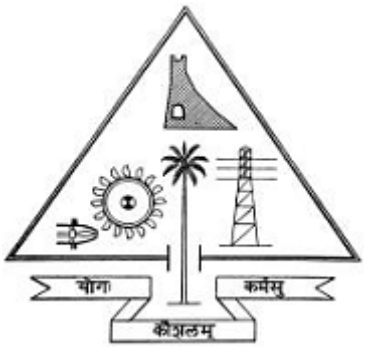
AUTHENTICATION BY ENCRYPTED NEGATIVE PASSWORD

Mini project submitted in partial fulfillment of the requirements for the award of the degree of ***Master of Computer Applications*** *of the* ***APJ Abdul Kalam Technological University***

submitted by

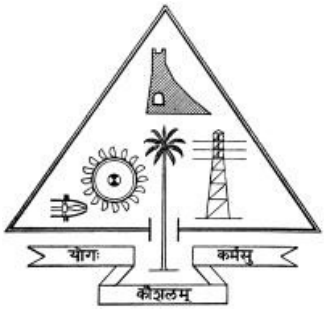
ANJANA R NAIR **(TCR20MCA008)**



DEPARTMENT OF COMPUTER APPLICATIONS
GOVERNMENT ENGINEERING COLLEGE
THRISSUR - 680009

DECEMBER 2021

DEPARTMENT OF COMPUTER APPLICATIONS
GOVERNMENT ENGINEERING COLLEGE, THRISSUR
THRISSUR, KERALA STATE, PIN 680009



CERTIFICATE

This is to certify that the mini project titled ***"AUTHENTICATION BY ENCRYPTED NEGATIVE PASSWORD"*** *is a bonafide work done by*
ANJANA R NAIR (TCR20MCA2008)
under my supervision and guidance, and is submitted in February 2022 in partial fulfillment of the requirements for the award of the Degree of Master of Computer Applications from APJ Abdul Kalam Technological University(KTU).

Project Coordinator	**Project Guide**	**Head of the Dept.**
Dr. Sminesh C N	Asst.Prof. Hussain Ahmed	Dr. Sminesh C N

Place : THRISSUR
Date :

DECLARATION

I hereby declare that the mini project named, **Authentication By Encrypted Negative Password**, is my own work and that, to the best of my knowledge and belief, it contains no material previously published another person nor material which has been accepted for the award of any other degree or course of the university or any other institute of higher learning, except where due acknowledgement and reference has been made in the text.

Signature

ANJANA R NAIR **(TCR20MCA008)**

Place : THRISSUR

Date :

ACKNOWLEDGEMENT

We would like to thank Computer Application Department of GEC Thrissur, for giving us this opportunity to pursue this project and successfully complete it.

We are highly indebted to our guide, **Asst.Prof.Hussain Ahmed** for his guidance and constant supervision as well as for providing necessary information regarding the project and also for his support in completing the project.

We express our heart-felt gratitude to project coordinator,
Dr. Sminesh C N, for her committed guidance, valuable suggestions, constructive criticisms and precious time that she invested throughout the work.

We also express special thanks to the Head of the Computer Applications Department, **Dr. Sminesh C N**, for his keen support and consistent encouragement in our academic activities.

We sincerely thank all other faculties of MCA department for guiding through the processes involved in the project .

ABSTRACT

Secure password storage is a vital aspect in systems based on password authentication, which is still the most widely used authentication technique, despite its some security flaws.The proposed password authentication framework is designed for secure password storage and could be easily integrated into existing authentication systems. In this framework, first, the received plain password from a client is hashed through a cryptographic hash function (e.g., SHA-256).Then, the hashed password is converted into a negative password.Finally, the negative password is encrypted into an Encrypted Negative Password (abbreviated as ENP) using a symmetric-key algorithm (e.g., AES).. The cryptographic hash function and symmetric encryption make it difficult to crack passwords from ENPs. The ENP could resist lookup table attack and provide stronger password protection under dictionary attack.The ENP does not introduce extra elements (e.g.,salt); besides this, the ENP could still resist precomputation attacks.The ENP is the first password protection scheme that combines the cryptographic hash function,the negative password and the symmetric-key algorithm, without the need for additional information except the plain password.

CONTENTS

LIST OF TABLES

LIST OF FIGURES

NOMENCLATURE

AES	Advanced Encryption Standard
DFD	Data Flow Diagram
ENP	Encrypted Negative Password
NDB	Negative Database
SHA	Secure Hash Algorithm
SQL	Structured Query Language

CHAPTER 1
INTRODUCTION

Owing to the development of the Internet, a vast number of online services have emerged, in which password authentication is the most widely used authentication technique, for it is available at a low cost and easy to deploy.Hence, password security always attracts great interest from academia and industry.

1.1 Background

Despite great research achievements on password security, passwords are still cracked since user's careless behaviors. For instance, many users often select weak passwords;they tend to reuse same passwords in different systems; they usually set their passwords using familiar vocabulary for its convenience to remember. In addition, system problems may cause password compromises.Adversaries may log into high security systems through cracked passwords from systems of low security.

1.2 Motivation

Typical password protection schemes include hashed password, salted password and key stretching. Among these schemes, hashed password would be gradually eliminated for its vulnerability for precomputation attacks.Although salted password could resist precomputation attacks, it introduces an extra element (i.e., salt) and could not resist dictionary attack.Key stretching schemes impose an extra burden on programmers for configuring more parameters. Some new password protection schemes were proposed, they are

similar to typical password protection schemes essentially.

1.3 Objective

The project was undertaken with the following objectives

- Introduce a password protection scheme.
- Resistance to Lookup Table attack
- Resistance to Dictionary attacks
- No dependence on salt

1.4 Contribution

A password protection scheme called Encrypted Negative Password (abbreviated as ENP) is proposed, which is based on the Negative Database (abbreviated as NDB), cryptographic hash function and symmetric encryption.The term ”Negative Password” means compression implemented using the wildcard ”*”.The cryptographic hash function and symmetric key encryption make it difficult to crack passwords from ENPs.

CHAPTER 2
EXISTING SYSTEM

Typical Password Protection Schemes:

1. HASHED PASSWORD

The simplest scheme to store passwords is to directly store plain passwords. However, this scheme presents a problem that once adversaries obtain the authentication data table, all passwords are immediately compromised.To safely store passwords, a common scheme is to hash passwords using a cryptographic hash function, because it is infeasible to directly recover plain passwords from hashed passwords. The cryptographic hash function quickly maps data of arbitrary size to a fixed-size sequence of bits. In the authentication system using the hashed password scheme, only hashed passwords are stored. However, hashed passwords cannot resist lookup table attack.Processor resources and storage resources are becoming richer,which makes the precomputed tables used in the above two attacks sufficiently large, so that adversaries could obtain a higher success rate of cracking hashed passwords.

