Secure Browser-to-Browser Communication using Cryptographic Algorithms and Socket Programming

CSE3502 – Information Security Management

(F1 - SLOT)

Project Based Component Report

By

Akshaj Gupta (19BCE0232)

Tanmayi Gupta (19BCE0610)

Samarth Singh Pundir (19BCE0804)

Under the Guidance of

Prof. Selvi M

School of Computer Science & Engineering (SCOPE)

VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

1.1 Abstract

Communication has become an important part in our day-to-day life. It can be either for business purposes, personal chatting or even covert communication between military personnel. Today there are many chat applications available on the app stores which can be downloaded on personal devices and used to communicate. Many of them offer end- to end encryption like WhatsApp. But there may be situations when you are using an unsecure non-personal device. You cannot download an app or any software on this device, or signup and use the device for communication.

1.2 Aim

To create a project in order to achieve Secure Browser-to-Browser Communication using Cryptographic Algorithms and Socket Programming. Our main focus is to create a chat room in which messages sent are secured through cryptographic algorithms and unique access key of the chat room.

1.3 Objective

In this project we are designing an interface in which the user will be asked for login username and password i.e., the information submitted during signup on the web page. After successful login the user will be asked if he/she wants to create a room or enter a room. After submitting the required inputs, the user types the message after clicking on the send button the message is encrypted with a access key of the room and the encrypted text will be sent to the intended computers where the other users can decrypt the message using the correct access key of the room. In the same way the other users can reply and both users can communicate with one another. The whole communication is secured using the room access key which is used to encrypt and decrypt the messages.

1.4 Scope/Applicability

After developing the application, the application will be tested against vulnerabilities. After careful planning and applying ethical hacking techniques, some vulnerabilities might be found. Our scope is, after exploiting the possible vulnerabilities present in the application, security and other preventive measures will be devised to cop-up with these vulnerabilities. The developed application will be very much secure in terms of integrity, confidentiality, authentication and non-repudiation. Security measures will be taken to prevent any type of database exploitation, SQL injection and cross-site scripting.

1.5 Introduction

The project proposes to design a web-based interface to communicate from one computer to another using TCP client- server paradigm through browser. The communication will be two-way communication which will be secured by any of the selected cryptographic algorithm such as Affine, Hill, Playfair, El Gamal, AES, DES etc. by the users on both the ends. The sending and receiving of the encrypted text on the web will be developed using socket programming based on TCP client- server

model and Web Sockets. The interface is designed for multiple users i.e., any number of users can connect in a room and chat with each other.

This is where this proposed web-based chat application comes to rescue. The proposed model allows any no of users to create a chat room with an access code. After creating the chat room any user having the room id and corresponding correct access code can enter the room and engage in the chat. This chat room is secured using the AES encryption i.e., all the incoming and outgoing messages are secured using the AES algorithm and the symmetric key for that algorithm is the access key of the room.

The web-based chat application is a one-time use application i.e., the messages are not stored on any of the devices but in a socket buffer. Once all the users log out from the chat room the messages cannot be retrieved by any means. This is one of the main advantages of this chat application.

2.1 Literature Review

Ref No.	Paper title	Journal name and year of publication	Work done	Technique used	Disadvantages
1.	The Reality of Applying Security in Web Applications in Academia	International Journal of Advanced Computer Science and Applications, Vol. 5, No. 10, 2014	We studied the possible reasons behind weakness of security in academic organizations. It also exposes the degree of security technologies in protecting the web applications against a set of known threats.	Scanning through the websites of each targeted destination and list all found vulnerabilities. each type of vulnerability was cross-checked with the list of top-ten vulnerabilities of OWASP [9] and if any of the vulnerabilities were matched, then a 10 pe	Robust program verification in early stage against a vector of security vulnerabilities can expose them to potential attacks.
2.	SQLrand: Preventing SQL Injection Attacks	Applied Cryptography and Network Security: Second International Conference, ACNS 2004, Yellow Mountain, China, June 8-	Practical protection mechanism against SQL injection attacks. Such attacks target databases that are accessible through a web front-end, and	It applies the concept of instruction-set randomization to SQL, creating instances of the language that are unpredictable to the attacker.	If we are not using a proxy for the de-randomization process then it would expose the randomization process and can leak data from the database itself

		11, 2004. Proceedings	take advantage of flaws in the input validation logic of Web components such as CGI scripts.		
3.	Web and Database Security	Security Enhanced Applications for Information Systems 2012 May	This paper introduces the types of attacks that target web applications. In addition, several examples on many attack scenarios are introduced.	The various techniques described are Integrity and Privacy, Common Gateway Interface (CGI) script, Access Control, Data transmission through TCP/IP, etc.	One of the major issues is the security and privacy of data and information transferred, stored and processed through at real time.
4.	Automated Initialization of Web Software Projects	Automated Initialization of Web Software Projects." (2018).	This thesis researches methods to design such a template and describes the implementation of a working tool for initializing a new project.	The project template contains a set of predefined components for different aspects of the continuous development pipeline.	Disadvantages of automated equipment include the high capital expenditure required to invest in automation
5.	Research of Web Real-Time Communication Based on Web Socket	Int. J. Communications, Network and System Sciences, 2012, 5, 797-801	In this paper it offer an approach to implement the Web Socket both in client and server side and can decrease network traffic and latency greatly.	Technologies, such as Flash, Comet, and Ajax long polling, have been applied to implement real-time communication between client and server.	It is very hard to figure out the frequency of data updating. Additionally, in the case of no data updating occurring during a period of time, browser's frequent request will generate unnecessary network traffic.
6.	Password Security: An Analysis of Password Strengths and Vulnerabilities	I. J. Computer Network and Information Security, 2016, 7, 23-30	This paper deals with password security, a close look at what goes into making a password strong and the difficulty involved in	It describes tests that were carried out to evaluate the resistance of passwords of varying strength against brute force attacks. It also discusses	Research shows that users tend to keep same or similar passwords for different accounts with little differences. Once a single password

			breaking a password.	overlooked parameters such as entropy and how it ties in to password strength.	becomes known, a number of accounts can be compromised.
7.	Blocking of Brute Force Attack	International Journal of Engineering Research & Technology (IJERT) ISSN: 2278-0181 Vol. 1 Issue 6, August - 2012	In this paper we have presented a system with high security by providing blocking methods to overcome the Brute Force attacks.	This paper shows different methods namely Locking of Accounts, Time bound login, Using CAPTCHA, Unique IP address Login, Query based authentication are used to block the accounts which will provide high security.	An attacker can always discover a password through a brute-force attack. Depending on the password's length and complexity, there could be trillions of possible combinations.
8.	An Efficient Brute Force Attack Handling Techniques for Server Virtualizatio n	Proceedings of the International Conference on Innovative Computing & Communication s (ICICC). 2020.	In this paper we will consider brute force attack and its tools with its implementation and prevention ways or techniques to avoid these types of attacks.	There are following methods and techniques such as password complexity, captcha and limit login access etc. to avoid this brute force attack. These methods are helpful to overcome the attacks which occur due to brute force.	While working on the servers numerous threats and attacks like cracking of passwords, knowing the root of machine, giving privilege to unauthorized users are common attacks that can harm the system and take access of servers.
9.	A New Approach to Hash Function Construction for Textual Data: A Comparison	2014 4th World Congress on Information and Communication Technologies (WICT 2014) 2014 Dec 8 (pp. 39-44). IEEE.	This paper presents a comparison of S-Hash hashing methods for textual data. The S-Hashing method offers a common approach to	The behaviour of the S-Hash function has been tested on Czech and English dictionaries as these two languages belong to	The shorter hash table can be easily constructed without need of all data processing, if shorten by a factor 1/2 or 1/4 etc. Of course, the bucket length will

