



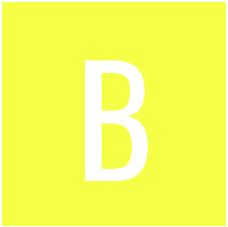
## ANDROID STATIC ANALYSIS REPORT








 Lloyds Bank (62.02)

File Name:

com.grppl.android.shell.CMBllloydsTSB73\_6202\_apps.evozi.com.apk

Package Name:	com.grppl.android.shell.CMBllloydsTSB73
Scan Date:	Nov. 7, 2024, 1:36 a.m.
App Security Score:	42/100 (MEDIUM RISK)
Grade:	
Trackers Detection:	6/432

## FINDINGS SEVERITY

 HIGH	 MEDIUM	 INFO	 SECURE	 HOTSPOT
7	18	3	2	1

## FILE INFORMATION

**File Name:** com.grppl.android.shell.CMBlloydsTSB73\_6202\_apps.evozi.com.apk

**Size:** 73.49MB

**MD5:** 0fba3895f0ed6d46ac7441d4e4702040

**SHA1:** 7f8500dda9063af96f979740c38f315871e62472

**SHA256:** a89bd627e72410b2786d382ca8a55bdb28e783fe296cdc54972cc6479730f0b6

## APP INFORMATION

**App Name:** Lloyds Bank

**Package Name:** com.grppl.android.shell.CMBlloydsTSB73

**Main Activity:**

**Target SDK:** 28

**Min SDK:** 21

**Max SDK:**

**Android Version Name:** 62.02

**Android Version Code:** 6202

## APP COMPONENTS

**Activities:** 54

Services: 11  
Receivers: 6  
Providers: 6  
Exported Activities: 5  
Exported Services: 2  
Exported Receivers: 2  
Exported Providers: 0

## CERTIFICATE INFORMATION

Binary is signed  
v1 signature: True  
v2 signature: True  
v3 signature: True  
v4 signature: False  
X.509 Subject: C=CA, ST=Ontario, L=Toronto, O=grapplemobile.com, OU=Grapple, CN=Grapple  
Signature Algorithm: dsa  
Valid From: 2010-04-12 15:46:27+00:00  
Valid To: 2039-12-31 15:46:27+00:00  
Issuer: C=CA, ST=Ontario, L=Toronto, O=grapplemobile.com, OU=Grapple, CN=Grapple  
Serial Number: 0x4bc34053  
Hash Algorithm: sha1  
md5: 44cc545f1486d6c0c27c8ec544102f4f  
sha1: 915f46b31757f66a927ab733cb0394f3413ac1f1  
sha256: cd5a3e3f58f2085896b72312166dd6e33a427baffa5ea73c7ba513c9fe1b91e2  
sha512: 6006320bc8db7a21670be9861cbe7a9110febac3a42e65e8a9a418801265a9aa6390513bed4675c4165d39ac251625425934f4b4d95c08e5312ba6e3bd0565e9  
PublicKey Algorithm: dsa  
Bit Size: 1024  
Fingerprint: bcde368f32d51961ce27583c893072afa49575f8a54010e95563405cfdd16171  
Found 1 unique certificates

## APPLICATION PERMISSIONS

PERMISSION	STATUS	INFO	DESCRIPTION
com.mobile.lloyds.retail.permission.DEEP_LINK	unknown	Unknown permission	Unknown permission from android reference

PERMISSION	STATUS	INFO	DESCRIPTION
android.permission.CALL_PHONE	dangerous	directly call phone numbers	Allows the application to call phone numbers without your intervention. Malicious applications may cause unexpected calls on your phone bill. Note that this does not allow the application to call emergency numbers.
android.permission.INTERNET	normal	full Internet access	Allows an application to create network sockets.
android.permission.USE_FINGERPRINT	normal	allow use of fingerprint	This constant was deprecated in API level 28. Applications should request USE_BIOMETRIC instead.
android.permission.USE_BIOMETRIC	normal	allows use of device-supported biometric modalities.	Allows an app to use device supported biometric modalities.
android.permission.READ_CONTACTS	dangerous	read contact data	Allows an application to read all of the contact (address) data stored on your phone. Malicious applications can use this to send your data to other people.
android.permission.WRITE_EXTERNAL_STORAGE	dangerous	read/modify/delete external storage contents	Allows an application to write to external storage.
android.permission.CAMERA	dangerous	take pictures and videos	Allows application to take pictures and videos with the camera. This allows the application to collect images that the camera is seeing at any time.
android.permission.READ_EXTERNAL_STORAGE	dangerous	read external storage contents	Allows an application to read from external storage.
android.permission.VIBRATE	normal	control vibrator	Allows the application to control the vibrator.
android.permission.ACCESS_WIFI_STATE	normal	view Wi-Fi status	Allows an application to view the information about the status of Wi-Fi.

PERMISSION	STATUS	INFO	DESCRIPTION
android.permission.ACCESS_NETWORK_STATE	normal	view network status	Allows an application to view the status of all networks.
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.RECEIVE_BOOT_COMPLETED	normal	automatically start at boot	Allows an application to start itself as soon as the system has finished booting. This can make it take longer to start the phone and allow the application to slow down the overall phone by always running.
android.permission.WAKE_LOCK	normal	prevent phone from sleeping	Allows an application to prevent the phone from going to sleep.
com.google.android.finsky.permission.BIND_GET_INSTALL_REFERRER_SERVICE	normal	permission defined by google	A custom permission defined by Google.
com.google.android.c2dm.permission.RECEIVE	normal	recieve push notifications	Allows an application to receive push notifications from cloud.
com.sec.android.provider.badge.permission.READ	normal	show notification count on app	Show notification count or badge on application launch icon for samsung phones.
com.sec.android.provider.badge.permission.WRITE	normal	show notification count on app	Show notification count or badge on application launch icon for samsung phones.
com.htc.launcher.permission.READ_SETTINGS	normal	show notification count on app	Show notification count or badge on application launch icon for htc phones.
com.htc.launcher.permission.UPDATE_SHORTCUT	normal	show notification count on app	Show notification count or badge on application launch icon for htc phones.

PERMISSION	STATUS	INFO	DESCRIPTION
com.sonyericsson.home.permission.BROADCAST_BADGE	normal	show notification count on app	Show notification count or badge on application launch icon for sony phones.
com.sonymobile.home.permission.PROVIDER_INSERT_BADGE	normal	show notification count on app	Show notification count or badge on application launch icon for sony phones.
com.anddoes.launcher.permission.UPDATE_COUNT	normal	show notification count on app	Show notification count or badge on application launch icon for apex.
com.majeur.launcher.permission.UPDATE_BADGE	normal	show notification count on app	Show notification count or badge on application launch icon for solid.
com.huawei.android.launcher.permission.CHANGE_BADGE	normal	show notification count on app	Show notification count or badge on application launch icon for huawei phones.
com.huawei.android.launcher.permission.READ_SETTINGS	normal	show notification count on app	Show notification count or badge on application launch icon for huawei phones.
com.huawei.android.launcher.permission.WRITE_SETTINGS	normal	show notification count on app	Show notification count or badge on application launch icon for huawei phones.
android.permission.READ_APP_BADGE	normal	show app notification	Allows an application to show app icon badges.
com.oppo.launcher.permission.READ_SETTINGS	normal	show notification count on app	Show notification count or badge on application launch icon for oppo phones.
com.oppo.launcher.permission.WRITE_SETTINGS	normal	show notification count on app	Show notification count or badge on application launch icon for oppo phones.
me.everything.badger.permission.BADGE_COUNT_READ	unknown	Unknown permission	Unknown permission from android reference
me.everything.badger.permission.BADGE_COUNT_WRITE	unknown	Unknown permission	Unknown permission from android reference

FILE	DETAILS	
lib/arm64-v8a/libTMXProfiling-6.0-138-jni.so	FINDINGS	DETAILS
	Obfuscator	Arxan
lib/armeabi-v7a/libTMXProfiling-6.0-138-jni.so	FINDINGS	DETAILS
	Obfuscator	Arxan
classes.dex	FINDINGS	DETAILS
	Anti-VM Code	possible Build.SERIAL check
	Obfuscator	Arxan
	Compiler	unknown (please file detection issue!)
classes2.dex	FINDINGS	DETAILS
	Compiler	unknown (please file detection issue!)



FILE	DETAILS	
classes3.dex	FINDINGS	DETAILS
	Compiler	unknown (please file detection issue!)
classes4.dex	FINDINGS	DETAILS
	Obfuscator	Arxan
	Compiler	unknown (please file detection issue!)
classes5.dex	FINDINGS	DETAILS
	Compiler	unknown (please file detection issue!)

## BROWSABLE ACTIVITIES

ACTIVITY	INTENT
com.grppl.android.shell.CMBIloydsTSB73.yiM	Schemes: https://, Hosts: authorise-api.lloydsbank.co.uk, Path Patterns: /prod01/lbg/lyds/personal/oidc-api.*/authorize,
com.grppl.android.shell.CMBIloydsTSB73.WaM	Schemes: @string/deeplink_scheme://, https://, Hosts: @string/accagg_deep_link_host, online.lloydsbank.co.uk, Path Patterns: /ib-access/cwa/aoob/index.html, /ib-access/ob/tpp/pisp,

## NETWORK SECURITY

NO	SCOPE	SEVERITY	DESCRIPTION
----	-------	----------	-------------

## CERTIFICATE ANALYSIS

HIGH: 0 | WARNING: 2 | 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.
Certificate algorithm might be vulnerable to hash collision	warning	Application is signed with SHA1withRSA. SHA1 hash algorithm is known to have collision issues. The manifest file indicates SHA256withRSA is in use.

## MANIFEST ANALYSIS

HIGH: 4 | WARNING: 9 | INFO: 0 | SUPPRESSED: 0

NO	ISSUE	SEVERITY	DESCRIPTION
1	App can be installed on a vulnerable upatched Android version Android 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.