2. SALTED PASSWORD

To resist precomputation attacks, the most common scheme is salted password. In this scheme, the concatenation of a plain password and a random data (called salt) is hashed through a cryptographic hash function.The greater the size of the salt is, the higher the password security is. However,

under dictionary attack, salted passwords are still weak.

3. KEY STRETCHING

To resist dictionary attack, key stretching, which converts weak passwords to enhanced passwords,was proposed. Key stretching could increase the time cost required to every password attempt, so that the power of defending against dictionary attack is increased. In the ENP ,like key stretching, multi-iteration encryption is used to further improve password security under dictionary attack, and compared with key stretching, the ENP does not introduce extra elements (e.g., salt).

CHAPTER 3
ENVIRONMENTAL STUDY

3.1 System Configuration

System configuration describe the hardware and software requirement of the system for development

3.1.1 Hardware Requirements

- Memory : 4 GB of RAM
- Processor : Intel Core i3 or equivalent CPU
- Speed : 2.4 GHz
- Proper Internet Connection

3.1.2 Software Requirements

- Operating system : Windows
- Front End :Java, HTML, CSS
- Backed End : MySQL
- IDE Used : NetBeans

3.2 Software Specification

3.2.1 JAVA

Java is platform independent because it is different from other languages like C, C++, etc. which are compiled into platform specific machines while

Java is a write once, run anywhere language. A platform is the hardware or software environment in which a program runs. Java is one of the world's most important and widely used computer languages, and it has held this distinction for many years. Object Oriented meaning the capability to reuse code. It is possible to develop a single application which can run on multiple platforms like Windows, UNIX, and Macintosh systems. Java does not support the use of pointers. It automatically manages memory garbage-collection routine is activated when system runs short of memory. Java provides the most secure programming environment. Java doesn't just fix security loopholes-it eliminates them, which makes Java the perfect language for programming on the Web.

FEATURES OF JAVA

- Simple: Java is an extension of C and C++ with the added feature of garbage collection and improved memory management.
- Object oriented: Object-oriented programming deals with objects and their behaviors and hence an analogy of the real world can be found in programs.
- Portable: There are no "implementation dependent" (machine/processor dependent) aspects of the specification. The sizes of the primitive data types (integer, float) are specified.
- Secure: Java is intended for use in networked/distribute environments. Toward that end, a lot of emphases have been placed on security. The changes to the semantics of pointers make it impossible for applications to forge access to the user's hard disk.
- Robust: Java is intended for writing programs that must be reliable in a variety of ways. Java puts a lot of emphasis on early checking for possible problems, later dynamic (runtime) checking, and eliminating situations that are error-prone.

3.2.2 MySQL

MySQL is an open source relational database management system. Its nameis combinations of “My”, the name of co-founder Michael Widenius’s daughter, and ”SQL”,the abbreviation for Structured QueryLanguage.MySQLruns on virtually all platforms, including Linux, UNIX and Windows. Although it can be used in a wide range of applications, MySQL is most often associated with web applications and online publishing.MySQL is an important component of an open source enterprise stack called LAMP. LAMP is a web development platform that uses Linux as the operating system, Apache as the web server ,and MySQL as the relational database management system and PHP as the object-oriented scripting language.

Originally conceived by the Swedish company MySQL AB, MySQL was acquired by Sun Microsystems in 2008 and then by Oracle when it bought Sun in 2010. Developers can use MySQL under the GNU General Public License (GPL), but enterprises must obtain a commercial license from Oracle.

MySQL is the RDBMS behind many of the top websites in the world and countless corporate and consumer-facing web-based applications, including Facebook, Twitter and YouTube.

3.2.3 HTML

Hyper Text Markup Language is a scripting language used for writing data in web pages. It specifies the layout and linking commands present in the hypertext documents themselves. The word hypertext refers to the non-linear information on the document, which helps to navigate through the pages. HTML was invented by Tim Burners LEE at CERN, the European laboratory for practical physics in Geneva. An HTML document is a plain ASCII text file created using any text editor with codes inserted in the text to define elements in the document. Users have to provide formatting through their browser platform combination. HTML publishing tools are used for making web pages in the net.

Markup is the process of talking extra ordinary text and extra signals.

Each of the signals used by the markup in the HTML is a command that tells the browser how to display the text. HTML defines the structure of a particular type of document via what is called a document type definition. It is a simple language used to design and describe the layout of web page. HTML also supports multimedia and document links HTML consists of special codes which embedded in text, adds formatting.

3.2.4 CSS

CSS stands for "Cascading Style Sheets". A single CSS file can contain positioning, layout, font,colors and style information for an entire web site. The file can be referenced by each html file on the site.CSS is a means of separating the content of an html document from the style and layout of that document.

3.2.5 NetBeans

NetBeans is an integrated development environment (IDE) for Java. NetBeans allows applications to be developed from a set of modular software components called modules. NetBeans runs on Windows, macOS, Linux and Solaris. In addition to Java development, it has extensions for other languages like PHP, C, C++, HTML5, and JavaScript. NetBeans IDE is an open-source integrated development environment. NetBeans IDE supports development of all Java application types (Java SE (including JavaFX), Java ME, web, EJB and mobile applications) out of the box. Among other features are an Ant-based project system, Maven support, refactorings, version control (supporting CVS, Subversion, Git, Mercurial and Clearcase).The NetBeans IDE Bundle for Web Java EE[22] provides complete tools for all the latest Java EE 6 standards, including the new Java EE 6 Web Profile, Enterprise Java Beans (EJBs), servlets, Java Persistence API, web services, and annotations. NetBeans also supports the JSF 2.0 (Facelets), JavaServer Pages (JSP), Hibernate, Spring, and Struts frameworks, and the Java EE 5 and J2EE 1.4 platforms. It includes GlassFish and Apache Tomcat.