			textual and geometrical data.	different language groups and on different databases including chemical and toxicological databases.	become longer.
10.	Password Cracking and Countermeasures in Computer Security: A Survey	Wong, D. F., & Chao, L. S. (2014)	The main objective of this work is offering the abecedarian IT security professionals and the common audiences with some knowledge about the computer security and password cracking, and promoting the development of this area.	The unalike techniques used in this paper are password cracking methods, import technologies of password cracking, and the countermeasures against password cracking that are usually designed at two stages including the password design stage and after the design.	The disadvantages are that reactive password checking consumes resources very much and the hackers can also use this strategy to find the weak passwords if they get the password file copy.
11.	Implementation of Web Security & Identity Scheme Based on Session & Online Table	Proceedings of 2009 4th International Conference on Computer Science & Education	This paper proposes a Web security & identity scheme based on session and online table.	Absolute secure system architecture or method is not existed. Security of Web-based application should refer to that attack and declassification to the system needs time long enough.	Web-based application collapses in hardware level, its data and services are not declassified.
12.	Secure web applications via automatic partitioning	ACM SIGOPS Operating Systems Review 41.6 (2007): 31-44	In this research paper we get to know that it is possible to build web applications that enforce security by construction. Not only is there greater assurance that the resulting application is		downsides derived from this paper are that Web applications are hard to build because code and data need to be partitioned for security and with application code split across differently trusted

			secure, but web applications are easier to build.		tiers, the developer is faced with a problem that is when is it secure to place code and data on the client server.
13.	Advanced Encryption Standard (AES) Algorithm to Encrypt and Decrypt Data	Ako Muhamad Abdullah June 16, 2017	This paper will provide an overview of AES algorithm and explaining the particular structure to encrypt and decrypt sensitive data of this algorithm in details and demonstration some previous researches that have done on it with comparing to other algorithms such as DES, 3DES, Blowfish etc.	AES algorithm enables to deal with different key sizes such as 128, 192, and 256 bits with 128 bits block cipher. It uses multiple matrix transformations and key expansions to encrypt and decrypt data.	It uses too simple algebraic structure. Every block is always encrypted in the same way. Hard to implement with software.
14.	Analysis of Step-Reduced SHA-256	Institute for Applied Information Processing and Communications (IAIK) Graz University of Technology, Austria (2008)	The article presented discusses the benefits and limitations of the SHA-256 and security against fast collision search And replacing the SHA 1 family, which was slowly losing strength against brute force attacks.	The SHA-256 uses a hash function method by dividing the clear text into several blocks of 512 multiples and digest these texts using the hash function and after several iterations, it creates a final hash digest of length 256 (as the name suggests).	Firstly, the shift operations in the message expansion of SHA-256 severely limit the usefulness of the perturbation correction approach. Also, the very low probability for one local collision in SHA-256 may give rise to a false feeling of security.
15.	Digital Signatures with RSA and Other Public Key Cryptosystems	Dorothy E. Denning (1984)	This paper discusses the concept of Digital Signatures using RSA. Digital signatures are a	Most public-key cryptosystems like RSA provide secure digital signature schemes The RSA public-key	A problem with digital signatures is that they are not linked to any event in the real world, even if enhanced with

			cryptographic tool to sign messages and verify message signatures in order to provide proof of authenticity and integrity for digital messages or electronic documents.	cryptosystem provides a cryptographically secure digital signature scheme based on the math of the modular exponentiations and discrete logarithms and the difficulty of the integer factorization problem (IFP).	time stamps and other confirmation information. Also, Cryptographic algorithms like RSA may sometimes have Longer key lengths and a longer signature, compared to elliptical curve-based signature.
16.	Phishing Website Detection using Machine Learning Algorithms	Ashit Kumar Dutta 2021	This paper aims to enhance detection method to detect phishing websites using machine learning technology. They achieved 97.14% detection accuracy using random forest algorithm with lowest false positive rate. Scikit-learn tool has been used to import Machine learning algorithms.	This paper deals with machine learning technology for detection of phishing URLs by extracting and analyzing various features of legitimate and phishing URLs. Decision Tree, random forest and Support vector machine algorithms are used to detect phishing websites.	Performance of classifiers has been evaluated by calculating classifier's accuracy score, false negative rate and false positive rate
17.	Detection of Phishing URLs using Machine Learning Techniques	International Conference on Control Communication & Computing India (ICCC) 2013		Several features are compared using various data mining algorithms. The results point to the efficiency that can be achieved using the lexical features. To protect end users from visiting these sites, they can try to identify phishing	On comparing Percentage split60% vs Percentage split90, it shows limitation in the choice of dataset we can use. They can instead try considering a diverse dataset for better results.

				URLs by analyzing their lexical and host-based features.	
18.	Phishing Detection Using Machine Learning Techniques	2020 First International Conference of Smart Systems and Emerging Technologies (SMARTTECH)	In this research, they have implemented and evaluated twelve classifiers on the phishing website dataset that consists of 6157 legitimate websites and 4898 phishing websites.	The main idea behind ensemble algorithms is to combine several weak learners into a stronger one, this is perhaps the primary reason why ensemble-based learning is used in practice for most of the classification problems.	However, for noisy data, the performance of AdaBoost is debated with some arguing that it generalizes well, while others show that noisy data leads to poor performance due to the algorithm spending too much time.
19.	Data Analytics: intelligent anti-phishing techniques based on Machine Learning	Journal of Information & Knowledge Management 2019	In this paper, phishing has been described in the classification context where website phishing is viewed as involving automatic categorization of websites into a predefined set of class values based on a number of available features (variables) and the class variable.	The paper critically analyzed the ways these anti-phishing methods work and showed their positive and negative aspects of the user and performance perspective.	This will equip individuals with knowledge e and skills that may prevent phishing on a wider context within the community.
20.	A framework of new hybrid features of Intelligent detection of Zero-hour Phishing Websites	International Conference on Computational Intelligence in Security for Information Systems (CISIS 2019)	proposed a framework of new hybrid features to predict zero hour phishing websites using machine learning. A total of 31 features,	Random Forest achieved an optimal accuracy of 98.45% and false negatives of 0.73%. The framework took 7.63s to predict a new webpage, suggesting that it	Further research on new potential features and the use of recent ma10 chine learning methodologies such as deep learning and online learning

			26 of them are novel, from five different types of webpages and third party related features were developed to learn the prediction model.	is promising for real time applications.	should be pursued to improve prediction performances and efficiency beyond those of the existing works.

2.2 Observations made and Gaps Identified from Literature Survey

From observing and gathering information from above literature survey we are proposing a method to enable secure communication in order to counteract security attacks such as: man in the middle attack. Moreover, SQL injection attacks are able to take advantage of the design flaws in poorly designed web applications to exploit SQL statements to execute malicious SQL code and the research also shows that users tend to keep same or similar passwords for different accounts. Once a single password becomes known, a number of accounts can be compromised which may lead to unauthorized access of private/confidential data. So, we will be implementing the provision of a digital signature therefore it was vulnerable to a man in the middle attack. Now the problems will be tackled by using digital signatures for both the users to verify each other and authenticate each other thereby making it impossible for a middleman to intercept and change anything in the message or change the message thereby ensuring integrity.

3.1 Proposed Project Work/Model

In this project we will be using different algorithms namely ADVANCED ENCRYPTION STANDARDS (AES), SHA-256 and RSA Digital Signatures.

