



## ANDROID STATIC ANALYSIS REPORT



androide Diva (1.0)

File Name: android.apk

Package Name: jakhar.aseem.diva

Scan Date: Dec. 26, 2025, 7:43 a.m.

App Security Score: **36/100 (HIGH RISK)**

Grade:



# FINDINGS SEVERITY

 HIGH	 MEDIUM	 INFO	 SECURE	 HOTSPOT
5	7	1	1	1

## FILE INFORMATION

**File Name:** android.apk

**Size:** 1.43MB

**MD5:** 82ab8b2193b3cfb1c737e3a786be363a

**SHA1:** 27e849d9d7b86a3a3357fb3e980433a91d416801

**SHA256:** 5cefc51fce9bd760b92ab2340477f4dda84b4ae0c5d04a8c9493e4fe34fab7c5

## APP INFORMATION

**App Name:** Diva

**Package Name:** jakhar.aseem.diva

**Main Activity:** jakhar.aseem.diva.MainActivity

**Target SDK:** 23

**Min SDK:** 15

**Max SDK:**

**Android Version Name:** 1.0

**Android Version Code:** 1

## APP COMPONENTS

Activities: 17

Services: 0

Receivers: 0

Providers: 1

Exported Activities: 2

Exported Services: 0

Exported Receivers: 0

Exported Providers: 1

## CERTIFICATE INFORMATION

Binary is signed

v1 signature: True

v2 signature: False

v3 signature: False

v4 signature: False

X.509 Subject: C=US, O=Android, CN=Android Debug

Signature Algorithm: rsassa\_pkcs1v15

Valid From: 2015-11-02 08:32:11+00:00

Valid To: 2045-10-25 08:32:11+00:00

Issuer: C=US, O=Android, CN=Android Debug

Serial Number: 0x218330df

Hash Algorithm: sha256

md5: d620162ac34ee974d7fd3a1862e7e4df

sha1: ae4ead5aeaba4e9e4fc928e7c7f7fd459f008031

sha256: 35d7f7ad35dfb826b70fa4b73187ed478540e32c8b8c5653b86568029fcd5840

sha512: e936169585893a7a248e393ddb296b2101432de5b07c3d7e2bdc8070dc0b58277b46e443eff3730b4f5a25b61f78c9078f54cc325cb86c17160b5bae13148d1e

Found 1 unique certificates

## APPLICATION PERMISSIONS

PERMISSION	STATUS	INFO	DESCRIPTION
android.permission.WRITE_EXTERNAL_STORAGE	dangerous	read/modify/delete external storage contents	Allows an application to write to external storage.

PERMISSION	STATUS	INFO	DESCRIPTION
android.permission.READ_EXTERNAL_STORAGE	dangerous	read external storage contents	Allows an application to read from external storage.
android.permission.INTERNET	normal	full Internet access	Allows an application to create network sockets.

## /APKID ANALYSIS

FILE	DETAILS	
	FINDINGS	DETAILS
classes.dex	Compiler	dx (possible dexmerge)
	Manipulator Found	dexmerge

## LOCK NETWORK SECURITY

NO	SCOPE	SEVERITY	DESCRIPTION

## CERTIFICATE ANALYSIS

HIGH: 2 | WARNING: 0 | INFO: 1

TITLE	SEVERITY	DESCRIPTION
Signed Application	info	Application is signed with a code signing certificate
Application vulnerable to Janus Vulnerability	high	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: 2 | WARNING: 4 | INFO: 0 | SUPPRESSED: 0

NO	ISSUE	SEVERITY	DESCRIPTION
1	App can be installed on a vulnerable unpatched Android version Android 4.0.3-4.0.4, [minSdk=15]	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	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.
3	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.
4	Activity (jakhar.aseem.diva.APICredsActivity) is not Protected. An intent-filter exists.	warning	An Activity is found to be shared with other apps on the device therefore leaving it accessible to any other application on the device. The presence of intent-filter indicates that the Activity is explicitly exported.

NO	ISSUE	SEVERITY	DESCRIPTION
5	Activity (jakhar.aseem.diva.APIcreds2Activity) is not Protected. An intent-filter exists.	warning	An Activity is found to be shared with other apps on the device therefore leaving it accessible to any other application on the device. The presence of intent-filter indicates that the Activity is explicitly exported.
6	Content Provider (jakhar.aseem.diva.NotesProvider) is not Protected. [android:exported=true]	warning	A Content Provider is found to be shared with other apps on the device therefore leaving it accessible to any other application on the device.

