M1: Improper Credential Usage

Threat Agents

Application Specific

Threat agents exploiting hardcoded credentials and improper credential usage in mobile applications can include automated attacks using publicly available or custom-built tools. Such agents could potentially locate and exploit hardcoded credentials or exploit weaknesses due to improper credential usage.

Attack Vectors

Exploitability EASY

Adversaries can exploit vulnerabilities in both hardcoded credentials and improper credential usage. Once these vulnerabilities are identified, an attacker can use hardcoded credentials to gain unauthorized access to sensitive functionalities of the mobile app. They can also misuse credentials, for instance by gaining access through improperly validated or stored credentials, thereby bypassing the need for legitimate access.

Security Weakness

Prevalence COMMON

The OWASP® Foundation works to improve the security of software through its community-led open source software projects, hundreds of chapters worldwide, tens of thousands of members, and by hosting local and global conferences.

Upcoming OWASP Global Events

OWASP Global AppSec EU 2026 - Vienna, Austria

- June 22-26, 2026

OWASP Global AppSec USA 2026 - San Francisco, CA

- November 2-6, 2026

Detectability EASY

Poor implementation of credential management, such as using hardcoded credentials and improper handling, can lead to severe security weaknesses. A comprehensive security testing process should aim to identify these issues. For instance, security testers should attempt to identify hardcoded credentials within the mobile app's source code or within any configuration files.

Technical Impacts

Impact SEVERE

Poor credential management can lead to several significant technical impacts. Unauthorized users might gain access to sensitive information or functionality within the mobile app or its backend systems. This can lead to data breaches, loss of user privacy, fraudulent activity, and potential access to administrative functionality.

Business Impacts

Impact SEVERE

The business impact of poor credential management, including hardcoded credentials and improper credential usage, can be substantial:

- Reputation Damage;
- Information Theft;
- Fraud;
- Unauthorized Access to Data.

Am I Vulnerable To 'Improper Credential Usage'?

Insecure credential management can occur when mobile apps use hardcoded credentials or when credentials are misused. Here are some indicators that your mobile app may be vulnerable:

- **Hardcoded Credentials** - If the mobile app contains hardcoded credentials within the app's source code or any configuration files, this is a clear indicator of vulnerability.
- **Insecure Credential Transmission** - If credentials are transmitted without encryption or through insecure channels, this could indicate a vulnerability.
- **Insecure Credential Storage** - If the mobile app stores user credentials on the device in an insecure manner, this could represent a vulnerability.
- **Weak User Authentication** - If user authentication relies on weak protocols or allows for easy bypassing, this could be a sign of vulnerability.

How Do I Prevent 'Improper Credentials Usage'?

Avoiding insecure credential management involves not using hardcoded credentials and properly handling user credentials.

Avoid Using Hardcoded Credentials

Hardcoded credentials can be easily discovered by attackers and provide an easy access point for unauthorized users. Always avoid using hardcoded credentials in your mobile app's code or configuration files.

Properly Handle User Credentials

User credentials should always be stored, transmitted, and authenticated securely:

- Encrypt credentials during transmission.
- Do not store user credentials on the device. Instead, consider using secure, revocable access tokens.
- Implement strong user authentication protocols.
- Regularly update and rotate any used API keys or tokens.

Example Attack Scenarios

The following scenarios showcase improper credentials usage in mobile apps:

Scenario #1: Hardcoded Credentials: An attacker discovers hardcoded credentials within the mobile app’s source code. They use these credentials to gain unauthorized access to sensitive functionality within the app or backend systems.

Scenario #2: Insecure Credential Transmission: An attacker intercepts insecurely transmitted credentials between the mobile app and its backend systems. They use these intercepted credentials to impersonate a legitimate user and gain unauthorized access.

Scenario #3: Insecure Credential Storage: An attacker gains physical access to a user’s device and extracts stored credentials from the mobile app. The attacker uses these credentials to gain unauthorized access to the user’s account.

References

- OWASP

    - OWASP
- External
    - External References

Edit on GitHub

Spotlight: Secure Code Warrior

Secure Code Warrior is a Developer Risk Management Platform enabling enterprises to implement new standards for secure code throughout the software development lifecycle. With an integrated approach of organizationally benchmarking secure code skills, governance by quality gates integrated during code changes, and upskilling with dynamic agile learning opportunities. Secure Code Warrior allows cybersecurity teams and CISOs to go beyond basic compliance initiatives. Efficiently measure, manage, and mitigate security risk to eliminate introduced vulnerabilities by up to 53%, reduce MTTR by 2x, ease tension between security and engineering teams, and provide continued development opportunities to help retain top talent.

Corporate Supporters

Become a corporate supporter

HOME PROJECTS CHAPTERS EVENTS ABOUT

PRIVACY SITEMAP CONTACT