CYBER SECURITY JULY MINOR PROJECT

PROBLEM STATEMENT

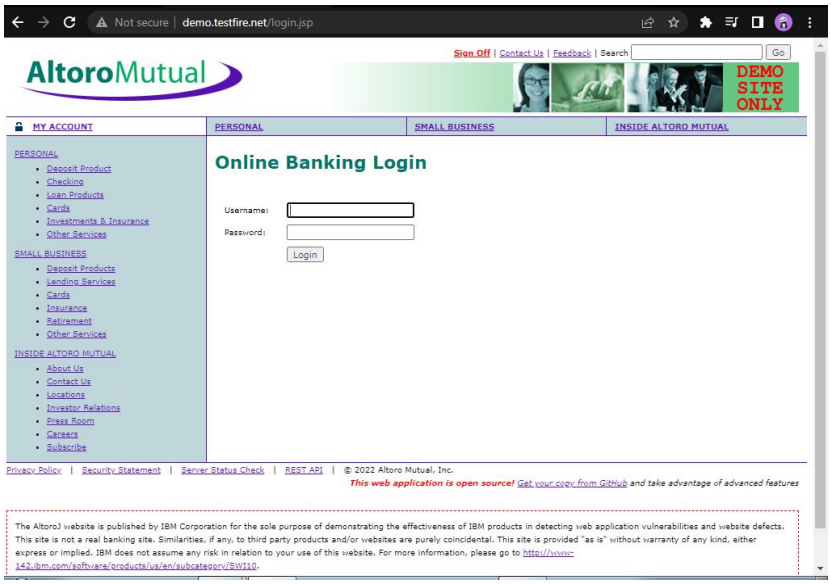
4.

Perform Bypass Authentication on http://demo.testfire.net website with different payloads and make report along with screenshots and mention to mitigation steps to protect.