NO	ISSUE	SEVERITY	DESCRIPTION
2	Activity (com.grppl.android.shell.CMBIlloydsTSB73.yiM) 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.
3	App Link assetlinks.json file not found [android:name=com.grppl.android.shell.CMBIlloydsTSB73.WaM] [android:host=https://online.lloydsbank.co.uk]	high	App Link asset verification URL (https://online.lloydsbank.co.uk/.well-known/assetlinks.json) not found or configured incorrectly. (Status Code: None). App Links allow users to redirect from a web URL/email to the mobile app. If this file is missing or incorrectly configured for the App Link host/domain, a malicious app can hijack such URLs. This may lead to phishing attacks, leak sensitive data in the URI, such as PII, OAuth tokens, magic link/password reset tokens and more. You must verify the App Link domain by hosting the assetlinks.json file and enabling verification via [android:autoVerify="true"] in the Activity intent-filter.
4	Activity (com.grppl.android.shell.CMBIlloydsTSB73.WaM) 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.
5	Activity (com.grppl.android.shell.CMBIlloydsTSB73.NOM) is vulnerable to StrandHogg 2.0	high	Activity is found to be vulnerable to StrandHogg 2.0 task hijacking vulnerability. When vulnerable, it is possible for other applications to place a malicious activity on top of the activity stack of the vulnerable application. This makes the application an easy target for phishing attacks. The vulnerability can be remediated by setting the launch mode attribute to "singleInstance" and by setting an empty taskAffinity (taskAffinity=""). You can also update the target SDK version (28) of the app to 29 or higher to fix this issue at platform level.
6	Activity (com.grppl.android.shell.CMBIlloydsTSB73.NOM) is not Protected. [android:exported=true]	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.

NO	ISSUE	SEVERITY	DESCRIPTION
7	Activity-Alias (com.halifax.halifax.GrappleActivity) is not Protected. An intent-filter exists.	warning	An Activity-Alias 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-Alias is explicitly exported.
8	Service (com.google.firebase.messaging.FirebaseMessagingService) is not Protected. [android:exported=true]	warning	A Service is found to be shared with other apps on the device therefore leaving it accessible to any other application on the device.
9	Broadcast Receiver (com.google.android.gms.measurement.AppMeasurementInstallReferrerReceiver) is Protected by a permission, but the protection level of the permission should be checked. Permission: android.permission.INSTALL_PACKAGES [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.
10	Broadcast Receiver (com.google.firebase.iid.FirebaseInstanceIdReceiver) is Protected by a permission, but the protection level of the permission should be checked. Permission: com.google.android.c2dm.permission.SEND [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.
11	Service (com.google.firebase.iid.FirebaseInstanceIdService) is not Protected. [android:exported=true]	warning	A Service is found to be shared with other apps on the device therefore leaving it accessible to any other application on the device.

NO	ISSUE	SEVERITY	DESCRIPTION
12	Activity (com.grppl.android.shell.CMBIlloydsTSB73.VYt) is vulnerable to StrandHogg 2.0	high	Activity is found to be vulnerable to StrandHogg 2.0 task hijacking vulnerability. When vulnerable, it is possible for other applications to place a malicious activity on top of the activity stack of the vulnerable application. This makes the application an easy target for phishing attacks. The vulnerability can be remediated by setting the launch mode attribute to "singleInstance" and by setting an empty taskAffinity (taskAffinity=""). You can also update the target SDK version (28) of the app to 29 or higher to fix this issue at platform level.
13	Activity (com.grppl.android.shell.CMBIlloydsTSB73.VYt) is not Protected. [android:exported=true]	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.

## </> CODE ANALYSIS

HIGH: 2 | WARNING: 6 | INFO: 2 | SECURE: 1 | SUPPRESSED: 0

NO	ISSUE	SEVERITY	STANDARDS	FILES
1	<a href="#">Insecure WebView Implementation. Execution of user controlled code in WebView is a critical Security Hole.</a>	warning	CWE: CWE-749: Exposed Dangerous Method or Function OWASP Top 10: M1: Improper Platform Usage OWASP MASVS: MSTG-PLATFORM-7	com/grppl/android/shell/view/C4731YnZ.java com/grppl/android/shell/view/KHD.java com/qualtrics/digital/QualtricsSurveyFragmen t.java
2	<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	com/grppl/android/shell/view/C12246rRp.java com/grppl/android/shell/view/C1955Kef.java com/grppl/android/shell/view/C2062Kt.java com/grppl/android/shell/view/C3805Ts.java com/grppl/android/shell/view/C5078aGt.java com/grppl/android/shell/view/C6765elp.java com/grppl/android/shell/view/C8918jRp.java com/grppl/android/shell/view/Mlf.java com/grppl/android/shell/view/Xlf.java

NO	ISSUE	SEVERITY	STANDARDS	butterknife/ButterKnife.java Files com/appdynamics/eum/reactnative/InstrumentationModule.java
				com/grppl/android/shell/view/AR.java com/grppl/android/shell/view/AS.java com/grppl/android/shell/view/AbstractC0657Dkt.java com/grppl/android/shell/view/AbstractC10562nPt.java com/grppl/android/shell/view/AbstractC13182tgt.java com/grppl/android/shell/view/AbstractC1514Hwt.java com/grppl/android/shell/view/AbstractC15636zbs.java com/grppl/android/shell/view/AbstractC4844Zc.java com/grppl/android/shell/view/AbstractC5584bRt.java com/grppl/android/shell/view/AbstractC7260fSs.java com/grppl/android/shell/view/AbstractC9788lWt.java com/grppl/android/shell/view/AbstractServiceC15613zZ.java com/grppl/android/shell/view/AbstractServiceConnectionC6470dY.java com/grppl/android/shell/view/AbstractSurfaceHolderCallbackC0977Fbs.java com/grppl/android/shell/view/ActivityC4215Vv.java com/grppl/android/shell/view/ActivityC8271hpM.java com/grppl/android/shell/view/AsyncTaskC6461dWt.java com/grppl/android/shell/view/AsyncTaskC9008jd.java com/grppl/android/shell/view/BE.java com/grppl/android/shell/view/BF.java com/grppl/android/shell/view/BR.java com/grppl/android/shell/view/BZZ.java com/grppl/android/shell/view/C0120App.java com/grppl/android/shell/view/C0125AqZ.java com/grppl/android/shell/view/C0144At.java

NO	ISSUE	SEVERITY	STANDARDS	FILES
				com/grppl/android/shell/view/C0147Atf.java com/grppl/android/shell/view/C0387Ca.java com/grppl/android/shell/view/C0423Cff.java com/grppl/android/shell/view/C0590Dbs.java com/grppl/android/shell/view/C0909Est.java com/grppl/android/shell/view/C0997Fef.java com/grppl/android/shell/view/C10056mDt.java com/grppl/android/shell/view/C10119mM.java com/grppl/android/shell/view/C10142mP.java com/grppl/android/shell/view/C10226mZf.java com/grppl/android/shell/view/C10378mst.java com/grppl/android/shell/view/C10383mtf.java com/grppl/android/shell/view/C10575nRs.java com/grppl/android/shell/view/C10642na.java com/grppl/android/shell/view/C10760np.java com/grppl/android/shell/view/C10899oFt.java com/grppl/android/shell/view/C10977oQ.java com/grppl/android/shell/view/C11058oa.java com/grppl/android/shell/view/C11158omt.java com/grppl/android/shell/view/C11243owt.java com/grppl/android/shell/view/C11365pM.java com/grppl/android/shell/view/C11381pNt.java com/grppl/android/shell/view/C11397pPt.java com/grppl/android/shell/view/C11421pSs.java com/grppl/android/shell/view/C1157Fzt.java com/grppl/android/shell/view/C11623psf.java com/grppl/android/shell/view/C11837qSs.java com/grppl/android/shell/view/C11854qV.java com/grppl/android/shell/view/C11920qct.java com/grppl/android/shell/view/C11932qef.java com/grppl/android/shell/view/C12022qq.java com/grppl/android/shell/view/C12065qv.java com/grppl/android/shell/view/C12217rNs.java com/grppl/android/shell/view/C12258rT.java com/grppl/android/shell/view/C12354rfp.java com/grppl/android/shell/view/C12673sU.java com/grppl/android/shell/view/C12681sV.java

NO	ISSUE	SEVERITY	STANDARDS	FILES
				com/grppl/android/shell/view/C12854sq.java com/grppl/android/shell/view/C12894sw.java com/grppl/android/shell/view/C13038tO.java com/grppl/android/shell/view/C13074tSs.java com/grppl/android/shell/view/C13136tat.java com/grppl/android/shell/view/C13222tlt.java com/grppl/android/shell/view/C13370uDt.java com/grppl/android/shell/view/C13536uZ.java com/grppl/android/shell/view/C13772vBt.java com/grppl/android/shell/view/C13787vDt.java com/grppl/android/shell/view/C13880vPt.java com/grppl/android/shell/view/C13967vat.java com/grppl/android/shell/view/C14069vnf.java com/grppl/android/shell/view/C1431HIZ.java com/grppl/android/shell/view/C14458wjt.java com/grppl/android/shell/view/C14507wqZ.jav a com/grppl/android/shell/view/C14762xX.java com/grppl/android/shell/view/C14964xw.java com/grppl/android/shell/view/C15130yQt.java com/grppl/android/shell/view/C15241yf.java com/grppl/android/shell/view/C15652zdt.java com/grppl/android/shell/view/C15701zk.java com/grppl/android/shell/view/C15791zv.java com/grppl/android/shell/view/C15816zxt.java com/grppl/android/shell/view/C1621lIZ.java com/grppl/android/shell/view/C1674lsZ.java com/grppl/android/shell/view/C1776jfp.java com/grppl/android/shell/view/C1842Jp.java com/grppl/android/shell/view/C2009Klp.java com/grppl/android/shell/view/C2206Lmt.java com/grppl/android/shell/view/C2447Ms.java com/grppl/android/shell/view/C2574Nit.java com/grppl/android/shell/view/C3025Ps.java com/grppl/android/shell/view/C3057Pw.java com/grppl/android/shell/view/C3245Qut.java com/grppl/android/shell/view/C3299Rbs.java com/grppl/android/shell/view/C3307Rct.java com/grppl/android/shell/view/C3314Rdt.java com/grppl/android/shell/view/C3448Rvt.java com/grppl/android/shell/view/C3562Sl.java com/grppl/android/shell/view/C3636Sut.java com/grppl/android/shell/view/C3776Tnt.java com/grppl/android/shell/view/C3777To.java



