ANDROID STATIC ANALYSIS REPORT

app_icon

VulnerableDemoApp (1.0)

File Name:	app-debug.apk
Package Name:	com.example.vulnerabledemoapp
Scan Date:	Feb. 11, 2026, 11:52 a.m.
App Security Score:	33/100 (HIGH RISK)
Grade:	C

FINDINGS SEVERITY

HIGH	MEDIUM	INFO	SECURE	HOTSPOT
4	3	0	1	1

FILE INFORMATION

File Name: app-debug.apk
Size: 5.56MB
MD5: f8ad4994f09d174387abffc597cf5fcb
SHA1: 822fdc3816d854e0b1f1a07dd37c7f3ca0d1a94d
SHA256: aa5934347cac20bd19a021f756224ef3d6c9a2177e5a905333dabf80acb7de96

APP INFORMATION

App Name: VulnerableDemoApp
Package Name: com.example.vulnerabledemoapp
Main Activity: com.example.vulnerabledemoapp.MainActivity
Target SDK: 36
Min SDK: 21
Max SDK:
Android Version Name: 1.0
Android Version Code: 1

APP COMPONENTS

Activities: 4
Services: 0
Receivers: 1
Providers: 1
Exported Activities: 0
Exported Services: 0
Exported Receivers: 1
Exported Providers: 0

CERTIFICATE INFORMATION

Binary is signed
v1 signature: True
v2 signature: True
v3 signature: False
v4 signature: False
X.509 Subject: CN=Android Debug, O=Android, C=US
Signature Algorithm: rsassa_pkcs1v15
Valid From: 2026-02-11 07:40:38+00:00
Valid To: 2056-02-04 07:40:38+00:00
Issuer: CN=Android Debug, O=Android, C=US
Serial Number: 0x1
Hash Algorithm: sha256
md5: 525d6c3c6a909fe0eef4281026f7d4e0
sha1: e0fcd410c660e8855c3beba1cc3c344464afc225
sha256: 375837d557f75b245a8f172fd378bc74e8125e567257b2086cfa4f1667ab2722
sha512: c2e15c0661e380535b1f716cc8951d4f1fee24a9c2652826249084c707096e99e6acfd355cfb7b0c837ef2bcfe10a62e05ea341636752f8240a13b2fc3a4ba1e
PublicKey Algorithm: rsa
Bit Size: 2048
Fingerprint: f00e217f18916af950b011ffcd9a046f18e0a3ecac5106abffe5afecca7e8175
Found 1 unique certificates

APPLICATION PERMISSIONS

PERMISSION	STATUS	INFO	DESCRIPTION
android.permission.ACCESS_FINE_LOCATION	dangerous	fine (GPS) location	Access fine location sources, such as the Global Positioning System on the phone, where available. Malicious applications can use this to determine where you are and may consume additional battery power.
android.permission.ACCESS_COARSE_LOCATION	dangerous	coarse (network-based) location	Access coarse location sources, such as the mobile network database, to determine an approximate phone location, where available. Malicious applications can use this to determine approximately where you are.
android.permission.INTERNET	normal	full Internet access	Allows an application to create network sockets.
com.example.vulnerabledemoapp.DYNAMIC_RECEIVER_NOT_EXPORTED_PERMISSION	unknown	Unknown permission	Unknown permission from android reference

APKID ANALYSIS

FILE	DETAILS

<table>
<tr><th>FILE</th><th>DETAILS</th></tr>
<tr><td>classes3.dex</td><td>
<table>
<tr><th>FINDINGS</th><th>DETAILS</th></tr>
<tr><td>Compiler</td><td>r8</td></tr>
</table>
</td></tr>
<tr><td>classes2.dex</td><td>
<table>
<tr><th>FINDINGS</th><th>DETAILS</th></tr>
<tr><td>Compiler</td><td>unknown (please file detection issue!)</td></tr>
</table>
</td></tr>
<tr><td>classes.dex</td><td>
<table>
<tr><th>FINDINGS</th><th>DETAILS</th></tr>
<tr><td>Anti-VM Code</td><td>Build.FINGERPRINT check
Build.MODEL check
Build.MANUFACTURER check</td></tr>
<tr><td>Compiler</td><td>r8</td></tr>
</table>
</td></tr>
</table>

NETWORK SECURITY

NO	SCOPE	SEVERITY	DESCRIPTION

CERTIFICATE ANALYSIS

HIGH: 1 | WARNING: 1 | INFO: 1

TITLE	SEVERITY	DESCRIPTION
Signed Application	info	Application is signed with a code signing certificate
Application vulnerable to Janus Vulnerability	warning	Application is signed with v1 signature scheme, making it vulnerable to Janus vulnerability on Android 5.0-8.0, if signed only with v1 signature scheme. Applications running on Android 5.0-7.0 signed with v1, and v2/v3 scheme is also vulnerable.
Application signed with debug certificate	high	Application signed with a debug certificate. Production application must not be shipped with a debug certificate.

MANIFEST ANALYSIS

HIGH: 3 | WARNING: 2 | INFO: 0 | SUPPRESSED: 0

NO	ISSUE	SEVERITY	DESCRIPTION
1	App can be installed on a vulnerable unpatched Android version 5.0-5.0.2, [minSdk=21]	high	This application can be installed on an older version of android that has multiple unfixed vulnerabilities. These devices won't receive reasonable security updates from Google. Support an Android version => 10, API 29 to receive reasonable security updates.
2	Clear text traffic is Enabled For App[android:usesCleartextTraffic=true]	high	The app intends to use cleartext network traffic, such as cleartext HTTP, FTP stacks, DownloadManager, and MediaPlayer. The default value for apps that target API level 27 or lower is "true". Apps that target API level 28 or higher default to "false". The key reason for avoiding cleartext traffic is the lack of confidentiality, authenticity, and protections against tampering; a network attacker can eavesdrop on transmitted data and also modify it without being detected.

