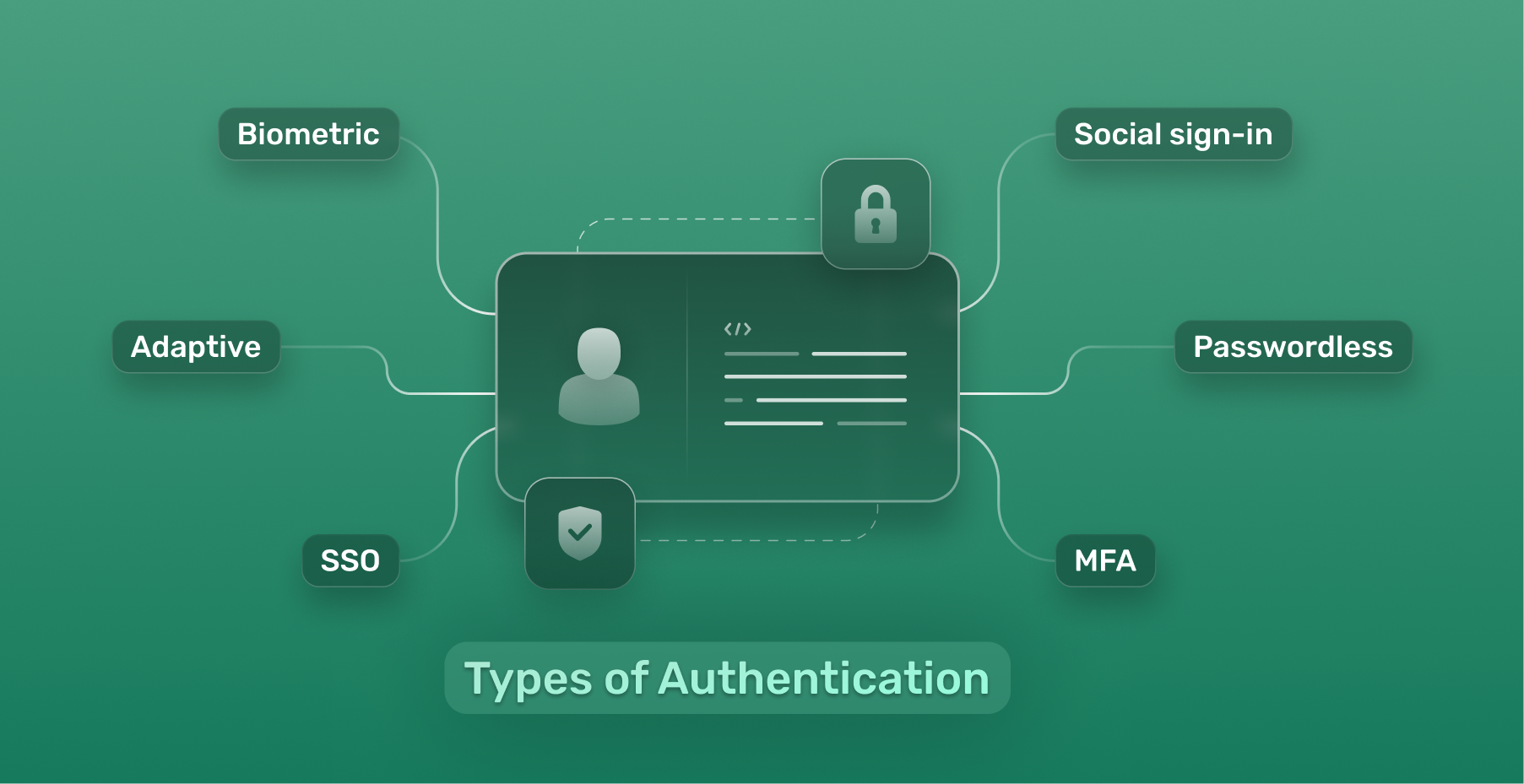
Authentication Security Research



[1. Introduction 2](#_Toc1528266147)

[2. Scope 3](#_Toc435369403)

[3. Determining threats 4](#_Toc291465122)

[Threat categorization – STRIDE Model 4](#_Toc1477484213)

[Use/Misuse cases 4](#_Toc1915226792)

[4. Countermeasures and mitigation 4](#_Toc257686237)

[Database security 5](#_Toc14613878)

[5. Assessment 5](#_Toc1882853562)

[Security controls checklist 6](#_Toc2064749671)

[Risk assessment 6](#_Toc1501861821)

[Autentication data flow 6](#_Toc1922208720)

[References 7](#_Toc1380723590)

# Introduction

This document describes the authentication design and security strategy for the secure programming minor. It outlines authentication methods, secure storage of credentials, and cryptographic principles applied, aligned with OWASP standards and best practices (OWASP Foundation, 2022).

