Name: Kushank Jain

Reg_no: 21BCE8549

Campus: VIT-AP

1— 1.1 . Vulnerability Name: Cross-Site Scripting (Stored)

CWE : CWE-79

OWASP Category: A03:2021 – Injection

Description: Untrusted data enters a web application, typically from a web request

Business Impact: The application stores dangerous data in a database, message forum, visitor log, or other trusted data store. At a later time, the dangerous data is subsequently read back into the application and included in dynamic content. From an attacker's perspective, the optimal place to inject malicious content is in an area that is displayed to either many users or particularly interesting users. Interesting users typically have elevated privileges in the application or interact with sensitive data that is valuable to the attacker. If one of these users executes malicious content, the attacker may be able to perform privileged operations on behalf of the user or gain access to sensitive data belonging to the user. For example, the attacker might inject XSS into a log message, which might not be handled properly when an administrator views the logs.

Vulnerability Path :http://192.168.0.109:8080/

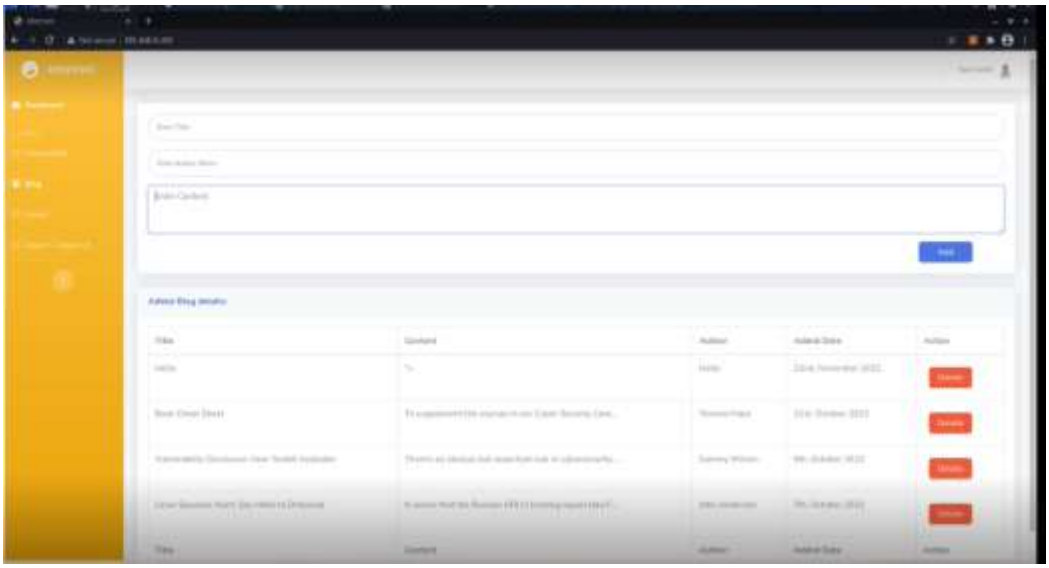
Vulnerability Parameter: http://192.168.0.109:8080/admin/en/blog

Steps to Reproduce :

Step 1. Access the URL

Step 2: Go to the login page and enter credentials

Step 3: Now you will be redirected to the dashboard where we will enter the script.



Step 3:-after entering the script content like” hacked” u will find the dialogue box as shown below.

Recommendation:

- Note that proper output encoding, escaping, and quoting is the most effective solution for preventing XSS, although input validation may provide some defense-in-depth.

1— 1.2 . Vulnerability Name: Cross-Site Scripting (Stored)

CWE : CWE-79

OWASP Category: A03:2021 – Injection

Description: Untrusted data enters a web application, typically from a web request

