Identity Management Testing

1. Introduction

Identity management testing focuses on how the application manages user accounts, roles, and permissions. This includes testing the registration, provisioning, and enumeration of user accounts, as well as the enforcement of username and password policies. Weaknesses in identity management can lead to unauthorized access, privilege escalation, and account takeover.

2. CVSS Score Summary

Test Case	Vulnerability	CVSS v3.1 Score	CVSS Vector	Severity
OTG-IDENT-001	Test Role Definitions	0.0	N/A	Informational
OTG-IDENT-002	Test User Registration Process	0.0	N/A	Informational
OTG-IDENT-003	Test Account Provisioning Process	0.0	N/A	Informational
OTG-IDENT-004	Account Enumeration	5.3	CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:N/A:N	Medium
OTG-IDENT-005	Testing for Weak or unenforced username policy	0.0	N/A	Informational

3. Summary of Findings

The most significant finding in this category is the presence of an account enumeration vulnerability. The application provides different error messages for valid and invalid usernames, which could allow an attacker to build a list of registered users. This information could then be used in further attacks, such as brute-forcing passwords or social engineering.

A key challenge during this assessment was the absence of a user registration page. This prevented the execution of tests for weak password policies and unenforced username policies. While the tests that were performed did not reveal any other major issues, the inability to fully assess the registration process leaves a potential gap in the understanding of the application's identity management security.

4. General Recommendation/Remediation

- **Implement Generic Error Messages:** The application should return a generic error message for all login failures, regardless of whether the username is valid or not. This will prevent attackers from being able to enumerate valid usernames.
- **Implement a User Registration Process:** A user registration process should be implemented to allow for the testing of password policies and username policies.
- **Enforce Strong Password Policies:** The application should enforce strong password policies, including minimum length, complexity, and expiration.
- **Enforce Strong Username Policies:** The application should enforce strong username policies, including restrictions on the use of special characters and common usernames.