## </> CODE ANALYSIS

HIGH: 1 | WARNING: 3 | INFO: 1 | SECURE: 0 | SUPPRESSED: 0

NO	ISSUE	SEVERITY	STANDARDS	FILES
1	<a href="#">The App logs information. Sensitive information should never be logged.</a>	info	CWE: CWE-532: Insertion of Sensitive Information into Log File OWASP MASVS: MSTG-STORAGE-3	jakhar/aseem/diva/AccessControl1Activity.java jakhar/aseem/diva/AccessControl2Activity.java jakhar/aseem/diva/InsecureDataStorage2Activity.java jakhar/aseem/diva/InsecureDataStorage3Activity.java jakhar/aseem/diva/InsecureDataStorage4Activity.java jakhar/aseem/diva/LogActivity.java jakhar/aseem/diva/SQLInjectionActivity.java
2	<a href="#">App can read/write to External Storage. Any App can read data written to External Storage.</a>	warning	CWE: CWE-276: Incorrect Default Permissions OWASP Top 10: M2: Insecure Data Storage OWASP MASVS: MSTG-STORAGE-2	jakhar/aseem/diva/InsecureDataStorage4Activity.java

NO	ISSUE	SEVERITY	STANDARDS	FILES
3	<a href="#">App uses SQLite Database and execute raw SQL query. Untrusted user input in raw SQL queries can cause SQL Injection. Also sensitive information should be encrypted and written to the database.</a>	warning	CWE: CWE-89: Improper Neutralization of Special Elements used in an SQL Command ('SQL Injection') OWASP Top 10: M7: Client Code Quality	jakhar/aseem/diva/InsecureDataStorage2Activity.java jakhar/aseem/diva/NotesProvider.java jakhar/aseem/diva/SQLInjectionActivity.java
4	<a href="#">Debug configuration enabled. Production builds must not be debuggable.</a>	high	CWE: CWE-919: Weaknesses in Mobile Applications OWASP Top 10: M1: Improper Platform Usage OWASP MASVS: MSTG-RESILIENCE-2	jakhar/aseem/diva/BuildConfig.java
5	App creates temp file. Sensitive information should never be written into a temp file.	warning	CWE: CWE-276: Incorrect Default Permissions OWASP Top 10: M2: Insecure Data Storage OWASP MASVS: MSTG-STORAGE-2	jakhar/aseem/diva/InsecureDataStorage3Activity.java

# SHARED LIBRARY BINARY ANALYSIS

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
----	---------------	----	-----	--------------	-------	-------	---------	---------	------------------