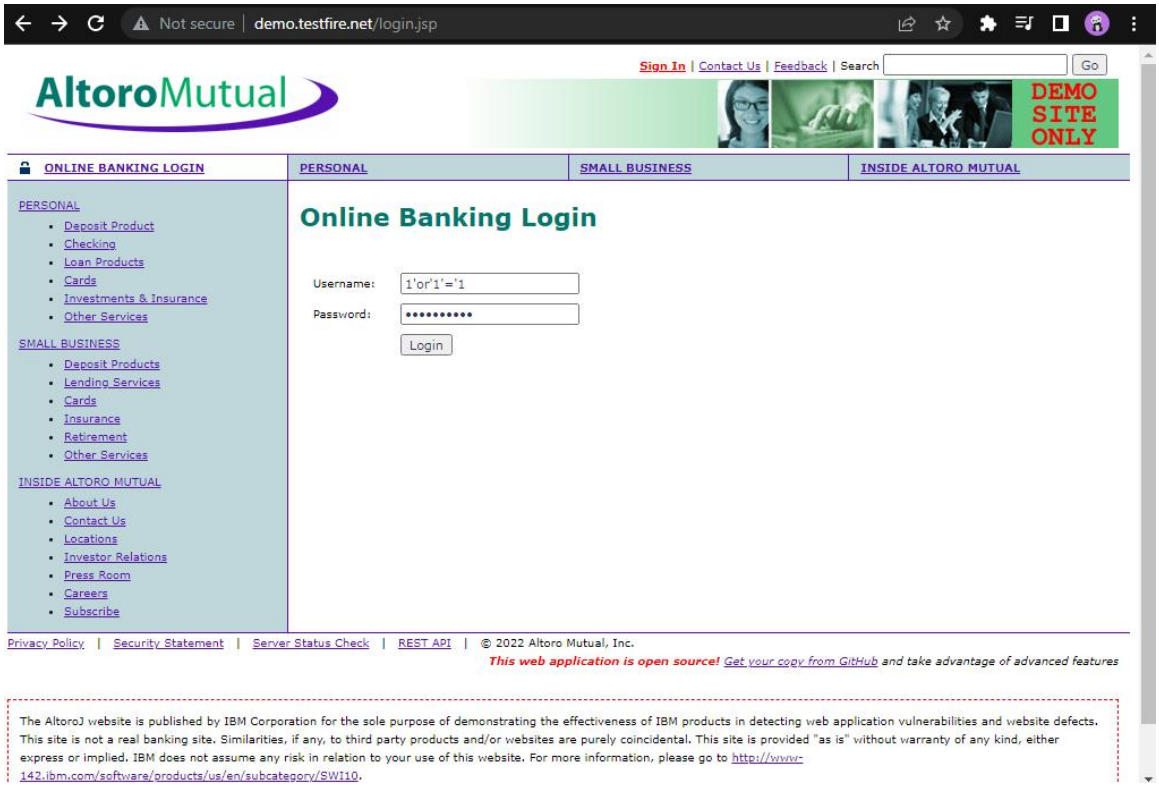
INTRODUCTION

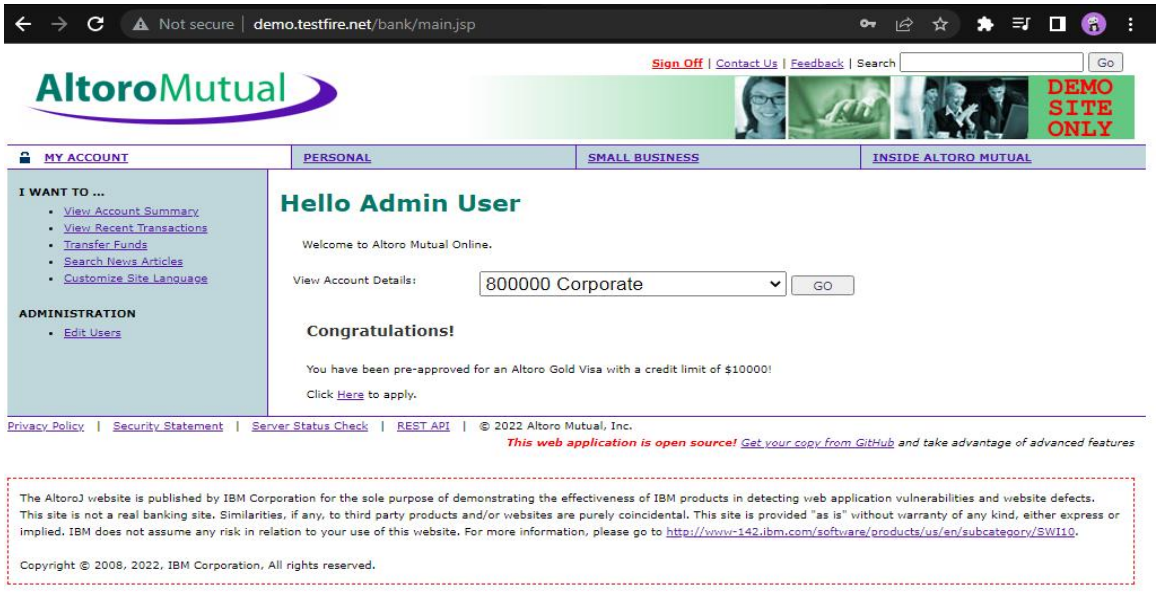
Bypass authentication refers to an attacker gaining access equivalent to an authenticated user without ever going through an authentication procedure. This is usually the result of the attacker using an unexpected access procedure that does not go through the proper checkpoints where authentication should occur.

