REVIEW 3

Project report

TITLE: HYBRID CRYPTOGRAPHY

NAME:LOGA RAKSHIKA .B

REG NO:18BIT0430

SLOT: G2

Team members:

LYDIA CHRISLIN PAUL 18BIT0439

SARADHA DEVI 18BIT0447

For review 2 , I implemented using aes-256 and sha256 algorithm for password hashing .

For review 3 , I implemented des and sha256 for password hashing.

APPLICATION DEVELOPED:
The application we developed is user registration login system.

This consist of registration page, login page and dashborad page.

Registration page running on localhost/project/reg2.php

Login page running on localhost/project/login.php

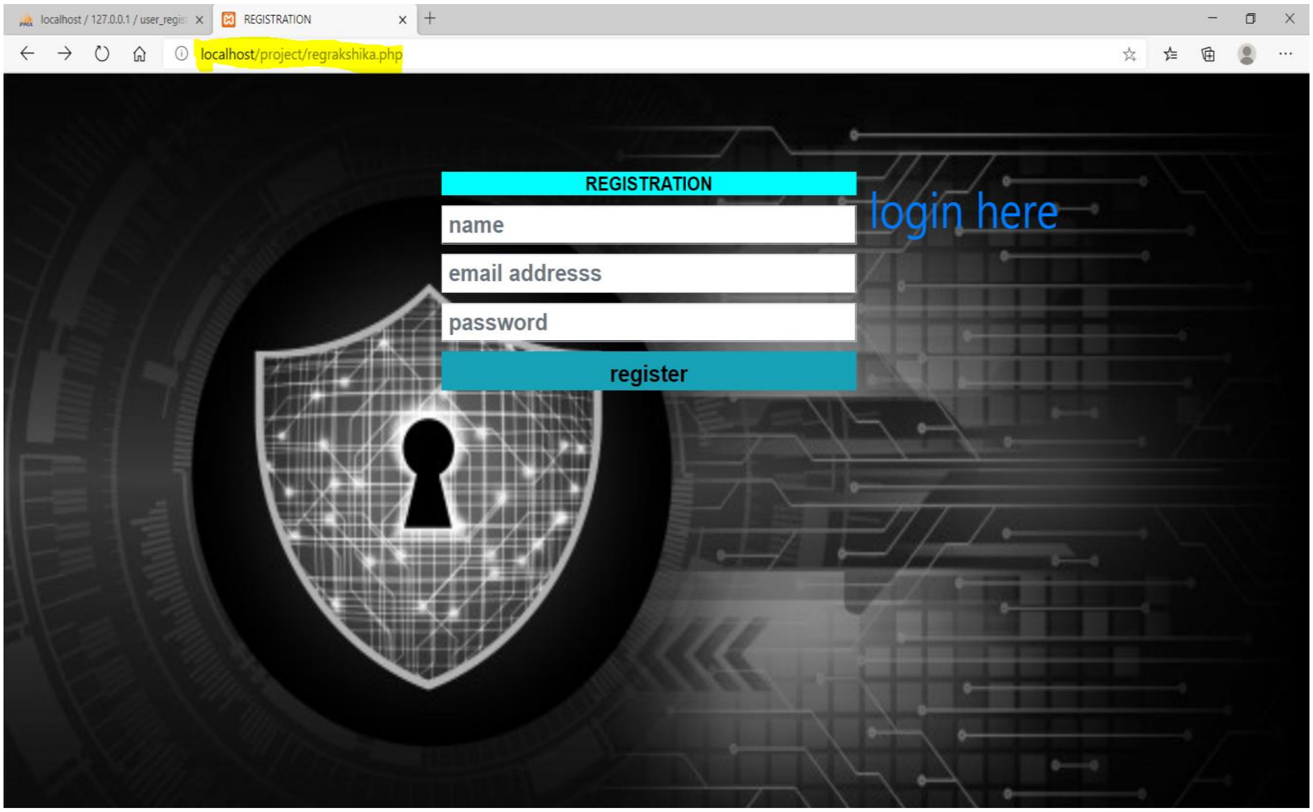
Dashboard page running on localhost/project/dashboard.html

