Summary of Findings

|  |  |  |  |
| --- | --- | --- | --- |
| **Code** | **Title** | **Risk Rating** | **Affected Function(s)** |
| F1 | Weak 2FA Implementation | **High** | Login, Reset password |
| F2 | Privilege Escalation via Broken Access Control | **Medium** | View account detail |

Vulnerability Details

# Weak 2FA Implementation

Code: F1

Rating: High

## Description:

Two Factor Authentication (2FA) adds an extra layer of security to user accounts by prompting them to enter a uniquely generated one-time password (OTP) after they have successfully inputted their username and password. The application's implementation of 2FA is weak which makes user accounts more susceptible to compromise. An attacker can take advantage of this weak 2FA implementation and potentially take over user accounts.

## Impact:

This vulnerability can lead to reputational damage and indirect financial loss to the company as customers may view the application as insecure.

## Recommendation:

There is no single technique to implement 2FA securely. However, the following best practices should be adhered to:  
  
- The 2FA code should be invalidated each time a new code is requested  
  
- 2FA should be implemented for users upon sensitive actions such as login, change of password or security questions, elevation of user session, change of email address or phone number, and disabling of 2FA.  
  
- The uniquely generated OTP should expire  
  
- The page behind the 2FA step should not be able to be accessed directly by manipulating the URL  
  
- 2FA should not be automatically disabled during the password recovery process  
  
- The 2FA procedure should not disclose any sensitive information. For example, do not disclose the unredacted phone number of a user  
  
- The 2FA process cannot depend on client-side modifiable header or status codes

## Affected Items:

|  |  |
| --- | --- |
| Function | URI |
| Login | https://example.com/login |
| Reset password | https://example.com/reset-password |

# Privilege Escalation via Broken Access Control

Code: F2

Rating: Medium

## Description:

Access controls can be bypassed through a variety of ways including, calling an internal post authentication page, modifying the given URL parameters, by manipulating the form, or by counterfeiting sessions. The access controls for this application can be bypassed by an attacker who can gain access to a privileged user’s account and functionality. As a result, the attacker has access to more resources or functionality within the application. This could include viewing or editing sensitive customer data, and viewing or editing other user permissions.

## Impact:

The impact of this vulnerability can vary in severity depending on the degree of access to resources or functionality the attacker is able to gain. An attacker with the ability to access, delete, or modify data from within the application could result in reputational damage for the business through the impact to customers’ trust. This can also result in indirect financial cost to the business through fines and regulatory bodies if sensitive data is accessed. The severity of the impact to the business is dependent on the sensitivity of the data being stored in, and transmitted by the application.

## Recommendation:

There are a number of best practices involved with implementing strong access controls, including the following:  
  
- Declare the type of access allowed for each resource and deny access by default, unless these specific access conditions are met by a user attempting to access the resource. To enforce these access controls, it’s important to use a single application wide mechanism.  
  
- It is recommended that all critical information is stored server-side, with only a session ID being passed back and forth between the server and client.  
  
- Similarly, the data being sent to the client should be encrypted to ensure that data cannot be viewed in transit, and it should be digitally signed to ensure the data's integrity.  
  
- Thoroughly test and audit the access controls during the development and deployment lifecycle to ensure the security and integrity of the controls.  
  
For further information, please see the following resources:  
  
- https://owasp.org/Top10/A01\_2021-Broken\_Access\_Control/  
  
- https://owasp.org/www-community/Access\_Control

## Affected Items:

|  |  |
| --- | --- |
| Function | URI |
| View account detail | https://example.com/account |