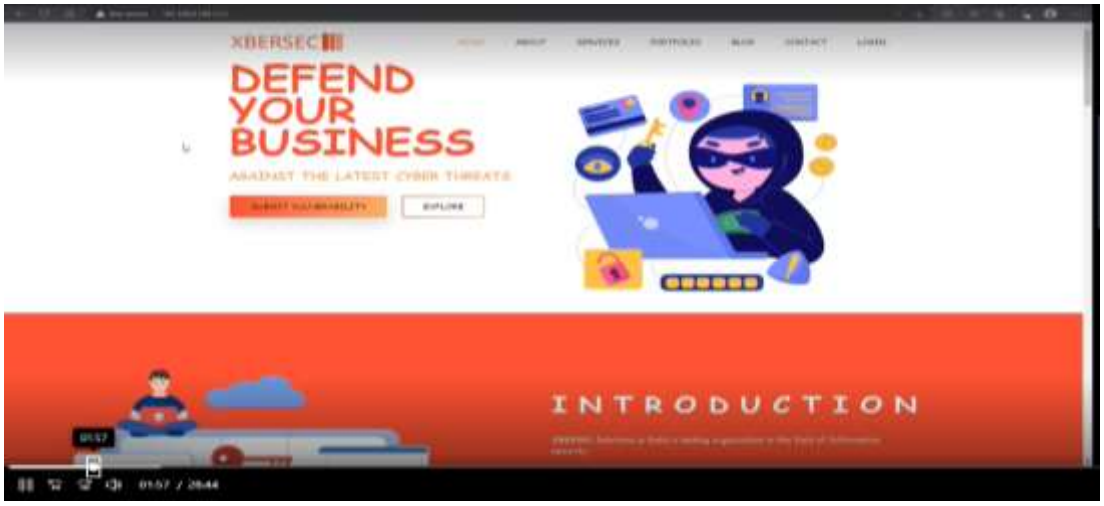
Business Impact: The application stores dangerous data in a database, message forum, visitor log, or other trusted data store. At a later time, the dangerous data is subsequently read back into the application and included in dynamic content. From an attacker's perspective, the optimal place to inject malicious content is in an area that is displayed to either many users or particularly interesting users. Interesting users typically have elevated privileges in the application or interact with sensitive data that is valuable to the attacker. If one of these users executes malicious content, the attacker may be able to perform privileged operations on behalf of the user or gain access to sensitive data belonging to the user. For example, the attacker might inject XSS into a log message, which might not be handled properly when an administrator views the logs.

Vulnerability Path :http://192.168.0.109:8080/

Vulnerability Parameter: http://192.168.0.109:8080/admin/en/report_contact_us

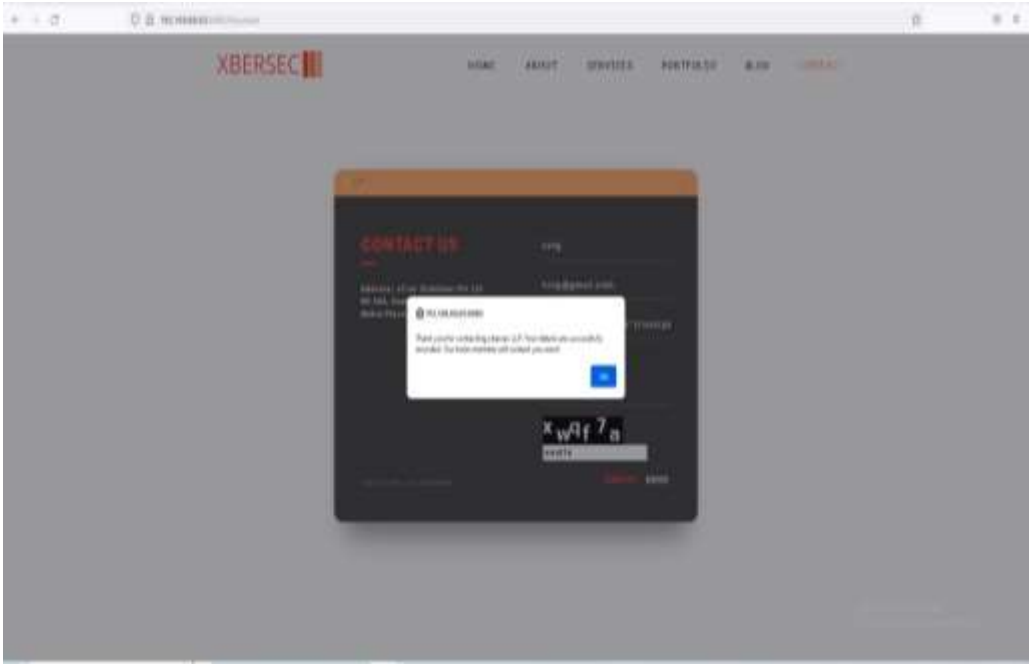
Steps to Reproduce :