Database name: user_register

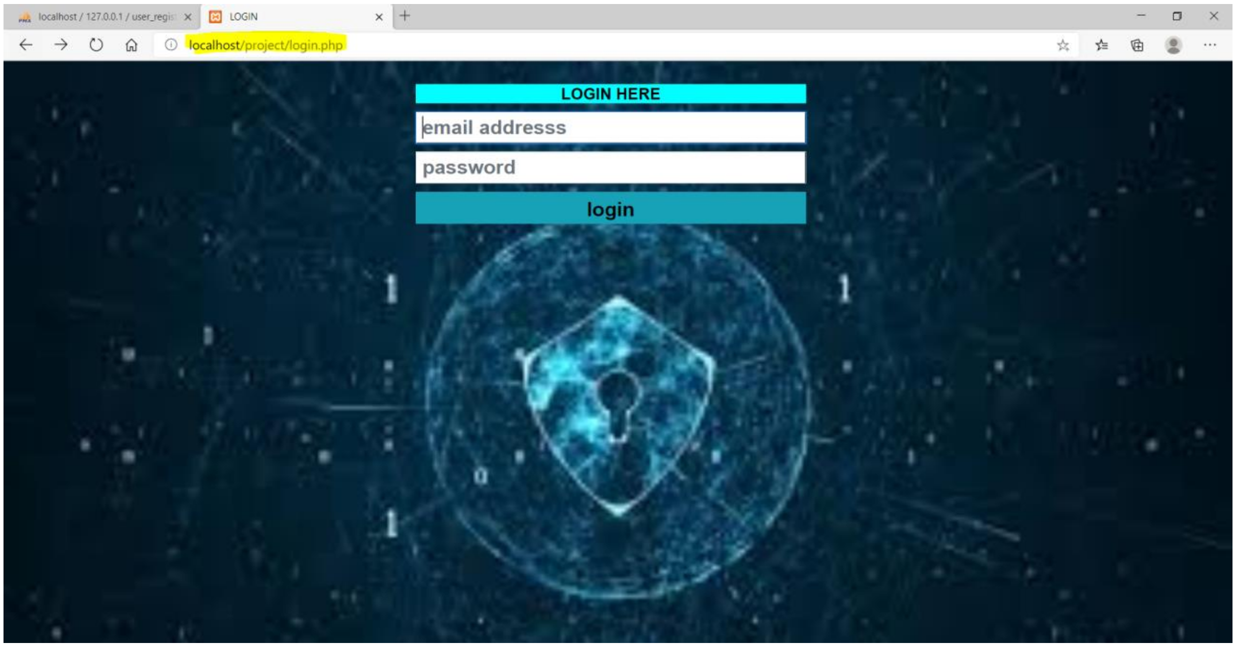
Table name:register

Table fields: id, name, email, password

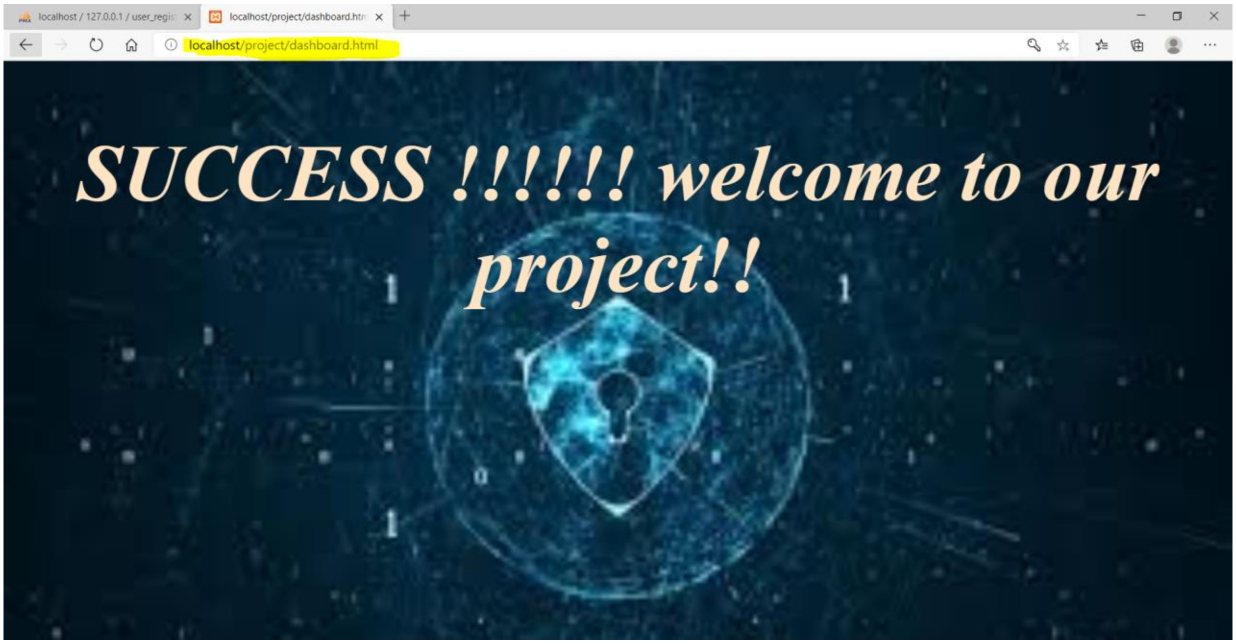
REGISTRATION PAGE:



LOGIN PAGE:


localhost / 127.0.0.1 / user_regis
LOGIN
localhost/project/login.php
LOGIN HERE
email addresss
password
login

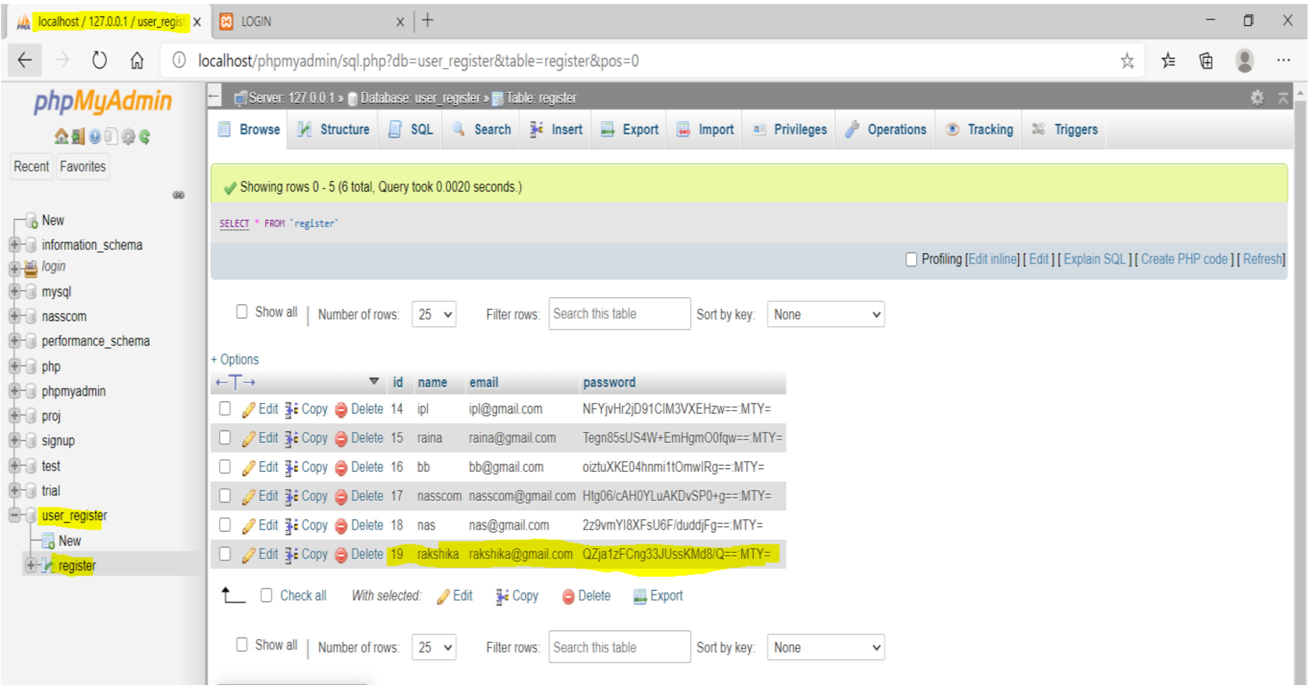
DASHBOARD PAGE:



localhost / 127.0.0.1 / user_regis
localhost/project/dashboard.htm
localhost/project/dashboard.html

SUCCESS !!!!!! welcome to our project!!

DATABASE WITH ENCRYPTED PASSWORD:



✓ Showing rows 0 - 5 (6 total, Query took 0.0020 seconds.)

```
SELECT * FROM `register`
```

□ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

□ Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None

+ Options

	id	name	email	password
□ Edit Copy Delete	14	ipl	ipl@gmail.com	NFYjvHr2jD91CIM3VXEHzw==:MTY=
□ Edit Copy Delete	15	raina	raina@gmail.com	Tegn85sUS4W+EmHgmO0fqw==:MTY=
□ Edit Copy Delete	16	bb	bb@gmail.com	oiztuXKE04hnmi1tOmwIRg==:MTY=
□ Edit Copy Delete	17	nasscom	nasscom@gmail.com	Htg06/cAH0YLuAKDvSP0+g==:MTY=
□ Edit Copy Delete	18	nas	nas@gmail.com	2z9vmYI8XFsU6F/duddjFg==:MTY=
□ Edit Copy Delete	19	rakshika	rakshika@gmail.com	QZja1zFCng33JUssKMd8/Q==:MTY=

□ Check all With selected: Edit Copy Delete Export

□ Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None

HASHING ALGORITHM USED: SHA-256

SHA-256 is one of the successor hash functions to SHA-1 (collectively referred to as SHA-2), and is one of the strongest hash functions available. SHA-256 is not much more complex to code than SHA-1, and has not yet been compromised in any way.