Authentication is critical to ensure that users are properly identified and unauthorized access is prevented. This study identifies potential risks, mitigation strategies, and recommendations for a secure authentication system.

# Scope

Authentication will be required to access the app's protected features. The authentication system will support:

* Username and password login
* Optional biometric authentication (Fingerprint/Face ID via Android BiometricPrompt API) (Android Developers, 2025)
* Secure session management

**Primary Assets to Protect:**

* User credentials (username, password)
* Session tokens (in case of session-based auth)
* Biometric authentication tokens
* Personal user data accessed post-authentication

**Entry Points:**

* Login screen
* Biometric prompt
* Forgot password/reset password feature
* Registration screen (if applicable)

**Trust levels:**

|  |  |  |
| --- | --- | --- |
| Id | Name | Description |
| 1 | Unauthenticated User | User not logged in |
| 2 | Authenticated User | User who has provided valid credentials |
| 3 | Authenticated with Biometrics | User authenticated using biometrics |

# Determining threats

## Threat categorization – STRIDE Model

|  |  |
| --- | --- |
| Stride Category | Potential threats in Authentication Context |
| Spoofing | Unauthorized access using stolen credentials or fake biometrics |
| Tampering | Modifying credential storage, tampering with network authentication requests |
| Repudiation | Users denying authentication actions |
| Information Disclosure | Leakage of credentials, session tokens, or biometric data |
| Denial of Service | Lockout attacks, biometric sensor overload |
| Elevevation of Privilege | Bypassing authentication to gain higher privileges |

## Use/Misuse cases

* **Use Case:** User enters correct credentials -> access granted
* **Misuse Case:** Attacker guesses/brute-forces password -> access unauthorized

# Countermeasures and mitigation

|  |  |
| --- | --- |
| Threat | Mitigation techniques |
| Credential theft | Use secure password hashing (e.g., Argon2, bcrypt) (NIST, 2020) |
| Biometric spoofing | Rely on trusted Android BiometricPrompt API (Android Developers, 2025) |
| Network tampering | Enforce HTTPS/TLS 1.3 for all communication, certificate pinning |
| Credential storage attack | Use Android Keystore System to store sensitive authentication tokens |
| Session hijacking | Implement secure session expiration, regeneration on login, token invalidation |
| Brute force attacks | Limit login attempts, use exponential back-off |
| Information disclosure | Encrypt sensitive local data with AES-256 |

## Database security

* No plain-text storage of passwords — store only strong salted+hashed versions
* Encrypted local database if any user data is stored (e.g., SQLCipher)