NO	ISSUE	SEVERITY	STANDARDS	FILES
3	<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	com/grppl/android/shell/view/C3785Tp.java com/grppl/android/shell/view/C3967Unf.java com/grppl/android/shell/view/C4090Vdt.java com/grppl/android/shell/view/C4141Vl.java com/grppl/android/shell/view/C4214Vut.java com/grppl/android/shell/view/C4222Vw.java com/grppl/android/shell/view/C4254Wa.java com/grppl/android/shell/view/C4371Wr.java com/grppl/android/shell/view/C4450Xbs.java com/grppl/android/shell/view/C4548Xot.java com/grppl/android/shell/view/C4564Xr.java com/grppl/android/shell/view/C4600Xvt.java com/grppl/android/shell/view/C4778Ytf.java com/grppl/android/shell/view/C4878Zh.java com/grppl/android/shell/view/C4907Zkt.java com/grppl/android/shell/view/C5271af.java com/grppl/android/shell/view/C5310ajt.java com/grppl/android/shell/view/C5561bOt.java com/grppl/android/shell/view/C5570bQ.java com/grppl/android/shell/view/C5747blt.java com/grppl/android/shell/view/C5954cLt.java com/grppl/android/shell/view/C6173cmt.java com/grppl/android/shell/view/C6292dB.java com/grppl/android/shell/view/C6337dH.java com/grppl/android/shell/view/C6345dl.java com/grppl/android/shell/view/C6482dZf.java com/grppl/android/shell/view/C6613dpt.java com/grppl/android/shell/view/C6791eLt.java com/grppl/android/shell/view/C6814eOt.java com/grppl/android/shell/view/C6829eQt.java com/grppl/android/shell/view/C6844eSs.java com/grppl/android/shell/view/C7174fHt.java com/grppl/android/shell/view/C7314fZf.java com/grppl/android/shell/view/C7580gG.java com/grppl/android/shell/view/C7610glt.java com/grppl/android/shell/view/C7734gZf.java com/grppl/android/shell/view/C7786gft.java com/grppl/android/shell/view/C7794ggt.java com/grppl/android/shell/view/C7931gxt.java com/grppl/android/shell/view/C8006hHs.java com/grppl/android/shell/view/C8063hP.java com/grppl/android/shell/view/C8176hct.java com/grppl/android/shell/view/C8253hn.java com/grppl/android/shell/view/C8301hst.java

NO	ISSUE	SEVERITY	STANDARDS	FILES
				com/grppl/android/shell/view/C8325hvt.java com/grppl/android/shell/view/C8616ift.java com/grppl/android/shell/view/C9039jgt.java com/grppl/android/shell/view/C9101jpM.java com/grppl/android/shell/view/C9272kK.java com/grppl/android/shell/view/C9580kwt.java com/grppl/android/shell/view/C9630lD.java com/grppl/android/shell/view/C9683lK.java com/grppl/android/shell/view/C9830lbt.java com/grppl/android/shell/view/C9987lt.java com/grppl/android/shell/view/CP.java com/grppl/android/shell/view/Cit.java com/grppl/android/shell/view/DH.java com/grppl/android/shell/view/Dlp.java com/grppl/android/shell/view/DSs.java com/grppl/android/shell/view/DialogInterface OnClickListenerC6282d.java com/grppl/android/shell/view/DialogInterface OnClickListenerC7914gw.java com/grppl/android/shell/view/EH.java com/grppl/android/shell/view/EZZ.java com/grppl/android/shell/view/EZt.java com/grppl/android/shell/view/Eft.java com/grppl/android/shell/view/FVt.java com/grppl/android/shell/view/FWp.java com/grppl/android/shell/view/Fit.java com/grppl/android/shell/view/GET.java com/grppl/android/shell/view/GOT.java com/grppl/android/shell/view/GPt.java com/grppl/android/shell/view/GV.java com/grppl/android/shell/view/HK.java com/grppl/android/shell/view/HMt.java com/grppl/android/shell/view/HandlerC1457 Hp.java com/grppl/android/shell/view/HandlerC1844J pM.java com/grppl/android/shell/view/HandlerC8612if f.java com/grppl/android/shell/view/lKt.java com/grppl/android/shell/view/lMt.java com/grppl/android/shell/view/lyt.java com/grppl/android/shell/view/JEt.java com/grppl/android/shell/view/JNs.java com/grppl/android/shell/view/JO.java

NO	ISSUE	SEVERITY	STANDARDS	FILES
				com/grppl/android/shell/view/JRt.java com/grppl/android/shell/view/KG.java com/grppl/android/shell/view/KUt.java com/grppl/android/shell/view/LT.java com/grppl/android/shell/view/LU.java com/grppl/android/shell/view/LayoutInflaterF actory2C0793Ect.java com/grppl/android/shell/view/MDt.java com/grppl/android/shell/view/MZ.java com/grppl/android/shell/view/MZf.java com/grppl/android/shell/view/MenuItemC245 1Msf.java com/grppl/android/shell/view/Mwt.java com/grppl/android/shell/view/NEt.java com/grppl/android/shell/view/NMt.java com/grppl/android/shell/view/NewHomeHub Fragment.java com/grppl/android/shell/view/OF.java com/grppl/android/shell/view/OHs.java com/grppl/android/shell/view/OTt.java com/grppl/android/shell/view/PZf.java com/grppl/android/shell/view/Pd.java com/grppl/android/shell/view/QEt.java com/grppl/android/shell/view/QHt.java com/grppl/android/shell/view/QN.java com/grppl/android/shell/view/Rbt.java com/grppl/android/shell/view/RunnableC088 6Ept.java com/grppl/android/shell/view/RunnableC100 73mG.java com/grppl/android/shell/view/RunnableC111 67oo.java com/grppl/android/shell/view/RunnableC117 34qG.java com/grppl/android/shell/view/RunnableC128 16sm.java com/grppl/android/shell/view/RunnableC132 23tm.java com/grppl/android/shell/view/RunnableC140 41vk.java com/grppl/android/shell/view/RunnableC149 14xq.java com/grppl/android/shell/view/RunnableC179 2Jht.java

NO	ISSUE	SEVERITY	STANDARDS	FILES
				com/grppl/android/shell/view/RunnableC314501.java com/grppl/android/shell/view/RunnableC8119hVt.java com/grppl/android/shell/view/SBp.java com/grppl/android/shell/view/ServiceConnectionC4016Utt.java com/grppl/android/shell/view/ServiceConnectionC7006en.java com/grppl/android/shell/view/TQ.java com/grppl/android/shell/view/UEt.java com/grppl/android/shell/view/ULT.java com/grppl/android/shell/view/UMZ.java com/grppl/android/shell/view/VF.java com/grppl/android/shell/view/VJt.java com/grppl/android/shell/view/VO.java com/grppl/android/shell/view/VR.java com/grppl/android/shell/view/VYt.java com/grppl/android/shell/view/Vbs.java com/grppl/android/shell/view/ViewGroupOnHierarchyChangeListenerC5838bx.java com/grppl/android/shell/view/ViewOnClickListenerC11508pdt.java com/grppl/android/shell/view/ViewOnClickListenerC8147hZZ.java com/grppl/android/shell/view/WB.java com/grppl/android/shell/view/WFt.java com/grppl/android/shell/view/XN.java com/grppl/android/shell/view/YD.java com/grppl/android/shell/view/YJ.java com/grppl/android/shell/view/YU.java com/grppl/android/shell/view/YZ.java com/grppl/android/shell/view/ZDt.java com/grppl/android/shell/view/ZG.java com/grppl/android/shell/view/Zlp.java com/grppl/android/shell/view/ZMf.java com/grppl/android/shell/view/ZSs.java com/grppl/android/shell/view/Za.java com/grppl/android/shell/view/rD.java com/qualtrics/digital/ActionSet.java com/qualtrics/digital/DateExpression.java com/qualtrics/digital/DayExpression.java com/qualtrics/digital/DurationExpression.java com/qualtrics/digital/InterceptDefinition.java

NO	ISSUE	SEVERITY	STANDARDS	FILES
				com/qualtrics/digital/InterceptManager.java com/qualtrics/digital/LatencyReportingService.java com/qualtrics/digital/Properties.java com/qualtrics/digital/Qualtrics.java com/qualtrics/digital/QualtricsPopOverFragment.java com/qualtrics/digital/QualtricsSurveyActivity.java com/qualtrics/digital/QualtricsSurveyExpression.java com/qualtrics/digital/QualtricsSurveyFragment.java com/qualtrics/digital/RequestInterceptor.java com/qualtrics/digital/ServiceInterceptor.java com/qualtrics/digital/SiteInterceptService.java com/qualtrics/digital/TimeExpression.java com/qualtrics/digital/VariableExpression.java com/qualtrics/digital/ViewCountExpression.java com/qualtrics/digital/WebViewInterface.java org/azeckoski/reflectutils/refmap/Finalizer.java org/opencv/android/StaticHelper.java
				com/grppl/android/shell/view/AbstractC1448wns.java com/grppl/android/shell/view/C0271BkM.java com/grppl/android/shell/view/C10048mD.java com/grppl/android/shell/view/C10627nYM.java com/grppl/android/shell/view/C10722nkM.java com/grppl/android/shell/view/C10874oCs.java com/grppl/android/shell/view/C11138okM.java com/grppl/android/shell/view/C11509peD.java com/grppl/android/shell/view/C12234rQD.java com/grppl/android/shell/view/C1286Grt.java com/grppl/android/shell/view/C13156tdZ.java com/grppl/android/shell/view/C13450uOM.java

