^{64}$ passwords
e. Users do not select passwords at random
f. Attacker has far less than 2^{63} passwords to try (**dictionary attack**)

Computers use the password to associate an identity with a trusted user, giving readers a clear understanding of what the organization needs to know its users reliably, and how to implement different authentication methods. Password is one of the basic building blocks of security. The computer system allows the distinction between legitimate users and others. Most sites provide passwords for authorized users.

The passwords which high priority in some Industries such as :

a. Banking & Finance
b. Share Trading
c. Military
d. Espionage
e. Social Media
f. Telecommunication & Gadgets
g. Corporate

Hackers always find new ways to circumvent passwords systems. The good news is that organizations now have a wide range of password alternatives and a variety of ways to make passwords safer. A well-designed authentication system allows users to easily identify themselves and access the network without compromising the integrity of organizations. The first of its kind, password authentication describes the full range of authentication methods used today. It examines situations in which some techniques fail and shows ways to strengthen them. Network specialists, designers, developers, supervisors, planners and managers

on these pages will find the authentication strategy to protect their value systems. Through graphs and examples .

One of the biggest problems with passwords is that they can be shared, guessed or misused. Organizations should educate users on how to properly handle their passwords. Among the most important password guidelines for users is that passwords should never been written down.

Password Authentication Protocols (PAP) is a password-based authentication protocol used by Point to Point Protocol (PPP) to validate users. Almost all network operating system remote servers support PAP.

PAP is considered a weak authentication scheme (weak schemes are simple and have lighter computational overhead but are much more vulnerable to attack; while weak schemes may have limited application in some constrained environments, they are avoided in general). Among PAP's deficiencies is the fact that it transmits unencrypted passwords over the network.

In fact, the password platforms are susceptible to different types of attacks, targeting different components, conducted from different domains, using different techniques. For better analyzing these attacks, it is useful to identify the main abstract components of a password, corresponding to different functional aspects of those systems. Attackers can target each of the different components, or they can target different levels, possibly with roughly the same logic.

2.8.1 There have Tools can be break the password such as :

2.8.1.1 Brute Force Attack

In this type of attack, the hacker tries to determine the password by trying every possible combination of characters.

The number of attempts get restricted by the number of characters and maximum length that is to be tried per position (or a byte if we are considering Unicode passwords too).

The time taken to complete is relatively more, but there are more chances of coverage of likely clear text value (all possibilities only if set to the maximum length

and every possible character is considered in every position). It is like a combination lock which requires three numbers to be taken in sequence; one tries every possible combination - e.g., First 1-2-3, then 1-2-4.

A brute force attack may not try all options in sequential order. An advanced brute force attack can make certain assumptions like complexity rules require uppercase, first character more likely to be upper than lower case.

2.8.1.2 Dictionary Attack

An attack which is based on estimation guessing using precompiled lists of options. Options which are likely to work are only tried in this attack and not all options are gone forward with.

The dictionary/possible combinations are based on some possible values and tend not to consider options of remote possibility. It may be based on the knowledge of one or a few key information about the target (family member names, birthday, etc.). The dictionary is based on the patterns or combinations that were observed across a massive number of users to determine the most commonly used patterns. The dictionary is more likely to include real words than random strings of characters.

The dictionary attack's execution time is reduced because the number of combinations is restricted only to those on the list. However, the coverage is less and a good password may not be on the list and will be missed .

2.8.1.3 Rainbow table

It is a recomputed table for reversing cryptographic hash functions, mostly used for cracking password hashes. This technique proves to be good for recovering plaintext passwords, debit card numbers, etc. up to a limited length which consists of a limited group of characters. Space & time's trade-off's practical example using less time in/for processing and extra storage capability than the brute force attack which computes a hash at every attempt, but involves more time for processing and lesser storage than table of lookup with an entry a hash .

2.8.1.4 Cain & Abel

Cain and Abel is a popular password cracking tool. Is can handle varying tasks. The most noticeable thing is the tool's availability only in Windows platforms. It can function as a sniffer on the network, for cracking of encrypted passwords by the dictionary attack, uncovering cached passwords, decoding scrambled passwords, brute attacks, recording VoIP conversations, password boxes revelation, cryptanalysis attacks, and analyzing protocols of routing.

Abel & Cain don't exploit any bugs or vulnerability. It covers only the security weakness of a protocol to grab the password [7].

2.8.1.5 THC Hydra

THC Hydra can be said to be the fast paced network logon tool for password cracking. In comparison to other similar tools, it is clearly shown why it is faster. New modules can be easy to install in the tool. One can easily enhance the features by adding modules .

2.8.1.6 Medusa

Medusa is another tool for password cracking like THC Hydra. It is known to be a speedy parallel, login brute forcing tool and modular. When cracking the password; host, password and username can be a flexible input while the performance of the attack.

Medusa is popular for being the command line tool, so one need to understand commands before utilizing the tool. Tool's efficiency depends on network's connectivity. It can test 2000 passwords per minute on a local system.

In this tool the attacker can also carry out parallel attacks at one time. It allows one to crack passwords of multiple email accounts simultaneously. He can specify the username list along with the password list .

2.8.1.7 OphCrack

OphCrack is available for free which is a rainbow-table based tool for password cracking on Windows. It is a popular Windows password cracking tool which can

also be used on Linux or Mac. It can crack LM and NTLM hashes. For cracking Windows 7, Vista or Windows XP, free rainbow-tables are made available.

A live CD of OphCrack is made available for the simplification of the cracking. One can utilize the Live CD of OphCrack to crack the Windows-based passwords .

2.8.1.8 L0phtCrack

L0phtCrack serves as substitute to OphCrack. It makes various attempts on cracking Windows passwords from hashes. For cracking these passwords, it utilizes the primary controllers of domain, workstations (windows), network server, also Active Directory. It also makes use of dictionary attack and brute force attacking in guessing and generating of passwords .

2.8.1.9 Aircrack-NG

Aircrack-NG is a tool for cracking of WiFi passwords that can crack WPA or WEP passwords. It analyses wireless encrypted packets also then tries to crack the passwords with cracking its algorithm. The FMS attack is utilized with other useful attacking methods for cracking of passwords. It is available on Linux and Windows systems [7].

We must be creating strong passwords and use best practices for their login credentials.

2.8.2 Some of the best practices for password requirements include:

a. A minimum length of eight characters with a limit of anywhere from 16 to 64 characters or possibly even higher; the inclusion of both uppercase and lowercase letters with case sensitivity , the use of at least one number; and the use of at least one special character.

b. Policies should prohibit certain characteristics in weak passwords. For instance, any recognizable personal information -- such as birthdates, names of children, or favorite sports teams -- should not be part of a password, as well as any words or phrases that are on a password blacklist.

c. Password blacklists are lists of passwords that are too easily cracked and thus are not secure enough to use. Common offenders that wind up on blacklists include "123456", "password", "football", "qwerty" and so on.

d. Strong password policies also include a time limit for user passwords. This means that passwords will expire after a set period of time -- such as 90 or 180 days -- and users will be forced to change their password to prevent the reuse of the same couple of passwords. The policy may also require the user to create a password that is different from any other they have used in the last six to 12 months.

e. While strong passwords are ideal, users often forget them. As a result, password recovery methods might vary depending upon access to an application, website or device. Methods might include answering security questions, confirming emails asking if users want to reset their passwords, or entering numerical security codes sent via text to a mobile phone to authenticate users who need to reset passwords or recover the original one.