# Assessment

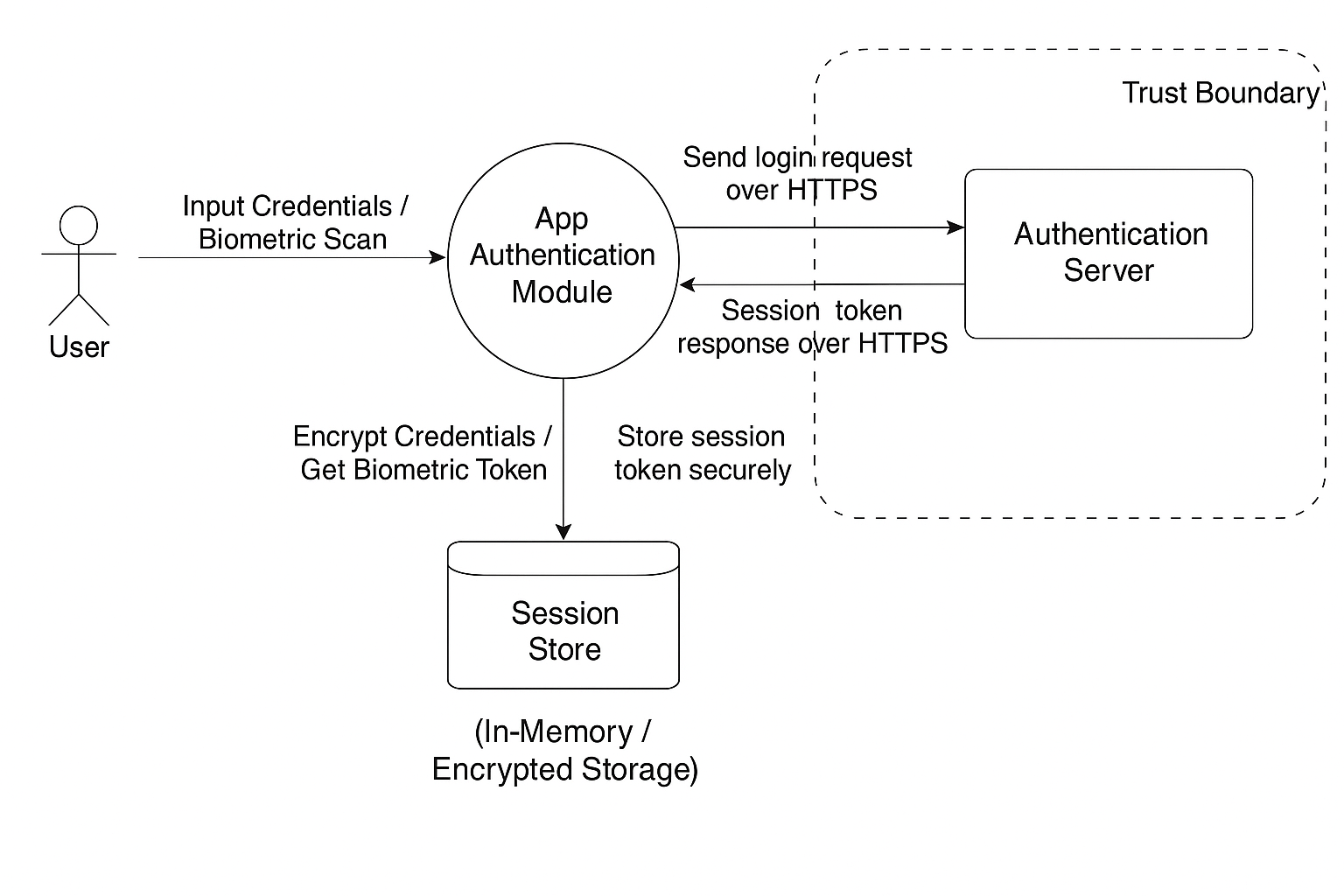
## Security controls checklist

* Passwords are hashed (bcrypt/Argon2)
* Biometric authentication uses secure hardware
* Network communication uses HTTPS and optionally certificate pinning
* Sensitive tokens stored using Android Keystore
* App has login attempt rate limiting
* Sessions are securely managed and expire appropriately

## Risk assessment

|  |  |  |  |
| --- | --- | --- | --- |
| Threat | Likelihood | Impact | Risk Level |
| Brute-force password attack | Medium | High | High |
| Biometric spoofing | Low | High | Medium |
| Session hijacking | Medium | Medium | Medium |
| Network sniffing | Low | High | Medium |
| Local database theft | Medium | High | High |

# Autentication data flow



# References

Android Developers. (2025). Biometric Authentication. From: <https://developer.android.com/training/sign-in/biometric-auth>

National Institute of Standards and Technology. (2020). NIST SP 800-63B Digital Identity Guidelines: Authentication and Lifecycle Management. U.S. Department of Commerce. From: <https://pages.nist.gov/800-63-3/sp800-63b.html>

OWASP Foundation. (2022). Authentication Cheat Sheet. From: <https://cheatsheetseries.owasp.org/cheatsheets/Authentication_Cheat_Sheet.html>

OWASP Foundation. (2022). Mobile Application Security Verification Standard (MASVS). From: <https://owasp.org/www-project-mobile-security-testing-guide/mstg/0x05c-security-verification-requirements/>