REPORT



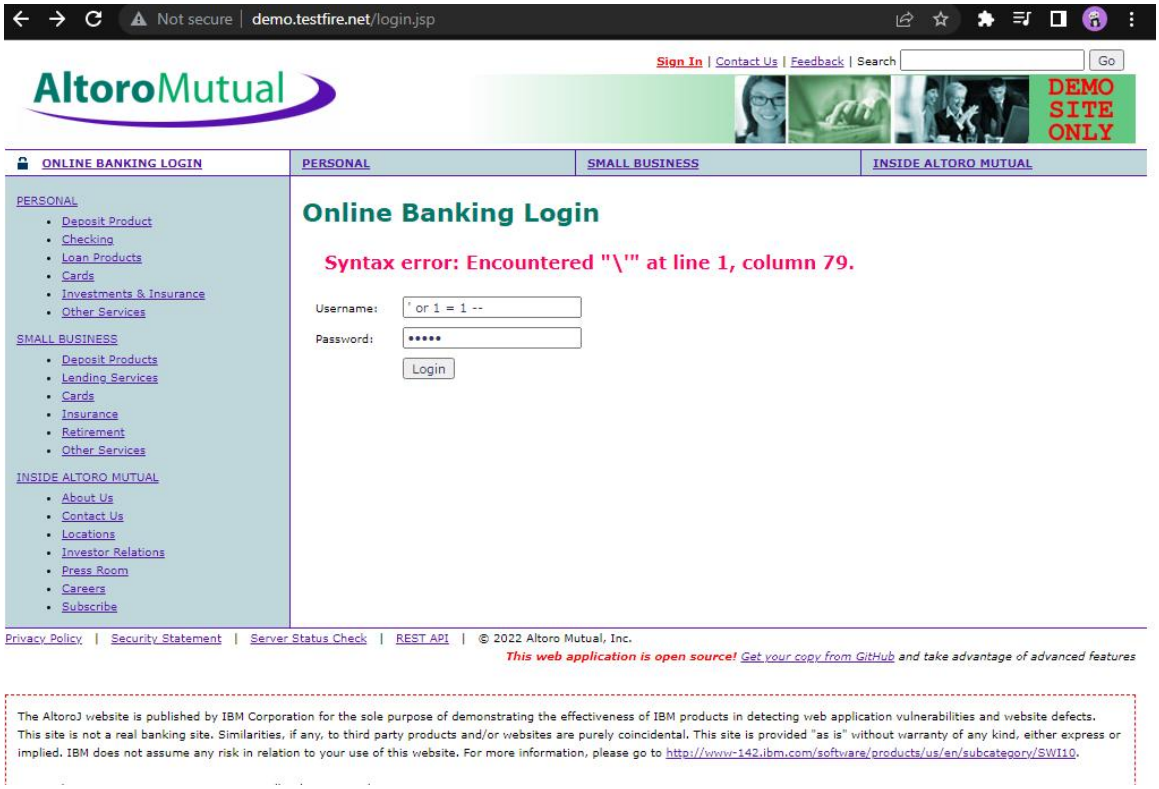
i)