2.8.3 Password weakness

The problem of people is lazy at creating effective passwords. Because they are expected to memories them, many people choose passwords that are easy to remember and store it's in a file .

2.8.4 Attacker could…

a. Target one particular account
b. Target any account on system
c. Target any account on any system
d. Attempt denial of service (DoS) attack

2.8.5 Common attack path

a. Outsider → normal user → administrator
b. May only require **one** weak password!

Password cracking is too easy , One weak password may break security , Users choose bad passwords , Social engineering attacks , Passwords are a BIG security problem and will continue to be a big problem .

2.9 Honey Encryption

Juels and Ristenpart proposed this concept of honey encryption specifically in the context of passwords. After a leakage of millions of real user passwords, it was observed that a significant number of people used weak, easily predictable, and repeated passwords.

The honey encryption (HE), a simple, general approach to encrypting messages or password using low min-entropy keys such as passwords and In the context of computer security, the term honey is used to describe a false resource designed to deflect or counteract attacker attempts of a system.

(HE) is designed to produce a ciphertext which, when decrypted with any of a number of incorrect keys, yields plausible-looking but bogus plaintexts called honey messages.

A key benefit of (HE) is that it provides security in cases where too little entropy is available to withstand brute-force attacks that try every key; in this sense, (HE) provides security beyond conventional brute-force bounds. (HE) can also provide a hedge against partial disclosure of high min-entropy keys.

(HE) significantly improves security in a number of practical settings. To showcase this improvement, we build concrete (HE) schemes for password-based encryption of one time pad secret keys.

The key challenges are development of appropriate instances of a new type of randomized message encoding scheme called a distribution-transforming encoder (DTE).

The core innovation of the honey encryption scheme is the distribution-transforming encoder (DTE), which gives the plain-text messages space the number from seeds of the n-bit string .

The (DTE) takes into account a probability distribution of the message space and assigns a corresponding ratio of bit strings to the message.

The intuition lies in the fact that all potential decryptions, regardless of correctness, map to some message and since possible decryptions are assigned via the expected probability distribution, the attacker gains no information.

Constructing a suitable (DTE) for various applications of honey encryption requires an understanding of the message space distribution.

The contributions of this project fall into two main categories.

Example for (HE) if an attacker tries to get a credit card number by making 1000 attempts, then for all the 1000 attempts he will be getting 1000 fake credit card numbers. Each decryption is going to look as plausible as other. The attacker has no way to distinguish a priori which is correct [4] .

2.9.1 Distribution Transforming Encoder (DTE)

(DTE) is a system for encrypting messages. Formally, (DTE) is binary encoded and decoded algorithms. (DTE) determines the message area of the seed area due to the above message space . We use the probabilities to discover the seed of the message and then randomly select the seeds within the range. The accompanying pseudo code shows this procedure:

Encoding (m):

a. Get a range of seed space to select a random series of possibilities in a selfish way
b. Select a random string of seeds corresponding to the original message .

Despite the fact that we consider these plans to be safe and with sufficient computer power, they cannot be defended against brutal and brutal attacks.In this chapter we will review the methodology and the tool used for implementation .

2.9.2 Seed Space

Seed space, S, is the space of all n-bit binary strings for some predetermined n. Each password in $\mathcal{M}$ is mapped to a seed in S. The size of the seed is directly proportional to how likely a particular password is to appear. Similar to $\mathcal{M}$, S is predefined by developer which can be based on personal judgment, research or sampling results. The distribution on set S is denoted as a map p: S $\rightarrow$ [0,1] . Subsequently, sampling according to such distribution is denoted as s $\leftarrow$ pS .

2.9.3 Honey Encryption Scheme Setup

We now depict the first Honey encryption plot proposed by Juels and Ristenpart. In this development, we have a message space M which contains every single conceivable message. We outline messages to a seed space S through the utilization of a conveyance changing encoder (DTE). The seed space is just the space of all n-bit parallel strings for some foreordained n. Each message in m $\in$ M is mapped to a

seed run in S. The span of the seed scope of m is straight forwardly corresponding to how likely m is in the message space M. We require some information about the message space M all together for the (DTE) to guide messages to seed ranges , (DTE) requires only the use of the probabilities process. Also, the seed space must be sufficiently huge so that even the message with littlest likelihood in the message space is alloted no less than one seed. With this data, we can locate the aggregate likelihood go comparing the message m and guide it to a similar percentile seed go in S. We outline the encryption procedure beneath with a fundamental case .[4]

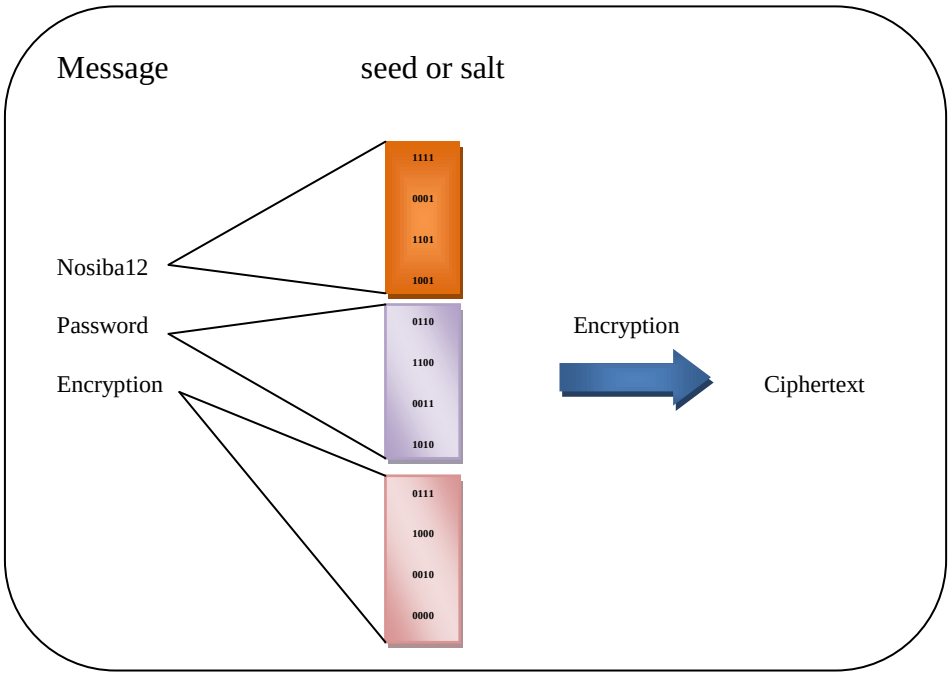

Figure (2.1) show Honey Encryption process with example . [4]

Consider the basic case of encoding dessert enhances in above figure. Our message space M comprises of various passwords, M = {nosiba12, password, encryption}. probabilities are doled out to each passwords. Consider a seed space S of 4-bit strings. With these probabilities, we can then guide each password to a seed run. For this situation, the passwords are requested one after another in order. Utilizing the (DTE), we haphazardly select a seed in the relating seed go. This seed as a salt in honeywords generation algorithm to create the ciphertext . Unscrambling is somewhat harder. The ciphertext is XORed with the mystery key is give back the plaintext . And the plaintext is not a real .