Step 1. Access the URL

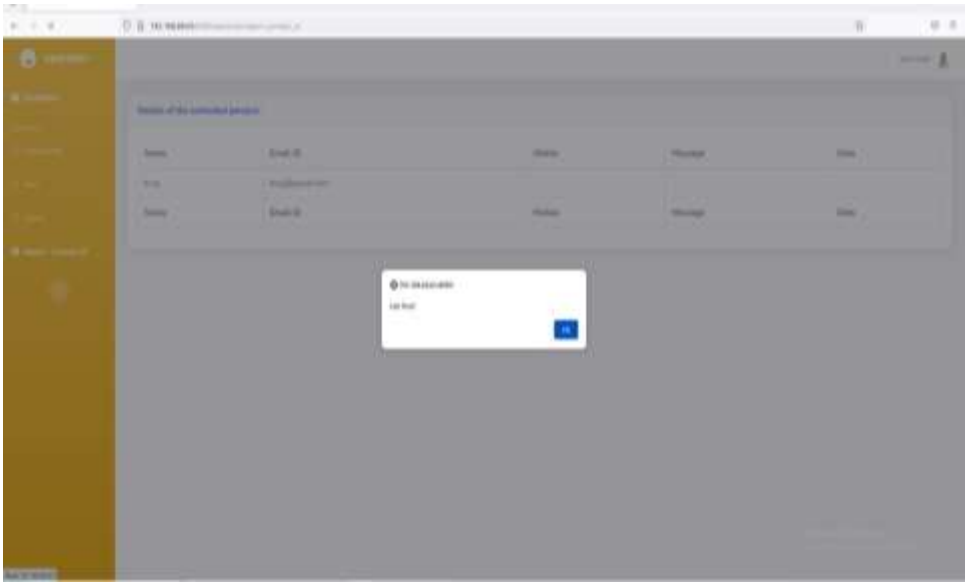


Step 2: Go to the login page and enter credentials

Step 3: Now you will be redirected to the dashboard where we enter the script in the contact_us page.



Step 4:- this is the pop up you get after you successfully inject the script in the contact page.



Recommendation:

- Note that proper output encoding, escaping, and quoting is the most effective solution for preventing XSS, although input validation may provide some defense-in-depth.

2 .Vulnerability Name:No Session Management

CWE : CWE-384

OWASP Category:A07:2021 –Identification and Authentication Failures

Description:An attacker is able to force a known session identifier on a user so that, once the user authenticates, the attacker has access to the authenticated session.

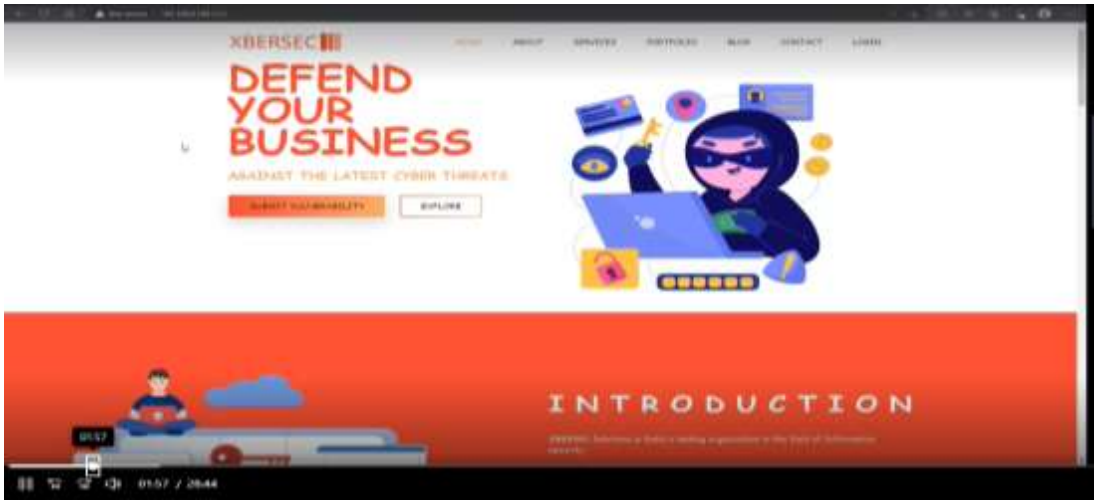
Business Impact:Without appropriate session management, you can run into several security problems, putting your users at risk. Common vulnerabilities caused by a lack of or poorly implemented session management include: Session hijacking