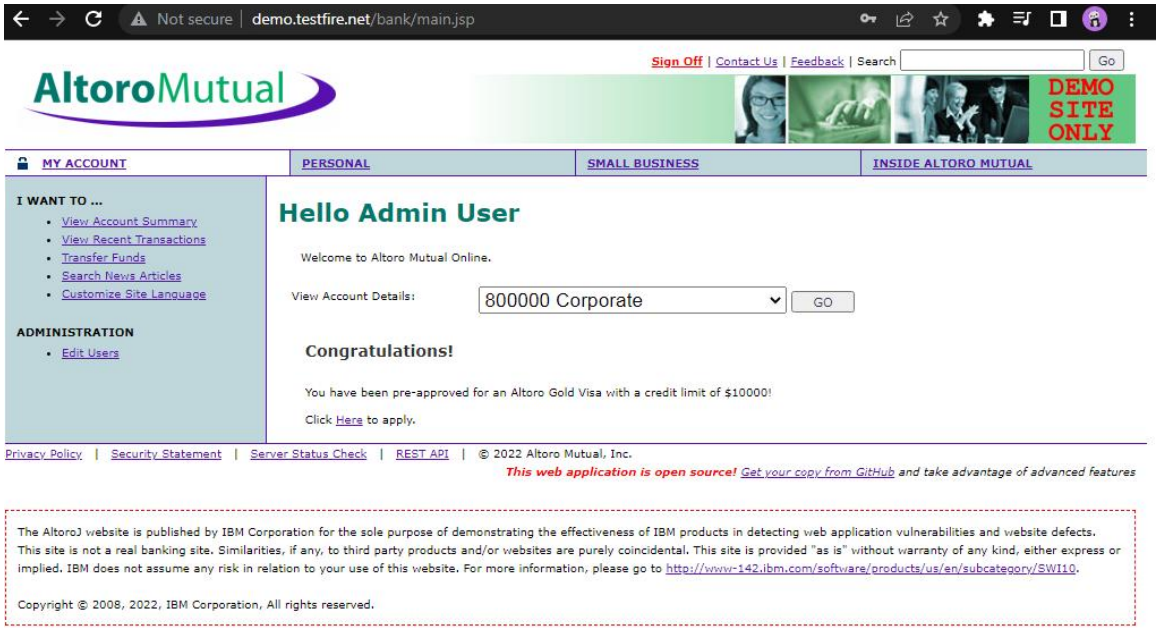




In above screenshots the username and password is same.

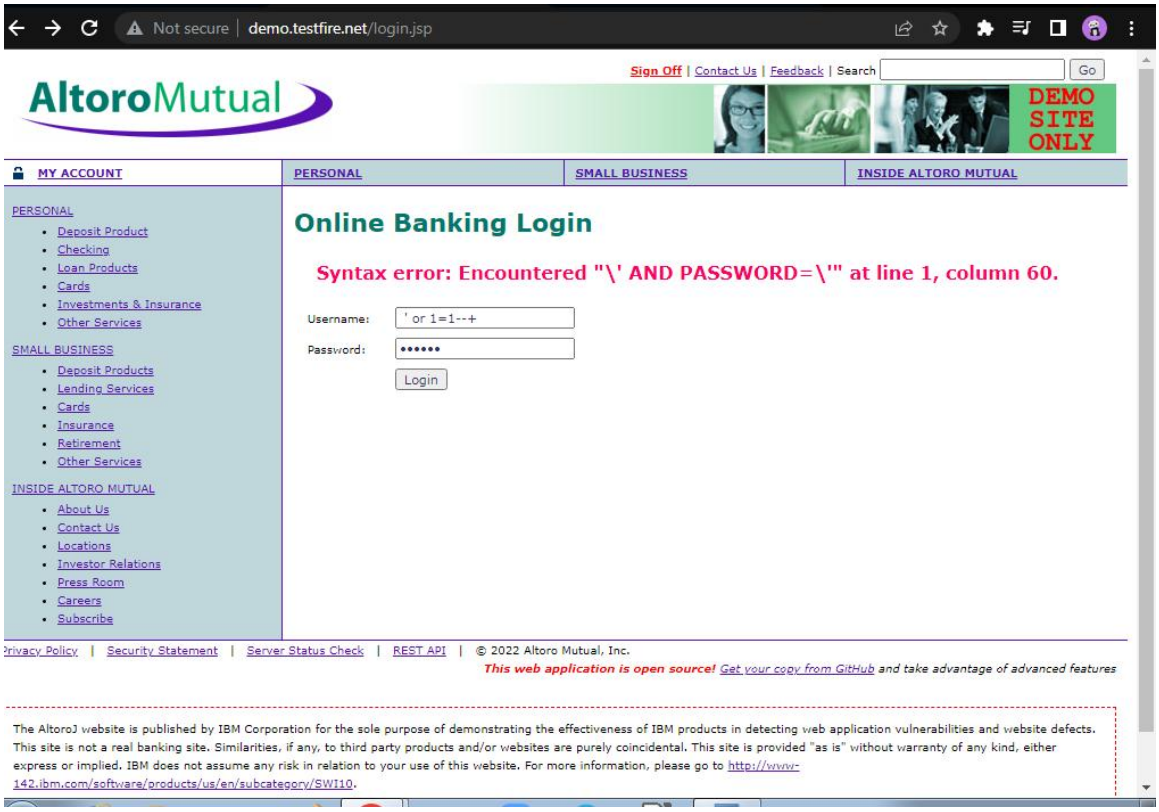
ii)