CHAPTER 4
SYSTEM DESIGN

4.1 Architecture

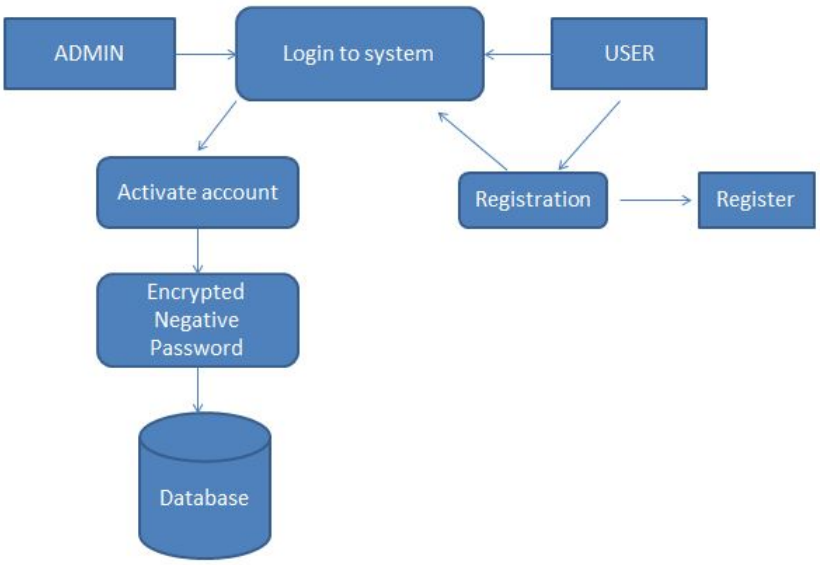


Fig. 4.1: Architecture diagram

An architecture describes the behavior of system, focused on how they interact with each other and with users.It is focused on the data consumed and produced by systems rather than their internal structure.

In this system,there are two types of users.One is Admin , and the other is User. The admin is responsible for activating account for each user who register.After activating account the password entered by the user is stored in the database in the form of Encrypted Negative Password.The registered user can access the account with valid credentials.

4.2 Module Design

4.2.1 Password Generation Module

The generation phase is divided into five steps.

(1) On the client side, a user enters his/her username and password.

(2) The received password is hashed using the selected cryptographic hash function (SHA-256).

(3) The hashed password is converted into a negative password using a Prefix Algorithm.

Here,very entry in a negative password is encoded as the concatenation of two-bit pairs.The two-bit pairs have four forms: 00, 01, 10, and 11, where 00 denotes the symbol '0', 01 denotes the symbol '1', and both 10 and 11 denote the symbol '*'.

(4) The negative password is encrypted to an ENP using the selected symmetric-key algorithm (here, AES) , where the key is the hash value of the plain password.

(5) The username and the resulting ENP are stored in the authentication data table and "Registration success" is returned, which means that the server has accepted the registration request.

4.2.2 Password Verification Module

The authentication phase is divided into five steps.

(1) On the client side, a user enters his/her username and password.

(2) If the received username does not exist in the authentication data table, then "Incorrect username or password!" is returned, which means that the server has rejected the authentication request, and the authentication phase is terminated; otherwise, go to Step (3)

(3) Search the authentication data table for the ENP corresponding to the received username.

(4) The ENP is decrypted using the selected symmetric-key algorithm (AES), where the key is the hash value of the plain password; thus, the negative

password is obtained.

(5) If the hash value of the received password is not the solution of the negative password then "Incorrect username or password!" is returned, which means that the server has rejected the authentication request, and the authentication phase is terminated; otherwise, "Authentication success" is returned, which means that the server has accepted the authentication request.

4.3 Database Design

It is the process of producing a detailed data model of a database. A database is a collection of interrelated data to serve many users quickly and efficiently. A properly designed database provides you with access to up-to-date, accurate information. Because a correct design is essential to achieving your goals in working with a database, investing the time required to learn the principles of good design makes sense.

4.3.1 Table Design

Name	Data type	Null	Constraints
id	int(11)	NO	PRIMARY KEY
name	VARCHAR(100)	NO	
email	VARCHAR(100)	NO	
mobile	VARCHAR(100)	NO	
username	VARCHAR(100)	NO	
secretkey	TEXT	NO	
status	TEXT	NO	
enp	TEXT	NO	

Table 4.1: Authentication Data Table

4.3.2 Input Design

Input design is the link between information system and the user. It comprises of determining set of inputs, validates the data, minimizes the data entries and thereby provides multiuser facilities. The input is designed in such a way that it provides security and ease of use with retaining privacy.Input Design is the process of converting a user-oriented description of the input into a computer-based system. This design is important to avoid errors in the data input process and show the correct direction to the management for getting correct information from the computerized system. It is achieved by creating user-friendly screens for the data entry to handle large volume of data. The goal of designing input is to make data entry easier and to be free from errors. Once identified, the appropriate input media are selected for processing. All the input data are validated and if any data violates any conditions, the user is warned by a message. If the data satisfies all the conditions, it is transferred to the appropriate tables in the database.

4.3.3 Output Design

A quality output is one, which meets the requirements of the end user and presents the information clearly. In any system results of processing are communicated to the users and to other system through outputs. In output design it is determined how the information is to be displaced for immediate need and also the hard copy output. It is the most important and direct source information to the user. Efficient and intelligent output design improves the system's relationship to help user decision-making. Designing computer output should proceed in an organized, well thought out manner; the right output must be developed while ensuring that each output element is designed so that people will find the system can use easily and effectively.

4.4 Data Flow Diagram

A data flow diagram is a graphical representation of the "flow" of data through an information system, modeling its process aspects. A DFD is often used as a preliminary step to create an overview of the system, which can later be elaborated. DFDs can also be used for the visualization of data processing (Structured Design). A DFD shows what kind of information will be input to and output from the system, where the data will come from and go to, and where the data will be stored. It does not show information about the timing of process or information about whether processes will operate in sequence or in parallel.

Data Flow Diagram

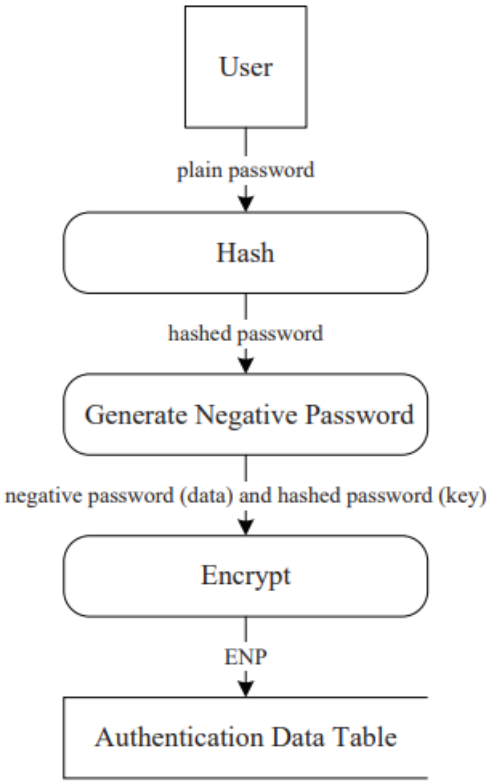


Fig. 4.2: The data flow diagram of the generation procedure of the ENP.