2.10 Related works

2.10.1 Lance Spitzner at.el. [8] proposed are fictitious words or records that are added to legitimate databases and ecoys distributed over a system. If any decoy is used, this means that a compromise is taking place. Their objective it's make decoys for the attackers and it's distributed over a system and allow administrators to track data in situations they wouldn't normally be able to track . The Strengths point in their proposal it's tells the system that there is something unusual and if data is stolen, honey tokens allow administrators to identify who it was stolen from or how it was leaked and the Weaknesses it's can't stop that process is unusual .

2.10.2 Ari Juels RSA Laboratories at.el. [9] proposed are passwords that are rarely used by normal users. Once a login attempt using a honeyword occurs, the system rises an alarm . Their objective it's decrypt codes for applications cross-platform to existing system . The Strengths point in their proposal it's alarm system with an intruder . and the Weaknesses it's alarm system with an intruder but can not stop it .

2.10.3 Margaret Rouse proposed are luring systems that present many vulnerabilities . His objective it's study attackers' motivations, tools, and techniques .The Strengths point in his proposal it's know the behavior , motives , the tools and techniques of attackers are used and the Weaknesses it's will be represented a Part of the system services , if the attacker sends a request and expects a specific response If different, the attacker knows that is a trap, and is used against the official or basic system .[10]

2.10.4 Wei Yen and Jadwiga Indulska in 2017 introduced a proposed is honey encryption and means it deceives attackers that the incorrectly guessed key is valid , aims to protects sensitive data in many applications by deceiving attackers that the key that you guessed incorrectly is valid , HE related to Format-Preserving Encryption (FPE) and Format-Transforming Encryption (FTE) .
the ciphertext message space is different from the message space It is therefore difficult to decrypt the password .

Chapter III

Methodology

3.1 Introduction

In this chapter details description of implementation of proposal system is introduced . Moreover, and describe case analysis of hashing and salting algorithm for securing password.

Many companies try to protect the password files using hashing and sating algorithms. For example, LinkedIn passwords were using the SHA-1 algorithm without a salt and the Harmonly passwords were also stored using unsalted MD5 hashes . If the password file is attacked by attacker using password cracking techniques, the attackers can easily get the password files. [12].

Honeyword generation method is one of the methods to defense against the stolen password file from attackers. In this approach, the system stores the list of password which contains the real user's password with honeywords from honey generation algorithm. Hackers who steal databases of user logins and passwords only have to guess a single correct password in order to get access to the data. The way they know they have the correct password is when the database or file becomes readable. To speed up the process, hackers have access to sophisticated software that can send thousands of passwords each minute to applications in an attempt to decrypt the data. Using higher speed, multi-core processors also shortens the time it can take to break encryption. With honey encryption (HE), decrypting with an incorrect password results attempt. For example, if a hacker made 100 password attempts, they would receive 100 plain text results. Even if one of the passwords were correct, the real data would be indistinguishable from the fake data . The core innovation of the honeywords generation scheme is to store the user's real password with honeyword and the classification of honeywords and user's real password. The current existing honeywords generation method has weakness in password storage and old password management problem .

Ziya Alper Genc, Suleyman Kardas, Mehmet Sabir Kiraz discussed that password is easy to crack with the improvement in the graphical processing unit (GPU) technology. An attacker can recover a user's password using brute-force attack on password hash. Once the password has been recovered no server can detect any invalid user authentication [13]. And this system can deceive to the hackers .

The example of existing system architecture design of honeywords generation algorithm is shown in following fig (3.1) In this figure, the user makes the registration process and the system generate honeywords for this user if the user is new user. The user's real password and generating honeywords are converted into hexadecimal form using hashing and salting algorithm and stored into the database. If the user is member of this system, the user can login with his username and password. In this section, the server checks the entire password exists in the database.

When the password exists in the database, the server sends this password to honeychecker for classification of honeyword and real password. Otherwise, the system sends unsuccessful login to user and quit the system. By checking the password from honeychecker, the system sends an alarm message to the system administrator if the password is honeyword. On the other hand, the user can enter the system successfully [14]. However, there are some limitations in honeywords generation process such as storage overhead problem, typo safety problem and so on.