Syntax for sha256 in php:

hash (string $algo , string $data [, bool $raw_output = **FALSE**]) : string

in $algo , we put sha256 or md5 or sha1 , as per our choice.

$data is the aes generated key.

$raw_output when true gives binary result.

Example: $hashedkey=hash(“sha256”,$aeskey);

ENCRYPTION ALGORITHM USED: DES

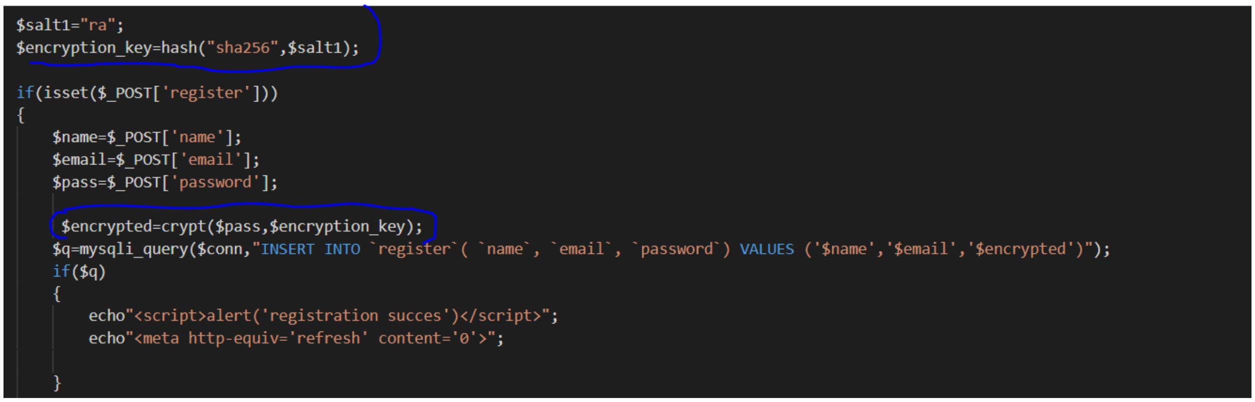
Data Encryption Standard (DES) is a block cipher algorithm that takes plain text in blocks of 64 bits and converts them to ciphertext using keys of 48 bits. It is a symmetric key algorithm, which means that the same key is used for encrypting and decrypting data.

CODE EXPLANATION:

The below figure shows the encryption done on the registration page.

In php , to implement DES algorithm , the crypt() function is used . for DES , crypt function used 2 length string as a salt parameter and in a way it’s a key for DES.

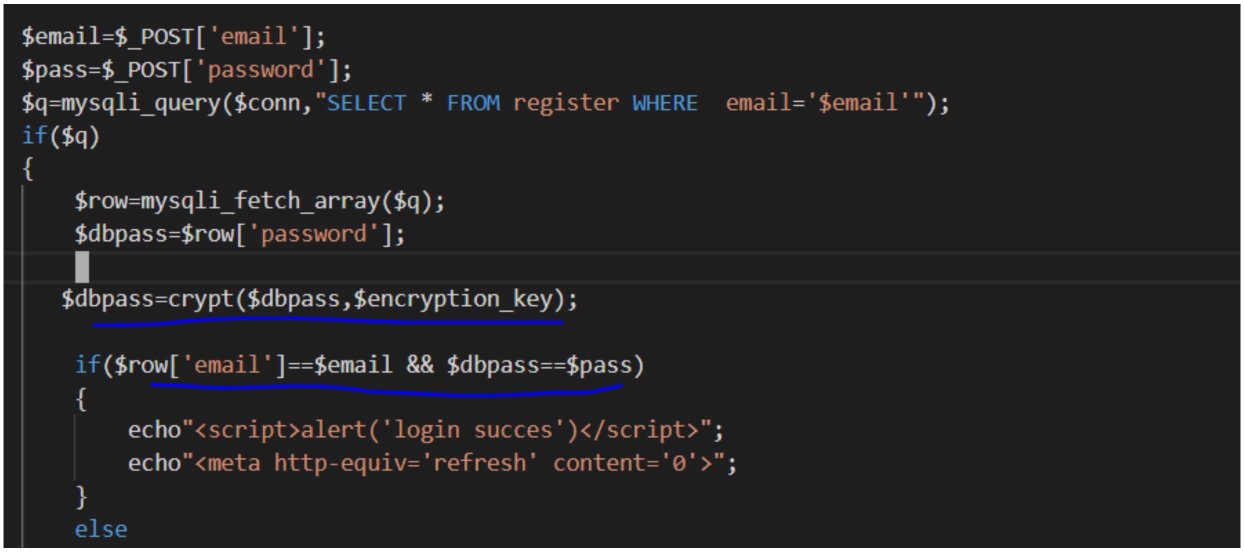
I encrypted the salt with SHA-256 algorithm. And that encrypted salt is being used for the encryption of the password.



```
$salt1="ra";
$encryption_key=hash("sha256",$salt1);

if(isset($_POST['register']))
{
    $name=$_POST['name'];
    $email=$_POST['email'];
    $pass=$_POST['password'];

     $encrypted=crypt($pass,$encryption_key);
    $q=mysqli_query($conn,"INSERT INTO `register`( `name`, `email`, `password`) VALUES ('$name','$email','$encrypted')");
    if($q)
    {
        echo"<script>alert('registration succes')</script>";
        echo"<meta http-equiv='refresh' content='0'>";

    }
```

The below picture shows the verification done on the login page.

First we get the password from the user . we encrypt the password with the salted key , then we compare the encrypted password with the database stored password . if it matches then login is success.



```
$email=$_POST['email'];
$pass=$_POST['password'];
$q=mysqli_query($conn,"SELECT * FROM register WHERE  email='$email'");
if($q)
{
    $row=mysqli_fetch_array($q);
    $dbpass=$row['password'];

   $dbpass=crypt($dbpass,$encryption_key);

    if($row['email']==$email && $dbpass==$pass)
    {
        echo"<script>alert('login succes')</script>";
        echo"<meta http-equiv='refresh' content='0'>";
    }
    else
```