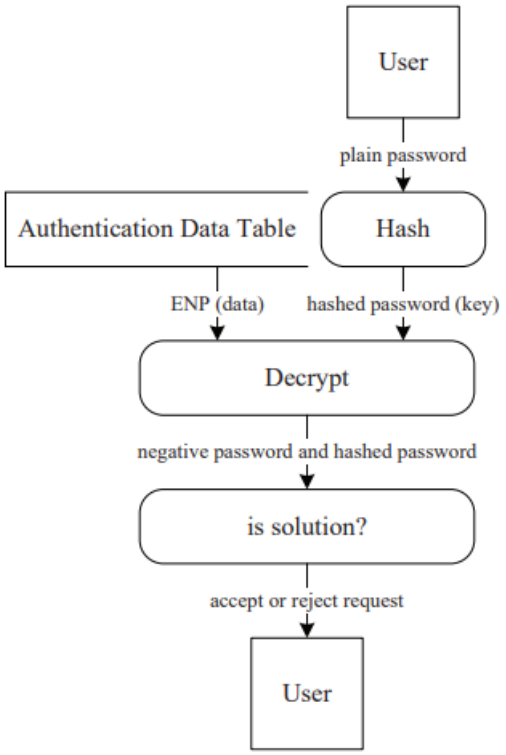


Fig. 4.3: The data flow diagram of the verification procedure of the ENP.

CHAPTER 5
ATTACK COMPLEXITY ANALYSIS

THE COMPARISONS OF ATTACK COMPLEXITY.

Schemes	Lookup table attack		Dictionary attack	
	Time complexity*	Space complexity	Time complexity	Space complexity
Hashed password	$O(N_d * N_p * T_{m\_hash})$	$O(N_d)$	$O(N_d * N_p * (T_h + T_{m\_hash}))$	$O(1)$
Salted password	$O(N_d * 2^l * N_p * T_{m\_hash})$	$O(N_d * 2^l)$	$O(N_d * N_p * (T_h + T_{m\_hash}))$	$O(l)$
Key stretching	$O(N_d * 2^l * N_p * T_{m\_ks})$	$O(N_d * 2^l)$	$O(N_d * N_p * (T_{ks} + T_{m\_ks}))$	$O(l)$
ENPI	$O(N_d * m! * N_p * T_{m\_NP})$	$O(N_d * m!)$	$O(N_d * N_p * (T_h + [n*]T_d + T_{m\_NP}))$	$O(m^2)$

Fig. 5.1: Attack Complexity Table

- N_d: the number of elements in a password list;
- N_p: the number of passwords to be cracked;
- T_h: the time spent on executing a cryptographic hash function;
- T_{ks}: the time spent on executing a key stretching algorithm;
- T_e: the time spent on executing the encryption of a symmetric-key algorithm;
- T_d: the time spent on executing the decryption of a symmetric-key algorithm;
- T_{m_hash}: the time spent on determining whether two hash values match;
- T_{m_ks}: the time spent on determining whether two passwords enhanced by a key stretching algorithm match;
- T_{m_NP}: the time spent on determining whether a hashed password matches a negative password, i.e., whether the hashed password is the solution of the negative password;
- l: the size of the salt (usually is sufficiently large);
- m: the size of the hash value (usually is 128, 160, 256, or 512 bits).

In order to clearly compare the attack complexity of hashed password, salted password, key stretching and the ENP, the time complexity and space complexity under lookup table attack and dictionary attack are listed in Table 6.1

1) Under Lookup Table Attack: Table 6.1 shows that, under lookup table attack, the space requirement of hashed password could be easily satisfied, and the time complexity is also low, so it is the most vulnerable password protection scheme among these schemes. However, it is difficult to meet the space requirements of salted password, key stretching, and the ENP; consequently, they could resist lookup table attack. In addition, the space complexity of the ENP is larger than that of salted password and key stretching when l = m, which makes the ENP better protected against lookup table attack than salted password and key stretching.The ENP could resist lookup table attack without introducing extra elements (e.g., salt).

2) Under Dictionary Attack: By comparison with lookup table attack, nothing needs to be precomputed and stored under dictionary attack; the only thing that adversaries need to do is to attempt every password in the password list for cracking passwords. Hence, our focus under dictionary attack is on the time complexity.From Table 6.1, we could see that the time complexity of the ENP is obviously higher than that of hashed password and salted password ;consequently, the ENP has higher security than hashed password and salted password under dictionary attack.

The ENP does that through multi-iteration encryption (i.e., increasing the n in the time complexity of cracking ENPs using dictionary attack; thus, at the time of cracking ENPs, adversaries need to execute the decryption the equal number of times with the encryption; the greater the number of encryption times is, the more difficult it is to crack ENPs using dictionary attack), while key stretching does that through raising higher resource demands, including processor resources and memory resources. Moreover, by comparison with the key stretching scheme, the ENP takes advantage of symmetric encryption to protect passwords, which further improves the strength of passwords; and the ENP does not introduce salt to resist precomputation attacks, which is the unique advantage of our scheme.