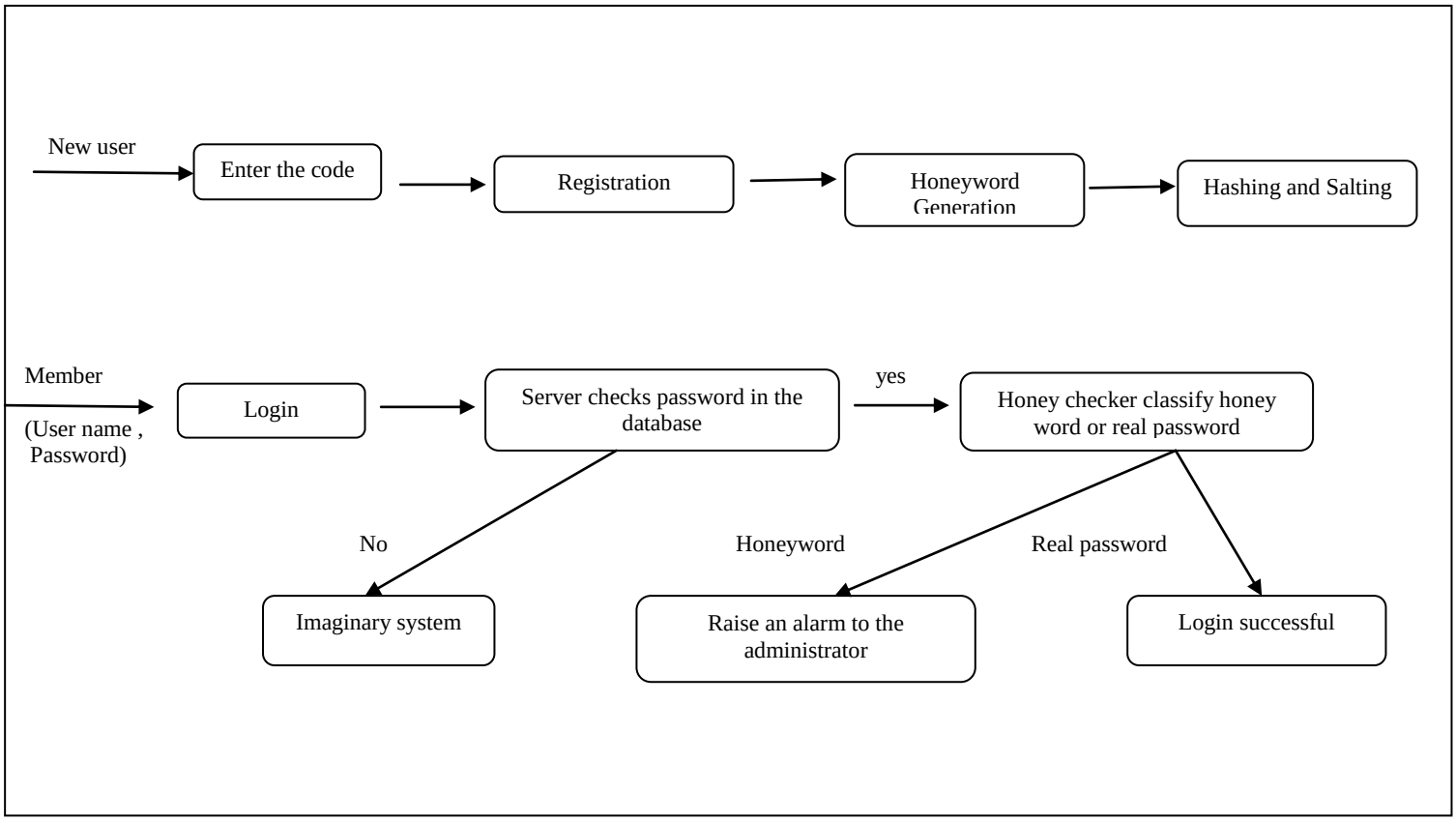

Figure (3.1): Existing System Architecture for Honeywords Generation method

3.2 Honeyword Generation Method

Passwords are notoriously weak authentication mechanism because users frequently choose poor and repeatedly Passwords. The attacker can easily know this poor password. So, the system stores the correct password with several honeywords for each account in the database to deceive attackers. So we generating for any new account allot of many honeyword of the user password and stored in indexes which we called honeyindexes . And then, if a new user that creates new account, he gets randomly index number for real password and honeywords .

For all users account, we create one password table in the database. The table has two attributes: first attribute is the regular id of the account and the second is the honeyindex (real password, honeywords).

Table (3.1) Example of password table in main server

Reg ID	Honeyindex set
1	3dd25e5250
1	9d75c525
1	9fd5c405a515
1	5fe4640405
1	5fe4640405

3.3 Hashing and Salting Algorithm

In our system, hashing and salting algorithm is considered to provide better security for key or password with faster time. This algorithm is as follows :

Step 1: Convert user's password into binary string.

Step 2: Adding padding bits to this binary string.

Step 3: Making flapping of the binary 1's and 0's in string.

Step 4: Perform XOR operation with salting binary string and string formed by combination of 0's and 1's Step.

Step 5: Rotate the string left or right by r character depending on the system.

Step 6: Convert the interchange binary string into hexadecimal string.

3.4 System flowchart

The flowchart for our system is shown in Scheme. In this flowchart, the user makes the registration process if the user is new user after receive the code in the existing email . Otherwise, the user can make the login process. If the registration process successful, the system creates honeywords for this user's password. And then, this user's real password and honeywords are stored into the database using hashing and salting algorithm. If the user is a member of the system and he or she tries to enter into system, the server checks whether this user's password exist or not into the database. If the password doesn't exist in the database, the server takes user to the imaginary system. If the password exists in the database, the server sends this user's password to honeychecker for classification of real password and honeyword. If the password is real password, the honeychecker allows this user to enter into the system. Otherwise, the honeychecker sends an alarm message to administrator for entering the honeywords into the system.

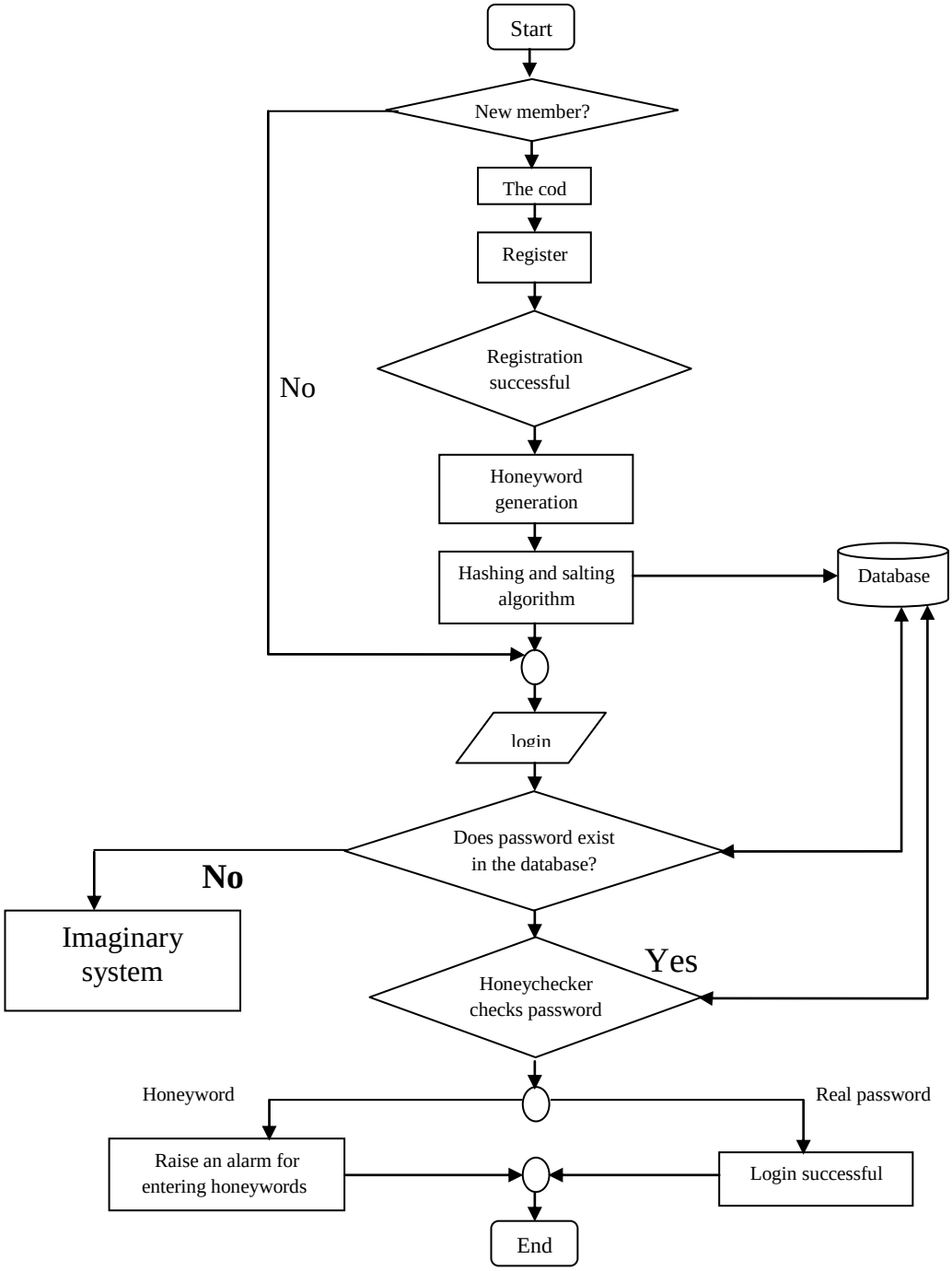

Figure (3.2) Flowchart of the system

3.5 PHP

PHP is a server side scripting language. that is used to develop Static websites or Dynamic websites or Web applications. PHP stands for Hypertext Pre-processor, that earlier stood for Personal Home Pages.