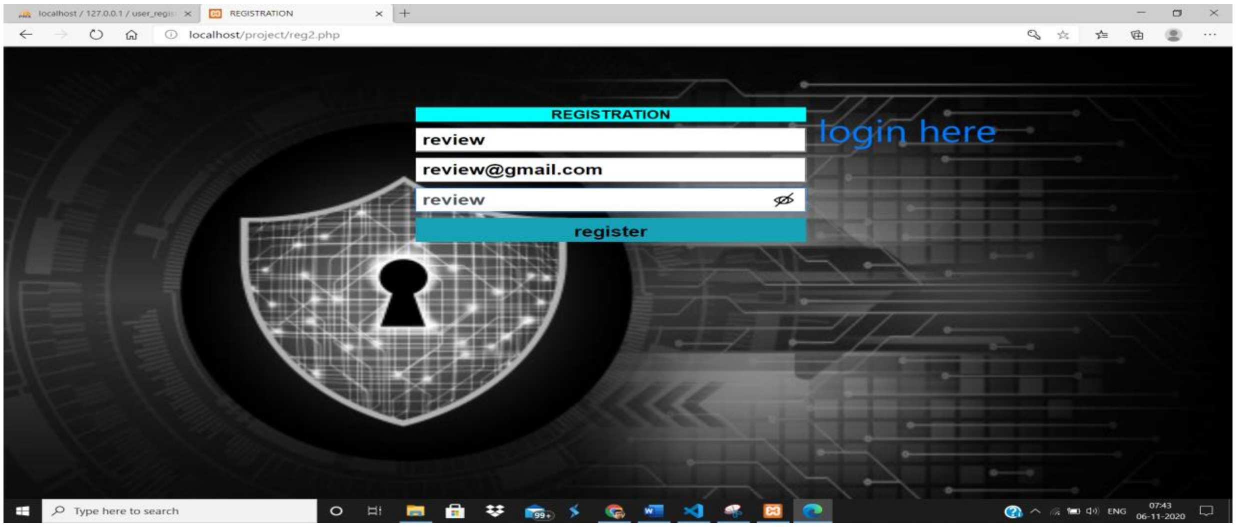
SAMPLE OUTPUT:

Username: review

Email:review@gmail.com

Password: review

REGISTRATION PAGE:



localhost/project/reg2.php

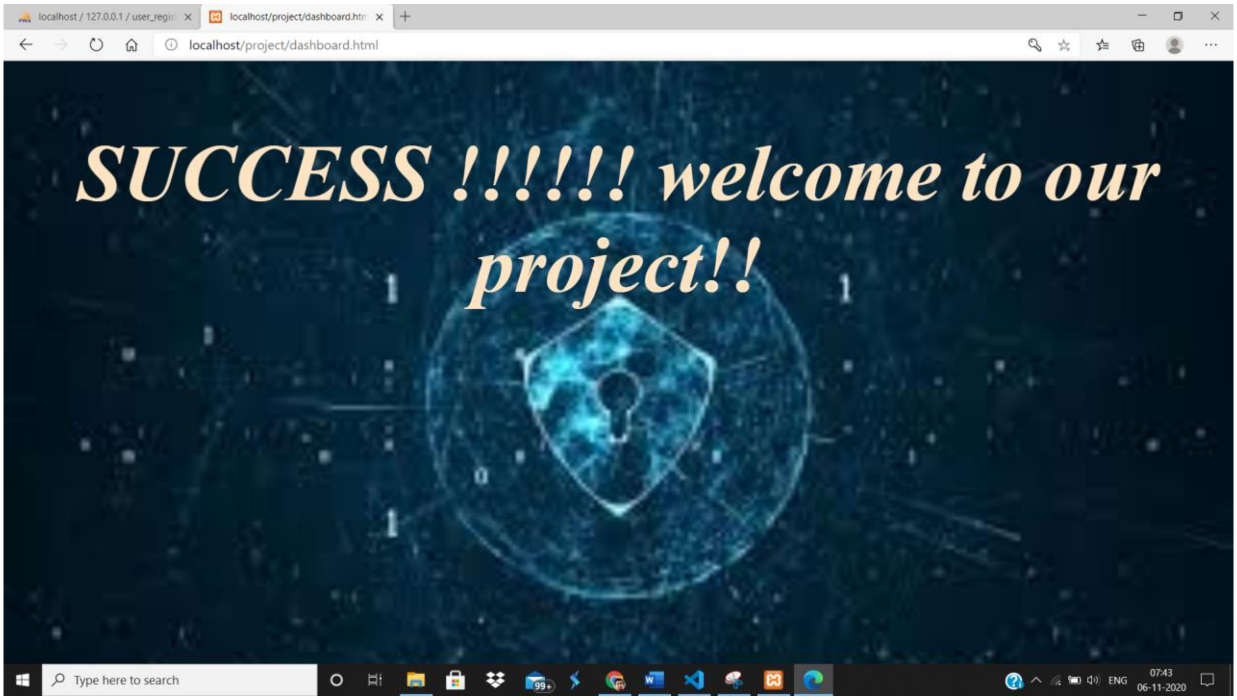
REGISTRATION

review

review@gmail.com

review

SUCCESS PAGE

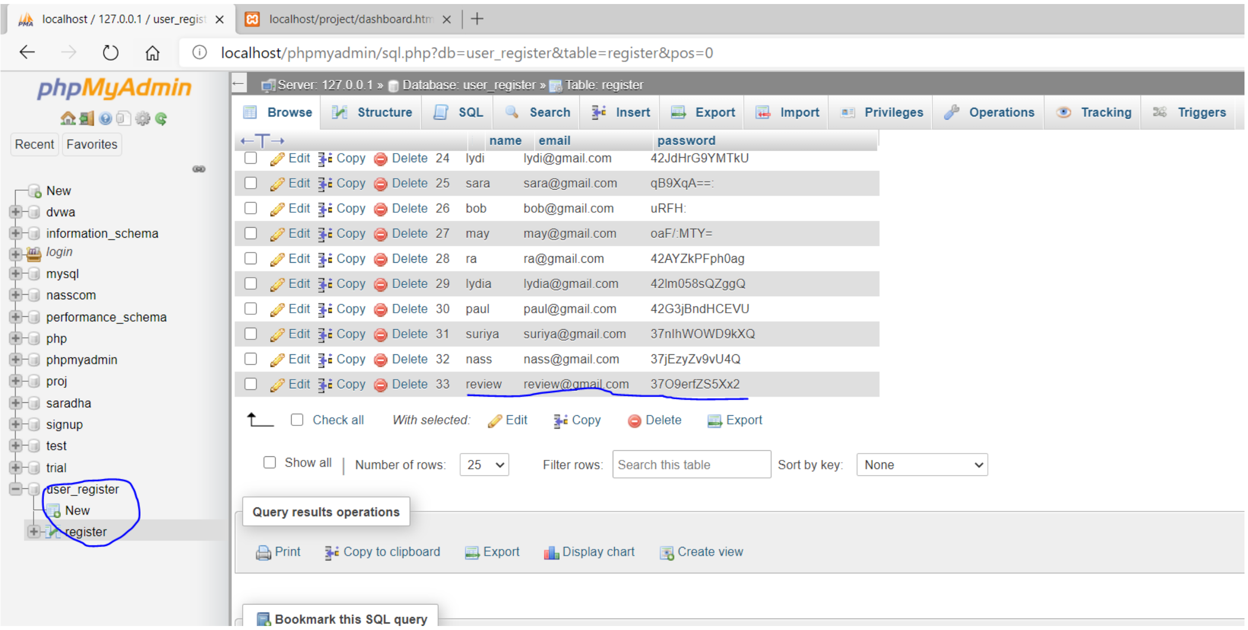


localhost/project/dashboard.html

SUCCESS !!!!!! welcome to our project!!