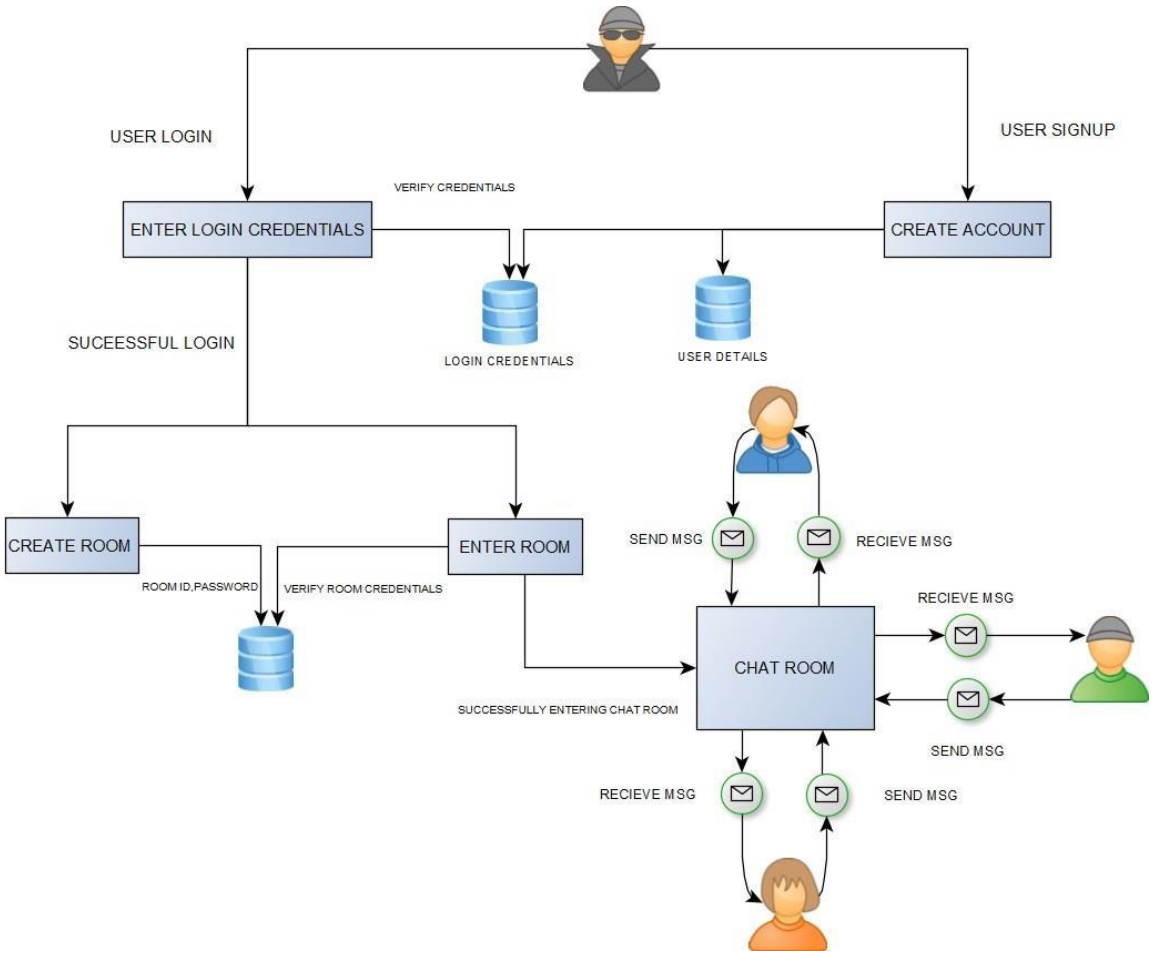


Fig. 3.1 Proposed Project Work/Model

3.1.1 The Advanced Encryption Standard, or AES.

AES is a symmetric block cipher and is implemented in software and hardware throughout the world to encrypt sensitive data. AES is an iterative rather than Feistel cipher. It is based on 'substitution–permutation network'. It comprises of a series of linked operations, some of which involve replacing inputs by specific outputs (substitutions) and others involve shuffling bits around (permutations).

3.1.2 SHA-256

SHA-256 is a member of the SHA-2 cryptographic hash functions designed by the NSA. SHA stands for Secure Hash Algorithm. Cryptographic hash functions are mathematical operations run on digital data; by comparing the computed "hash" (the output from execution of the algorithm) to a known and expected hash value, a person can determine the data's integrity. A one-way hash can be generated from any piece of data, but the data cannot be generated from the hash.

3.1.3 Digital signatures

Digital Signatures are the public-key primitives of message authentication. A digital signature is a technique that binds a person/entity to the digital data. This binding can be independently verified by receiver as well as any third party. Digital signature is a cryptographic value that is calculated from the data and a secret key known only by the signer.

3.2 Security Aspects

3.2.1 Confidentiality

Confidentiality refers to protecting the information from being accessed by unauthorized parties. AES is being used for encrypting data. Even when the transmission or storage medium has been compromised, the encrypted information is practically useless to unauthorized persons without the proper keys for decryption. If intercepted interceptor will not be able to crack unless they know the key. Advanced Encryption Standards (AES) is used for encrypting and decrypting the data using room access key.

3.2.2 Access Control

Users are provided with passcodes to grant access for chat group/room that they have the passcode to access among the different chat groups. It ensures that only the people eligible to be the part of a particular chat group are granted permission and able to log in to the chat room created.

3.2.3 Integrity

Digital signatures being used for ensuring integrity and user authentication. Public and private key pair being used for digital signature. Sender creates a message digest (md1) using a hash function. The message and message digest are encrypted. Transmitted to receiver. Receiver decrypts message and message digest (md1). Receiver applies same hash function to message to create message digest (md2). The two message digests must match to ensure data integrity. RSA alongside message digest is also used for digital signature to verify the user's authentication.

3.2.4 Non-Repudiation

Non repudiation ensures that the sender cannot deny sending something. Chat once posted cannot be deleted and sender cannot deny sending something to the receiver and chats will be stored once the session between the users in a group is terminated/ended. In the chat application non-repudiation is ensured so that the user cannot delete his messages.

3.2.5 Authentication

User ID and password are being used for authenticating that the user belongs to registered set of people who can access the page. All the passwords are encrypted using SHA-256 algorithm. So, in case of any attack on databases, passwords are not compromised.

3.3 Algorithm Description

3.3.1 Advanced Encryption Standard (AES)

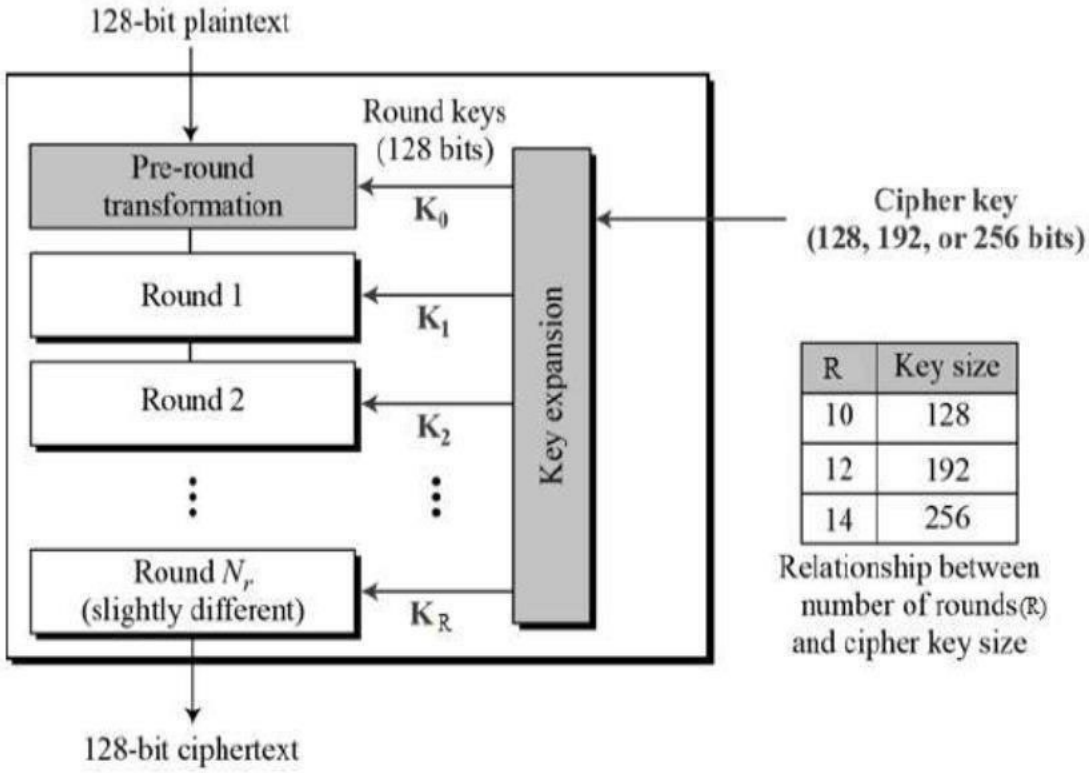
The Advanced Encryption Standard, or AES, is a symmetric block cipher chosen by the U.S. government to protect classified information and is implemented in software and hardware throughout the world to encrypt sensitive data.