PHP started out as a small open source project that evolved as more and more people found out how useful it was. Rasmus Lerdorf unleashed the first version of PHP way back in 1994.

a. PHP is a recursive acronym for "PHP: Hypertext Preprocessor".
b. PHP is a server side scripting language that is embedded in HTML. It is used to manage dynamic content, databases, session tracking, even build entire e-commerce sites.
c. It is integrated with a number of popular databases, including MySQL, PostgreSQL, Oracle, Sybase, Informix, and Microsoft SQL Server.
d. PHP scripts can only be interpreted on a server that has PHP installed. The client computers accessing the PHP scripts require a web browser only.
e. A PHP file contains PHP tags and ends with the extension ".php".
f. PHP is pleasingly zippy in its execution, especially when compiled as an Apache module on the Unix side. The MySQL server, once started, executes even very complex queries with huge result sets in record-setting time.
g. PHP supports a large number of major protocols such as POP3, IMAP, and LDAP. PHP4 added support for Java and distributed object architectures (COM and CORBA), making n-tier development a possibility for the first time.
h. PHP is forgiving: PHP language tries to be as forgiving as possible.
i. PHP Syntax is C-Like.

3.5.1 Common uses of PHP

a. PHP performs system functions, i.e. from files on a system it can create, open, read, write, and close them.
b. PHP can handle forms, i.e. gather data from files, save data to a file, through email you can send data, return data to the user.
c. You add, delete, modify elements within your database through PHP.
d. Access cookies variables and set cookies.

e. Using PHP, you can restrict users to access some pages of your website.
f. It can encrypt data.

3.5.2 Characteristics of PHP

Five important characteristics make PHP's practical nature possible −

a. Simplicity
b. Efficiency
c. Security
d. Flexibility
e. Familiarity

Chapter IV

Result & Analysis

4.1 Introduction

In this chapter details description of snapshot of proposed referral system , analysis and results obtained from the system design .

Firstly, the admin enters the system using the user name and password that in Figure (4.1), then Figure (4.2) includes the manager to add the employees using the full name and e-mail of the user and then the system send a message containing the code to the user in the email that in the Figure (4.3) The user enters the email and the code sent that in the Figure (4.4) then the user enter the code in Figure (4.5) and then the registration screen appears for the user who fills it in the Figure (4.6). Then the system takes the user password and generates 4 passwords, one is the real password , the other is honeywords and encrypt it using hashing and salting algorithm and storages in the database that in the Figure (4.7). This is done by an account the user .

To enter in the system the user must write the user name and password that in the Figure (4.8) and the system checks the password Is there in the database or not, if the system find password it will do another check to detect whether the user is real or not, if yes, the user enters the system and if not The system sends a message to the manager in the existence of an attacker that in the scheme (4.9) and the message contains the name and password of the attacker, and if there is no password in the database, the system enters the attacker into the imaginary system .

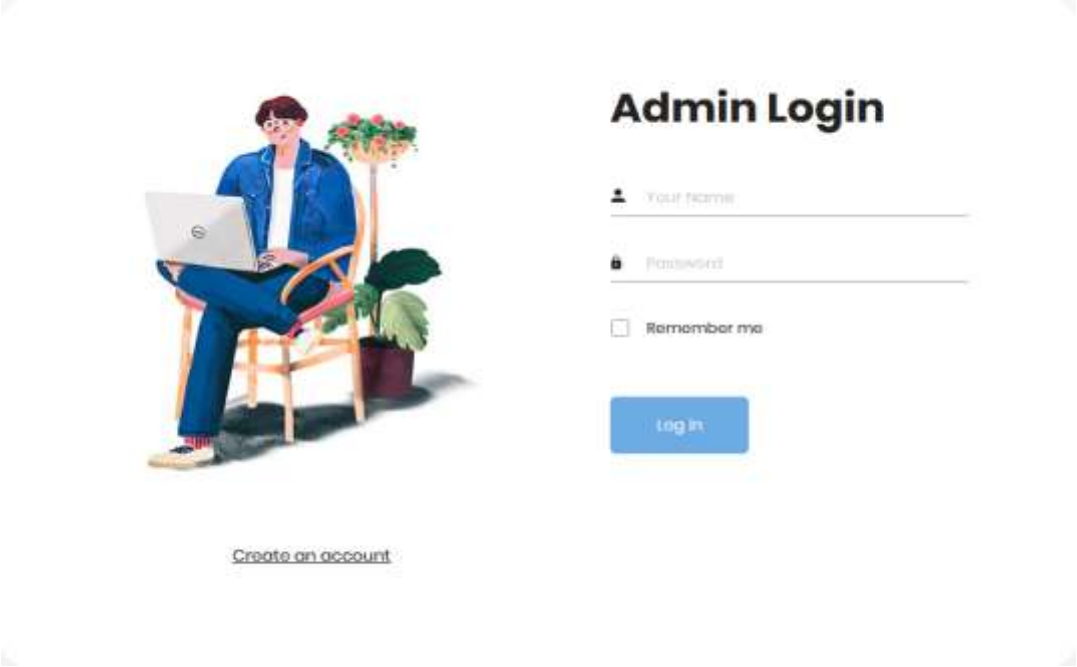

Figure (4.1) Shows the administrator login screen to the system.

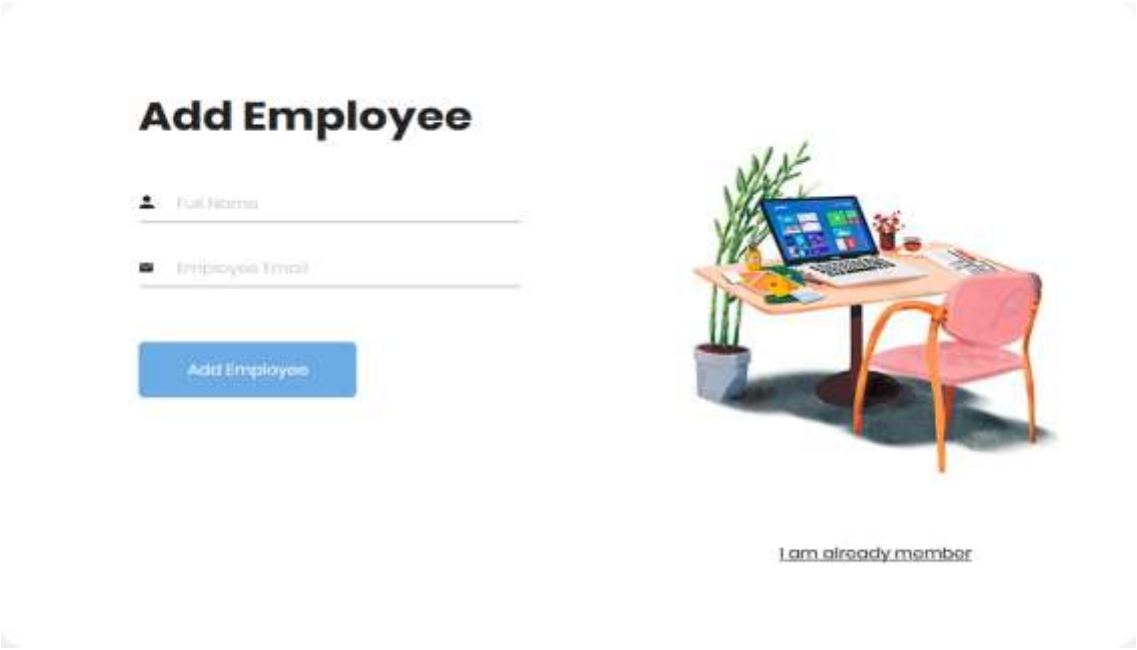

Figure (4.2) Shows the Manager screen to add employees after login.