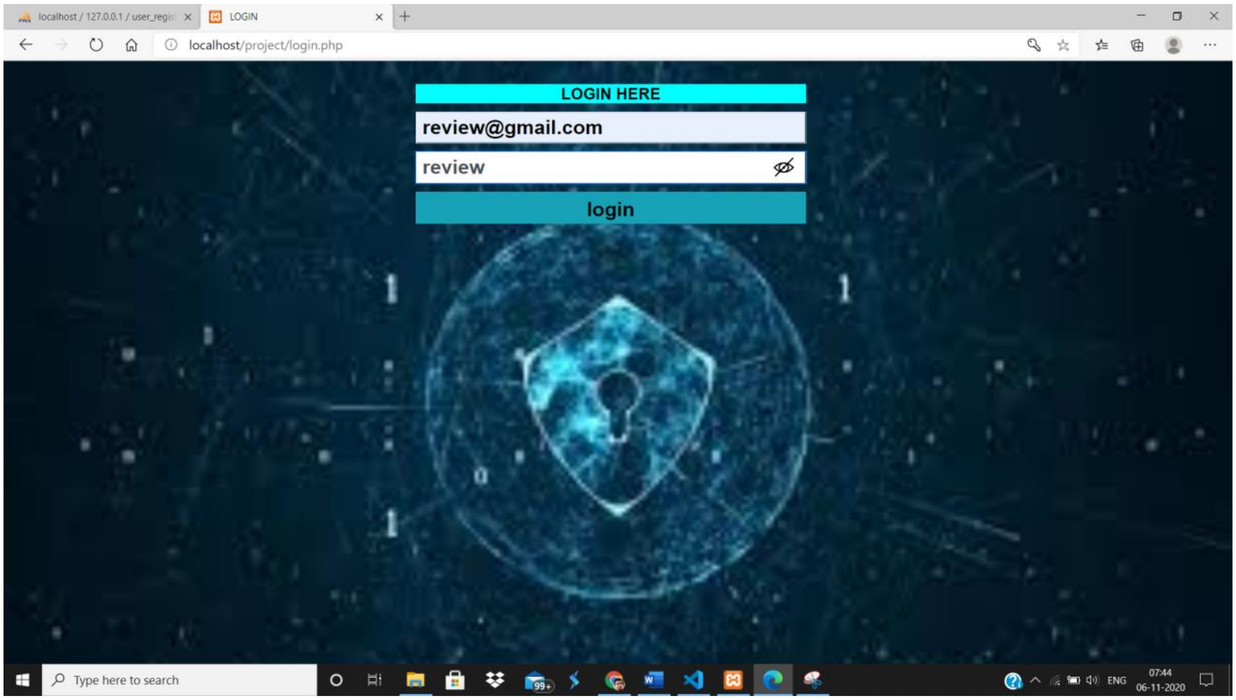
localhost / 127.0.0.1 / user_regi
localhost/project/dashboard.htm
Type here to search
ENG
07:43
06-11-2020

PASSWORD STORED IN HASHED FORMAT



				id	name	email	password
□	Edit	Copy	Delete	24	lydi	lydi@gmail.com	42JdHrG9YMTkU
□	Edit	Copy	Delete	25	sara	sara@gmail.com	qB9XqA==:
□	Edit	Copy	Delete	26	bob	bob@gmail.com	uRFH:
□	Edit	Copy	Delete	27	may	may@gmail.com	oaF/:MTY=
□	Edit	Copy	Delete	28	ra	ra@gmail.com	42AYZkPFph0ag
□	Edit	Copy	Delete	29	lydia	lydia@gmail.com	42lm058sQZggQ
□	Edit	Copy	Delete	30	paul	paul@gmail.com	42G3jBndHCEVU
□	Edit	Copy	Delete	31	suriya	suriya@gmail.com	37nlhWOWD9kXQ
□	Edit	Copy	Delete	32	nass	nass@gmail.com	37jEzyZv9vU4Q
□	Edit	Copy	Delete	33	review	review@gmail.com	37O9erfZS5Xx2

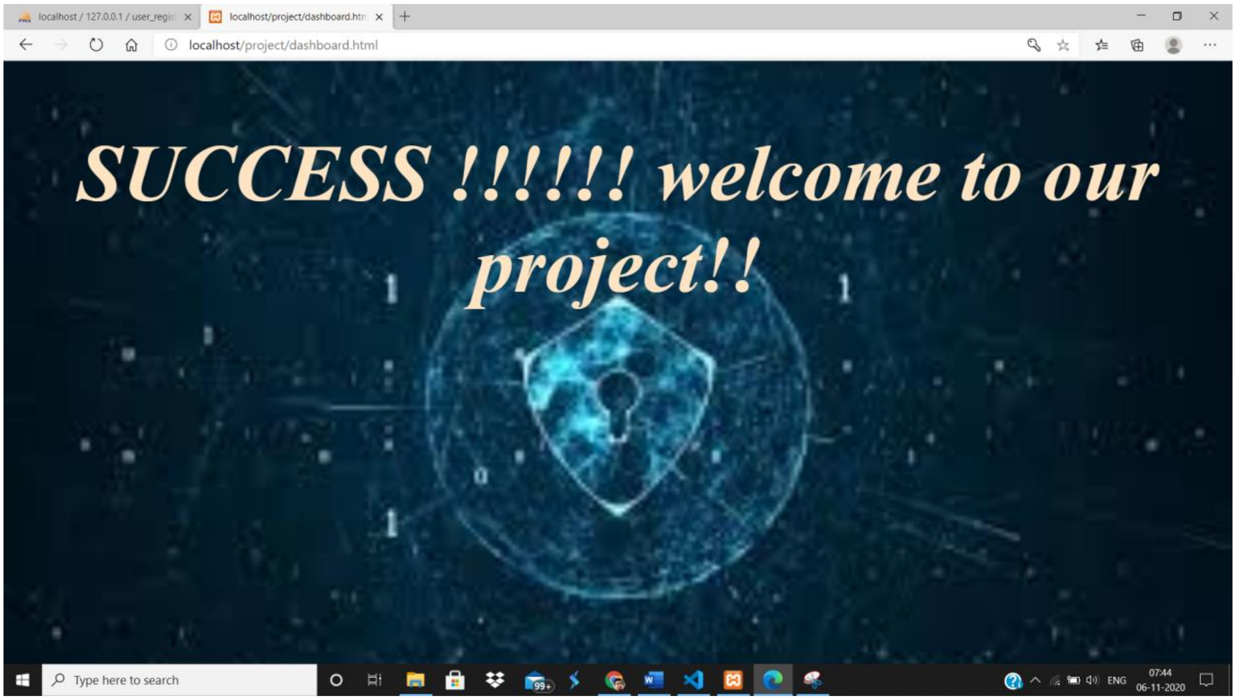
LOGIN PAGE


localhost / 127.0.0.1 / user_regis
LOGIN
localhost/project/login.php
LOGIN HERE
login
Type here to search
ENG
07:44
06-11-2020

review@gmail.com

review

LOGIN SUCCESS


localhost / 127.0.0.1 / user_regis
localhost/project/dashboard.htm
localhost/project/dashboard.html
Type here to search
ENG
07:44
06-11-2020