AES is an iterative rather than Feistel cipher. It is based on "substitution–permutation network". It comprises of a series of linked operations, some of which involve replacing inputs by specific outputs (substitutions) and others involve shuffling bits around (permutations).

Interestingly, AES performs all its computations on bytes rather than bits. Hence, AES treats the 128 bits of a plaintext block as 16 bytes. These 16 bytes are arranged in four columns and four rows for processing as a matrix.

Unlike DES, the number of rounds in AES is variable and depends on the length of the key. AES uses 10 rounds for 128-bit keys, 12 rounds for 192-bit keys and 14 rounds for 256-bit keys. Each of these rounds uses a different 128-bit round key, which is calculated from the original AES key.

The schematic of AES structure is given in the following illustration:



R	Key size
10	128
12	192
14	256

Relationship between number of rounds(R) and cipher key size

Fig 3.1 AES Block Diagram

ENCRYPTION PROCESS:

Here, we restrict to description of a typical round of AES encryption. Each round comprises of four sub-processes. The first-round process is depicted below:

1. Byte Substitution (SubBytes)

   The 16 input bytes are substituted by looking up a fixed table (S-box) given in design. The result is in a matrix of four rows and four columns.

2. Shiftrows

   Each of the four rows of the matrix is shifted to the left. Any entries that 'fall off' are re-inserted on the right side of row. Shift is carried out as follows:

   - First row is not shifted.
   - Second row is shifted one (byte) position to the left.
   - Third row is shifted two positions to the left.
   - Fourth row is shifted three positions to the left.
   - The result is a new matrix consisting of the same 16 bytes but shifted with respect to each other.

3. MixColumns

   Each column of four bytes is now transformed using a special mathematical function. This function takes as input the four bytes of one column and outputs four completely new bytes, which replace the original column. The result is another new matrix consisting of 16 new bytes. It should be noted that this step is not performed in the last round.

4. Addroundkey

   The 16 bytes of the matrix are now considered as 128 bits and are XORed to the 128 bits of the round key. If this is the last round then the output is the ciphertext. Otherwise, the resulting 128 bits are interpreted as 16 bytes and we begin another similar round.

DECRYPTION PROCESS:

The process of decryption of an AES ciphertext is similar to the encryption process in the reverse order. Each round consists of the four processes conducted in the reverse order:

- Add round key
- Mix columns
- Shift rows
- Byte Substitution

Since sub-processes in each round are in reverse manner, unlike for a Feistel Cipher, the encryption and decryption algorithms need to be separately implemented, although they are very closely related.

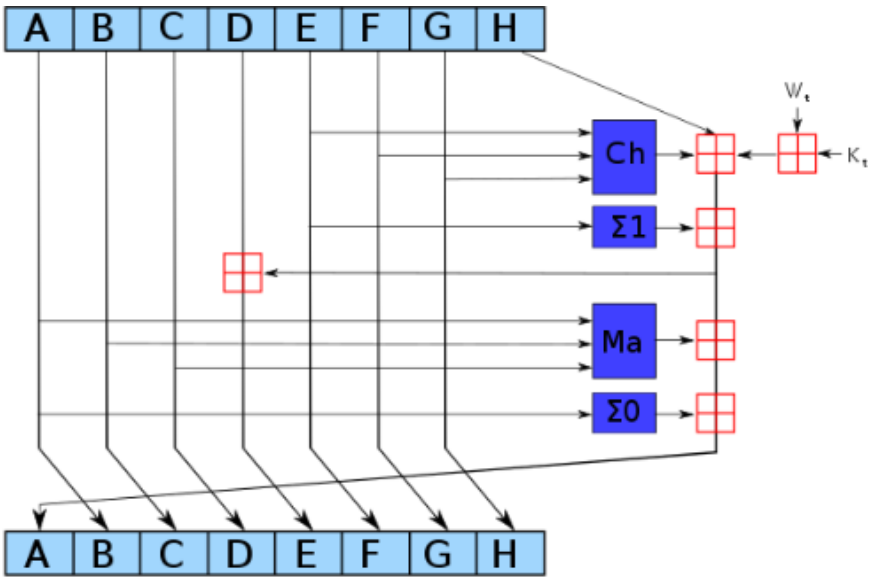
3.3.2 Secure Hash Algorithm (SHA – 256)

SHA-256 is a member of the SHA-2 cryptographic hash functions designed by the NSA. SHA stands for Secure Hash Algorithm. Cryptographic hash functions are mathematical operations run on digital data; by comparing the computed "hash" (the output from execution of the algorithm) to a known and expected hash value, a person can determine the data's integrity. A one-way hash can be generated from any piece of data, but the data cannot be generated from the hash.

SHA-256 has quite good technical parameters:

- Block size indicator (byte): 64
- Maximum allowed message length (bytes): 33
- Characteristics of the message digest size (bytes): 32
- Standard word size (bytes): 4
- Internal position length parameter (byte): 32
- Number of iterations in one cycle: 64
- Speed achieved by the Protocol (MiB/s): approximately 140

The SHA-256 algorithm is based on the Merkle-Damgard construction method, according to which the initial index is divided into blocks immediately after the change is made, and those, in turn, into 16 words.



One iteration in a SHA-2 family compression function. The blue components perform the following operations:

$$\mathrm{Ch}(E,F,G) = (E \wedge F) \oplus (\neg E \wedge G)$$
$$\mathrm{Ma}(A,B,C) = (A \wedge B) \oplus (A \wedge C) \oplus (B \wedge C)$$
$$\Sigma_0(A) = (A \ggg 2) \oplus (A \ggg 13) \oplus (A \ggg 22)$$
$$\Sigma_1(E) = (E \ggg 6) \oplus (E \ggg 11) \oplus (E \ggg 25)$$

The bitwise rotation uses different constants for SHA-512. The given numbers are for SHA-256.

The red ⊞ is addition modulo 2^{32} for SHA-256, or 2^{64} for SHA-512.

Fig 3.2 SHA-256 Block Diagram

3.4 Tech Stack

3.4.1 Technical Overview

The project has used a variety of front and back-end frameworks for implementations such as:

- HTML: For front-end development
- CSS: For front-end development
- JS: For animations and socket programming
- PHP: For front and back-end connections, session creation and user authentication
- NodeJS: Implementation of socket programming and server handling

3.4.2 Hardware & Software Requirements

Hardware

- Client/User Side: Any device with internet connection.
- Server Side: Apache Server to host this site.

Software

- Client/User Side: Internet Browser
- Server Side: Apache Tomcat, PHP, Socket.io

3.5 List of Modules

3.5.1 Sign Up Page:

This page is for the first-time user to sign-up with their personal details. This page ensures that no two users have same username and e-mails.