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
1	armeabi-v7a/libdivajni.so	<p>False <b>high</b> The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. Use option --noexecstack or -z noexecstack to mark stack as non executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a> The shared object is build with -fPIC flag which enables Position independent code. This makes Return Oriented Programming (ROP) attacks much more difficult to execute reliably.</p>	<p>False <b>high</b> This binary does not have a stack canary value added to the stack. Stack canaries are used to detect and prevent exploits from overwriting return address. Use the option -fstack-protector-all to enable stack canaries. Not applicable for Dart/Flutter libraries unless Dart FFI is used.</p>	<p>Full RELRO info This shared object has full RELRO enabled. RELRO ensures that the GOT cannot be overwritten in vulnerable ELF binaries. In Full RELRO, the entire GOT (.got and .got.plt both) is marked as read-only.</p>	<p>None <a href="#">info</a> The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a> The binary does not have RUNPATH set.</p>	<p>False <b>warning</b> The binary does not have any fortified functions. Fortified functions provides buffer overflow checks against glibc's commons insecure functions like strcpy, gets etc. Use the compiler option -D_FORTIFY_SOURCE=2 to fortify functions. This check is not applicable for Dart/Flutter libraries.</p>	<p>True <a href="#">info</a> Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
2	arm64-v8a/libdivajni.so	<p>False <b>high</b>            The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. Use option --noexecstack or -z noexecstack to mark stack as non executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a>            The shared object is build with -fPIC flag which enables Position independent code. This makes Return Oriented Programming (ROP) attacks much more difficult to execute reliably.</p>	<p>False <b>high</b>            This binary does not have a stack canary value added to the stack. Stack canaries are used to detect and prevent exploits from overwriting return address. Use the option -fstack-protector-all to enable stack canaries. Not applicable for Dart/Flutter libraries unless Dart FFI is used.</p>	<p>Full RELRO <a href="#">info</a>            This shared object has full RELRO enabled. RELRO ensures that the GOT cannot be overwritten in vulnerable ELF binaries. In Full RELRO, the entire GOT (.got and .got.plt both) is marked as read-only.</p>	<p>None <a href="#">info</a>            The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a>            The binary does not have RUNPATH set.</p>	<p>False <b>warning</b>            The binary does not have any fortified functions. Fortified functions provides buffer overflow checks against glibc's commons insecure functions like strcpy, gets etc. Use the compiler option -D_FORTIFY_SOURCE=2 to fortify functions.            This check is not applicable for Dart/Flutter libraries.</p>	<p>True <a href="#">info</a>            Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
3	mips64/libdivajni.so	<p>False <b>high</b> The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. Use option --noexecstack or -z noexecstack to mark stack as non executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a> The shared object is build with -fPIC flag which enables Position independent code. This makes Return Oriented Programming (ROP) attacks much more difficult to execute reliably.</p>	<p>False <b>high</b> This binary does not have a stack canary value added to the stack. Stack canaries are used to detect and prevent exploits from overwriting return address. Use the option -fstack-protector-all to enable stack canaries. Not applicable for Dart/Flutter libraries unless Dart FFI is used.</p>	<p>Full RELRO info This shared object has full RELRO enabled. RELRO ensures that the GOT cannot be overwritten in vulnerable ELF binaries. In Full RELRO, the entire GOT (.got and .got.plt both) is marked as read-only.</p>	<p>None <a href="#">info</a> The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a> The binary does not have RUNPATH set.</p>	<p>False <b>warning</b> The binary does not have any fortified functions. Fortified functions provides buffer overflow checks against glibc's commons insecure functions like strcpy, gets etc. Use the compiler option -D_FORTIFY_SOURCE=2 to fortify functions. This check is not applicable for Dart/Flutter libraries.</p>	<p>True <a href="#">info</a> Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
4	x86_64/libdivajni.so	<p>False <b>high</b> The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. Use option --noexecstack or -z noexecstack to mark stack as non executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a> The shared object is build with -fPIC flag which enables Position independent code. This makes Return Oriented Programming (ROP) attacks much more difficult to execute reliably.</p>	<p>False <b>high</b> This binary does not have a stack canary value added to the stack. Stack canaries are used to detect and prevent exploits from overwriting return address. Use the option -fstack-protector-all to enable stack canaries. Not applicable for Dart/Flutter libraries unless Dart FFI is used.</p>	<p>Full RELRO info This shared object has full RELRO enabled. RELRO ensures that the GOT cannot be overwritten in vulnerable ELF binaries. In Full RELRO, the entire GOT (.got and .got.plt both) is marked as read-only.</p>	<p>None <a href="#">info</a> The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a> The binary does not have RUNPATH set.</p>	<p>False <b>warning</b> The binary does not have any fortified functions. Fortified functions provides buffer overflow checks against glibc's commons insecure functions like strcpy, gets etc. Use the compiler option -D_FORTIFY_SOURCE=2 to fortify functions. This check is not applicable for Dart/Flutter libraries.</p>	<p>True <a href="#">info</a> Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