Vulnerability Path :http://192.168.0.109:8080/

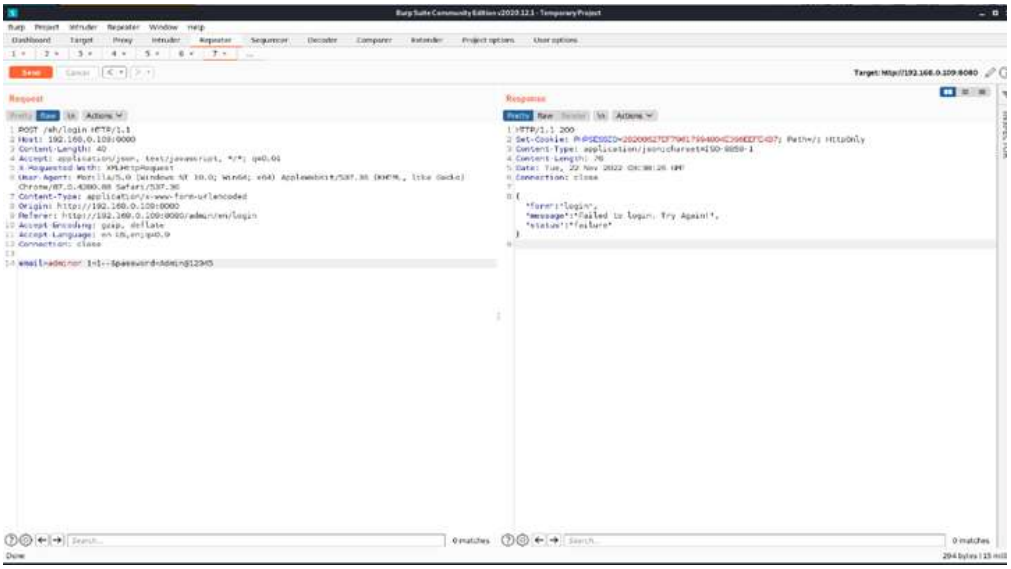
Vulnerability Parameter:http://192.168.0.100:8080/sh/admclient

Steps to Reproduce :

Step 1. Access the URL



Step 2.without the proper session management the burp can still access the request of session as shown.



Recommendation:

- Invalidate any existing session identifiers prior to authorizing a new user session.

3 . Vulnerability Name:Login Captcha Bypass

CWE : CWE-804

OWASP Category:A06:2021-Vulnerable and Outdated Components

Description:An automated attacker could bypass the intended protection of the CAPTCHA challenge and perform actions at a higher frequency than humanly possible, such as launching spam attacks.

Business Impact:When authorization, authentication, or another protection mechanism relies on CAPTCHA entities to ensure that only human actors can access certain functionality, then an automated attacker such as a bot may access the restricted functionality by guessing the CAPTCHA.

Vulnerability Path :http://192.168.0.109:8080/

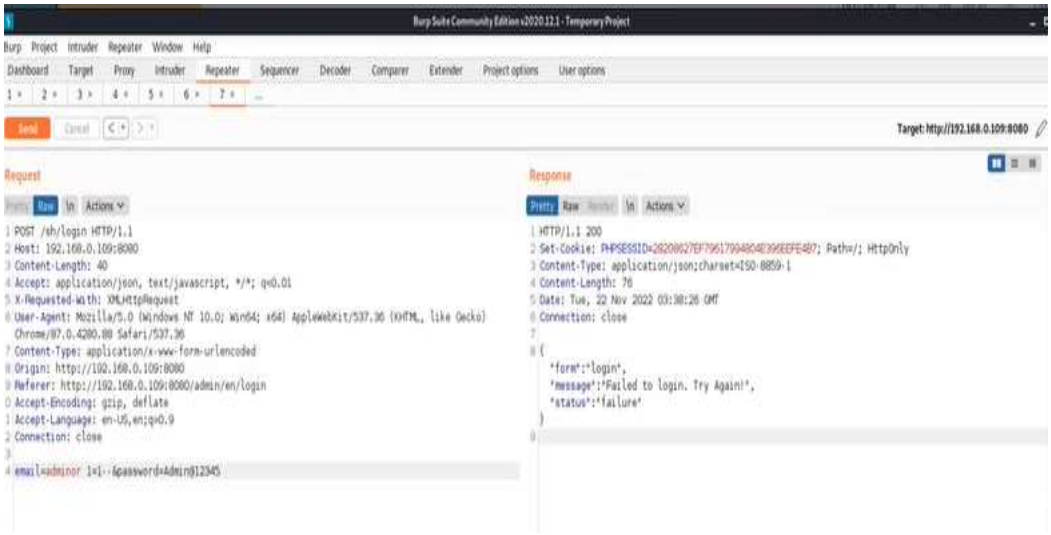
Vulnerability Parameter:**http://192.168.0.109:8080/sh/login**