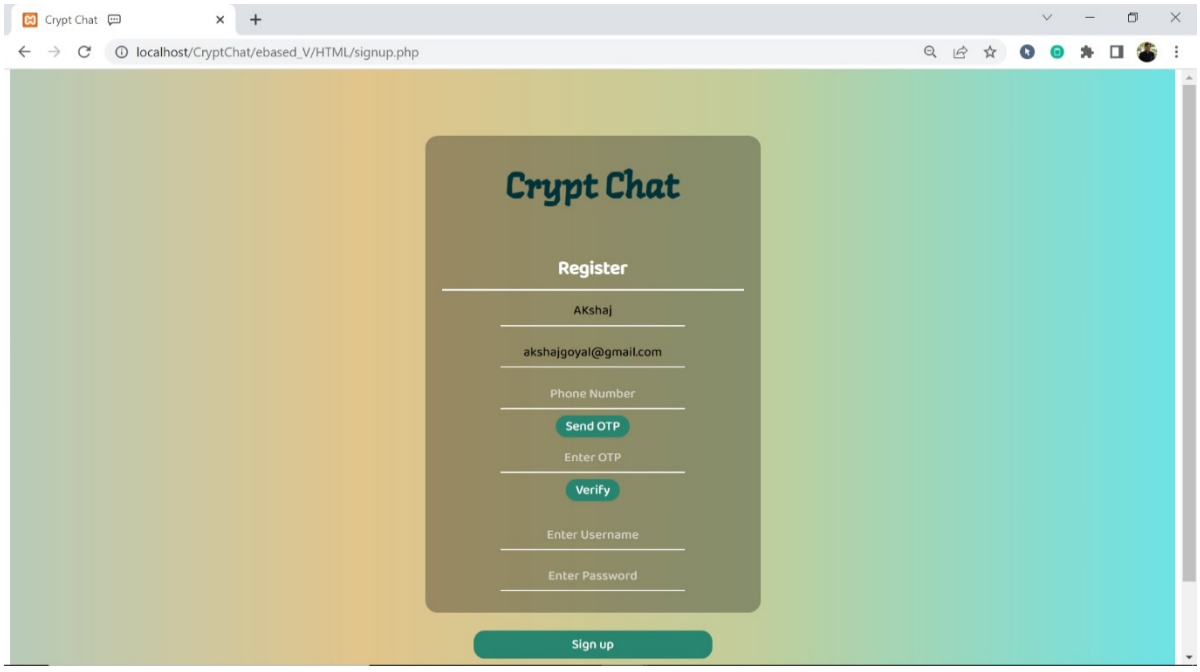


Fig. 3.3 Sign Up Page

3.5.2 Login Page:

This is a user authentication page to verify the registered users and prevent invalid users. This also takes an input by the users to enter a predefined chat room by room administrator.

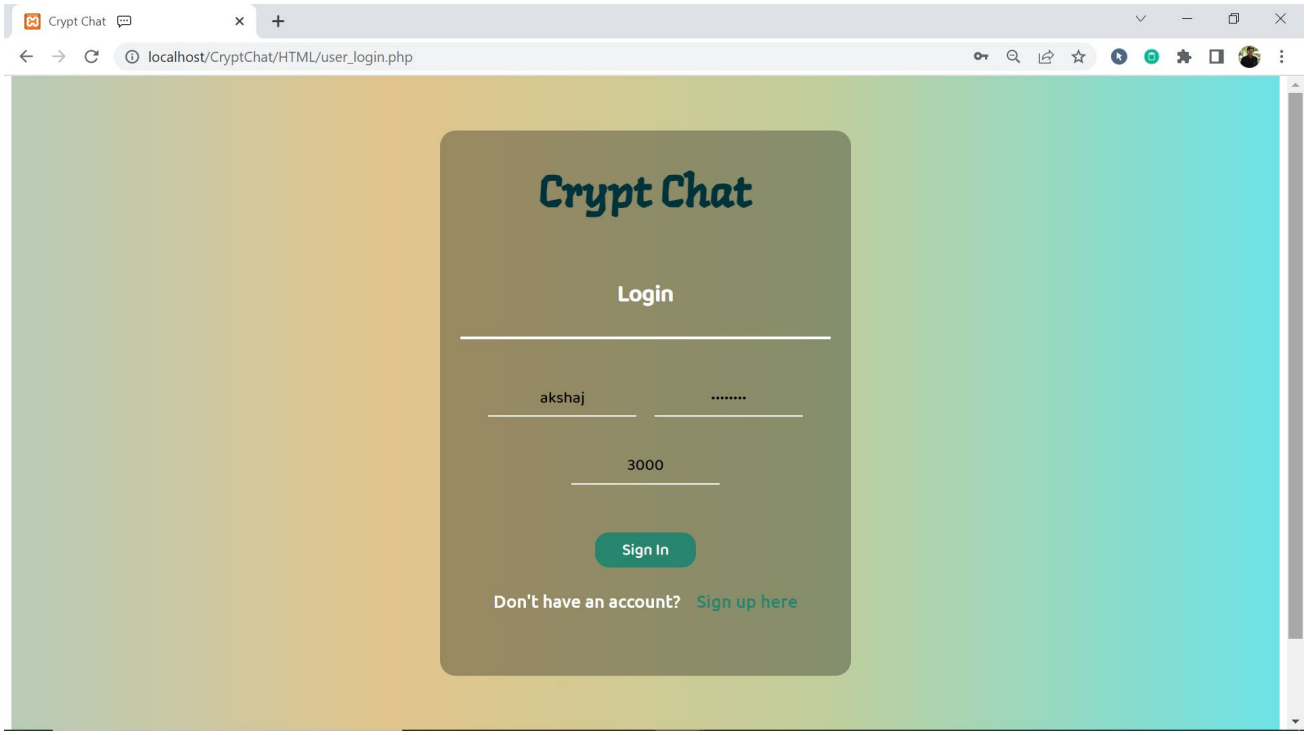


Fig. 3.4 Login Page

3.5.3 Chat Room & Login Page:

This is the chat room login credential i.e., the access code which allows the user to enter into the requested chat room.

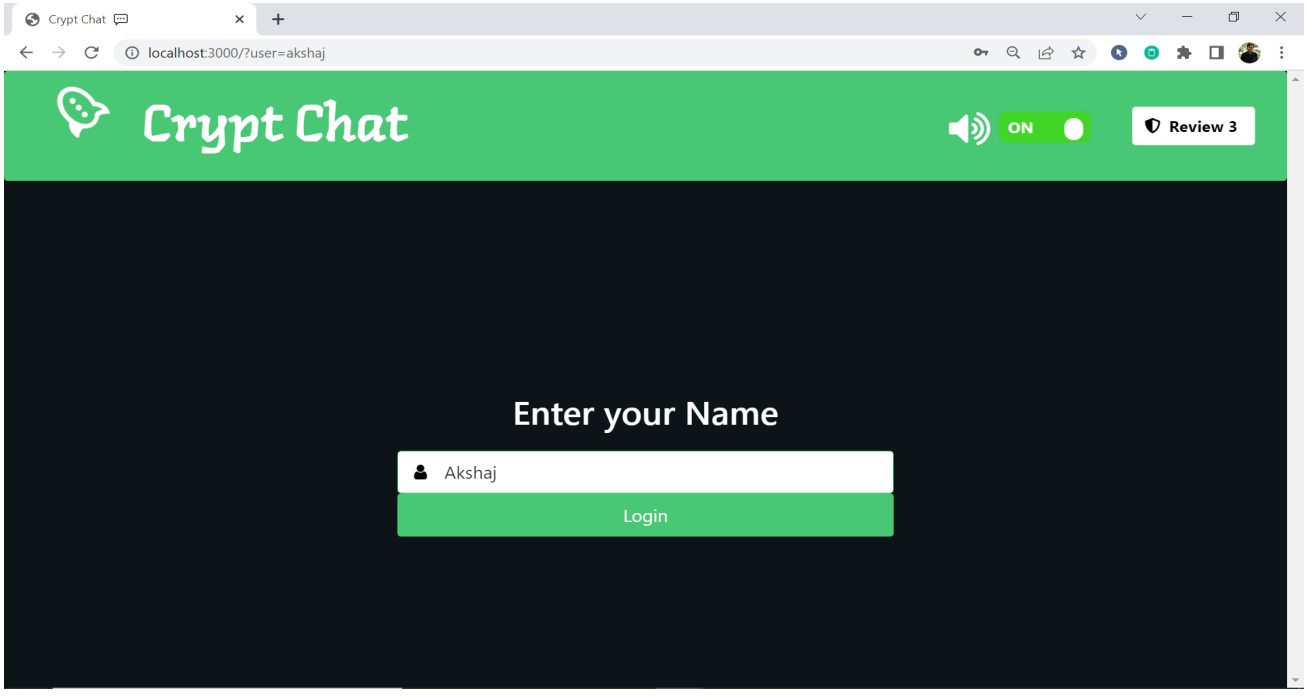


Fig. 3.5 Chat Room & Login Page

3.5.4 Chat Page:

This is the main chat page which is two way encrypted and where the users can chat anonymously by using the username they entered in previous screen. Each user can see the online users in their room. But they cannot see users in the other room.