After entering the employee's name and e-mail by admin , the employee receiving a message in an email containing a code .

←T→	id	fullName	employeeCode	email	used
Edit Copy Delete	4	doaa mohammed	966471	doaa@gmail.com	0

Figure (4.3) Shows the code that send to user in an email .

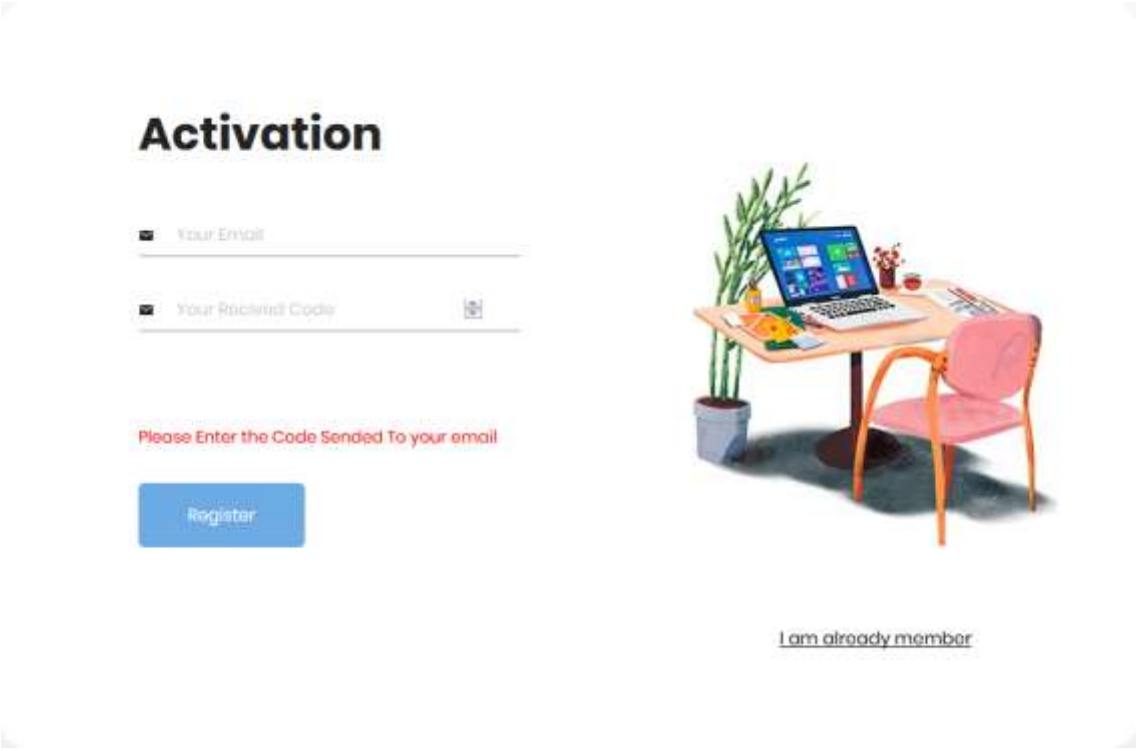

Figure (4.4) : Shows the activation screen that contains the code that was sent and the email .

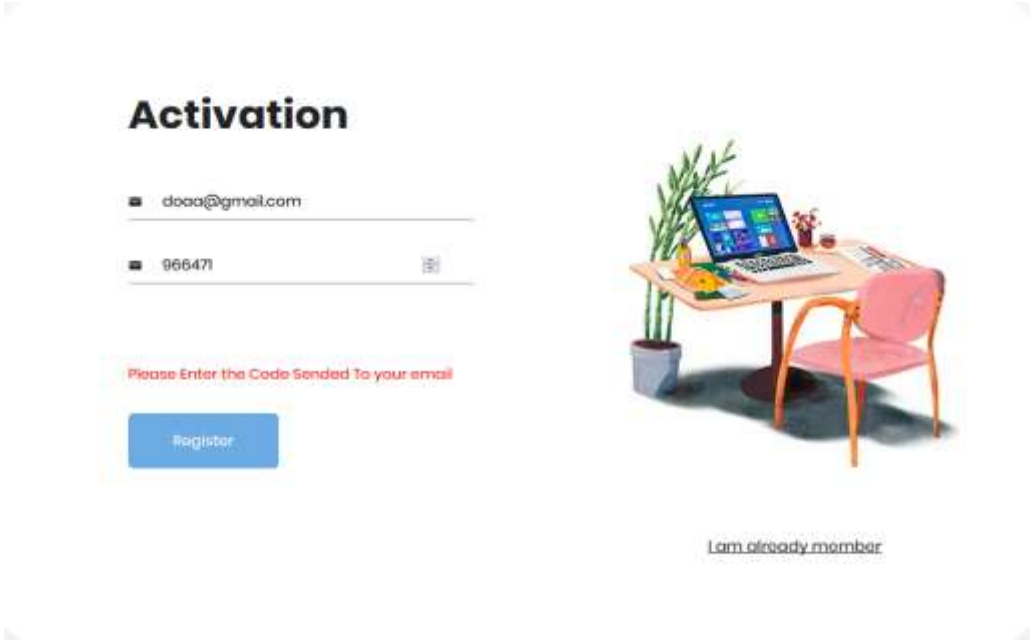

Figure (4.5) Shows the activation screen after input email and code.