Steps to Reproduce :

Step 1. Access the URL

Step :- in the backend this login page is by default giving the access without the captcha.

Step 3:- these the backend in the burp where you can see that without the captcha validation it is given the access to session.



Recommendation:

Ensure that the CAPTCHA value stored in the session is verified against the user's input and is removed when the request is submitted.

4 . Vulnerability Name:Admin Login SQL Injection

CWE : CWE-284

OWASP Category:A03:2021-Injections

Description:Without sufficient removal or quoting of SQL syntax in user-controllable inputs, the generated SQL query can cause those inputs to be interpreted as SQL instead of ordinary user data. This can be used to alter query logic to bypass security checks, or to insert additional statements that modify the back-end database, possibly including execution of system commands.

Business Impact:If authorization information is held in a SQL database, it may be possible to change this information through the successful exploitation of a SQL injection vulnerability

Vulnerability Path :http://192.168.0.109:8080/

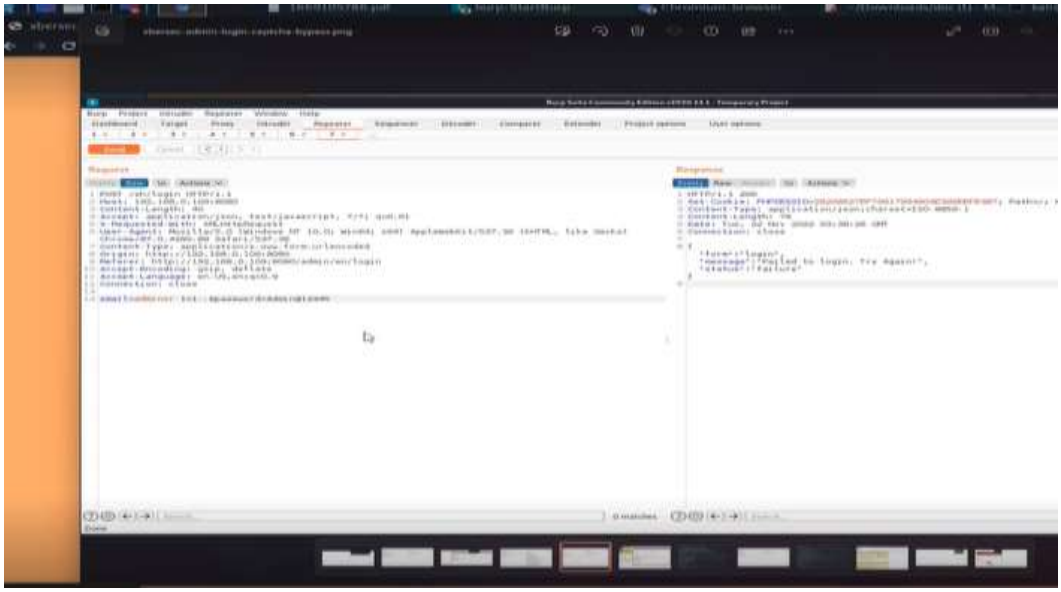
Vulnerability Parameter:**http://192.168.0.109:8080/sh/login**

Steps to Reproduce :