5	armeabi/libdivajni.so	<p>False <b>high</b> The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. Use option --noexecstack or -z noexecstack to mark stack as non executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a> The shared object is build with -fPIC flag which enables Position independent code. This makes Return Oriented Programming (ROP) attacks much more difficult to execute reliably.</p>	<p>False <b>high</b> This binary does not have a stack canary value added to the stack. Stack canaries are used to detect and prevent exploits from overwriting return address. Use the option -fstack-protector-all to enable stack canaries. Not applicable for Dart/Flutter libraries unless Dart FFI is used.</p>	<p>Full RELRO info This shared object has full RELRO enabled. RELRO ensures that the GOT cannot be overwritten in vulnerable ELF binaries. In Full RELRO, the entire GOT (.got and .got.plt both) is marked as read-only.</p>	<p>None <a href="#">info</a> The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a> The binary does not have RUNPATH set.</p>	<p>False <b>warning</b> The binary does not have any fortified functions. Fortified functions provides buffer overflow checks against glibc's commons insecure functions like strcpy, gets etc. Use the compiler option -D_FORTIFY_SOURCE=2 to fortify functions. This check is not applicable for Dart/Flutter libraries.</p>	<p>True <a href="#">info</a> Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
6	mips/libdivajni.so	<p>False <b>high</b>            The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. Use option --noexecstack or -z noexecstack to mark stack as non executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a>            The shared object is build with -fPIC flag which enables Position independent code. This makes Return Oriented Programming (ROP) attacks much more difficult to execute reliably.</p>	<p>False <b>high</b>            This binary does not have a stack canary value added to the stack. Stack canaries are used to detect and prevent exploits from overwriting return address. Use the option -fstack-protector-all to enable stack canaries. Not applicable for Dart/Flutter libraries unless Dart FFI is used.</p>	<p>Full RELRO <a href="#">info</a>            This shared object has full RELRO enabled. RELRO ensures that the GOT cannot be overwritten in vulnerable ELF binaries. In Full RELRO, the entire GOT (.got and .got.plt both) is marked as read-only.</p>	<p>None <a href="#">info</a>            The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a>            The binary does not have RUNPATH set.</p>	<p>False <b>warning</b>            The binary does not have any fortified functions. Fortified functions provides buffer overflow checks against glibc's commons insecure functions like strcpy, gets etc. Use the compiler option -D_FORTIFY_SOURCE=2 to fortify functions.            This check is not applicable for Dart/Flutter libraries.</p>	<p>True <a href="#">info</a>            Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
7	x86/libdivajni.so	<p>False <b>high</b>            The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. Use option --noexecstack or -z noexecstack to mark stack as non executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a>            The shared object is build with -fPIC flag which enables Position independent code. This makes Return Oriented Programming (ROP) attacks much more difficult to execute reliably.</p>	<p>True <a href="#">info</a>            This binary has a stack canary value added to the stack so that it will be overwritten by a stack buffer that overflows the return address. This allows detection of overflows by verifying the integrity of the canary before function return.</p>	<p>Full RELRO <a href="#">info</a>            This shared object has full RELRO enabled. RELRO ensures that the GOT cannot be overwritten in vulnerable ELF binaries. In Full RELRO, the entire GOT (.got and .got.plt both) is marked as read-only.</p>	<p>None <a href="#">info</a>            The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a>            The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a>            The binary does not have any fortified functions. Fortified functions provides buffer overflow checks against glibc's commons insecure functions like strcpy, gets etc. Use the compiler option -D_FORTIFY_SOURCE=2 to fortify functions.            This check is not applicable for Dart/Flutter libraries.</p>	<p>True <a href="#">info</a>            Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