NO	ISSUE	SEVERITY	STANDARDS	FILES
4	<a href="#">The App uses an insecure Random Number Generator.</a>	warning	CWE: CWE-330: Use of Insufficiently Random Values OWASP Top 10: M5: Insufficient Cryptography OWASP MASVS: MSTG-CRYPTO-6	com/grppl/android/shell/view/C14930xs.java com/grppl/android/shell/view/C15494zKM.java com/grppl/android/shell/view/C2689NyM.java com/grppl/android/shell/view/C4424WyM.java com/grppl/android/shell/view/C4816YyM.java com/grppl/android/shell/view/C5556bOM.java com/grppl/android/shell/view/C5887cDD.java com/grppl/android/shell/view/C6152ckM.java com/grppl/android/shell/view/C6305dCs.java com/grppl/android/shell/view/C6569dkM.java com/grppl/android/shell/view/C7221fNs.java com/grppl/android/shell/view/C7821gkM.java com/grppl/android/shell/view/C8550iXs.java com/grppl/android/shell/view/C8862jKf.java com/grppl/android/shell/view/C9785IWf.java com/grppl/android/shell/view/ExecutorC5214aXs.java com/grppl/android/shell/view/IKM.java com/grppl/android/shell/view/IYs.java com/grppl/android/shell/view/QCs.java com/grppl/android/shell/view/VKM.java com/grppl/android/shell/view/XAs.java com/grppl/android/shell/view/XKM.java com/grppl/android/shell/view/YJf.java com/grppl/android/shell/view/ZYM.java com/qualtrics/digital/SamplingUtil.java
5	<a href="#">App can read/write to External Storage.</a> <a href="#">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	com/grppl/android/shell/view/C0701DqZ.java com/grppl/android/shell/view/C10349mpf.java com/grppl/android/shell/view/C11920qct.java com/grppl/android/shell/view/C4521XIZ.java com/grppl/android/shell/view/C9249kHD.java com/grppl/android/shell/view/MKt.java com/grppl/android/shell/view/NK.java com/grppl/android/shell/view/PSD.java com/grppl/android/shell/view/SurfaceHolderC allbackC2814OpM.java com/grppl/android/shell/view/VHt.java

NO	ISSUE	SEVERITY	STANDARDS	FILES
6	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	com/grppl/android/shell/view/AsyncTaskC0810Eff.java com/grppl/android/shell/view/C1272Gq.java com/grppl/android/shell/view/C13288ttt.java com/grppl/android/shell/view/C1883Juf.java com/grppl/android/shell/view/C6361dJt.java com/grppl/android/shell/view/C6791eLt.java com/grppl/android/shell/view/HED.java com/grppl/android/shell/view/Wyt.java
7	<a href="#">The file or SharedPreferences is World Writable. Any App can write to the file</a>	high	CWE: CWE-276: Incorrect Default Permissions OWASP Top 10: M2: Insecure Data Storage OWASP MASVS: MSTG-STORAGE-2	com/grppl/android/shell/view/C15135yRf.java com/grppl/android/shell/view/C6791eLt.java
8	<a href="#">Ensure that user controlled URLs never reaches the Webview. Enabling file access from URLs in WebView can leak sensitive information from the file system.</a>	warning	CWE: CWE-200: Information Exposure OWASP Top 10: M1: Improper Platform Usage OWASP MASVS: MSTG-PLATFORM-7	com/grppl/android/shell/view/C5875cBZ.java
9	<a href="#">This App copies data to clipboard. Sensitive data should not be copied to clipboard as other applications can access it.</a>	info	OWASP MASVS: MSTG-STORAGE-10	com/grppl/android/shell/view/C9249kHD.java com/grppl/android/shell/view/RunnableC12162rH.java
10	<a href="#">Remote WebView debugging is enabled.</a>	high	CWE: CWE-919: Weaknesses in Mobile Applications OWASP Top 10: M1: Improper Platform Usage OWASP MASVS: MSTG-RESILIENCE-2	com/grppl/android/shell/view/AbstractApplicationC3951UIM.java
11	<a href="#">This App uses SSL certificate pinning to detect or prevent MITM attacks in secure communication channel.</a>	secure	OWASP MASVS: MSTG-NETWORK-4	com/grppl/android/shell/view/C8118hVs.java

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
1	x86_64/libfbjni.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>



NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
2	x86_64/libimagepipeline.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
3	x86_64/libreactnativeblob.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
4	x86_64/libnative-filters.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
5	x86_64/libjscexecutor.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
6	x86_64/libfolly_json.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
7	x86_64/libjsijniProfiler.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
8	x86_64/libreactnativejni.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
9	x86_64/libhermes-executor-debug.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>



NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
10	x86_64/libyoga.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
11	x86_64/libhermes-executor-release.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
12	x86_64/libglog_init.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
13	x86_64/libhermes-inspector.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
14	x86_64/libfolly_futures.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
15	x86_64/libglog.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
16	x86_64/libc++_shared.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
17	x86_64/libfb.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>



NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
18	x86_64/libTMXProfiling-6.0-138-jni.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
19	x86_64/libjsinspector.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
20	x86_64/libnative-imagetranscoder.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
21	x86_64/libjsc.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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>Partial RELRO <a href="#">warning</a></p> <p>This shared object has partial RELRO enabled. RELRO ensures that the GOT cannot be overwritten in vulnerable ELF binaries. In partial RELRO, the non-PLT part of the GOT section is read only but .got.plt is still writeable. Use the option -z,relro,-z,now to enable full RELRO.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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>False <a href="#">warning</a></p> <p>Symbols are available.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
22	armeabi-v7a/libfbjni.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
23	armeabi-v7a/libimagepipeline.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
24	armeabi-v7a/libreactnativeblob.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
25	armeabi-v7a/libnative-filters.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>



NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
26	armeabi-v7a/libjscexecutor.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
27	armeabi-v7a/libfolly_json.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
28	armeabi-v7a/libjsijni profiler.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
29	armeabi-v7a/libreactnativejni.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
30	armeabi-v7a/libhermes-executor-debug.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
31	armeabi-v7a/libyoga.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
32	armeabi-v7a/libtaz_full.4.9.0.5.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
33	armeabi-v7a/libhermes-executor-release.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>



NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
34	armeabi-v7a/libsecure_core.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
35	armeabi-v7a/libtaz_native.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
36	armeabi-v7a/libglog_init.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
37	armeabi-v7a/libhermes-inspector.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
38	armeabi-v7a/libfolly_futures.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
39	armeabi-v7a/libglog.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
40	armeabi-v7a/libc++_shared.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
41	armeabi-v7a/libfb.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>



NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
42	armeabi-v7a/libTMXProfiling-6.0-138-jni.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
43	armeabi-v7a/libjsinspector.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
44	armeabi-v7a/libnative-imagetranscoder.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
45	armeabi-v7a/libmobicheck.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
46	armeabi-v7a/libjsc.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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>Partial RELRO <a href="#">warning</a></p> <p>This shared object has partial RELRO enabled. RELRO ensures that the GOT cannot be overwritten in vulnerable ELF binaries. In partial RELRO, the non-PLT part of the GOT section is read only but .got.plt is still writeable. Use the option -z,relro,-z,now to enable full RELRO.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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>False <a href="#">warning</a></p> <p>Symbols are available.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
47	armeabi-v7a/libopencv_java3.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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 <a href="#">high</a></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
48	x86/libfbjni.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
49	x86/libimagepipeline.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>



NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
50	x86/libreactnativeblob.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
51	x86/libnative-filters.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
52	x86/libjscexecutor.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
53	x86/libfolly_json.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
54	x86/libjsijniprofiler.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
55	x86/libreactnativejni.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
56	x86/libhermes-executor-debug.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
57	x86/libyoga.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>



NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
58	x86/libhermes-executor-release.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
59	x86/libglog_init.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
60	x86/libhermes-inspector.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
61	x86/libfolly_futures.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
62	x86/libglog.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
63	x86/libc++_shared.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
64	x86/libfb.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
65	x86/libTMXProfiling-6.0-138-jni.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>



NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
66	x86/libjsinspector.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
67	x86/libnative-imagetranscoder.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
68	x86/libjsc.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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>Partial RELRO <a href="#">warning</a></p> <p>This shared object has partial RELRO enabled. RELRO ensures that the GOT cannot be overwritten in vulnerable ELF binaries. In partial RELRO, the non-PLT part of the GOT section is read only but .got.plt is still writeable. Use the option -z,relro,-z,now to enable full RELRO.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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>False <a href="#">warning</a></p> <p>Symbols are available.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
69	arm64-v8a/libfbjni.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
70	arm64-v8a/libimagepipeline.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
71	arm64-v8a/libreactnativeblob.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
72	arm64-v8a/libnative-filters.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
73	arm64-v8a/libjscexecutor.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>



NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
74	arm64-v8a/libfolly_json.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
75	arm64-v8a/libjsijniProfiler.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
76	arm64-v8a/libreactnativejni.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
77	arm64-v8a/libhermes-executor-debug.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
78	arm64-v8a/libyoga.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
79	arm64-v8a/libtaz_full.4.9.0.5.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
80	arm64-v8a/libhermes-executor-release.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
81	arm64-v8a/libsecure_core.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>



NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
82	arm64-v8a/libtaz_native.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
83	arm64-v8a/libglog_init.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
84	arm64-v8a/libhermes-inspector.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
85	arm64-v8a/libfolly_futures.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
86	arm64-v8a/libglog.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
87	arm64-v8a/libc++_shared.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
88	arm64-v8a/libfb.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
89	arm64-v8a/libTMXProfiling-6.0-138-jni.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>



NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
90	arm64-v8a/libjsinspector.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
91	arm64-v8a/libnative-imagetranscoder.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
92	arm64-v8a/libmobicheck.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
93	arm64-v8a/libjsc.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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>Partial RELRO <a href="#">warning</a></p> <p>This shared object has partial RELRO enabled. RELRO ensures that the GOT cannot be overwritten in vulnerable ELF binaries. In partial RELRO, the non-PLT part of the GOT section is read only but .got.plt is still writeable. Use the option -z,relro,-z,now to enable full RELRO.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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>False <a href="#">warning</a></p> <p>Symbols are available.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
94	arm64-v8a/libopencv_java3.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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 <a href="#">high</a></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
95	x86_64/libfbjni.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
96	x86_64/libimagepipeline.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
97	x86_64/libreactnativeblob.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>



NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
98	x86_64/libnative-filters.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
99	x86_64/libjscexecutor.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
100	x86_64/libfolly_json.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
101	x86_64/libjsijniProfiler.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
102	x86_64/libreactnativejni.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
103	x86_64/libhermes-executor-debug.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
104	x86_64/libyoga.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
105	x86_64/libhermes-executor-release.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>



NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
106	x86_64/libglog_init.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
107	x86_64/libhermes-inspector.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
108	x86_64/libfolly_futures.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
109	x86_64/libglog.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
110	x86_64/libc++_shared.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
111	x86_64/libfb.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
112	x86_64/libTMXProfiling-6.0-138-jni.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
113	x86_64/libjsinspector.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>



NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
114	x86_64/libnative-imagetranscoder.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
115	x86_64/libjsc.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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>Partial RELRO <a href="#">warning</a></p> <p>This shared object has partial RELRO enabled. RELRO ensures that the GOT cannot be overwritten in vulnerable ELF binaries. In partial RELRO, the non-PLT part of the GOT section is read only but .got.plt is still writeable. Use the option -z,relro,-z,now to enable full RELRO.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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>False <a href="#">warning</a></p> <p>Symbols are available.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
116	armeabi-v7a/libfbjni.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
117	armeabi-v7a/libimagepipeline.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
118	armeabi-v7a/libreactnativeblob.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
119	armeabi-v7a/libnative-filters.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
120	armeabi-v7a/libjscexecutor.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
121	armeabi-v7a/libfolly_json.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>



NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
122	armeabi-v7a/libjsijni profiler.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
123	armeabi-v7a/libreactnativejni.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
124	armeabi-v7a/libhermes-executor-debug.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
125	armeabi-v7a/libyoga.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
126	armeabi-v7a/libtaz_full.4.9.0.5.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
127	armeabi-v7a/libhermes-executor-release.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
128	armeabi-v7a/libsecure_core.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
129	armeabi-v7a/libtaz_native.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>



NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
130	armeabi-v7a/libglog_init.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
131	armeabi-v7a/libhermes-inspector.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
132	armeabi-v7a/libfolly_futures.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
133	armeabi-v7a/libglog.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
134	armeabi-v7a/libc++_shared.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
135	armeabi-v7a/libfb.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
136	armeabi-v7a/libTMXProfiling-6.0-138-jni.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
137	armeabi-v7a/libjsinspector.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>



NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
138	armeabi-v7a/libnative-imagetranscoder.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
139	armeabi-v7a/libmobicheck.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
140	armeabi-v7a/libjsc.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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>Partial RELRO <a href="#">warning</a></p> <p>This shared object has partial RELRO enabled. RELRO ensures that the GOT cannot be overwritten in vulnerable ELF binaries. In partial RELRO, the non-PLT part of the GOT section is read only but .got.plt is still writeable. Use the option -z,relro,-z,now to enable full RELRO.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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>False <a href="#">warning</a></p> <p>Symbols are available.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
141	armeabi-v7a/libopencv_java3.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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 <a href="#">high</a></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
142	x86/libfbjni.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
143	x86/libimagepipeline.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
144	x86/libreactnativeblob.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
145	x86/libnative-filters.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>



NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
146	x86/libjscexecutor.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
147	x86/libfolly_json.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
148	x86/libjsijniprofiler.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
149	x86/libreactnativejni.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
150	x86/libhermes-executor-debug.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
151	x86/libyoga.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
152	x86/libhermes-executor-release.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
153	x86/libglog_init.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>



NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
154	x86/libhermes-inspector.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
155	x86/libfolly_futures.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
156	x86/libglog.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
157	x86/libc++_shared.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
158	x86/libfb.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
159	x86/libTMXProfiling-6.0-138-jni.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
160	x86/libjsinspector.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
161	x86/libnative-imagetranscoder.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>



NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
162	x86/libjsc.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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>Partial RELRO <a href="#">warning</a></p> <p>This shared object has partial RELRO enabled. RELRO ensures that the GOT cannot be overwritten in vulnerable ELF binaries. In partial RELRO, the non-PLT part of the GOT section is read only but .got.plt is still writeable. Use the option -z,relro,-z,now to enable full RELRO.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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>False <a href="#">warning</a></p> <p>Symbols are available.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
163	arm64-v8a/libfbjni.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
164	arm64-v8a/libimagepipeline.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
165	arm64-v8a/libreactnativeblob.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
166	arm64-v8a/libnative-filters.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
167	arm64-v8a/libjscexecutor.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
168	arm64-v8a/libfolly_json.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
169	arm64-v8a/libjsijni profiler.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>



NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
170	arm64-v8a/libreactnativejni.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
171	arm64-v8a/libhermes-executor-debug.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
172	arm64-v8a/libyoga.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
173	arm64-v8a/libtaz_full.4.9.0.5.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
174	arm64-v8a/libhermes-executor-release.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
175	arm64-v8a/libsecure_core.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
176	arm64-v8a/libtaz_native.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
177	arm64-v8a/libglog_init.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>



NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
178	arm64-v8a/libhermes-inspector.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
179	arm64-v8a/libfolly_futures.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
180	arm64-v8a/libglog.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
181	arm64-v8a/libc++_shared.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
182	arm64-v8a/libfb.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
183	arm64-v8a/libTMXProfiling-6.0-138-jni.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
184	arm64-v8a/libjsinspector.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
185	arm64-v8a/libnative-imagetranscoder.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>



NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
186	arm64-v8a/libmobicheck.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
187	arm64-v8a/libjsc.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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></p> <p>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>Partial RELRO <a href="#">warning</a></p> <p>This shared object has partial RELRO enabled. RELRO ensures that the GOT cannot be overwritten in vulnerable ELF binaries. In partial RELRO, the non-PLT part of the GOT section is read only but .got.plt is still writeable. Use the option -z,relro,-z,now to enable full RELRO.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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>False <a href="#">warning</a></p> <p>Symbols are available.</p>

NO	SHARED OBJECT	NX	PIE	STACK CANARY	RELRO	RPATH	RUNPATH	FORTIFY	SYMBOLS STRIPPED
188	arm64-v8a/libopencv_java3.so	<p>True <a href="#">info</a></p> <p>The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.</p>	<p>Dynamic Shared Object (DSO) <a href="#">info</a></p> <p>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 <a href="#">high</a></p> <p>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></p> <p>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></p> <p>The binary does not have run-time search path or RPATH set.</p>	<p>None <a href="#">info</a></p> <p>The binary does not have RUNPATH set.</p>	<p>False <a href="#">warning</a></p> <p>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></p> <p>Symbols are stripped.</p>

# NIAP ANALYSIS v1.3

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

## FIREBASE DATABASES ANALYSIS

TITLE	SEVERITY	DESCRIPTION
App talks to a Firebase database	info	The app talks to Firebase database at <a href="https://nga3-cloud-messaging.firebaseio.com">https://nga3-cloud-messaging.firebaseio.com</a>
Firebase Remote Config disabled	secure	Firebase Remote Config is disabled for <a href="https://firebaseremoteconfig.googleapis.com/v1/projects/694308978164/namespaces/firebase:fetch?key=AlzaSyDFOLGzB6B0_daP5W6a-qWcnLQ7wf3sWm4">https://firebaseremoteconfig.googleapis.com/v1/projects/694308978164/namespaces/firebase:fetch?key=AlzaSyDFOLGzB6B0_daP5W6a-qWcnLQ7wf3sWm4</a> . This is indicated by the response: The response code is 403

## ABUSED PERMISSIONS

TYPE	MATCHES	PERMISSIONS
Malware Permissions	11/24	android.permission.INTERNET, android.permission.READ_CONTACTS, android.permission.WRITE_EXTERNAL_STORAGE, android.permission.CAMERA, android.permission.READ_EXTERNAL_STORAGE, android.permission.VIBRATE, android.permission.ACCESS_WIFI_STATE, android.permission.ACCESS_NETWORK_STATE, android.permission.ACCESS_FINE_LOCATION, android.permission.RECEIVE_BOOT_COMPLETED, android.permission.WAKE_LOCK
Other Common Permissions	3/45	android.permission.CALL_PHONE, com.google.android.finsky.permission.BIND_GET_INSTALL_REFERRER_SERVICE, com.google.android.c2dm.permission.RECEIVE

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
www.halifaxurcatravelinsurance.co.uk	ok	IP: 85.159.154.30 Country: United Kingdom of Great Britain and Northern Ireland Region: England City: London Latitude: 51.508530 Longitude: -0.125740 View: <a href="#">Google Map</a>
www.bankofscotland.co.uk	ok	IP: 23.2.212.211 Country: Netherlands Region: Noord-Holland City: Amsterdam Latitude: 52.374031 Longitude: 4.889690 View: <a href="#">Google Map</a>