CHAPTER 6

RESULTS AND SCREENSHOTS

6.1 Source code screenshots

HashedPassword.java

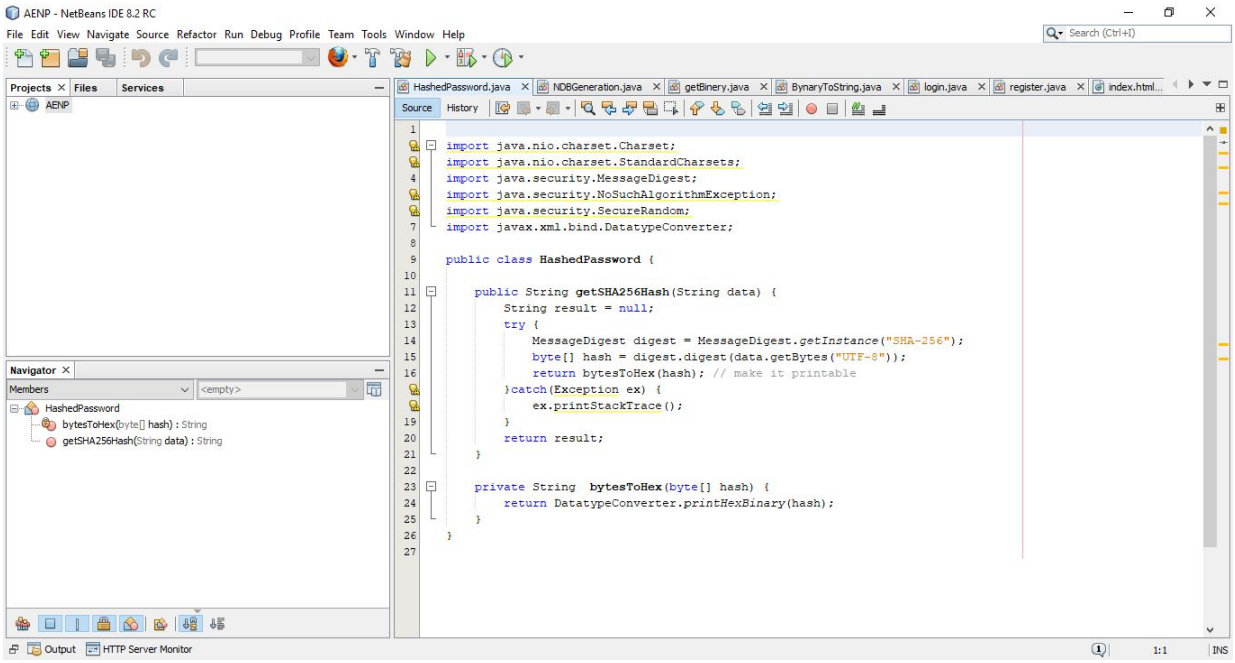


Fig. 6.1: Source code sample screenshots.1

NDBGeneration.java

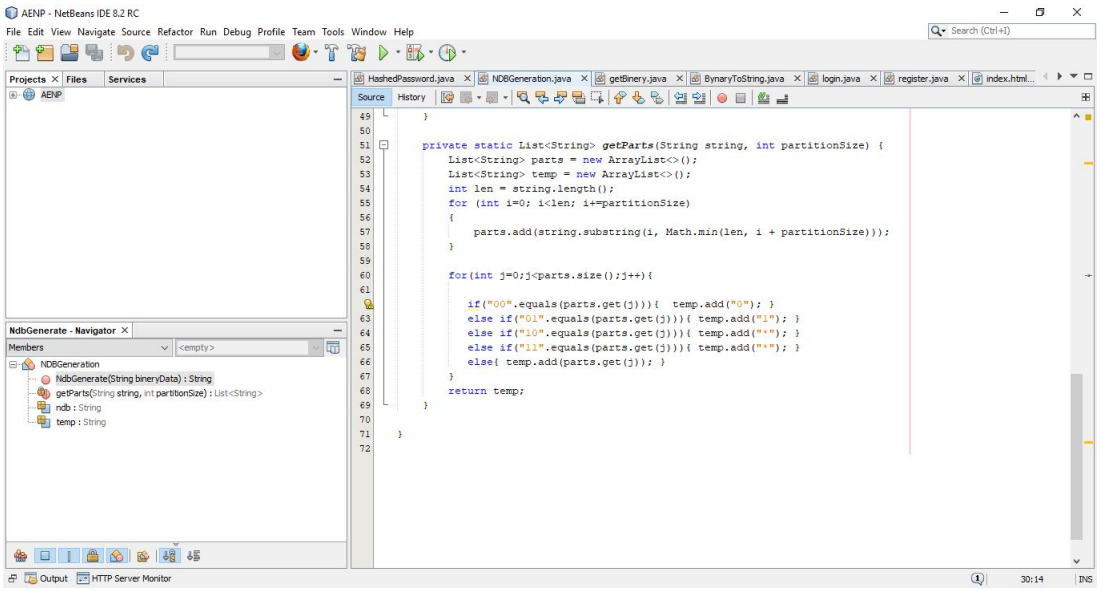


Fig. 6.2: Source code sample screenshots.2

index.html

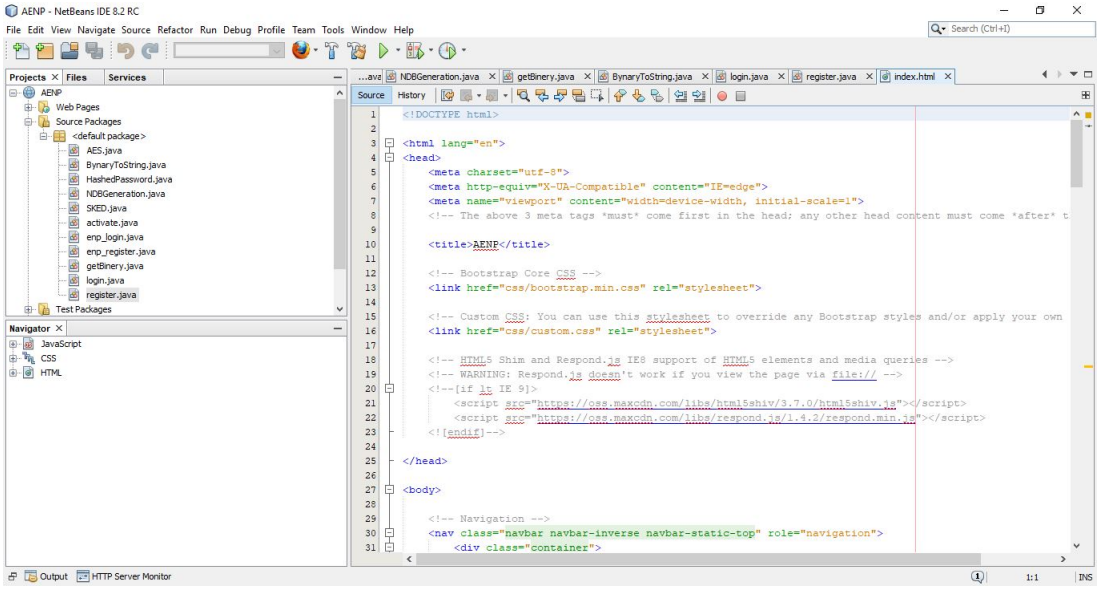


Fig. 6.3: Source code sample screenshots.3

register.java

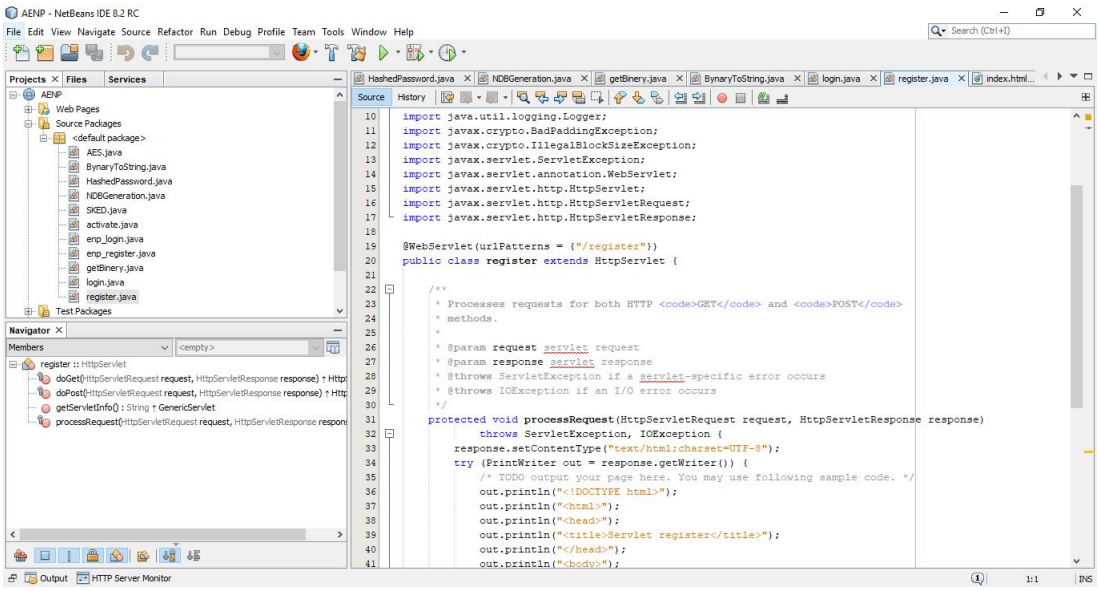


Fig. 6.4: Source code sample screenshots.4

adminhome.java

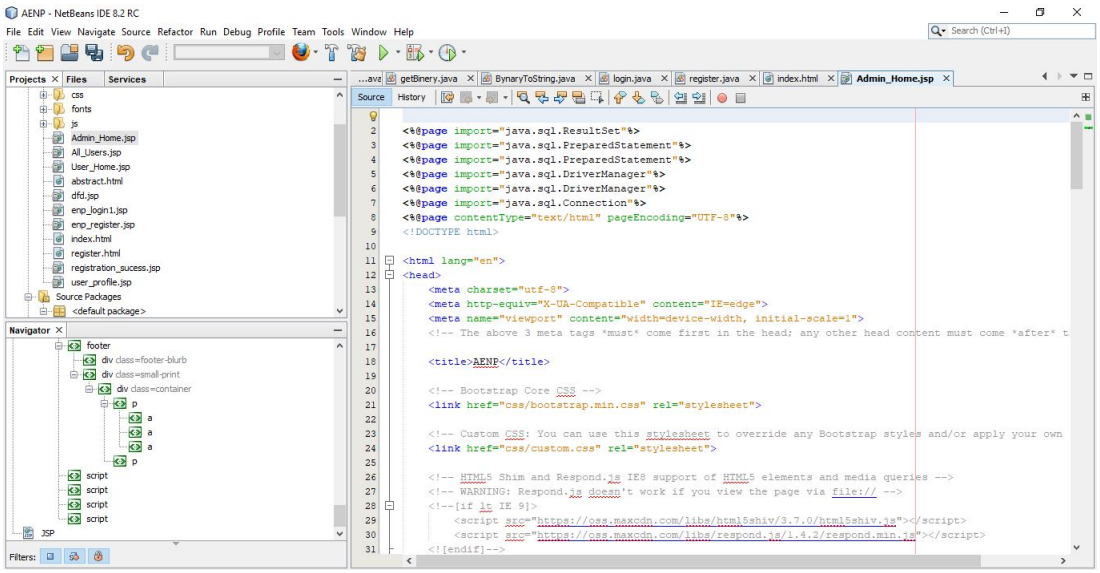


Fig. 6.5: Source code sample screenshots.5

6.2 Results

Fig. 6.6: Index Page

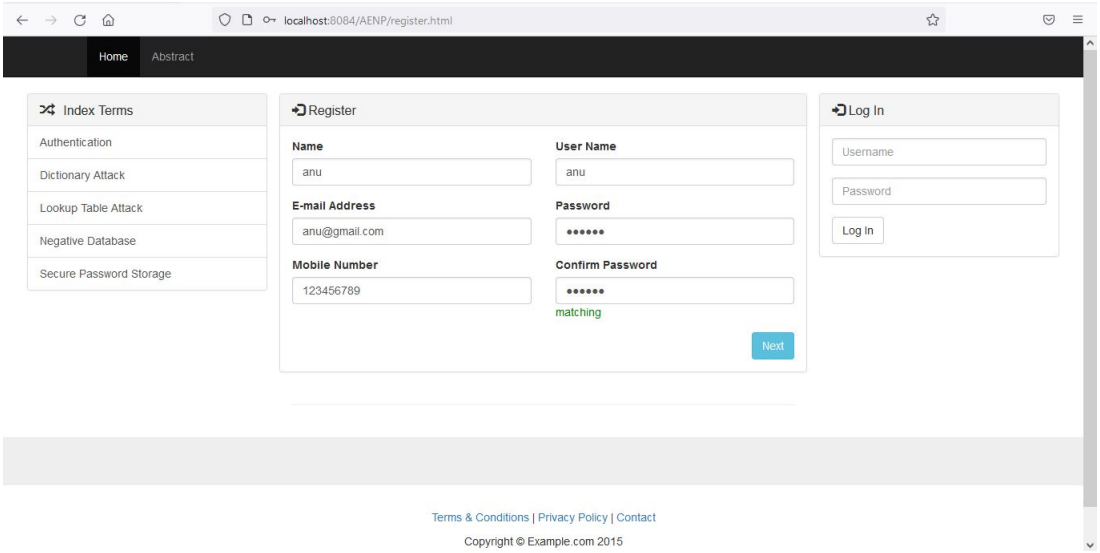


Fig. 6.7: Registration Page