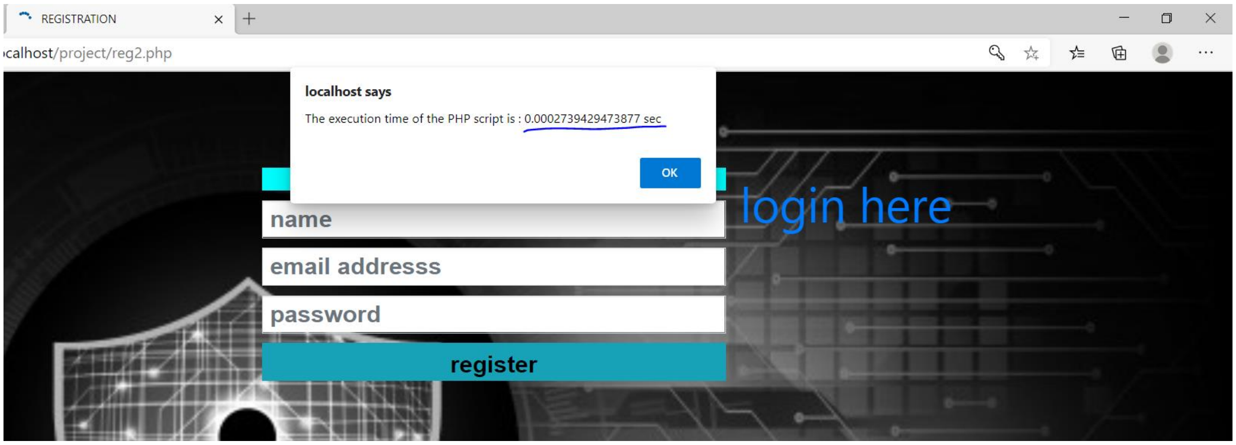
SUCCESS !!!!!! welcome to our project!!

COMPARISON ANALYSIS:

As a 1st step of comparision analysis , I did it based on execution time algorithm to find out which algorithm is faster .for that purpose I calculated execution time for few sample test cases shown below.

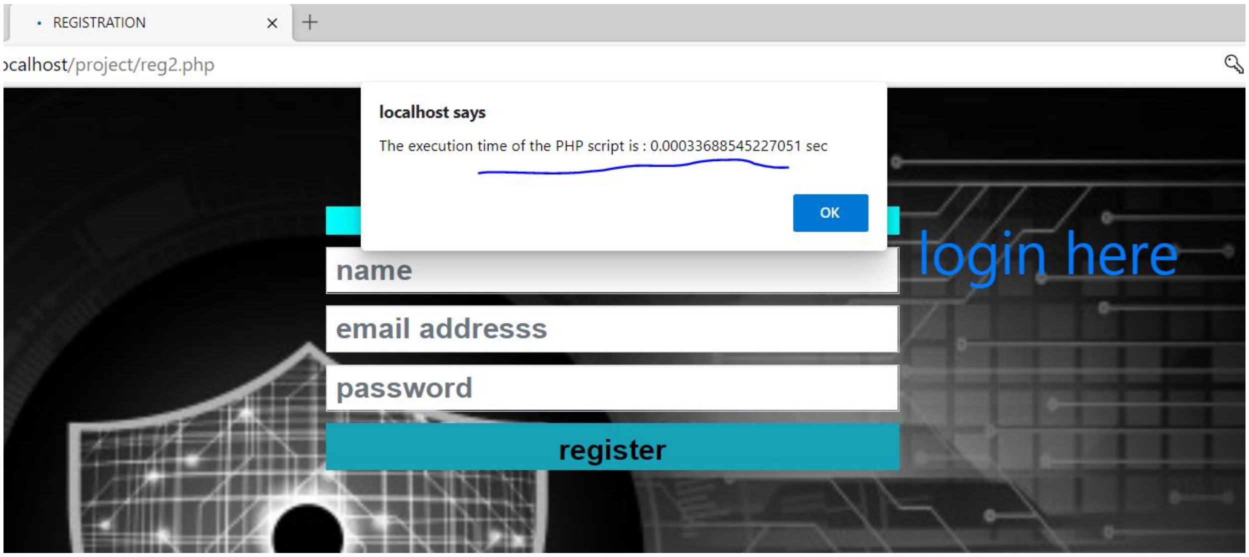
EXECUTION TIME OF SCRIPTS:

For AES and sha256



localhost says

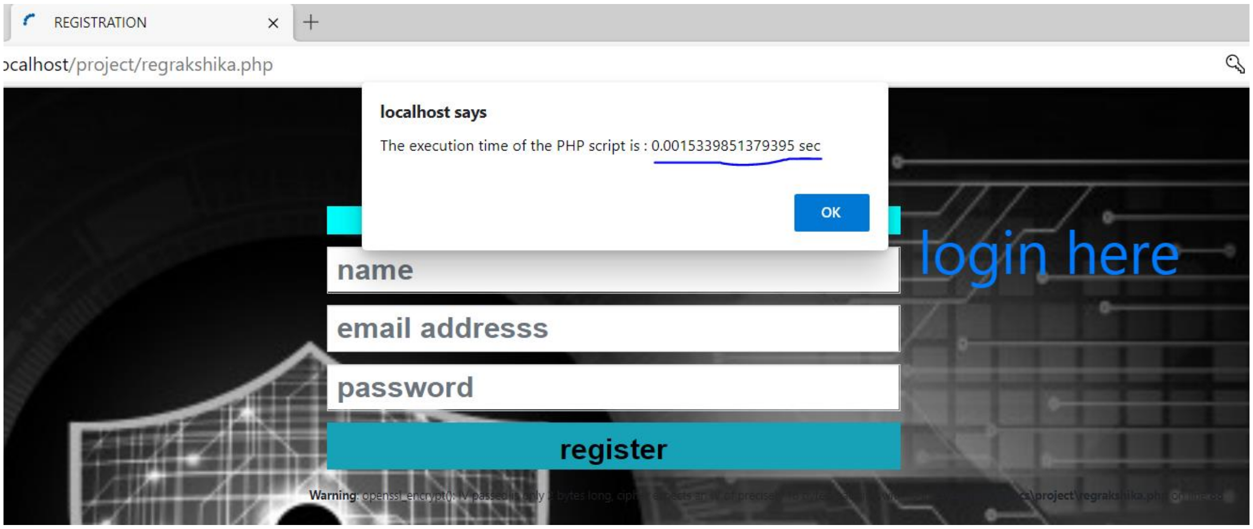
The execution time of the PHP script is : 0.0002739429473877 sec



