PENETRATION TEST REPORT

Mobile Application Penetration Test Report

CLIENT	Acme Fintech
ENGAGEMENT ID	demo-mobile
TESTER	Komal Prasad
TARGET	com.acmepay.app (Android)
START DATE	2026-08-03
END DATE	2026-08-06
MODULE	Mobile Pentest
OWASP FRAMEWORK	OWASP Mobile Top 10 — 2024

HIGH 1 | MEDIUM 1 | LOW 1

Table of Contents

1. Executive Summary

During the engagement, a total of **3** findings were identified against com.acmepay.app (Android): 1 high, 1 medium and 1 low.

The most significant issue is **Hardcoded AWS credentials in native library** (High, CVSS 7.1), which should be remediated as a priority.

The overall risk is assessed as **High**. Remediation of the high-severity findings should be scheduled promptly.

Detailed technical findings, evidence, and remediation guidance are provided in the following sections. All testing was performed with appropriate authorization.

Severity Breakdown

Severity	Count	CVSS Range
Critical	0	9.0 – 10.0
High	1	7.0 – 8.9
Medium	1	4.0 – 6.9
Low	1	0.1 – 3.9
Info	0	0.0
TOTAL	**3**	

2. Engagement Overview

Client	Acme Fintech
Engagement ID	demo-mobile
Tester	Komal Prasad
Start Date	2026-08-03
End Date	2026-08-06
Target	com.acmepay.app (Android)
Module	Mobile Pentest
OWASP Framework	OWASP Mobile Top 10 — 2024

3. Scope & Methodology

In-Scope

- com.acmepay.app (Android)
- com.acmepay.app (Android)/docs/
- com.acmepay.app (Android)/api/

Out of Scope

- com.acmepay.app (Android)/admin-panel/

Testing Checklist

Step	Status
Static analysis (manifest/plist, binary protections, permissions)	Complete
Insecure data storage review (local DB, logs, backups, keystore)	Complete
Insecure communication review (TLS, ATS/NSC, certificate pinning)	Complete
Authentication & session handling (local & remote)	Complete
Authorization & access control on device resources	Complete
Input validation (deep links, intents, WebViews, IPC)	Complete
Cryptography review (weak algorithms, key management)	Complete
Binary protections (tampering, debugger, repackaging, obfuscation)	Complete
Supply chain review (SDKs, third-party libraries, dependencies)	Pending
Privacy controls review (data collection, consent, third-party sharing)	Pending
Dynamic / runtime testing (traffic interception, hooking)	Pending
Retesting of confirmed findings	Pending

4. Findings Summary

ID	Title	Severity	CVSS	Status
F-001	Hardcoded AWS credentials in native library	High	7.1	open
F-002	Insecure data storage — plaintext session token in SharedPreferences	Medium	4.4	open
F-003	Certificate pinning not implemented	Low	4.3	open

5. Detailed Findings

F-001 — Hardcoded AWS credentials in native library [High]

Category	M10:2024 — Insufficient Cryptography
Severity	High
CVSS v3.1	7.1 · CVSS:3.1/AV:L/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:N
Status	open
Platform	Android
App Version / Build	v3.2.1 (build 88)
Package / Bundle ID	com.acmepay.app
Component / Screen	libacmepay.so

DESCRIPTION

Plaintext AWS access key and secret were extracted from libacmepay.so during static analysis.

EVIDENCE / PROOF OF CONCEPT

strings libacmepay.so | grep AKIA — AKIAIOSFODNN7EXAMPLE / wJalrXUtnFEMI…

IMPACT

Compromise of the cloud account, data exposure, and resource abuse.

REMEDIATION

Remove embedded secrets; move to a backend-mediated credential service; rotate the exposed keys immediately.

F-002 — Insecure data storage — plaintext session token in SharedPreferences [Medium]

Category	M9:2024 — Insecure Data Storage
Severity	Medium
CVSS v3.1	4.4 · CVSS:3.1/AV:L/AC:L/PR:L/UI:N/S:U/C:H/I:N/A:N
Status	open
Platform	Android
App Version / Build	v3.2.1 (build 88)
Package / Bundle ID	com.acmepay.app
Component / Screen	SharedPreferences/session.xml

DESCRIPTION

The app stores the OAuth refresh token in world-readable SharedPreferences without encryption.

EVIDENCE / PROOF OF CONCEPT

adb shell run-as com.acmepay.app cat shared_prefs/session.xml — token in plaintext.

IMPACT

Token theft on rooted devices or via backup extraction enables account takeover.

REMEDIATION

Store tokens in Android Keystore with hardware-backed encryption; disable backups for sensitive data.

F-003 — Certificate pinning not implemented [Low]

Category	M5:2024 — Insecure Communication
Severity	Low
CVSS v3.1	4.3 · CVSS:3.1/AV:A/AC:L/PR:N/UI:N/S:U/C:L/I:N/A:N
Status	open
Platform	Android
App Version / Build	v3.2.1 (build 88)
Package / Bundle ID	com.acmepay.app
Component / Screen	Network security config

DESCRIPTION

No certificate pinning in OkHttp/TrustKit; the app accepts any CA-issued certificate for api.acmepay.example.

EVIDENCE / PROOF OF CONCEPT

Mitmproxy successfully intercepted TLS traffic with a custom CA on a test device.

IMPACT

Man-in-the-middle interception of API traffic on untrusted networks.

REMEDIATION

Implement certificate pinning for production endpoints with a rotation strategy.

6. Conclusion & Recommendations

The engagement identified 3 finding(s). All findings should be tracked to closure with defined owners and target dates. High and critical findings should be remediated before the application is exposed to untrusted users. A retest should be scheduled once fixes are deployed.

Remediation Priority

Severity	Recommended Timeline
Critical	Immediate (within days)
High	Short term (within 2 weeks)
Medium	Near term (next release)
Low	Scheduled (next cycle)
Info	Informational

Appendix A — OWASP Category Reference

OWASP Mobile Top 10 — 2024

Code	Category
M1:2024	Improper Credential Usage
M2:2024	Inadequate Supply Chain Security
M3:2024	Insecure Authentication/Authorization
M4:2024	Insufficient Input/Output Validation
M5:2024	Insecure Communication
M6:2024	Inadequate Privacy Controls
M7:2024	Insufficient Binary Protections
M8:2024	Security Misconfiguration
M9:2024	Insecure Data Storage
M10:2024	Insufficient Cryptography