DOMAIN	STATUS	GEOLOCATION
www.google.com	ok	<b>IP:</b> 142.251.36.36 <b>Country:</b> United States of America <b>Region:</b> California <b>City:</b> Mountain View <b>Latitude:</b> 37.405991 <b>Longitude:</b> -122.078514 <b>View:</b> <a href="#">Google Map</a>
www.lloydsbank.com	ok	<b>IP:</b> 23.2.212.211 <b>Country:</b> Netherlands <b>Region:</b> Noord-Holland <b>City:</b> Amsterdam <b>Latitude:</b> 52.374031 <b>Longitude:</b> 4.889690 <b>View:</b> <a href="#">Google Map</a>
nga3-cloud-messaging.firebaseio.com	ok	<b>IP:</b> 34.120.206.254 <b>Country:</b> United States of America <b>Region:</b> Missouri <b>City:</b> Kansas City <b>Latitude:</b> 39.099731 <b>Longitude:</b> -94.578568 <b>View:</b> <a href="#">Google Map</a>
survey.euro.confermit.com	ok	<b>IP:</b> 212.100.250.8 <b>Country:</b> United Kingdom of Great Britain and Northern Ireland <b>Region:</b> England <b>City:</b> London <b>Latitude:</b> 51.508530 <b>Longitude:</b> -0.125740 <b>View:</b> <a href="#">Google Map</a>

DOMAIN	STATUS	GEOLOCATION
lloydsbankinggroup.eu.qualtrics.com	ok	<b>IP:</b> 104.123.41.128 <b>Country:</b> United States of America <b>Region:</b> Massachusetts <b>City:</b> Cambridge <b>Latitude:</b> 42.363598 <b>Longitude:</b> -71.085205 <b>View:</b> <a href="#">Google Map</a>
www.google.co.uk	ok	<b>IP:</b> 142.251.39.99 <b>Country:</b> United States of America <b>Region:</b> California <b>City:</b> Mountain View <b>Latitude:</b> 37.405991 <b>Longitude:</b> -122.078514 <b>View:</b> <a href="#">Google Map</a>
static.halifax.co.uk	ok	<b>IP:</b> 23.197.155.80 <b>Country:</b> United States of America <b>Region:</b> Florida <b>City:</b> Miami <b>Latitude:</b> 25.774269 <b>Longitude:</b> -80.193657 <b>View:</b> <a href="#">Google Map</a>
www.moneyadviceservice.org.uk	ok	<b>IP:</b> 3.33.249.164 <b>Country:</b> United States of America <b>Region:</b> Washington <b>City:</b> Seattle <b>Latitude:</b> 47.627499 <b>Longitude:</b> -122.346199 <b>View:</b> <a href="#">Google Map</a>

DOMAIN	STATUS	GEOLOCATION
register.fca.org.uk	ok	<b>IP:</b> 104.17.107.103 <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 <b>View:</b> <a href="#">Google Map</a>
www.scottishwidows.co.uk	ok	<b>IP:</b> 2.16.6.24 <b>Country:</b> Germany <b>Region:</b> Hessen <b>City:</b> Frankfurt am Main <b>Latitude:</b> 50.115520 <b>Longitude:</b> 8.684170 <b>View:</b> <a href="#">Google Map</a>
online.lloydsbank.co.uk	ok	<b>IP:</b> 23.197.158.232 <b>Country:</b> United States of America <b>Region:</b> Florida <b>City:</b> Miami <b>Latitude:</b> 25.774269 <b>Longitude:</b> -80.193657 <b>View:</b> <a href="#">Google Map</a>
www.pensionwise.gov.uk	ok	<b>IP:</b> 104.22.43.197 <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 <b>View:</b> <a href="#">Google Map</a>



DOMAIN	STATUS	GEOLOCATION
www.youtube.com	ok	<b>IP:</b> 142.250.179.206 <b>Country:</b> United States of America <b>Region:</b> California <b>City:</b> Mountain View <b>Latitude:</b> 37.405991 <b>Longitude:</b> -122.078514 <b>View:</b> <a href="#">Google Map</a>
www.apache.org	ok	<b>IP:</b> 151.101.2.132 <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 <b>View:</b> <a href="#">Google Map</a>
www.lloydsbankinggroup.com	ok	<b>IP:</b> 23.2.212.211 <b>Country:</b> Netherlands <b>Region:</b> Noord-Holland <b>City:</b> Amsterdam <b>Latitude:</b> 52.374031 <b>Longitude:</b> 4.889690 <b>View:</b> <a href="#">Google Map</a>

## EMAILS

EMAIL	FILE
n0joc@d.gvwfml	com/grppl/android/shell/view/C10496nHp.java
l@46.g04597a441	com/grppl/android/shell/view/C13092tV.java

EMAIL	FILE
td77@o.96864	com/grppl/android/shell/view/C5139aOp.java
9@dnd.84	com/grppl/android/shell/view/C15347ysZ.java
2@39-gk9rh6a.522	com/grppl/android/shell/view/C15351yss.java
nk@idt.gvwfml	com/grppl/android/shell/view/C2936Pft.java
df@kgs.riucvmykwn 5sb@5r.mnu a@_v19xp.mhr u0016@iti.ohz6 u001eb@_mp.6uy96 km@ae3j.fsbhcxgk yp@du6t.w19xn	com/grppl/android/shell/view/C4510Xjt.java
td77@o.96864	com/grppl/android/shell/view/C11152omD.java
46@l01.007bd 7-@d.825	com/grppl/android/shell/view/C7848go.java
ggm@jqq.oqvlr	com/grppl/android/shell/view/UPD.java
u0013a3.@0.pp	com/grppl/android/shell/view/ComponentCallbacksC6298dBt.java
n0joc@d.gvwfml	com/grppl/android/shell/view/BHp.java
u0013cb0g@y.99a	com/grppl/android/shell/view/C7949hAD.java
k@hoo.brtr06	com/grppl/android/shell/view/C8590ics.java
6-i7-@b.c6di9g5j	com/grppl/android/shell/view/C14953xup.java
u0019ed@bn1.87	com/grppl/android/shell/view/C11571pmD.java

EMAIL	FILE
mobileapps@lloydsbanking.com mobilecolleaguebeta@lloydsbanking.com	Android String Resource

## TRACKERS

TRACKER	CATEGORIES	URL
Adobe Experience Cloud		<a href="https://reports.exodus-privacy.eu.org/trackers/229">https://reports.exodus-privacy.eu.org/trackers/229</a>
Appdynamics	Profiling, Analytics	<a href="https://reports.exodus-privacy.eu.org/trackers/194">https://reports.exodus-privacy.eu.org/trackers/194</a>
Google Analytics	Analytics	<a href="https://reports.exodus-privacy.eu.org/trackers/48">https://reports.exodus-privacy.eu.org/trackers/48</a>
Google Firebase Analytics	Analytics	<a href="https://reports.exodus-privacy.eu.org/trackers/49">https://reports.exodus-privacy.eu.org/trackers/49</a>
Google Tag Manager	Analytics	<a href="https://reports.exodus-privacy.eu.org/trackers/105">https://reports.exodus-privacy.eu.org/trackers/105</a>
Qualtrics		<a href="https://reports.exodus-privacy.eu.org/trackers/306">https://reports.exodus-privacy.eu.org/trackers/306</a>

## HARDCODED SECRETS

POSSIBLE SECRETS
"in_app_search_keyword_authorised" : "authorised"
"google_api_key" : "AlzaSyDFOLGzB6B0_daP5W6a-qWcnLQ7wf3sWm4"
"liberty_api_channel_header" : "IBL"

POSSIBLE SECRETS
"cwa_radio_auth" : "Auth"
"google_crash_reporting_api_key" : "AlzaSyDFOLGzB6B0_daP5W6a-qWcnLQ7wf3sWm4"
"firebase_database_url" : "https://nga3-cloud-messaging.firebaseio.com"
"lp_google_maps_key" : ""
"liberty_api_brand_header" : ""
"pre_auth_menu_item_font" : ""
"failedToInitalizeSession" : ""
"in_app_search_keyword_authority" : "authority"

## PLAYSTORE INFORMATION

**Title:** Lloyds Mobile Banking

**Score:** 4.55 **Installs:** 10,000,000+ **Price:** 0 **Android Version Support:** Category: Finance **Play Store URL:** [com.grppl.android.shell.CMBllloydsTSB73](http://com.grppl.android.shell.CMBllloydsTSB73)

**Developer Details:** Lloyds Banking Group PLC, Lloyds+Banking+Group+PLC, None, <http://www.lloydsbank.com/app>, [mobileapps@lloydsbanking.com](mailto:mobileapps@lloydsbanking.com),

**Release Date:** May 18, 2016 **Privacy Policy:** [Privacy link](#)

### Description:

Welcome to the Lloyds Mobile Banking app. Your money on the go. Easy access • Fast log on with fingerprint logon, or memorable information. • Summary view of all your accounts. • Discover and apply for new products in the app. • View your PIN, freeze your card, and more in the app. • Get around the app with the tabs at the top of your screen. Brilliant everyday • See, spend and manage your current and credit card accounts. • Block and cancel subscriptions. • Stay on top of your money with tools and spending insights. • Know what's ahead with upcoming payments. A total view of your savings & investments • Set savings goals and build a safety net. • Plan ahead with a handy new calculator. • Useful reads to help you save for the future. Stay on top of your borrowing • Free credit score checks that give you a view of your financial well-being. • Know what you need to pay and when on your credit card. • Check and manage your credit limit. • Keep track of your personal loan. All your home needs in one place • View your mortgage balance and home value. • Discover potential savings on your bills and compare deals. • Browse rewards for sustainable home improvements. Protect what's important • View your home, car and life insurance cover. • Explore your cover options and request a quote. Getting started You'll need: • The phone number that you've registered with us. • A Lloyds personal, Lloyds Bank Islands personal, or Sterling

international account. Keeping you safe online We use the latest online security measures to protect your money, your personal information and your privacy. How we'll contact you Using our app won't affect how we contact you. Our emails will address you by your title and surname, and include the last four digits of your account number, or the last three digits of your postcode. Any texts we send will come from LLOYDSBANK. Be wary of any message that differs from this – it could be a scam. Important information We don't charge you to use our services but your mobile operator might charge for some things (such as downloading or using the app), so please check with them. Services may be affected by phone signal and functionality. When you use this app we collect anonymous location data to help combat fraud, fix bugs and improve future services. Our app is available to customers with a UK personal account and valid registered phone number. Minimum operating systems apply, so check the Google Play for details. Device registration required. Terms and conditions apply. Freeze and unfreeze certain types of transactions securely 24/7 for extra peace of mind or if you've temporarily misplaced your card. Fingerprint Logon requires a compatible mobile running Android 7.0 or higher and may not work on some tablets. Lloyds and Lloyds Bank are trading names of Lloyds Bank plc (registered in England and Wales (no. 2065), registered office: 25 Gresham Street, London EC2V 7HN). Authorised by the Prudential Regulation Authority and regulated by the Financial Conduct Authority and the Prudential Regulation Authority under registration number 119278.

