# Keamanan Aplikasi Berbasis Mobile

Aplikasi berbasis web memiliki sejumlah kerentanan umum yang dapat di eksploitasi. Open Web Application Security Project (OWASP) Foundation telah merilis 10 (sepuluh) jenis kerentanan keamanan aplikasi mobile paling kritis dan umum terjadi. Daftar ini dibuat berdasarkan penelitian dan analisis mendalam terhadap data yang dikumpulkan dari berbagai sumber di seluruh dunia.

Badan Siber dan Sandi Negara (BSSN) juga telah mengeluarkan pedoman teknis keamanan aplikasi berbasis mobile yang termuat dalam Peraturan BSSN Nomor 4 Tahun 2021 tentang Pedoman Manajemen Keamanan Informasi Sistem Pemerintahan Berbasis Elektronik dan Standar Teknis dan Prosedur Keamanan Sistem Pemerintahan Berbasis Elektronik.


- Mobile Apps Vulnerability
  - M1: Improper Credential Usage
  - M2: Inadequate Supply Chain Security
  - M3: Insecure Authentication/Authorization
  - M4: Insufficient Input/Output Validation
  - M5: Insecure Communication
  - M6: Inadequate Privacy Controls
  - M7: Insufficient Binary Protections
  - M8: Security Misconfiguration
  - M9: Insecure Data Storage
  - M10: Insufficient Cryptography

# Mobile Apps Vulnerability

Open Web Application Security Project (OWASP) Foundation telah merilis daftar kerentanan umum pada aplikasi berbasis web Top 10 adalah daftar yang dibuat oleh Open Web Application Security Project (OWASP) yang berisi sepuluh jenis kerentanan keamanan aplikasi mobile paling kritis (Mobile Top 10 - 2024 ) dan umum terjadi. Daftar ini dibuat berdasarkan penelitian dan analisis mendalam terhadap data yang dikumpulkan dari berbagai sumber di seluruh dunia. Dari daftar ini, kita bisa mengetahui ancaman yang paling memiliki dampak besar atau serius pada keamanan aplikasi mobile

Referensi: https://owasp.org/www-project-mobile-top-10/

# M1: Improper Credential Usage

**Deskripsi**

Kerentanan terjadi ketika aplikasi gagal mengelola kredensial pengguna, misalnya dengan menyimpannya pada source code aplikasi, atau menyimpan kredensial pada tempat yang tidak aman. Threat agent dapat mendeteksi lokasi penyimpanan kredensial untuk eksploitasi lebih lanjut.

Kerentanan disebabkan antara lain:

- **Hardcoded Credentials** - Aplikasi menyimpan kredensial pengguna di dalam source code atau file konfigurasi.
- **Insecure Credential Transmission** - Kredensial dikirimkan tanpa enkripsi atau melalui saluran yang tidak aman.
- **Insecure Credential Storage** - Aplikasi menyimpan krendensial pengguna pada perangkat yang tidak diamankan.
- **Weak User Authentication** -Otentikasi pengguna menggunakan protokol yang lemah atau mudah dilewati.

**Dampak**

- teknis - Pengelolaan kredensial yang buruk dapat mengakibatkan dampak teknis yang signifikan. Pihak yang tidak sah dapat mengakses informasi sensitif atau fungsi aplikasi atau sistem backend sehingga menyebabkan kebocoran data, kehilangan privasi, tindakan fraud, atau akses administratif.
- Bisnis - Berdampak signifikan antara lain menurunnya reputatasi, pencurian informasi, fraud dan akses tidak sah terhadap data.

**Mitigasi**

- Tidak menyimpan kredensial pengguna dalam source aplikasi atau file konfigurasi tertentu pada aplikasi/
- Tidak menyimpan kredensial pengguna pada perangkat.
- Menerapkan protokol otentikasi pengguna yang kua.
- Memperbarui dan mengganti API key atau token secara berkala.

# M2: Inadequate Supply Chain Security

# M3: Insecure Authentication/Authorization

# M4: Insufficient Input/Output Validation

# M5: Insecure Communication

# M6: Inadequate Privacy Controls

# M7: Insufficient Binary Protections

# M8: Security Misconfiguration

# M9: Insecure Data Storage

# M10: Insufficient Cryptography