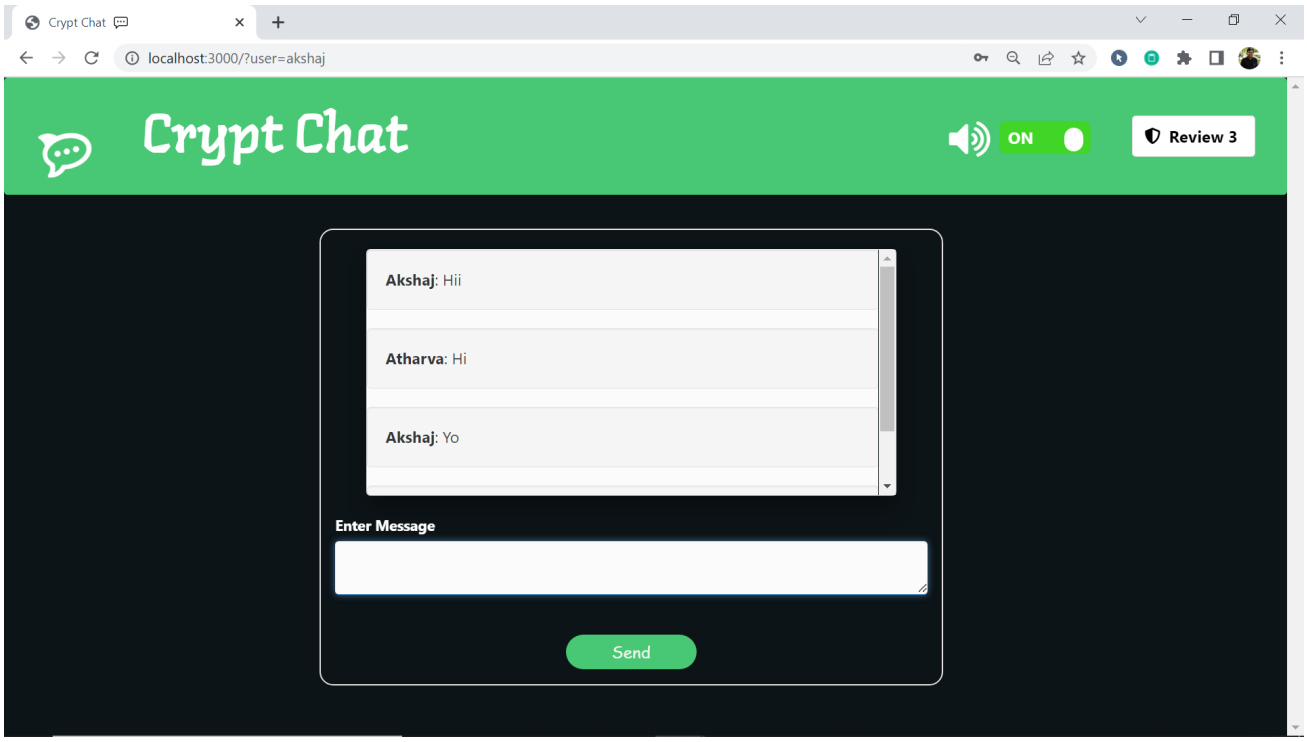


Fig. 3.6 Chat Page

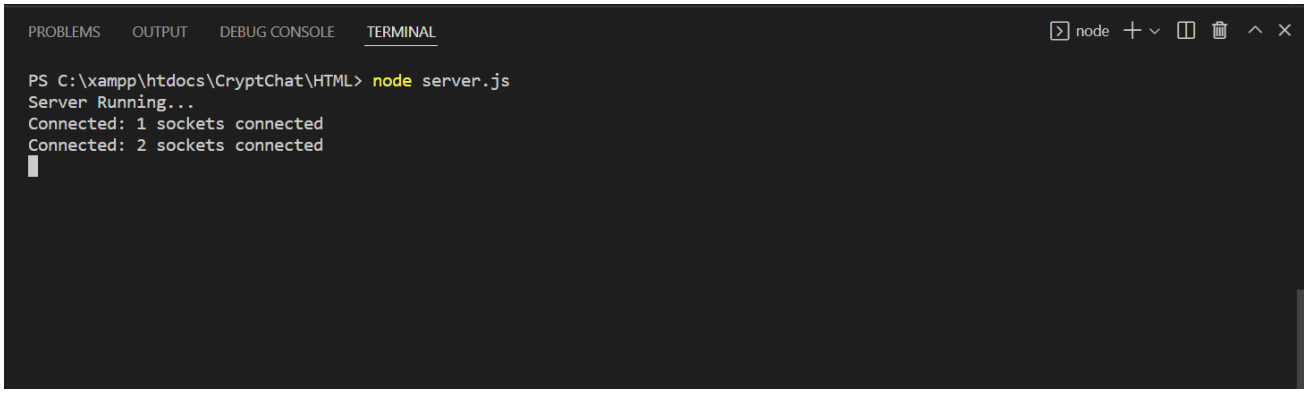


Fig. 3.7 Backend Socket Initialization

4.1 Results and Discussion

4.1.1 Vulnerability Analysis

After developing the application, the application was tested against vulnerabilities. After careful planning and applying ethical hacking techniques, following vulnerabilities were found.

1. SQL Injection attack (SQLi):

SQL Injection is an attack that poisons dynamic SQL statements to comment out certain parts of the statement or appending a condition that will always be true. It takes advantage of the design flaws in poorly designed web applications to exploit SQL statements to execute malicious SQL code. The types of attacks that can be performed using SQL injection vary depending on the type of database engine. The attack works on dynamic SQL statements. A dynamic statement is a statement that is generated at run time using parameters password from a web form or URI query string.

In our application, in the login page we were authenticating the users by verifying credentials from the database. We were using the following query for authentication.

```
$res=mysqli_query($conn,"SELECT username, password, email FROM login WHERE username='$usrid'"); $row=mysqli_fetch_array($res);
$count = mysqli_num_rows($res);

if( $count == 1 && $row['password']==$pass && $row['is_staff']==0) {
$_SESSION['chat'] = $row['email']; header("Location: http://localhost:3000");

} else {
$errMSG1 = "Incorrect Credentials, Try again..."; }
```

Now, the problem with the query was that any hacker intelligently can supply an input which can bypass the user authentication and allows the user to proceed even without correct credentials.

For e.g.
The above code can be exploited by commenting out the password part and appending a condition that will always be true.

```
SELECT * FROM login WHERE username = 'xxx@xxx.xxx' OR 1 = 1 LIMIT 1 -- ' ] AND password = md5('1234');
```

HERE,
• xxx@xxx.xxx ends with a single quote which completes the string quote
• OR 1 = 1 LIMIT 1 is a condition that will always be true and limits the returned results to only one record.
• -- ' AND ... is a SQL comment that eliminates the password part.

Even when the username xxx@xxx.xxx is not in the database, it will show a result which will allow the hacker to bypass the login page even without incorrect credentials.