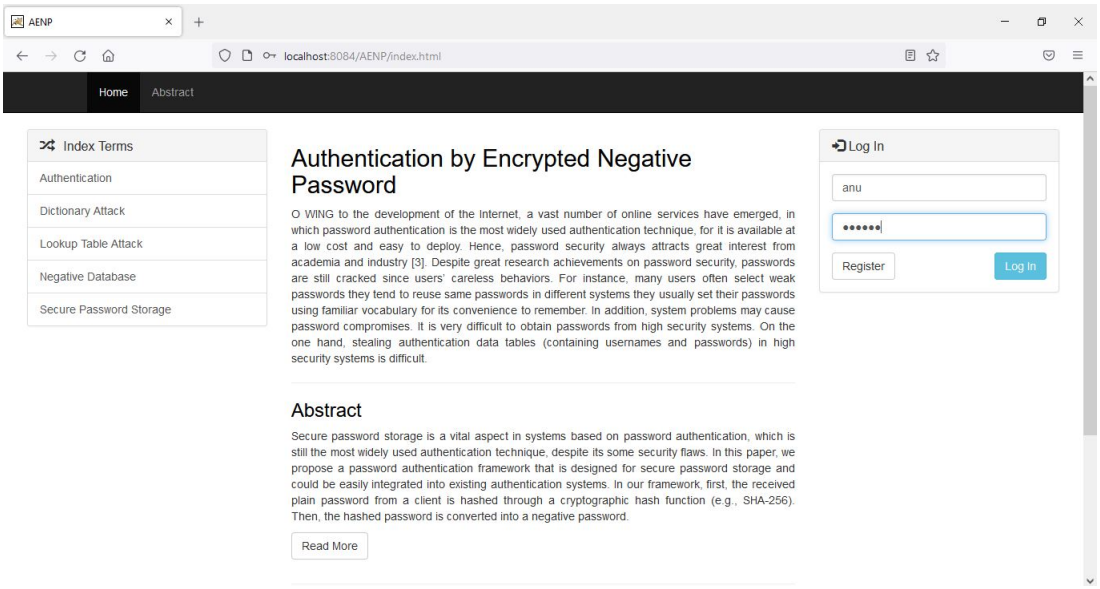


Fig. 6.8: Login Page

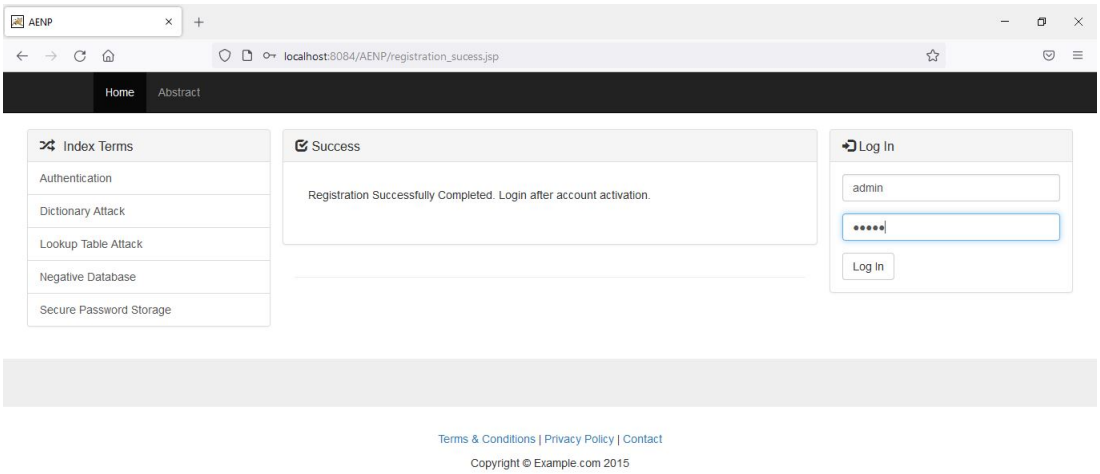


Fig. 6.9: Registration Success Page

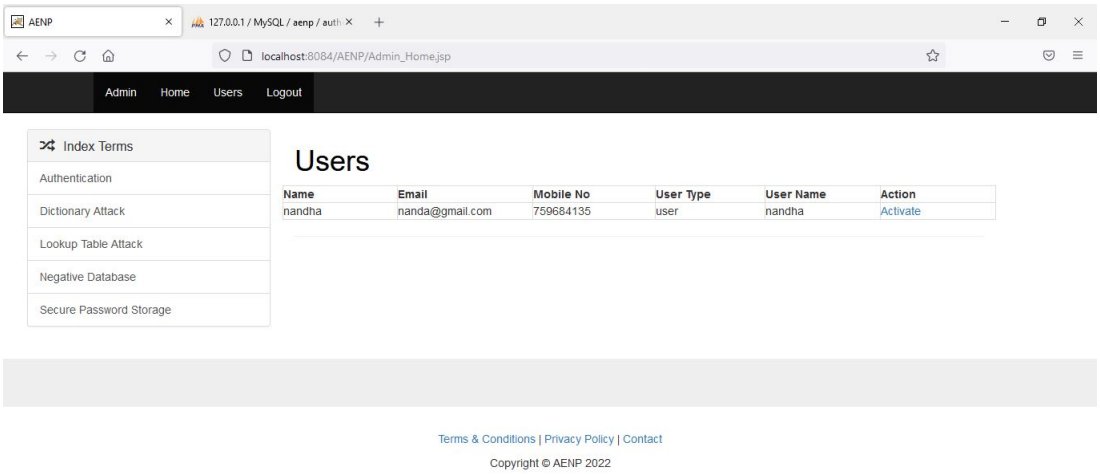


Fig. 6.10: Admin Page

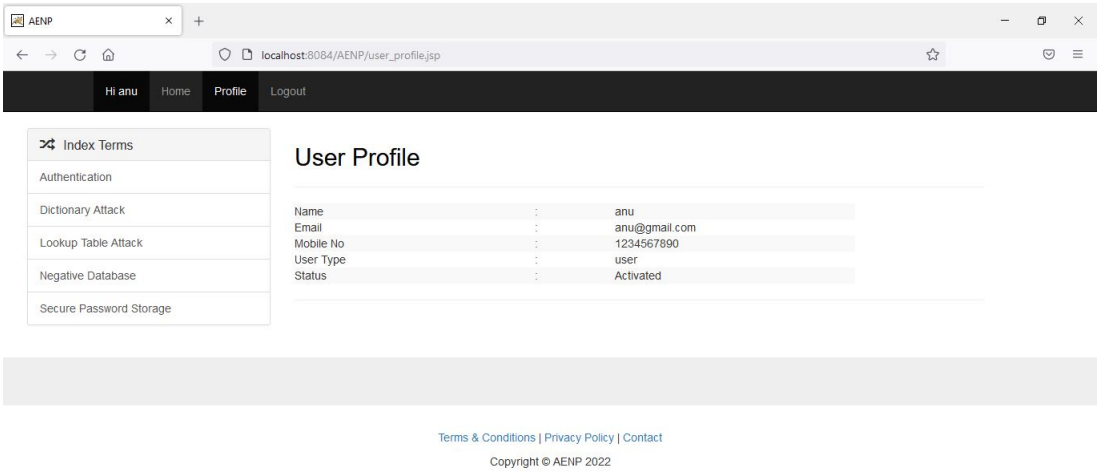
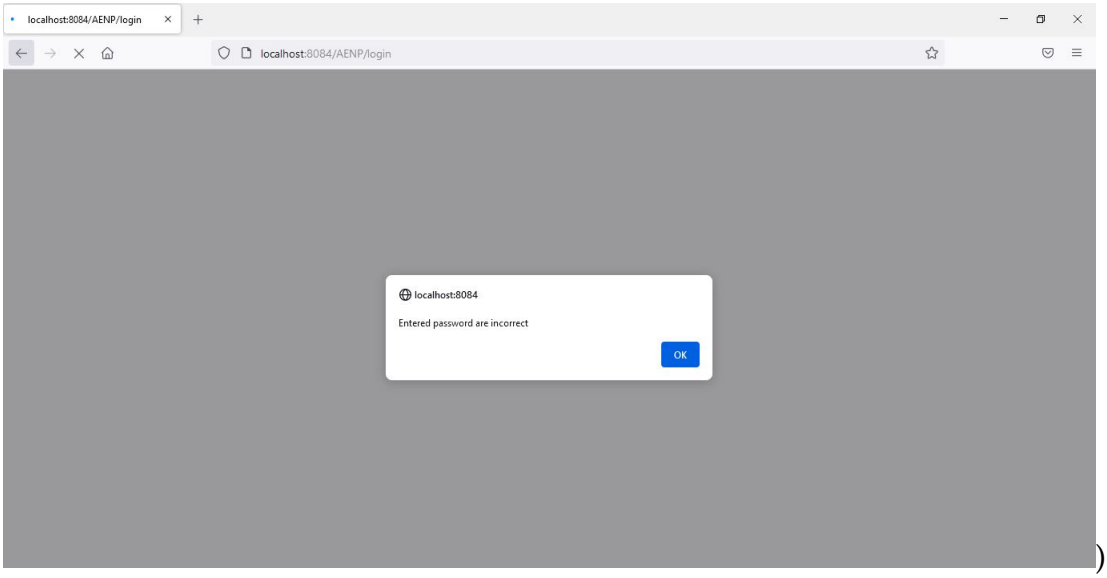


Fig. 6.11: User Profile Page



)

Fig. 6.12: Password Authentication

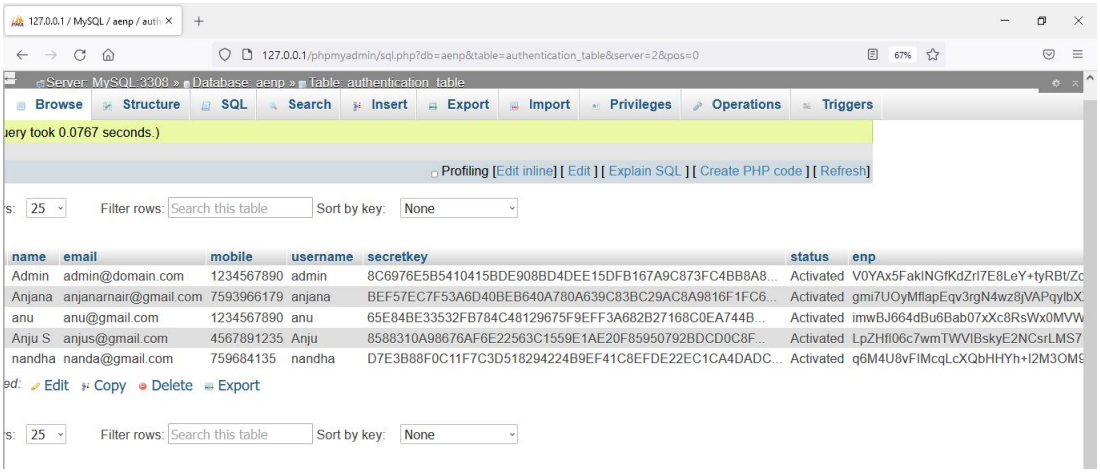


Fig. 6.13: Authentication Data Table

CHAPTER 7
CONCLUSION

A password protection scheme called ENP is proposed, and presented a password authentication framework based on the ENP. In our framework, the entries in the authentication data table are ENPs. After analyzing and comparing the attack complexity