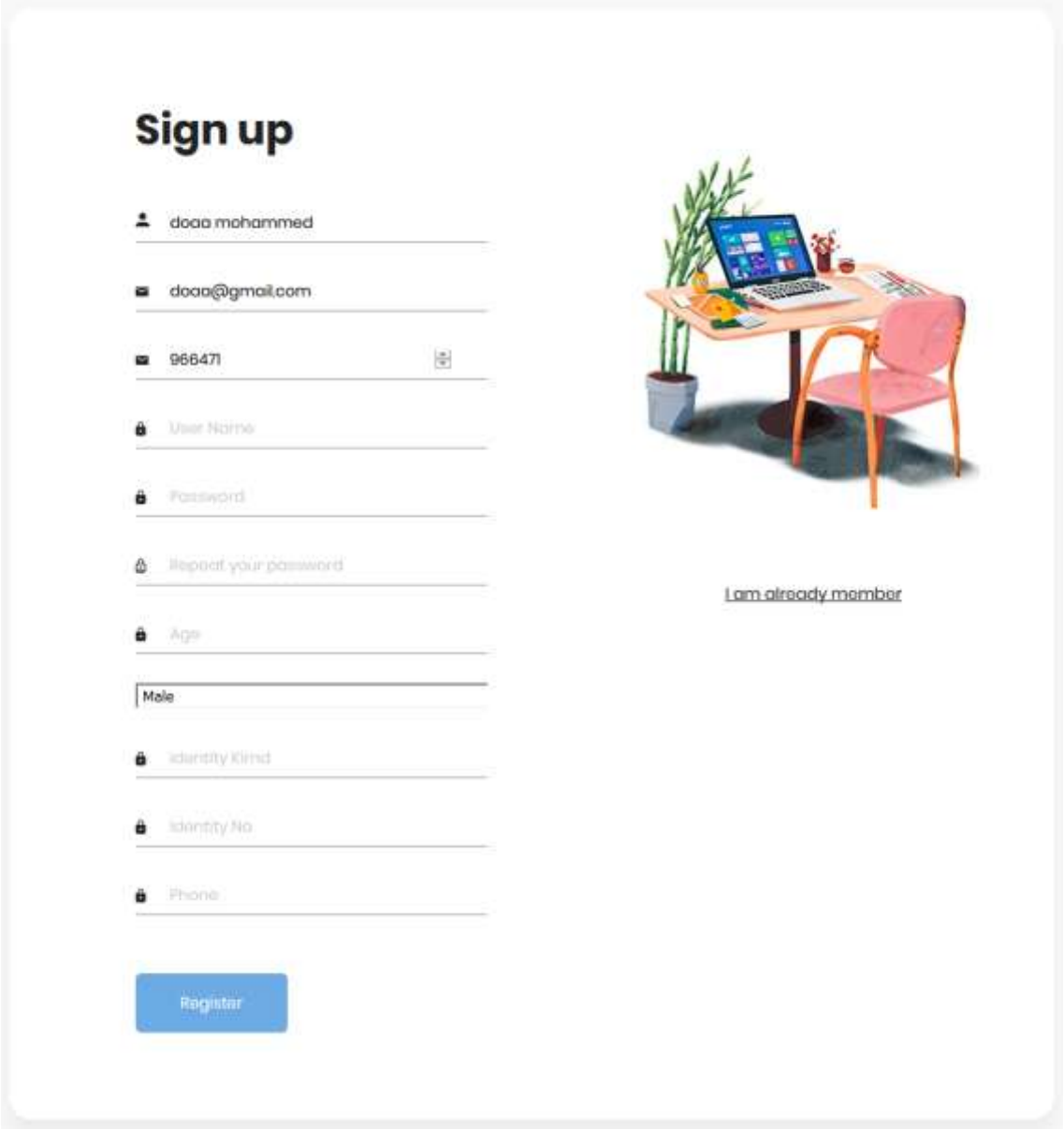

Figure (4.6) Shows the registration screen of employee .

After this screen, the system will take the password input and give a number of seeds the size of N bits and password encryption by hashing and salting algorithm and the improduce of 4 passwords from each .

	Id	regId	Password
Edit Copy Delete	1	1	9d75c525
Edit Copy Delete	2	1	b575c525
Edit Copy Delete	3	1	b575c525
Edit Copy Delete	4	1	b575c525

Figure (4.7) Shows the `Produce` of 4 passwords of employee .

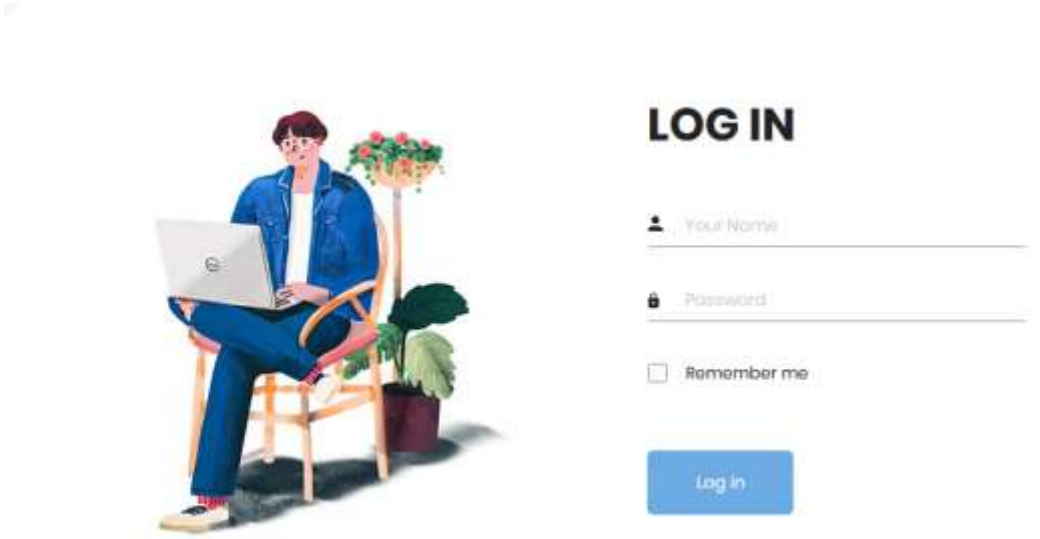

Figure (4.8) Shows the login screen of employee to system .

After this screen, the system takes the password for that user to verify that the server exists in the database. If the password is present in the database, the server sends that user's password to the Honeycomb to classify the password is real password . If the password is a real password, the honeycomb scanner allows that user to access the system. Otherwise, the Honeycomb sends an alert message to the administrator to enter his or her words into the system.

If the password is not present in the database, the server will take the user to imaginary system and send a message to the administrator that an attack exists and the message contains the name and password which were entered.