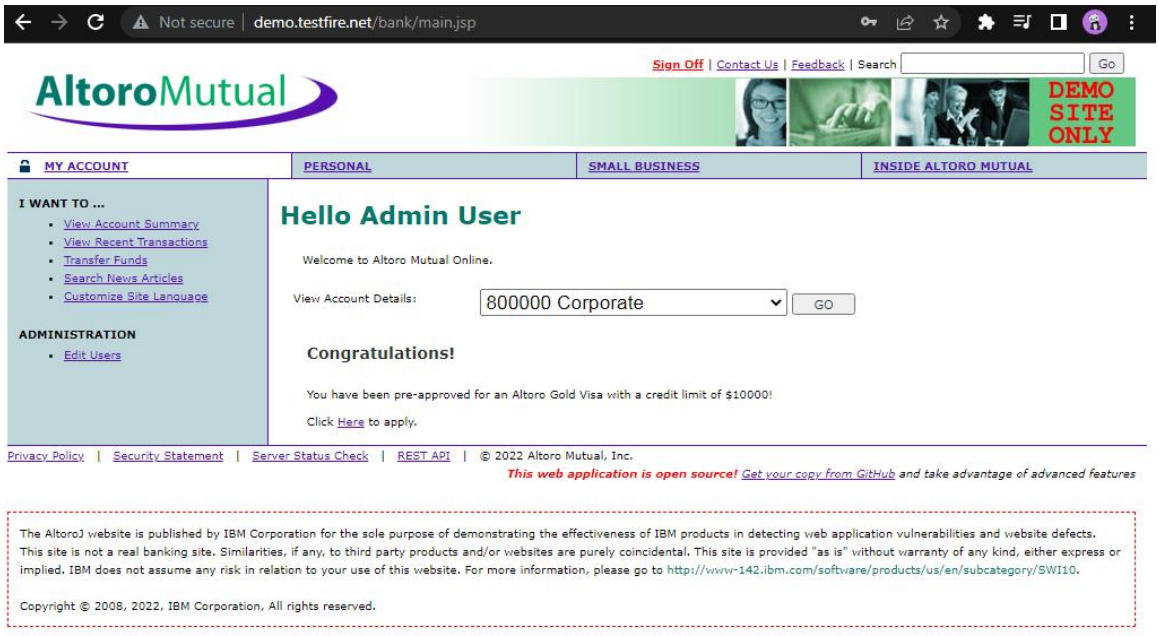




In above screenshots the password is 'foo'.

iii)





In above screenshots the password is 'foo'.

MITIGATION STEPS FOR PROTECTION:

This vulnerability can be eliminated by fixing the SQL injection vulnerability in the application's authentication mechanism.

The authentication bypass vulnerability is a special case of SQL injection, specifically located in your authentication routines.

The following recommendations will help to mitigate the risk of Authentication Bypass attacks:

- Keep up to date on patches and security fixes as they are released by the vendor or maintainer
- You always check for all vulnerabilities and always install the best antivirus software and are always free from bugs.
- To Avoid the special character '=' 'or' to bypass authentication, you can use the " mysqli_real_escape_string() ".
- It is best to have a secure and strong authentication policy in place.
- Avoid using external SQL interpreters.
- It is best to ensure all systems, folders, apps, are password protected.
- Audit your applications frequently for points where HTML input can access interpreters.
- Security experts recommend resetting default passwords with unique strong passwords and periodically rotate passwords.
- It is suggested to not expose authentication protocol in the client-side web browser script.
- They suggest ensuring that user session IDs and cookies are encrypted.
- It is recommended to validate all user input on the server side.
- Avoid the use of dynamic SQL or PL/SQL and use bound variables whenever possible.

Enforce strict limitations on the rights to database access.

Remove any sample applications or demo scripts that allow remote database queries.