8	armeabi-v7a/libdivajni.so	<p>False <b>high</b> The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. Use option --noexecstack or -z noexecstack to mark stack as non executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a> The shared object is build with -fPIC flag which enables Position independent code. This makes Return Oriented Programming (ROP) attacks much more difficult to execute reliably.</p>	<p>False <b>high</b> This binary does not have a stack canary value added to the stack. Stack canaries are used to detect and prevent exploits from overwriting return address. Use the option -fstack-protector-all to enable stack canaries. Not applicable for Dart/Flutter libraries unless Dart FFI is used.</p>	<p>Full RELRO info This shared object has full RELRO enabled. RELRO ensures that the GOT cannot be overwritten in vulnerable ELF binaries. In Full RELRO, the entire GOT (.got and .got.plt both) is marked as read-only.</p>	<p>None <a href="#">info</a> The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a> The binary does not have RUNPATH set.</p>	<p>False <b>warning</b> The binary does not have any fortified functions. Fortified functions provides buffer overflow checks against glibc's commons insecure functions like strcpy, gets etc. Use the compiler option -D_FORTIFY_SOURCE=2 to fortify functions. This check is not applicable for Dart/Flutter libraries.</p>	<p>True <a href="#">info</a> Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
9	arm64-v8a/libdivajni.so	<p>False <b>high</b>            The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. Use option --noexecstack or -z noexecstack to mark stack as non executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a>            The shared object is build with -fPIC flag which enables Position independent code. This makes Return Oriented Programming (ROP) attacks much more difficult to execute reliably.</p>	<p>False <b>high</b>            This binary does not have a stack canary value added to the stack. Stack canaries are used to detect and prevent exploits from overwriting return address. Use the option -fstack-protector-all to enable stack canaries. Not applicable for Dart/Flutter libraries unless Dart FFI is used.</p>	<p>Full RELRO <a href="#">info</a>            This shared object has full RELRO enabled. RELRO ensures that the GOT cannot be overwritten in vulnerable ELF binaries. In Full RELRO, the entire GOT (.got and .got.plt both) is marked as read-only.</p>	<p>None <a href="#">info</a>            The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a>            The binary does not have RUNPATH set.</p>	<p>False <b>warning</b>            The binary does not have any fortified functions. Fortified functions provides buffer overflow checks against glibc's commons insecure functions like strcpy, gets etc. Use the compiler option -D_FORTIFY_SOURCE=2 to fortify functions.            This check is not applicable for Dart/Flutter libraries.</p>	<p>True <a href="#">info</a>            Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
10	mips64/libdivajni.so	<p>False <b>high</b> The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. Use option --noexecstack or -z noexecstack to mark stack as non executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a> The shared object is build with -fPIC flag which enables Position independent code. This makes Return Oriented Programming (ROP) attacks much more difficult to execute reliably.</p>	<p>False <b>high</b> This binary does not have a stack canary value added to the stack. Stack canaries are used to detect and prevent exploits from overwriting return address. Use the option -fstack-protector-all to enable stack canaries. Not applicable for Dart/Flutter libraries unless Dart FFI is used.</p>	<p>Full RELRO info This shared object has full RELRO enabled. RELRO ensures that the GOT cannot be overwritten in vulnerable ELF binaries. In Full RELRO, the entire GOT (.got and .got.plt both) is marked as read-only.</p>	<p>None <a href="#">info</a> The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a> The binary does not have RUNPATH set.</p>	<p>False <b>warning</b> The binary does not have any fortified functions. Fortified functions provides buffer overflow checks against glibc's commons insecure functions like strcpy, gets etc. Use the compiler option -D_FORTIFY_SOURCE=2 to fortify functions. This check is not applicable for Dart/Flutter libraries.</p>	<p>True <a href="#">info</a> Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
11	x86_64/libdivajni.so	<p>False <b>high</b> The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. Use option --noexecstack or -z noexecstack to mark stack as non executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a> The shared object is build with -fPIC flag which enables Position independent code. This makes Return Oriented Programming (ROP) attacks much more difficult to execute reliably.</p>	<p>False <b>high</b> This binary does not have a stack canary value added to the stack. Stack canaries are used to detect and prevent exploits from overwriting return address. Use the option -fstack-protector-all to enable stack canaries. Not applicable for Dart/Flutter libraries unless Dart FFI is used.</p>	<p>Full RELRO info This shared object has full RELRO enabled. RELRO ensures that the GOT cannot be overwritten in vulnerable ELF binaries. In Full RELRO, the entire GOT (.got and .got.plt both) is marked as read-only.</p>	<p>None <a href="#">info</a> The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a> The binary does not have RUNPATH set.</p>	<p>False <b>warning</b> The binary does not have any fortified functions. Fortified functions provides buffer overflow checks against glibc's commons insecure functions like strcpy, gets etc. Use the compiler option -D_FORTIFY_SOURCE=2 to fortify functions. This check is not applicable for Dart/Flutter libraries.</p>	<p>True <a href="#">info</a> Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