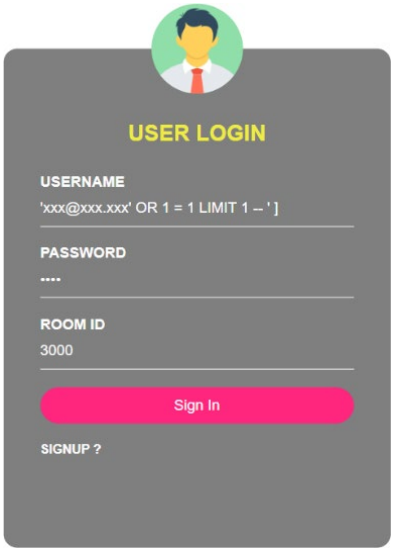


Fig. 4.1

username	password	email
akshaj	2584fa1ab8d6d6f9ab3fd481fd7f7a21b35040220dfffd74e6...	akshajgoyal@gmail.com
atharva	5bf9388be236593288e0c4310f1dfafa5d5096c82a954a5c7b...	atharva@gmail.com
somil	5f42800d94ad4fe3cb725209928b8872ce4508f11398b28192...	somil@gmail.com
test21	fab10377db597db1ab94a6935d5896f8498ca6ec365c663e17...	test21@gmail.com
test22	39602bbfdde1f76a2631dbbcb1ad02b6bf2fa700fb92a57e58...	test22@gmail.com
a	ef797c8118f02dfb649607dd5d3f8c7623048c9c063d532cc9...	a@g.com
sam	a40c43b41bffac09312e8614b6c5e8a5465ffb739f4cf1edf4...	sam@gmail.com
testuser	d79aeec77b3b6e6ec5f45ae5e040173380262e1ea51b387471...	testing@gmail.com
snehil	ab1f57a3ccaad5ef94e6b76376d4d62f001bd58281a173ed17...	snehil@gmail.com
sankalp	ca2ee954d72e56fbfec56eb4858e7216e5fa7ee3a66eeaf89b...	sankalp@gmail.com
vinit	b352826bc4755191125ba9f30bf678adc3bad4a2c427e41604...	vinit@gmail.com
testr3	cd0d3feb59b82b68f14920cc3fea43814a31c39da13eee19ad...	testr3@gmail.com

Fig. 4.2

2. Password Transmission and Storage Attacks:

Another potential threat to the application was in case of an attack on the server and user authentication database. In case of such attack the user passwords and information may be at a risk from the hackers. Not only from the database the passwords are at risk during transmission and authentication using PhP.

3. Digital Signatures:

The previous version did not have the provision of a digital signature therefore it was vulnerable to a man in the middle attack. Now this problem was tackled by using digital signatures for both the users to verify each other and authenticate each other thereby making it impossible for a middleman to intercept and change anything in the message or change the message there by ensuring integrity.

4.1.2 Preventative Measures

After exploiting the possible vulnerabilities present in the application, security and other preventive measures were devised to cop-up with these vulnerabilities.

1. Prevent SQLi Attack:

With user input channels being the main vector for SQL injection attacks, most of the defensive methods involve controlling and vetting user input for attack patterns. Validation is the process of making sure the right type of input is provided by users and to neutralize any potential malicious commands that might be embedded in input string. For instance, in PHP, you can use the mysql_real_escape_string() to escape characters that might change the nature of the SQL command.

```
$usrid = trim($_POST['usrid']); $usrid = strip_tags($usrid);
$usrid = htmlspecialchars($usrid);

$pass = trim($_POST['pass']);
$pass = strip_tags($pass);
$pass = htmlspecialchars($pass); if (!$error) { $res=mysqli_query($conn,"SELECT username,
password, email FROM login WHERE username='$usrid'"); $row=mysqli_fetch_array($res);
$count = mysqli_num_rows($res);

if( $count == 1 && $row['password']==$) { $_SESSION['chat'] = $row['email']; header("Location:
http://localhost:3000");
} else {
$errMSG1 = "Incorrect Credentials, Try again";
}}
```

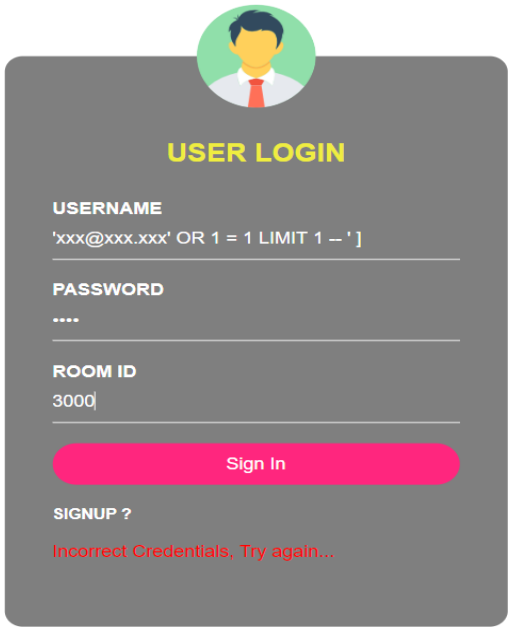


Fig. 4.3

Modified SQL Commands:

```
$usrid = mysqli_real_escape_string($con,
 $_POST[‘usrid’]); $pass = mysqli_real_escape_string($con, $_POST[‘pass’]);
 $sql_command = “select * from login where username= ‘”.$useid;
$sql_command .=”’AND password = ‘” . $password . “’”;
```

2. SHA-256 hash to store and authenticate password:

For storage during sign-up:

```
$pass = trim($_POST['pass']);
$pass = strip_tags($pass);
$pass = htmlspecialchars($pass);
$password = hash('sha256', $pass);
$query2 = "INSERT INTO login(username,password,email) VALUES('$usr', '$password', '$email')";
```

For authentication during login:

```
$pass = trim($_POST['pass']); $pass = strip_tags($pass);
$pass = htmlspecialchars($pass);

if(empty($pass)){
$error = true;
$errMSG1 = "Please enter your password.";
}
else{
$pass = hash('sha256', $pass);
}
```

The hashing ensures that the passwords are stored as hash values and cannot be exploited by any suspicious user.

4.1.3 Performance Table:

Table 4.1 – Performance Table

S. No	Algorithm	Packet Size (KB)	Encrypt Time (Sec)	Decrypt time (Sec)	Buffer Size
1.	DES	153	3.0	1	157
	AES		1.6	1.1	152
	RSA		7.3	4.9	222
2.	DES	118	3.2	1.2	121
	AES		1.7	1.2	110
	RSA		10.0	5.0	188
3.	DES	196	2.0	1.4	201
	AES		1.7	1.24	200
	RSA		8.5	5.9	257