of hashed password, salted password, key stretching and the ENP. The results show that the ENP could resist lookup table attack and provide stronger password protection under dictionary attack. It is worth mentioning that the ENP does not need extra elements (e.g., salt) while resisting lookup table attack.

In the future, other NDB generation algorithms will be studied and introduced to the ENP to further improve password security. Furthermore, other techniques, such as multi–factor authentication and challenge–response authentication, can be introduced into our password authentication framework.

BIBLIOGRAPHY

[1] E. H. Spafford,"*Opus:Preventing weak password choices," Computers Security, vol.11, no. 3, pp. 273–278, 1992.* https://www.sciencedirect.com/science/article/pii/0167404892902078

[2] Wenjian Luo, Senior Member, IEEE, Yamin Hu, Hao Jiang, and Junteng Wang,*"Authentication by encrypted negative password"*, https://www.ijert.org/research/authentication-by-encrypted-negativepassword-IJERTCONV8IS12006.pdf

[3] R. Liu, W. Luo, and X. Wang *A hybrid of the prefix algorithm and the q-hidden algorithm for generating single negative databases," in Proceedings of 2011 IEEE Symposium on Computational Intelligence in Cyber Security, Apr. 2011, pp. 31–38.*
https://www.researchgate.net/publication/304558058 Afinegrainedalgorithmforgeneratinghardtoreversenegativedatabases Appendix 25/