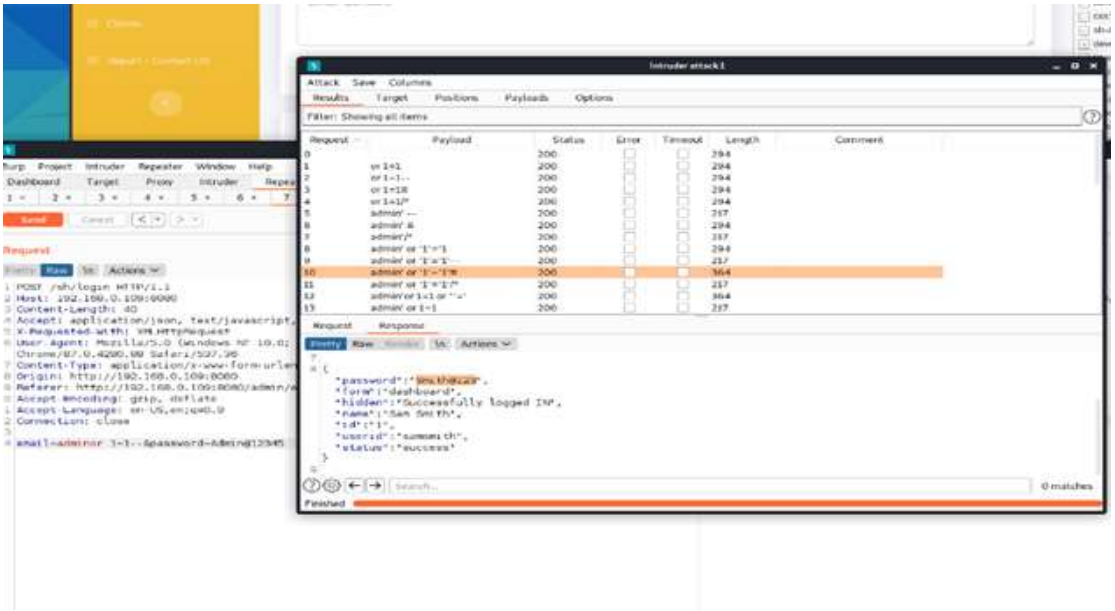
Step 1. Access the URL

Step 2:- Enter the login credentials in and try to validate as shown below.

Step 3:- in the backend we need to use the burp to configure the user with sql injection.



Step 4: - this is the place where we upload injection then you will gain access to one of the user accounts.



Recommendation:

Employ a layered approach to security that includes utilizing parameterized queries when accepting user input, ensuring that only expected data (white listing) is accepted by an application, and harden the database server to prevent data from being accessed inappropriately.

5 . Vulnerability Name:Improper access control

CWE : CWE-284

OWASP Category:A01:2021 – Broken Access Control

Description:The SQL Injection occurs when user-controllable input is interpreted as a SQL command, rather than as normal data, by the backend database.Exploitation of this vulnerability can have critical implications, including creation, modification, or exfiltration of database content.

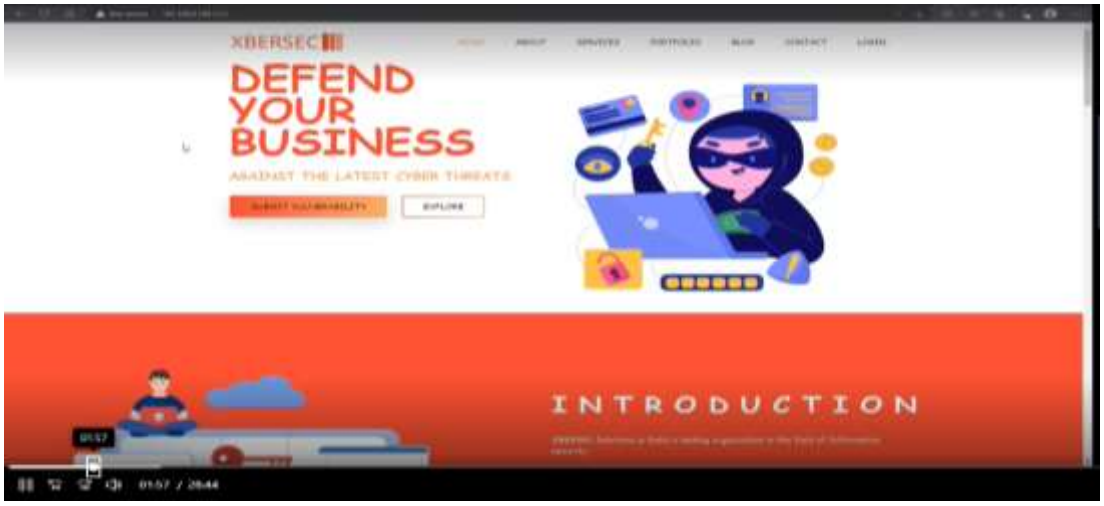
Business Impact:It may be possible to read sensitive information, it is also possible to make changes or even delete this information with a SQL injection attack.