## SCAN LOGS

Timestamp	Event	Error
2024-11-07 01:36:13	Generating Hashes	OK
2024-11-07 01:36:13	Extracting APK	OK
2024-11-07 01:36:13	Unzipping	OK
2024-11-07 01:36:15	Getting Hardcoded Certificates/Keystores	OK
2024-11-07 01:36:20	Parsing AndroidManifest.xml	OK
2024-11-07 01:36:20	Parsing APK with androguard	OK
2024-11-07 01:36:21	Extracting Manifest Data	OK

2024-11-07 01:36:21	Performing Static Analysis on: Lloyds Bank (com.grppl.android.shell.CMBllloydsTSB73)	OK
2024-11-07 01:36:21	Fetching Details from Play Store: com.grppl.android.shell.CMBllloydsTSB73	OK
2024-11-07 01:36:21	Manifest Analysis Started	OK
2024-11-07 01:36:22	Checking for Malware Permissions	OK
2024-11-07 01:36:22	Fetching icon path	OK
2024-11-07 01:36:22	Library Binary Analysis Started	OK
2024-11-07 01:36:22	Analyzing lib/x86_64/libfbjni.so	OK
2024-11-07 01:36:22	Analyzing lib/x86_64/libimagepipeline.so	OK
2024-11-07 01:36:22	Analyzing lib/x86_64/libreactnativeblob.so	OK
2024-11-07 01:36:22	Analyzing lib/x86_64/libnative-filters.so	OK
2024-11-07 01:36:22	Analyzing lib/x86_64/libjscexecutor.so	OK

2024-11-07 01:36:22	Analyzing lib/x86_64/libfolly_json.so	OK
2024-11-07 01:36:22	Analyzing lib/x86_64/libjsijniprofiler.so	OK
2024-11-07 01:36:22	Analyzing lib/x86_64/libreactnativejni.so	OK
2024-11-07 01:36:22	Analyzing lib/x86_64/libhermes-executor-debug.so	OK
2024-11-07 01:36:22	Analyzing lib/x86_64/libyoga.so	OK
2024-11-07 01:36:22	Analyzing lib/x86_64/libhermes-executor-release.so	OK
2024-11-07 01:36:22	Analyzing lib/x86_64/libglog_init.so	OK
2024-11-07 01:36:22	Analyzing lib/x86_64/libhermes-inspector.so	OK
2024-11-07 01:36:22	Analyzing lib/x86_64/libfolly_futures.so	OK
2024-11-07 01:36:22	Analyzing lib/x86_64/libglog.so	OK
2024-11-07 01:36:22	Analyzing lib/x86_64/libc++_shared.so	OK

2024-11-07 01:36:22	Analyzing lib/x86_64/libfb.so	OK
2024-11-07 01:36:22	Analyzing lib/x86_64/libTMXProfiling-6.0-138-jni.so	OK
2024-11-07 01:36:22	Analyzing lib/x86_64/libjsinspector.so	OK
2024-11-07 01:36:22	Analyzing lib/x86_64/libnative-image-transcoder.so	OK
2024-11-07 01:36:22	Analyzing lib/x86_64/libjsc.so	OK
2024-11-07 01:36:22	Analyzing lib/armeabi-v7a/libfbjni.so	OK
2024-11-07 01:36:23	Analyzing lib/armeabi-v7a/libimagepipeline.so	OK
2024-11-07 01:36:23	Analyzing lib/armeabi-v7a/libreactnativeblob.so	OK
2024-11-07 01:36:23	Analyzing lib/armeabi-v7a/libnative-filters.so	OK
2024-11-07 01:36:23	Analyzing lib/armeabi-v7a/libjscexecutor.so	OK
2024-11-07 01:36:23	Analyzing lib/armeabi-v7a/libfolly_json.so	OK



2024-11-07 01:36:23	Analyzing lib/armeabi-v7a/libjsjniprofiler.so	OK
2024-11-07 01:36:23	Analyzing lib/armeabi-v7a/libreactnativejni.so	OK
2024-11-07 01:36:23	Analyzing lib/armeabi-v7a/libhermes-executor-debug.so	OK
2024-11-07 01:36:23	Analyzing lib/armeabi-v7a/libyoga.so	OK
2024-11-07 01:36:23	Analyzing lib/armeabi-v7a/libtaz_full.4.9.0.5.so	OK
2024-11-07 01:36:23	Analyzing lib/armeabi-v7a/libhermes-executor-release.so	OK
2024-11-07 01:36:23	Analyzing lib/armeabi-v7a/libsecure_core.so	OK
2024-11-07 01:36:23	Analyzing lib/armeabi-v7a/libtaz_native.so	OK
2024-11-07 01:36:23	Analyzing lib/armeabi-v7a/libglog_init.so	OK
2024-11-07 01:36:23	Analyzing lib/armeabi-v7a/libhermes-inspector.so	OK
2024-11-07 01:36:23	Analyzing lib/armeabi-v7a/libfolly_futures.so	OK

2024-11-07 01:36:23	Analyzing lib/armeabi-v7a/libglog.so	OK
2024-11-07 01:36:23	Analyzing lib/armeabi-v7a/libc++_shared.so	OK
2024-11-07 01:36:23	Analyzing lib/armeabi-v7a/libfb.so	OK
2024-11-07 01:36:23	Analyzing lib/armeabi-v7a/libTMXProfiling-6.0-138-jni.so	OK
2024-11-07 01:36:23	Analyzing lib/armeabi-v7a/libjsinspector.so	OK
2024-11-07 01:36:23	Analyzing lib/armeabi-v7a/libnative-image-transcoder.so	OK
2024-11-07 01:36:23	Analyzing lib/armeabi-v7a/libmobilecheck.so	OK
2024-11-07 01:36:23	Analyzing lib/armeabi-v7a/libjsc.so	OK
2024-11-07 01:36:24	Analyzing lib/armeabi-v7a/libopencv_java3.so	OK
2024-11-07 01:36:24	Analyzing lib/x86/libfbjni.so	OK
2024-11-07 01:36:24	Analyzing lib/x86/libimagepipeline.so	OK

2024-11-07 01:36:24	Analyzing lib/x86/libreactnativeblob.so	OK
2024-11-07 01:36:24	Analyzing lib/x86/libnative-filters.so	OK
2024-11-07 01:36:24	Analyzing lib/x86/libjscexecutor.so	OK
2024-11-07 01:36:24	Analyzing lib/x86/libfolly_json.so	OK
2024-11-07 01:36:24	Analyzing lib/x86/libjsijni profiler.so	OK
2024-11-07 01:36:24	Analyzing lib/x86/libreactnativejni.so	OK
2024-11-07 01:36:24	Analyzing lib/x86/libhermes-executor-debug.so	OK
2024-11-07 01:36:24	Analyzing lib/x86/libyoga.so	OK
2024-11-07 01:36:24	Analyzing lib/x86/libhermes-executor-release.so	OK
2024-11-07 01:36:24	Analyzing lib/x86/libglog_init.so	OK
2024-11-07 01:36:24	Analyzing lib/x86/libhermes-inspector.so	OK

2024-11-07 01:36:24	Analyzing lib/x86/libfolly_futures.so	OK
2024-11-07 01:36:24	Analyzing lib/x86/libglog.so	OK
2024-11-07 01:36:24	Analyzing lib/x86/libc++_shared.so	OK
2024-11-07 01:36:24	Analyzing lib/x86/libfb.so	OK
2024-11-07 01:36:24	Analyzing lib/x86/libTMXProfiling-6.0-138-jni.so	OK
2024-11-07 01:36:24	Analyzing lib/x86/libjsinspector.so	OK
2024-11-07 01:36:24	Analyzing lib/x86/libnative-image-transcoder.so	OK
2024-11-07 01:36:24	Analyzing lib/x86/libjsc.so	OK
2024-11-07 01:36:25	Analyzing lib/arm64-v8a/libfbjni.so	OK
2024-11-07 01:36:25	Analyzing lib/arm64-v8a/libimagepipeline.so	OK
2024-11-07 01:36:25	Analyzing lib/arm64-v8a/libreactnativeblob.so	OK

2024-11-07 01:36:25	Analyzing lib/arm64-v8a/libnative-filters.so	OK
2024-11-07 01:36:25	Analyzing lib/arm64-v8a/libjscexecutor.so	OK
2024-11-07 01:36:25	Analyzing lib/arm64-v8a/libfolly_json.so	OK
2024-11-07 01:36:25	Analyzing lib/arm64-v8a/libjsijni profiler.so	OK
2024-11-07 01:36:25	Analyzing lib/arm64-v8a/libreactnativejni.so	OK
2024-11-07 01:36:25	Analyzing lib/arm64-v8a/libhermes-executor-debug.so	OK
2024-11-07 01:36:25	Analyzing lib/arm64-v8a/libyoga.so	OK
2024-11-07 01:36:25	Analyzing lib/arm64-v8a/libtaz_full.4.9.0.5.so	OK
2024-11-07 01:36:25	Analyzing lib/arm64-v8a/libhermes-executor-release.so	OK
2024-11-07 01:36:25	Analyzing lib/arm64-v8a/libsecure_core.so	OK
2024-11-07 01:36:25	Analyzing lib/arm64-v8a/libtaz_native.so	OK