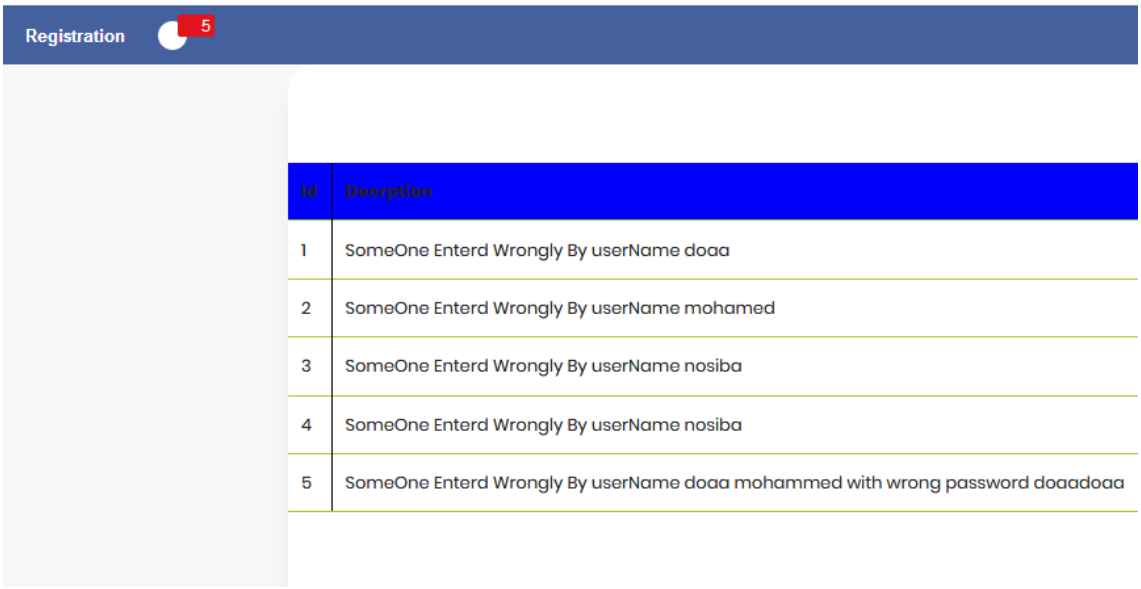

Figure (4.9) Shows the alerts message screen were send to administrator .

4.2 Decryption

When the attacker gets the ciphertext from the database to decrypt , the encrypted password (in figure (4.9)) and the secret key (in figure (4.9)) must be entered, the attacker enters the ciphertext and the secret key is guessed. The system generates a valid message. Not only is the message produced valid, but the probability produced is the same as expected. In this way, the honey encryption system protects against low entropy passwords as well as low entropy message space.

In this section, a screen appears where the encrypted text obtained from the database is entered, the second screen is displayed on the secret key that the attacker is entering and the last screen that contains the decrypted password appears.

Figure (4.10) Shows the screen of decryption password .

Figure (4.11) secret key screen

Figure (4.11) Shows the screen after entering the ciphertext .

After this screen comes the screen containing the password that has been decrypted .

4.3 Analysis and results:

This password hashing scheme is implemented in PHP**.** This programming language was designed to meet all the real world requirements with its key features and easy for the programmers to learn and use efficiently. Unlike other programming system, PHP provides a small number of clear ways to achieve a given task .

Password based encryption (PBE) algorithm plays an important role in various areas . But PBE is not much safe to provide the security to the users because large number of attacks. Here it examines the security of the system against some possible attack which is follows :

4.3.1 Brute Force Attack

Brute force attack is method used by attackers to decode encrypted data such as passwords or data encryption standard key. This attack consists of systematically checking all possible combination of password or keys. But in this encryption technique, this attack is not possible due to the honeywords generation methods. Brute force attacks are very time consuming because of hashing algorithm. Moreover, the attackers cannot classify which is real key or password if they get password file.

4.3.2 DoS Attack

Denial of service attack is a kind of attack to make a machine or network resources unavailable to its users. The purpose of this proposed design is to minimize the DoS vulnerabilities. When a person tries to login into the system, the system automatically check login of the user to detect Dos attack. If the attacker tries to login with his attacking password file, the system raises alarm for user and attack detected. And then, the system can convert the attacker to Imaginary system .

4.3.3 Password Guessing Attack

In this attack, the attacker is stealing the password file from the main server and the honey checker. Then tries to get the original password by decryption but can not tell the user password that belongs to the user account because it is not directly connected to a specific user or account. Otherwise, when the attacker randomly

chooses a password and tries to sign in using key, there can be two successful or unsuccessful possibilities. The first is guessing the correct key so that you can get the correct password, or if the key is incorrect, the system drops one of the honey words. The system alerts the administrator to the user account and can detect the attack .

Chapter V

Conclusion and Recommendations

5.1 Conclusion

Most users choose very simple passwords, so they are easy to remember. So the problem if it's easy to remember, it's also easy to predict an attacker. Most users want to choose and use one password for all their accounts, never change them, write them in future references, and password-based encryption methods (PBE) that are used to protect private data vulnerable to attack by brute force.

When a PBE system is decrypted with an incorrectly key, the system displayed an message confirming the incorrect key, which makes the attacker continue to try to guess the key that makes the system vulnerable to brute force attack.

In this system was used Honeyencryption when the attacker to decrypt the ciphertext with incorrect key The system gives the attacker password existing in the database for another user and when the attacker enters the system does not give a message that the password is wrong , the user is switching to the imaginary system.

This type of encryption technologies are maintained security and integrity . But in the future most of the standard encryption algorithms that exist today can be broken in seconds because the effectiveness of cracking passwords depends to a large extent on two independent things: strength and efficiency. strength is simply the power of computing and efficiency is the ability to guess passwords intelligently. It does not make sense to pass through all eight characters of "aaaaaaaa" to "zzzzzzzz" in such an order. This will be 200 billion passwords possible .

5.2 Recommendations

In the future, can be apply this honeywords generation method in real world application system that is needed for security such as message transmission process in military, government offices and can be made in the complexity of calculating the probability of the message in the process of salt conversion and selecting better cryptographic algorithms .

Although there are limitations and complications in the worst-case scenario, the security of honeycomb encryption declines to the normal PBE security, thus using the alternative.

References

1. Huang, E. Ayday, J. Fellay, J.-P. Hubaux, and A. Juels, "GenoGuard: Protecting genomic data against brute-force attacks," in Proceedings of the 36th IEEE Symposium on Security and Privacy (SP '15), pp. 447–462, Washington, DC, USA, May 2015. View at Publisher · View at Google Scholar · View at Scopus
2. P. Owezarski, "A near real-time algorithm for autonomous identification and characterization of honeypot attacks," in Proceedings of the 10th ACM Symposium on Information, Computer and Communications Security, ASIACCS 2015, pp. 531–542, Singapore, April 2015. View at Publisher · View at Google Scholar · View at Scopus
3. Juels and R. L. Rivest, "Honeywords: making password-cracking detectable," in Proceedings of the ACM SIGSAC Conference on Computer and Communications Security (CCS '13), pp. 145–160, ACM, November 2013. View at Publisher · View at Google Scholar ·
4. Applied cryptography , second Edition: protocols Algorithms , and source code in C(cloth) (publisher: john wiley & sons , Inc .):Bruce Schneier
5. http://elearningatria.files.wordpress.com/2013/10/ise-viii-information-and-network-security-06is835-notes.pdf .
6. https://www.greycampus.com/blog/information-security/what-are-the-best-password-cracking-tools .
7. Spitzner, Lance. Honeypots: tracking hackers. Vol. 1. Reading: Addison-Wesley, 2003.
8. Ari Jules, Ronald L. Rivest. " Honeywords: Making Password-e Cracking Detectable." MIT CSAIL,
   May 2, 2013 .
9. ^ a b c d x.800 : security architecture for open systems interconnection for CCITT applications .
10. Ziya Alper Genc, Suleyman Kardas, Mehmet Sabir Kiraz. "Examination of a New Defence Mechanism: Honeywords," Inernational Journal of Engineering Trends and Technology (IJETT), vol. 27, Number 4, Sept. 2015.