4.1.4 Graphical Representation of Performance Table:

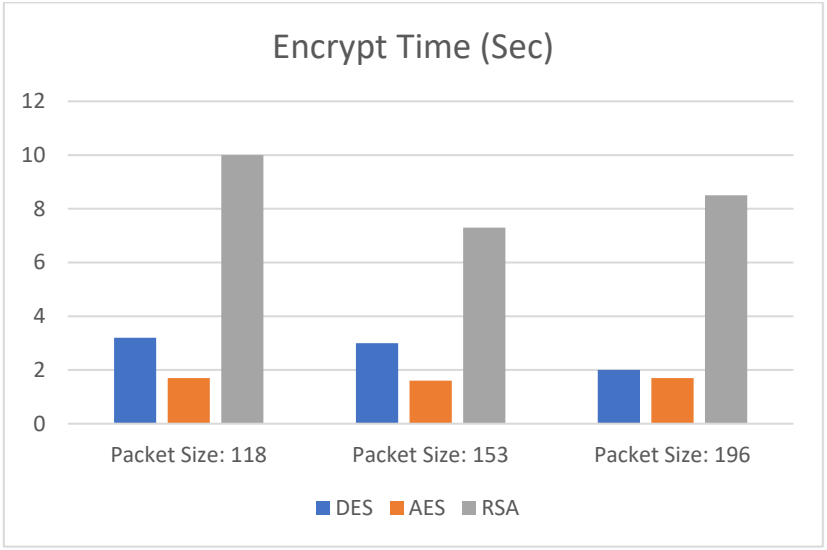


Fig. 4.4

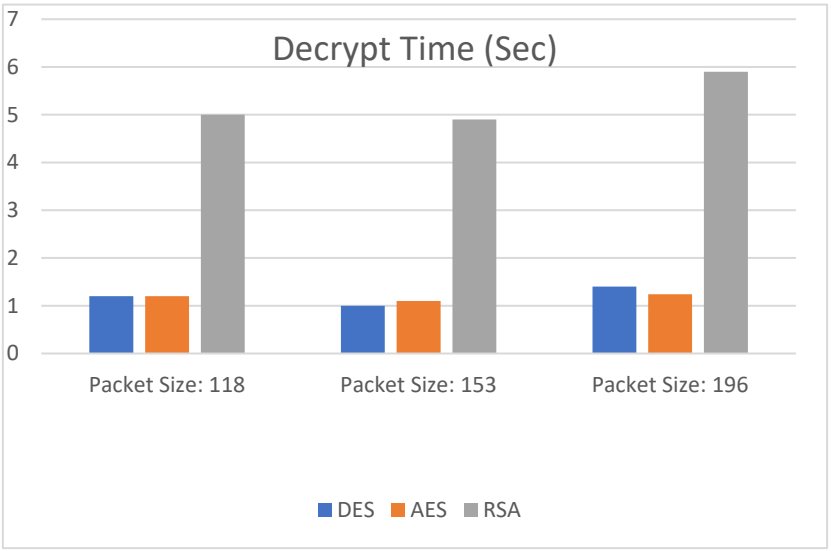


Fig. 4.5

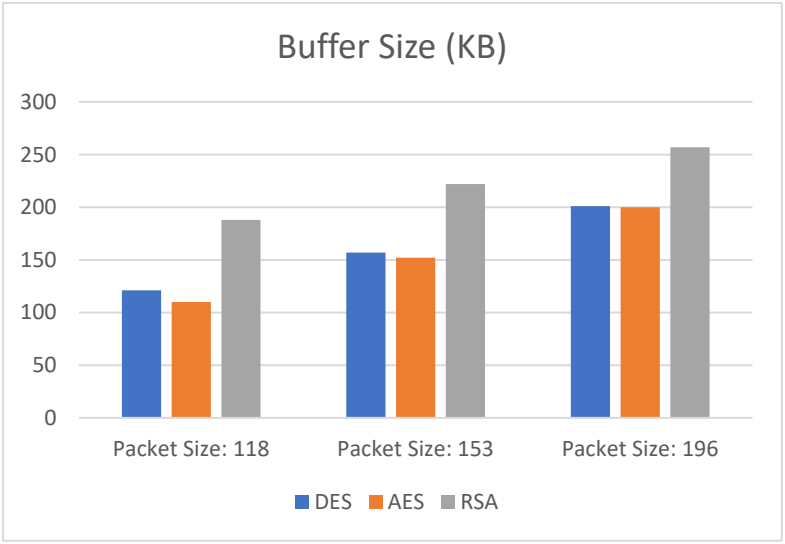


Fig. 4.6

5.1 Conclusion and Future Work

The developed application is very much secure in terms of integrity, confidentiality, authentication and non-repudiation. Security measures have been taken to prevent any type of database exploitation, SQL injection and cross-site scripting. The application is in its initial stages. There is wide scope of improvement in the application in terms of user interface, features and security point of view. Still some vulnerabilities may exist which may be hard to find after first analysis of the application. But with trials and errors and with repeating security analysis the vulnerabilities can be found and with proper knowledge can be rectified with appropriate security measures.

REFERENCES

[1] Kumari, Sarita. "A research paper on cryptography encryption and compression techniques." *International Journal Of Engineering And Computer Science* 6.4 (2017): 20915-20919.

[2] Abdullah, AkoMuhamad. "Advanced encryption standard (AES) algorithm to encrypt and decrypt data." *Cryptography and Network Security* 16 (2017): 1-11.

[3] Al-Ibrahim, Mohamed, and Yousef Shams Al-Deen. "The reality of applying security in Web applications in Academia." *International Journal of Advanced Computer Science and Applications* (2014).

[4] Boyd, Stephen W., and Angelos D. Keromytis. "SQLrand: Preventing SQL injection attacks." *International conference on applied cryptography and network security*. Springer, Berlin, Heidelberg, 2004.

[5] Cao, Min, Tianyang Xing, and Chun Wang. "Implementation of web security & identity scheme based on session & online table." *2009 4th International Conference on Computer Science & Education*. IEEE, 2009.

[6] Chanda, Katha. "Password security: an analysis of password strengths and vulnerabilities." *International Journal of Computer Network and Information Security* 8.7 (2016): 23.

[7] Chong, Stephen, et al. "Secure web applications via automatic partitioning." *ACM SIGOPS Operating Systems Review* 41.6 (2007): 31-44.

[8] Grover, Varsha. "An Efficient Brute Force Attack Handling Techniques for Server Virtualization." *Proceedings of the International Conference on Innovative Computing & Communications (ICICC)*. 2020.

[9] Haixia, Yang, and Nan Zhihong. "A database security testing scheme of web application." *2009 4th International Conference on Computer Science & Education*. IEEE, 2009.

[10] Han, Aaron L-F., Derek F. Wong, and Lidia S. Chao. "Password cracking and countermeasures in computer security: A survey." *arXiv preprint arXiv:1411.7803* (2014).

[11] Liu, Muyang, Ke Li, and Tao Chen. "Security testing of web applications: a search-based approach for detecting SQL injection vulnerabilities." *Proceedings of the Genetic and Evolutionary Computation Conference Companion*. 2019.

[12] Liu, Qigang, and Xiangyang Sun. "Research of web real-time communication based on web socket." (2012).

[13] Mendel, Florian, et al. "Analysis of step-reduced SHA-256." *International workshop on fast software encryption*. Springer, Berlin, Heidelberg, 2006.

[14] Skala, Vaclav, and Radek Petruska. "A new approach to hash function construction for textual data: a comparison." *2014 4th World Congress on Information and Communication Technologies (WICT 2014)*. IEEE, 2014.

[15] Sowmya, G., D. Jamuna, and M. Venkata Krishna Reddy. "Blocking of brute force attack." *International Journal of Engineering Research and Technology* 1.6 (2012).

[16] Tuomi, Jan-Sebastian. "Automated Initialization of Web Software Projects." (2018).

[17] Dutta AK (2021) Detecting phishing websites using machine learning technique. PLoS ONE 16(10): e0258361. https://doi.org/10.1371/journal.pone.0258361.

[18] J. James, Sandhya L. and C. Thomas, "Detection of phishing URLs using machine learning techniques," 2013 International Conference on Control Communication and Computing (ICCC), 2013, pp. 304-309, doi: 10.1109/ICCC.2013.6731669.

[19] J. Rashid, T. Mahmood, M. W. Nisar and T. Nazir, "Phishing Detection Using Machine Learning Technique," 2020 First International Conference of Smart Systems and Emerging Technologies (SMARTTECH), 2020, pp. 43-46, doi: 10.1109/SMART-TECH49988.2020.00026.

[20] Nagunwa, Thomas and Naqvi, S. and Fouad, Shereen and Shah, H. (2019) A Framework of New Hybrid Features for Intelligent Detection of Zero Hour Phishing Websites. In: International Conference on Computational Intelligence in Security for Information Systems (CISIS 2019), 13-15 May 2019, Seville, Spain.

APPENDIX (CODE)

The code is attached below:

https://drive.google.com/file/d/18YUOoD2DfToE2qKwBXiUeBAk8nc44bR5/view?usp=sharing