Vulnerability Path :http://192.168.0.109:8080/

Vulnerability Parameter:http://192.168.0.109:8080/admin/en/vulnerability

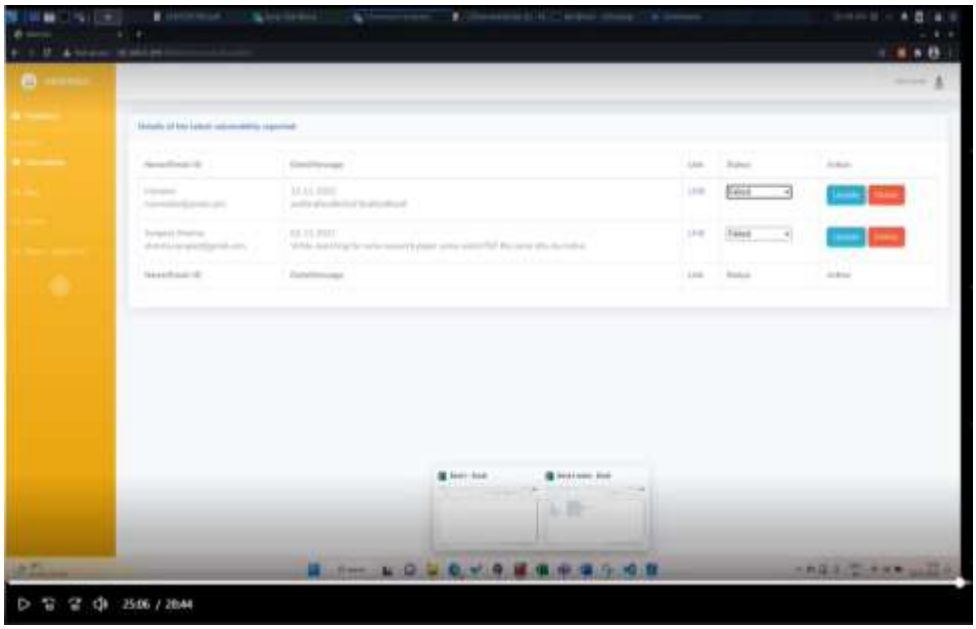
Steps to Reproduce :