12	armeabi/libdivajni.so	<p>False <b>high</b>            The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. Use option --noexecstack or -z noexecstack to mark stack as non executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a>            The shared object is build with -fPIC flag which enables Position independent code. This makes Return Oriented Programming (ROP) attacks much more difficult to execute reliably.</p>	<p>False <b>high</b>            This binary does not have a stack canary value added to the stack. Stack canaries are used to detect and prevent exploits from overwriting return address. Use the option -fstack-protector-all to enable stack canaries. Not applicable for Dart/Flutter libraries unless Dart FFI is used.</p>	<p>Full RELRO <a href="#">info</a>            This shared object has full RELRO enabled. RELRO ensures that the GOT cannot be overwritten in vulnerable ELF binaries. In Full RELRO, the entire GOT (.got and .got.plt both) is marked as read-only.</p>	<p>None <a href="#">info</a>            The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a>            The binary does not have RUNPATH set.</p>	<p>False <b>warning</b>            The binary does not have any fortified functions. Fortified functions provides buffer overflow checks against glibc's commons insecure functions like strcpy, gets etc. Use the compiler option -D_FORTIFY_SOURCE=2 to fortify functions.            This check is not applicable for Dart/Flutter libraries.</p>	<p>True <a href="#">info</a>            Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
13	mips/libdivajni.so	<p>False <b>high</b>            The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. Use option --noexecstack or -z noexecstack to mark stack as non executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a>            The shared object is build with -fPIC flag which enables Position independent code. This makes Return Oriented Programming (ROP) attacks much more difficult to execute reliably.</p>	<p>False <b>high</b>            This binary does not have a stack canary value added to the stack. Stack canaries are used to detect and prevent exploits from overwriting return address. Use the option -fstack-protector-all to enable stack canaries. Not applicable for Dart/Flutter libraries unless Dart FFI is used.</p>	<p>Full RELRO <a href="#">info</a>            This shared object has full RELRO enabled. RELRO ensures that the GOT cannot be overwritten in vulnerable ELF binaries. In Full RELRO, the entire GOT (.got and .got.plt both) is marked as read-only.</p>	<p>None <a href="#">info</a>            The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a>            The binary does not have RUNPATH set.</p>	<p>False <b>warning</b>            The binary does not have any fortified functions. Fortified functions provides buffer overflow checks against glibc's commons insecure functions like strcpy, gets etc. Use the compiler option -D_FORTIFY_SOURCE=2 to fortify functions.            This check is not applicable for Dart/Flutter libraries.</p>	<p>True <a href="#">info</a>            Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
14	x86/libdivajni.so	<p>False <b>high</b>            The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. Use option --noexecstack or -z noexecstack to mark stack as non executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a>            The shared object is build with -fPIC flag which enables Position independent code. This makes Return Oriented Programming (ROP) attacks much more difficult to execute reliably.</p>	<p>True <a href="#">info</a>            This binary has a stack canary value added to the stack so that it will be overwritten by a stack buffer that overflows the return address. This allows detection of overflows by verifying the integrity of the canary before function return.</p>	<p>Full RELRO <a href="#">info</a>            This shared object has full RELRO enabled. RELRO ensures that the GOT cannot be overwritten in vulnerable ELF binaries. In Full RELRO, the entire GOT (.got and .got.plt both) is marked as read-only.</p>	<p>None <a href="#">info</a>            The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a>            The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a>            The binary does not have any fortified functions. Fortified functions provides buffer overflow checks against glibc's commons insecure functions like strcpy, gets etc. Use the compiler option -D_FORTIFY_SOURCE=2 to fortify functions.            This check is not applicable for Dart/Flutter libraries.</p>	<p>True <a href="#">info</a>            Symbols are stripped.</p>