localhost says

The execution time of the PHP script is : 0.00033688545227051 sec

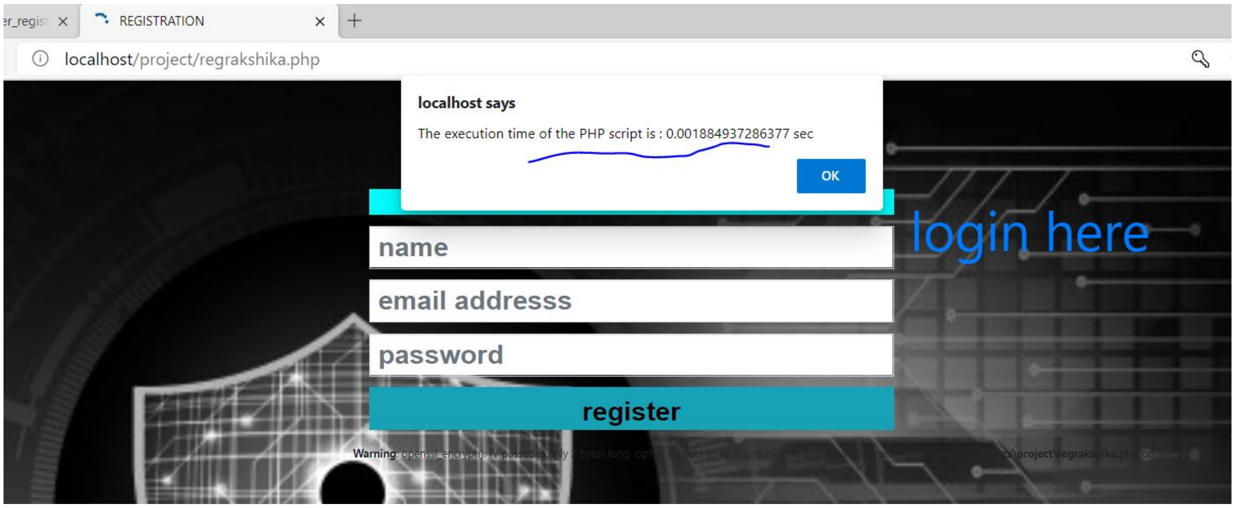
For des and sha256



localhost says

The execution time of the PHP script is : 0.0015339851379395 sec

Warning: openssl_encrypt(): IV passed is only 2 bytes long, cipher expects an IV of precisely [illegible] **cs\project\regrakshika.php** on line [illegible]



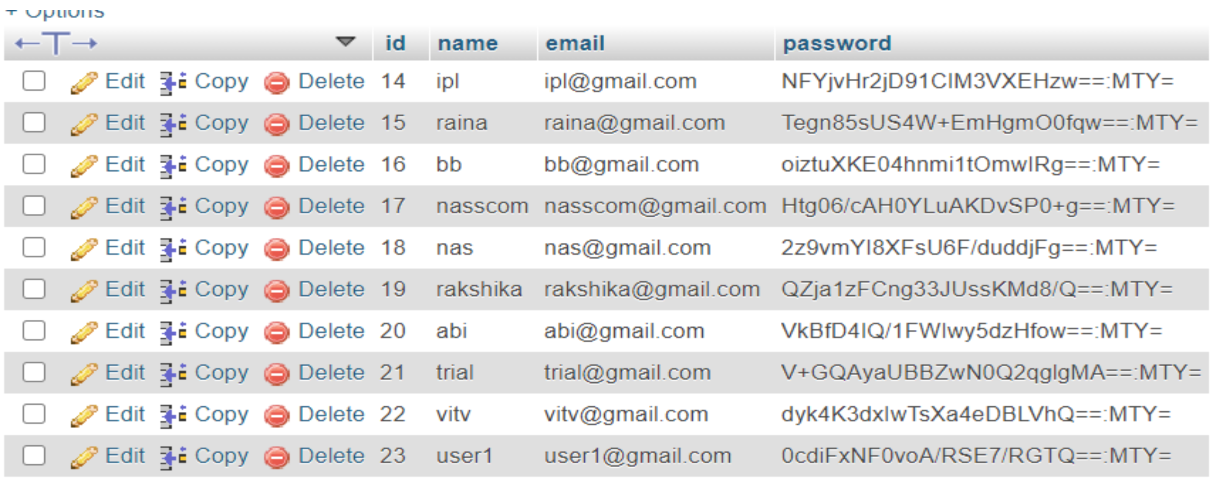
localhost says

The execution time of the PHP script is : 0.001884937286377 sec

Warning: openssl_encrypt(): IV passed is only 2 bytes long, cipher expects an IV of precisely [illegible] **cs\project\regrakshika.php** on line [illegible]

The next comparison is based on hashed format and its complexiety.