Step 1. Access the URL

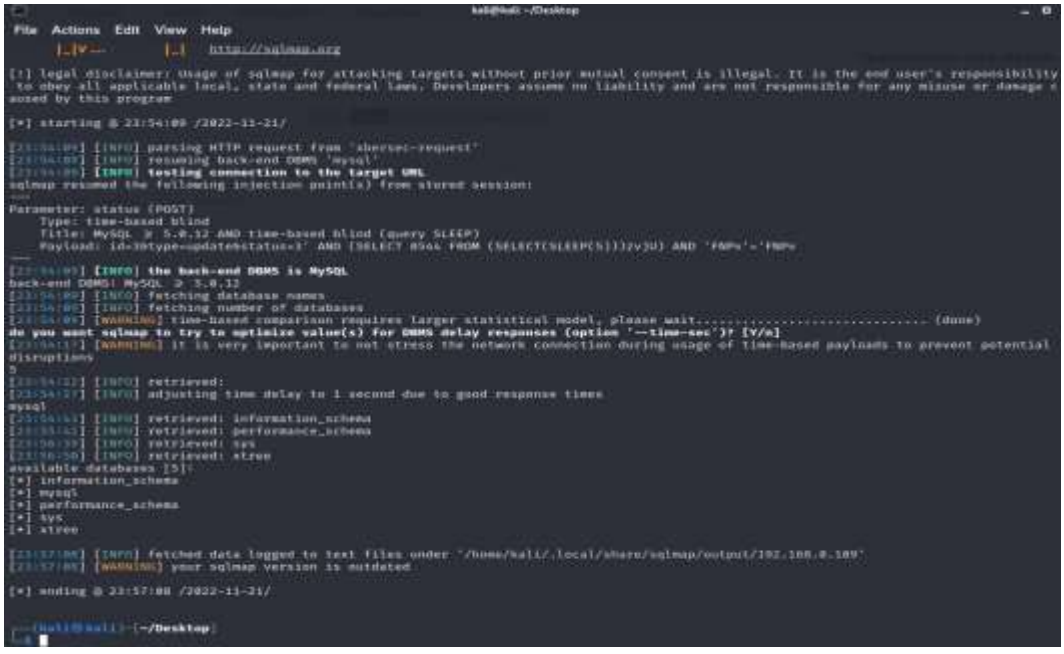


Step 2:- Try to enter the login credentials

Step 3 :- After that with the default credentials assessed we can gain access to the database with entering the default credentials.



Step 4:- this is the map where status url remain unchanged



Recommendation:

- Applications should not incorporate any user-controllable data directly into SQL queries.
- Parameterized queries (also known as prepared statements) should be used to safely insert data into predefined queries.

6 . Vulnerability Name:Default Credentials

CWE : CWE-1392

OWASP Category:A07:2021-Identification and Authentication Failures

Description:It is common practice for products to be designed to use default keys, passwords, or other mechanisms for authentication. The rationale is to simplify the manufacturing process or the system administrator's task of installation and deployment into an enterprise. However, if admins do not change the defaults, it is easier for attackers to bypass authentication quickly across multiple organizations.

Business Impact:Attackers can easily obtain default passwords and identify internet-connected target systems. Passwords can be found in product documentation and compiled lists available on the internet.

Vulnerability Path :http://192.168.0.109:8080/

Vulnerability Parameter:http://192.168.0.109:8080/manager/html

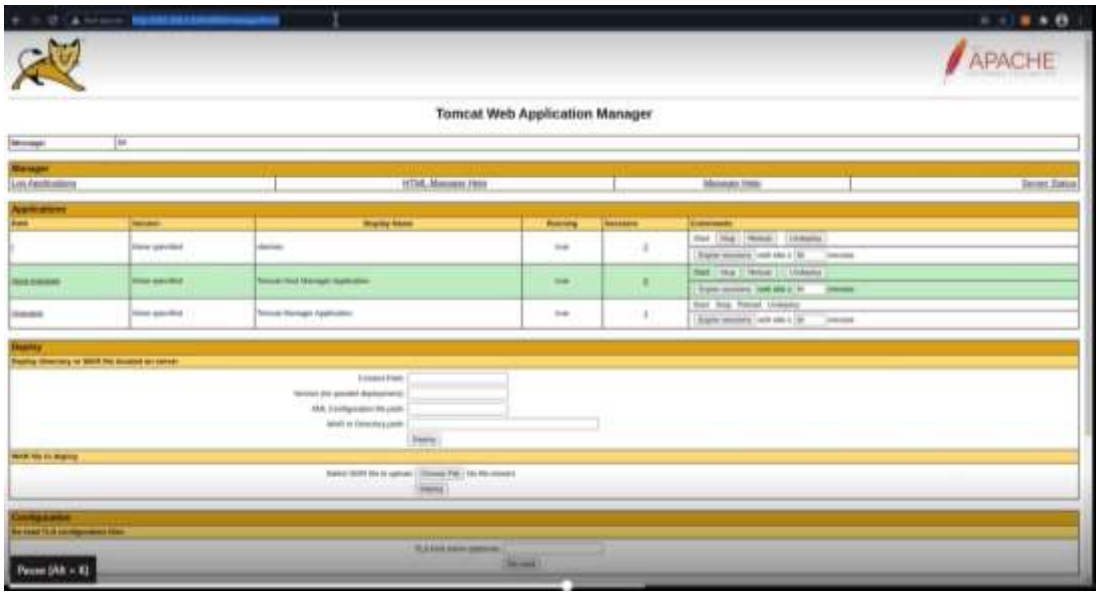
Steps to Reproduce :Step 1. Access the URL

Step 2: By changing the URL parameters with tomcat configurations we find the below page.

Step 3:- By closing the dialog box without canceling it this will give you the access to

the default credentials

Step4 :- Here in this step the default credentials are printed on the screen which helps to gain access to the DB.



Recommendation:

- Prohibit use of default, hard-coded, or other values that do not vary for each installation of the product - especially for separate organizations.