## NIAP ANALYSIS v1.3

NO	IDENTIFIER	REQUIREMENT	FEATURE	DESCRIPTION
----	------------	-------------	---------	-------------

# BEHAVIOUR ANALYSIS

RULE ID	BEHAVIOUR	LABEL	FILES
00022	Open a file from given absolute path of the file	file	jakhar/aseem/diva/InsecureDataStorage4Activity.java

## ABUSED PERMISSIONS

TYPE	MATCHES	PERMISSIONS
Malware Permissions	3/25	android.permission.WRITE_EXTERNAL_STORAGE, android.permission.READ_EXTERNAL_STORAGE, 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
payatu.com	ok	<b>IP:</b> 172.67.72.183 <b>Country:</b> United States of America <b>Region:</b> California <b>City:</b> San Francisco <b>Latitude:</b> 37.775700 <b>Longitude:</b> -122.395203 View: <a href="#">Google Map</a>

## HARDCODED SECRETS

POSSIBLE SECRETS
"pkey" : "notespin"

## SCAN LOGS

Timestamp	Event	Error
2025-12-26 07:43:03	Generating Hashes	OK
2025-12-26 07:43:03	Extracting APK	OK

2025-12-26 07:43:03	Unzipping	OK
2025-12-26 07:43:03	Parsing APK with androguard	OK
2025-12-26 07:43:04	Extracting APK features using aapt/aapt2	OK
2025-12-26 07:43:04	Getting Hardcoded Certificates/Keystores	OK
2025-12-26 07:43:07	Parsing AndroidManifest.xml	OK
2025-12-26 07:43:07	Extracting Manifest Data	OK
2025-12-26 07:43:07	Manifest Analysis Started	OK
2025-12-26 07:43:08	Performing Static Analysis on: Diva (jakhar.aseem.diva)	OK
2025-12-26 07:43:09	Fetching Details from Play Store: jakhar.aseem.diva	OK
2025-12-26 07:43:09	Checking for Malware Permissions	OK
2025-12-26 07:43:09	Fetching icon path	OK

2025-12-26 07:43:09	Library Binary Analysis Started	OK
2025-12-26 07:43:09	Analyzing apktool_out/lib/armeabi-v7a/libdivajni.so	OK
2025-12-26 07:43:09	Analyzing apktool_out/lib/arm64-v8a/libdivajni.so	OK
2025-12-26 07:43:09	Analyzing apktool_out/lib/mips64/libdivajni.so	OK
2025-12-26 07:43:09	Analyzing apktool_out/lib/x86_64/libdivajni.so	OK
2025-12-26 07:43:09	Analyzing apktool_out/lib/armeabi/libdivajni.so	OK
2025-12-26 07:43:09	Analyzing apktool_out/lib/mips/libdivajni.so	OK
2025-12-26 07:43:09	Analyzing apktool_out/lib/x86/libdivajni.so	OK
2025-12-26 07:43:09	Analyzing lib/armeabi-v7a/libdivajni.so	OK
2025-12-26 07:43:09	Analyzing lib/arm64-v8a/libdivajni.so	OK
2025-12-26 07:43:09	Analyzing lib/mips64/libdivajni.so	OK

2025-12-26 07:43:09	Analyzing lib/x86_64/libdivajni.so	OK
2025-12-26 07:43:09	Analyzing lib/armeabi/libdivajni.so	OK
2025-12-26 07:43:09	Analyzing lib/mips/libdivajni.so	OK
2025-12-26 07:43:09	Analyzing lib/x86/libdivajni.so	OK
2025-12-26 07:43:09	Reading Code Signing Certificate	OK
2025-12-26 07:43:10	Running APKiD 3.0.0	OK
2025-12-26 07:43:14	Detecting Trackers	OK
2025-12-26 07:43:15	Decompiling APK to Java with JADX	OK
2025-12-26 07:43:28	Converting DEX to Smali	OK
2025-12-26 07:43:28	Code Analysis Started on - java_source	OK
2025-12-26 07:43:29	Android SBOM Analysis Completed	OK

2025-12-26 07:43:29	Android SAST Completed	OK
2025-12-26 07:43:29	Android API Analysis Started	OK
2025-12-26 07:43:31	Android API Analysis Completed	OK
2025-12-26 07:43:33	Android Permission Mapping Started	OK
2025-12-26 07:43:33	Android Permission Mapping Completed	OK
2025-12-26 07:43:34	Android Behaviour Analysis Started	OK
2025-12-26 07:43:34	Android Behaviour Analysis Completed	OK
2025-12-26 07:43:34	Extracting Emails and URLs from Source Code	OK
2025-12-26 07:43:34	Email and URL Extraction Completed	OK
2025-12-26 07:43:34	Extracting String data from APK	OK
2025-12-26 07:43:34	Extracting String data from SO	OK

2025-12-26 07:43:34	Extracting String data from Code	OK
2025-12-26 07:43:34	Extracting String values and entropies from Code	OK
2025-12-26 07:43:35	Performing Malware check on extracted domains	OK
2025-12-26 07:43:36	Saving to Database	OK

---

## Report Generated by - MobSF v4.4.4

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.

© 2025 Mobile Security Framework - MobSF | [Ajin Abraham](#) | [OpenSecurity](#).