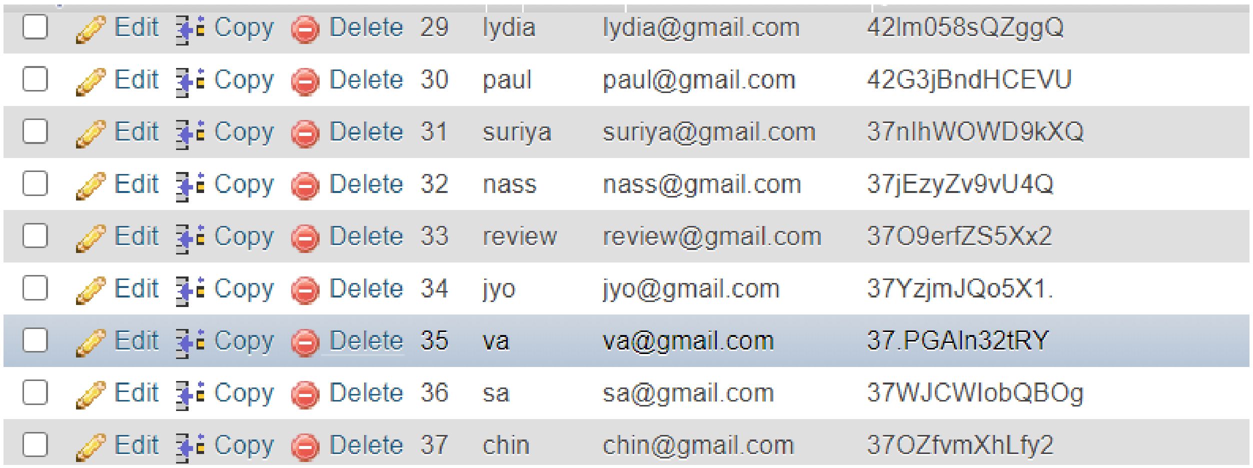
This pic below is for aes and sha256



+ Options

←T→				id	name	email	password
☐	Edit	Copy	Delete	14	ipl	ipl@gmail.com	NFYjvHr2jD91CIM3VXEHzw==:MTY=
☐	Edit	Copy	Delete	15	raina	raina@gmail.com	Tegn85sUS4W+EmHgmO0fqw==:MTY=
☐	Edit	Copy	Delete	16	bb	bb@gmail.com	oiztuXKE04hnmi1tOmwIRg==:MTY=
☐	Edit	Copy	Delete	17	nasscom	nasscom@gmail.com	Htg06/cAH0YLuAKDvSP0+g==:MTY=
☐	Edit	Copy	Delete	18	nas	nas@gmail.com	2z9vmYI8XFsU6F/duddjFg==:MTY=
☐	Edit	Copy	Delete	19	rakshika	rakshika@gmail.com	QZja1zFCng33JUssKMd8/Q==:MTY=
☐	Edit	Copy	Delete	20	abi	abi@gmail.com	VkBfD4IQ/1FWIwy5dzHfow==:MTY=
☐	Edit	Copy	Delete	21	trial	trial@gmail.com	V+GQAyaUBBZwN0Q2qglgMA==:MTY=
☐	Edit	Copy	Delete	22	vitv	vitv@gmail.com	dyk4K3dxIwTsXa4eDBLVhQ==:MTY=
☐	Edit	Copy	Delete	23	user1	user1@gmail.com	0cdiFxNF0voA/RSE7/RGTQ==:MTY=

This pic below is for des and sha256



☐	Edit	Copy	Delete	29	lydia	lydia@gmail.com	42Im058sQZggQ
☐	Edit	Copy	Delete	30	paul	paul@gmail.com	42G3jBndHCEVU
☐	Edit	Copy	Delete	31	suriya	suriya@gmail.com	37nIhWOWD9kXQ
☐	Edit	Copy	Delete	32	nass	nass@gmail.com	37jEzyZv9vU4Q
☐	Edit	Copy	Delete	33	review	review@gmail.com	37O9erfZS5Xx2
☐	Edit	Copy	Delete	34	jyo	jyo@gmail.com	37YzjmJQo5X1.
☐	Edit	Copy	Delete	35	va	va@gmail.com	37.PGAIn32tRY
☐	Edit	Copy	Delete	36	sa	sa@gmail.com	37WJCWIobQBOg
☐	Edit	Copy	Delete	37	chin	chin@gmail.com	37OZfvmXhLfy2

ANALYSIS TABLE: (for my algorithms)

Categories	**AES-SHA256**	**DES-SHA256**
Run time	0.0003 sec	0.001 and above
Key length	256 bits	56 bits
Rounds	12	16
Complexiety	Hard to crack	Not harder than AES
Secure	Most secure	Less secure
Brute force attack	Not possibility	Possible

possibility		
Man in the middle attack	Not possible	vulnerable
Cryptanalysis Resistance	Strong against differential, truncated differential, linear, interpolation and square attacks	Vulnerable to differential and linear cryptanalysis; weak substitution tables
Hash size	256 bits	256 bits
Encryption time	0.00012 sec	0.001 sec
Decryption time	0.0001 sec	0.004 sec

COMPARISION ANALYSIS ALONG WITH MY TEAM MATES ALGORITHM:

Categories	**AES-SHA256**	**DES-SHA256**	**AES-MD5**	**BLOWFISH-SHA256**
Run time	Fastest	Slow	Faster	faster
Key length	256 bits	56 bits	128 bits	32 bits
Rounds	12	16	10	16
Complexiety	Hard to crack	Not harder than AES	Hard to crack	Harder because of sha256
Secure	Most secure	Less secure	Secure	Secure
Brute force attack possibility	Not possibility	Possible	No possible	Not possible
Man in the middle attack	Not possible	vulnerable	Not possible	Not possible
Hash size	256 bits	256 bits	128 bits	256 bits
Encryption time	0.00012 sec	0.001 sec	0.002	0.002
Decryption time	0.0001 sec	0.004 sec	0.003	0.001

The above mentioned encryption and decryption time was calculated by giving sample test cases and we got the script execution time .with the script execution time , we wrote encryption and decryption time.

COMPARISION WITH SURVEY PAPERS:

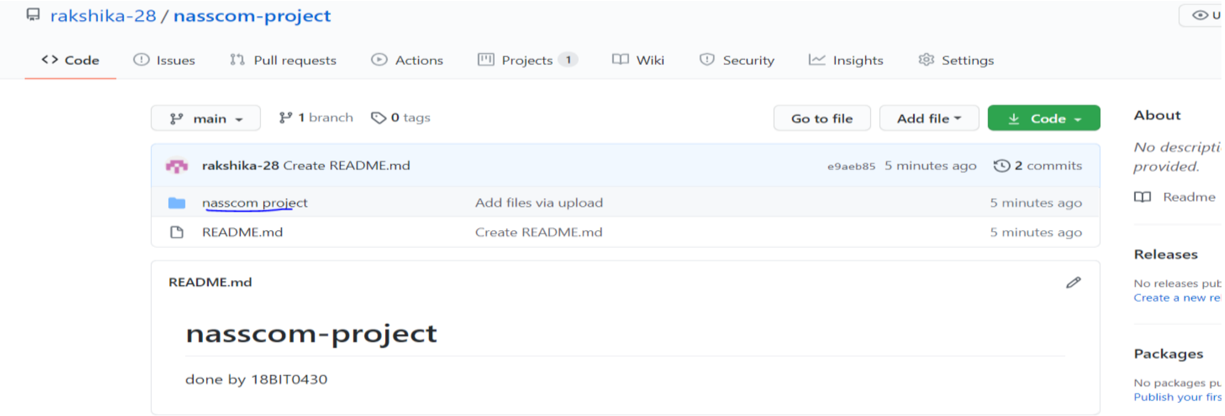
In most of the survey papers , they used SHA-512 algorithm .though it was introduced latest , it has many vulnerabilities . still many companies use SHA-256 because of its stability and security .in that way the use of sha 256 was a better choice.

The second factor is that , most of the papers used asymmetric and hashing . in our project we have implemented with symmetric and hashing .

The fact that symmetric is best because , our aim is to secure database . for data in rest , symmetric key is good.when it comes to data transfer or data in motion alone , asymmetric key is preferred. Hence , the implementation of symmetric key and hashing was a good idea to some extent.

GITHUB LINK:

https://github.com/rakshika-28/nasscom-project



rakshika-28 / nasscom-project

rakshika-28 Create README.md e9aeb85 5 minutes ago 2 commits

nasscom project	Add files via upload	5 minutes ago
README.md	Create README.md	5 minutes ago

README.md

nasscom-project

done by 18BIT0430

About

No descripti provided.

Readme

Releases

No releases pub
Create a new re

Packages

No packages pu
Publish your firs

VIDEO LINK:

https://drive.google.com/file/d/13YtGZ3TkRXNoXdulSn3fCyE5k8WMhGFQ/view?usp=sharing