NO	ISSUE	SEVERITY	DESCRIPTION
3	Debug Enabled For App [android:debuggable=true]	high	Debugging was enabled on the app which makes it easier for reverse engineers to hook a debugger to it. This allows dumping a stack trace and accessing debugging helper classes.
4	Application Data can be Backed up [android:allowBackup=true]	warning	This flag allows anyone to backup your application data via adb. It allows users who have enabled USB debugging to copy application data off of the device.
5	Broadcast Receiver (androidx.profileinstaller.ProfileInstallReceiver) is Protected by a permission, but the protection level of the permission should be checked. Permission: android.permission.DUMP [android:exported=true]	warning	A Broadcast Receiver is found to be shared with other apps on the device therefore leaving it accessible to any other application on the device. It is protected by a permission which is not defined in the analysed application. As a result, the protection level of the permission should be checked where it is defined. If it is set to normal or dangerous, a malicious application can request and obtain the permission and interact with the component. If it is set to signature, only applications signed with the same certificate can obtain the permission.

CODE ANALYSIS

NO	ISSUE	SEVERITY	STANDARDS	FILES

NIAP ANALYSIS v1.3

NO	IDENTIFIER	REQUIREMENT	FEATURE	DESCRIPTION

BEHAVIOUR ANALYSIS

RULE ID	BEHAVIOUR	LABEL	FILES
00096	Connect to a URL and set request method	command network	com/example/vulnerabledemoapp/NetworkActivity.java
00089	Connect to a URL and receive input stream from the server	command network	com/example/vulnerabledemoapp/NetworkActivity.java

ABUSED PERMISSIONS

TYPE	MATCHES	PERMISSIONS
Malware Permissions	3/25	android.permission.ACCESS_FINE_LOCATION, android.permission.ACCESS_COARSE_LOCATION, android.permission.INTERNET
Other Common Permissions	0/44	

Malware Permissions:
Top permissions that are widely abused by known malware.
Other Common Permissions:
Permissions that are commonly abused by known malware.

OFAC SANCTIONED COUNTRIES

This app may communicate with the following OFAC sanctioned list of countries.

DOMAIN	COUNTRY/REGION

DOMAIN MALWARE CHECK

DOMAIN	STATUS	GEOLOCATION
example.com	ok	**IP:** 104.18.26.120 **Country:** United States of America **Region:** California **City:** San Francisco **Latitude:** 37.775700 **Longitude:** -122.395203 **View:** Google Map

SCAN LOGS

Timestamp	Event	Error
2026-02-11 11:52:53	Generating Hashes	OK
2026-02-11 11:52:53	Extracting APK	OK
2026-02-11 11:52:53	Unzipping	OK
2026-02-11 11:52:53	Parsing APK with androguard	OK

2026-02-11 11:52:54	Extracting APK features using aapt/aapt2	OK
2026-02-11 11:52:54	Getting Hardcoded Certificates/Keystores	OK
2026-02-11 11:53:00	Parsing AndroidManifest.xml	OK
2026-02-11 11:53:00	Extracting Manifest Data	OK
2026-02-11 11:53:00	Manifest Analysis Started	OK
2026-02-11 11:53:01	Performing Static Analysis on: VulnerableDemoApp (com.example.vulnerabledemoapp)	OK
2026-02-11 11:53:02	Fetching Details from Play Store: com.example.vulnerabledemoapp	OK
2026-02-11 11:53:02	Checking for Malware Permissions	OK
2026-02-11 11:53:02	Fetching icon path	OK
2026-02-11 11:53:02	Library Binary Analysis Started	OK
2026-02-11 11:53:02	Reading Code Signing Certificate	OK

2026-02-11 11:53:03	Running APKiD 3.0.0	OK
2026-02-11 11:53:05	Detecting Trackers	OK
2026-02-11 11:53:07	Decompiling APK to Java with JADX	OK
2026-02-11 11:53:49	Converting DEX to Smali	OK
2026-02-11 11:53:49	Code Analysis Started on - java_source	OK
2026-02-11 11:53:51	Android SBOM Analysis Completed	OK
2026-02-11 11:54:00	Android SAST Completed	OK
2026-02-11 11:54:00	Android API Analysis Started	OK
2026-02-11 11:54:03	Android API Analysis Completed	OK
2026-02-11 11:54:04	Android Permission Mapping Started	OK
2026-02-11 11:54:06	Android Permission Mapping Completed	OK

2026-02-11 11:54:07	Android Behaviour Analysis Started	OK
2026-02-11 11:54:09	Android Behaviour Analysis Completed	OK
2026-02-11 11:54:09	Extracting Emails and URLs from Source Code	OK
2026-02-11 11:54:10	Email and URL Extraction Completed	OK
2026-02-11 11:54:10	Extracting String data from APK	OK
2026-02-11 11:54:10	Extracting String data from Code	OK
2026-02-11 11:54:10	Extracting String values and entropies from Code	OK
2026-02-11 11:54:11	Performing Malware check on extracted domains	OK
2026-02-11 11:54:13	Saving to Database	OK

Report Generated by - MobSF v4.4.5

Mobile Security Framework (MobSF) is an automated, all-in-one mobile application (Android/iOS/Windows) pen-testing, malware analysis and security assessment framework capable of performing static and dynamic analysis.