# Mobile Application Security Assessment Report

## 1. Introduction

Objective: Conduct a security assessment on a mobile application to identify common vulnerabilities, including insecure data storage, communication, and authentication mechanisms.  
  
Scope: This report covers the primary security mechanisms and data protection implementations of the application.  
  
Tools Used: MobSF, Burp Suite, and Frida.

## 2. Assessment Methodology

This assessment uses the following approaches:  
1. Static Analysis: Reviewing the app's code and configurations to find security weaknesses.  
2. Dynamic Analysis: Testing the app in runtime to detect vulnerabilities in real-time.  
3. Reverse Engineering: Disassembling the app’s code for deeper insights into its data handling.

## 3. Vulnerability Assessment Categories

### 3.1 Insecure Data Storage

Objective: Verify that sensitive data is not stored insecurely.  
  
Steps:  
- Check for storage of sensitive data in Shared Preferences, external storage, or databases.  
- Ensure encryption is used for sensitive data.  
  
Findings:  
Example: User credentials were found stored in plain text within Shared Preferences.  
  
Recommendations:  
- Encrypt sensitive data before storage using Android's Keystore or iOS's Keychain.

### 3.2 Insecure Communication

Objective: Ensure secure data transmission between the app and server.  
  
Steps:  
- Verify the use of HTTPS/TLS for all communications.  
- Check for certificate pinning to prevent MITM attacks.  
  
Findings:  
Example: Some API calls were observed using HTTP instead of HTTPS.  
  
Recommendations:  
- Enforce HTTPS with TLS 1.2 or higher and enable certificate pinning.

### 3.3 Insecure Authentication Mechanisms

Objective: Evaluate the security of user authentication.  
  
Steps:  
- Inspect login flows for secure transmission of credentials.  
- Verify secure session management for token handling.  
  
Findings:  
Example: Weak password policies were observed, allowing passwords with fewer than 8 characters.  
  
Recommendations:  
- Implement strong password policies and secure session management.

## 4. Results Summary

|  |  |  |  |
| --- | --- | --- | --- |
| Vulnerability | Severity | Description | Recommendation |
| Insecure Data Storage | High | Sensitive data stored in plain text within Shared Preferences. | Use Android Keystore or iOS Keychain for data encryption. |
| Insecure Communication | Medium | Some endpoints use HTTP instead of HTTPS. | Enforce HTTPS and enable certificate pinning. |
| Weak Authentication | Medium | Weak password policies allow short, simple passwords. | Enforce stronger password policies and secure session handling. |

## 5. Conclusion

This assessment identified several security vulnerabilities, including insecure data storage, communication, and weak authentication mechanisms. To enhance application security, it is recommended to implement encryption, enforce HTTPS with certificate pinning, and adopt stronger password policies.  
  
A re-assessment is advised post-implementation of the recommended security measures.