2024-11-07 01:36:25	Analyzing lib/arm64-v8a/libglog_init.so	OK
2024-11-07 01:36:25	Analyzing lib/arm64-v8a/libhermes-inspector.so	OK
2024-11-07 01:36:25	Analyzing lib/arm64-v8a/libfolly_futures.so	OK
2024-11-07 01:36:25	Analyzing lib/arm64-v8a/libglog.so	OK
2024-11-07 01:36:25	Analyzing lib/arm64-v8a/libc++_shared.so	OK
2024-11-07 01:36:25	Analyzing lib/arm64-v8a/libfb.so	OK
2024-11-07 01:36:25	Analyzing lib/arm64-v8a/libTMXProfiling-6.0-138-jni.so	OK
2024-11-07 01:36:25	Analyzing lib/arm64-v8a/libjsinspector.so	OK
2024-11-07 01:36:25	Analyzing lib/arm64-v8a/libnative-image-transcoder.so	OK
2024-11-07 01:36:25	Analyzing lib/arm64-v8a/libmobilecheck.so	OK
2024-11-07 01:36:25	Analyzing lib/arm64-v8a/libjsc.so	OK

2024-11-07 01:36:26	Analyzing lib/arm64-v8a/libopencv_java3.so	OK
2024-11-07 01:36:26	Analyzing apktool_out/lib/x86_64/libfbjni.so	OK
2024-11-07 01:36:26	Analyzing apktool_out/lib/x86_64/libimagepipeline.so	OK
2024-11-07 01:36:26	Analyzing apktool_out/lib/x86_64/libreactnativeblob.so	OK
2024-11-07 01:36:26	Analyzing apktool_out/lib/x86_64/libnative-filters.so	OK
2024-11-07 01:36:26	Analyzing apktool_out/lib/x86_64/libjscexecutor.so	OK
2024-11-07 01:36:26	Analyzing apktool_out/lib/x86_64/libfolly_json.so	OK
2024-11-07 01:36:26	Analyzing apktool_out/lib/x86_64/libjsijniProfiler.so	OK
2024-11-07 01:36:26	Analyzing apktool_out/lib/x86_64/libreactnativejni.so	OK
2024-11-07 01:36:26	Analyzing apktool_out/lib/x86_64/libhermes-executor-debug.so	OK
2024-11-07 01:36:26	Analyzing apktool_out/lib/x86_64/libyoga.so	OK

2024-11-07 01:36:26	Analyzing apktool_out/lib/x86_64/libhermes-executor-release.so	OK
2024-11-07 01:36:26	Analyzing apktool_out/lib/x86_64/libglog_init.so	OK
2024-11-07 01:36:26	Analyzing apktool_out/lib/x86_64/libhermes-inspector.so	OK
2024-11-07 01:36:26	Analyzing apktool_out/lib/x86_64/libfolly_futures.so	OK
2024-11-07 01:36:26	Analyzing apktool_out/lib/x86_64/libglog.so	OK
2024-11-07 01:36:26	Analyzing apktool_out/lib/x86_64/libc++_shared.so	OK
2024-11-07 01:36:26	Analyzing apktool_out/lib/x86_64/libfb.so	OK
2024-11-07 01:36:26	Analyzing apktool_out/lib/x86_64/libTMXProfiling-6.0-138-jni.so	OK
2024-11-07 01:36:26	Analyzing apktool_out/lib/x86_64/libjsinspector.so	OK
2024-11-07 01:36:26	Analyzing apktool_out/lib/x86_64/libnative-image-transcoder.so	OK
2024-11-07 01:36:26	Analyzing apktool_out/lib/x86_64/libjsc.so	OK



2024-11-07 01:36:26	Analyzing apktool_out/lib/armeabi-v7a/libfbjni.so	OK
2024-11-07 01:36:27	Analyzing apktool_out/lib/armeabi-v7a/libimagepipeline.so	OK
2024-11-07 01:36:27	Analyzing apktool_out/lib/armeabi-v7a/libreactnativeblob.so	OK
2024-11-07 01:36:27	Analyzing apktool_out/lib/armeabi-v7a/libnative-filters.so	OK
2024-11-07 01:36:27	Analyzing apktool_out/lib/armeabi-v7a/libjscexecutor.so	OK
2024-11-07 01:36:27	Analyzing apktool_out/lib/armeabi-v7a/libfolly_json.so	OK
2024-11-07 01:36:27	Analyzing apktool_out/lib/armeabi-v7a/libjsijni profiler.so	OK
2024-11-07 01:36:27	Analyzing apktool_out/lib/armeabi-v7a/libreactnativejni.so	OK
2024-11-07 01:36:27	Analyzing apktool_out/lib/armeabi-v7a/libhermes-executor-debug.so	OK
2024-11-07 01:36:27	Analyzing apktool_out/lib/armeabi-v7a/libyoga.so	OK
2024-11-07 01:36:27	Analyzing apktool_out/lib/armeabi-v7a/libtaz_full.4.9.0.5.so	OK
2024-11-07 01:36:27	Analyzing apktool_out/lib/armeabi-v7a/libhermes-executor-release.so	OK

2024-11-07 01:36:27	Analyzing apktool_out/lib/armeabi-v7a/libsecure_core.so	OK
2024-11-07 01:36:27	Analyzing apktool_out/lib/armeabi-v7a/libtaz_native.so	OK
2024-11-07 01:36:27	Analyzing apktool_out/lib/armeabi-v7a/libglog_init.so	OK
2024-11-07 01:36:27	Analyzing apktool_out/lib/armeabi-v7a/libhermes-inspector.so	OK
2024-11-07 01:36:27	Analyzing apktool_out/lib/armeabi-v7a/libfolly_futures.so	OK
2024-11-07 01:36:27	Analyzing apktool_out/lib/armeabi-v7a/libglog.so	OK
2024-11-07 01:36:27	Analyzing apktool_out/lib/armeabi-v7a/libc++_shared.so	OK
2024-11-07 01:36:27	Analyzing apktool_out/lib/armeabi-v7a/libfb.so	OK
2024-11-07 01:36:27	Analyzing apktool_out/lib/armeabi-v7a/libTMXProfiling-6.0-138-jni.so	OK
2024-11-07 01:36:27	Analyzing apktool_out/lib/armeabi-v7a/libjsinspector.so	OK
2024-11-07 01:36:27	Analyzing apktool_out/lib/armeabi-v7a/libnative-image-transcoder.so	OK

2024-11-07 01:36:27	Analyzing apktool_out/lib/armeabi-v7a/libmobicheck.so	OK
2024-11-07 01:36:27	Analyzing apktool_out/lib/armeabi-v7a/libjsc.so	OK
2024-11-07 01:36:27	Analyzing apktool_out/lib/armeabi-v7a/libopencv_java3.so	OK
2024-11-07 01:36:28	Analyzing apktool_out/lib/x86/libfbjni.so	OK
2024-11-07 01:36:28	Analyzing apktool_out/lib/x86/libimagepipeline.so	OK
2024-11-07 01:36:28	Analyzing apktool_out/lib/x86/libreactnativeblob.so	OK
2024-11-07 01:36:28	Analyzing apktool_out/lib/x86/libnative-filters.so	OK
2024-11-07 01:36:28	Analyzing apktool_out/lib/x86/libjscexecutor.so	OK
2024-11-07 01:36:28	Analyzing apktool_out/lib/x86/libfolly_json.so	OK
2024-11-07 01:36:28	Analyzing apktool_out/lib/x86/libjsijni profiler.so	OK
2024-11-07 01:36:28	Analyzing apktool_out/lib/x86/libreactnativejni.so	OK

2024-11-07 01:36:28	Analyzing apktool_out/lib/x86/libhermes-executor-debug.so	OK
2024-11-07 01:36:28	Analyzing apktool_out/lib/x86/libyoga.so	OK
2024-11-07 01:36:28	Analyzing apktool_out/lib/x86/libhermes-executor-release.so	OK
2024-11-07 01:36:28	Analyzing apktool_out/lib/x86/libglog_init.so	OK
2024-11-07 01:36:28	Analyzing apktool_out/lib/x86/libhermes-inspector.so	OK
2024-11-07 01:36:28	Analyzing apktool_out/lib/x86/libfolly_futures.so	OK
2024-11-07 01:36:28	Analyzing apktool_out/lib/x86/libglog.so	OK
2024-11-07 01:36:28	Analyzing apktool_out/lib/x86/libc++_shared.so	OK
2024-11-07 01:36:28	Analyzing apktool_out/lib/x86/libfb.so	OK
2024-11-07 01:36:28	Analyzing apktool_out/lib/x86/libTMXProfiling-6.0-138-jni.so	OK
2024-11-07 01:36:28	Analyzing apktool_out/lib/x86/libjsinspector.so	OK

2024-11-07 01:36:28	Analyzing apktool_out/lib/x86/libnative-imagetranscoder.so	OK
2024-11-07 01:36:28	Analyzing apktool_out/lib/x86/libjsc.so	OK
2024-11-07 01:36:28	Analyzing apktool_out/lib/arm64-v8a/libfbjni.so	OK
2024-11-07 01:36:28	Analyzing apktool_out/lib/arm64-v8a/libimagepipeline.so	OK
2024-11-07 01:36:28	Analyzing apktool_out/lib/arm64-v8a/libreactnativeblob.so	OK
2024-11-07 01:36:28	Analyzing apktool_out/lib/arm64-v8a/libnative-filters.so	OK
2024-11-07 01:36:28	Analyzing apktool_out/lib/arm64-v8a/libjscexecutor.so	OK
2024-11-07 01:36:29	Analyzing apktool_out/lib/arm64-v8a/libfolly_json.so	OK
2024-11-07 01:36:29	Analyzing apktool_out/lib/arm64-v8a/libjsijni profiler.so	OK
2024-11-07 01:36:29	Analyzing apktool_out/lib/arm64-v8a/libreactnativejni.so	OK
2024-11-07 01:36:29	Analyzing apktool_out/lib/arm64-v8a/libhermes-executor-debug.so	OK

2024-11-07 01:36:29	Analyzing apktool_out/lib/arm64-v8a/libyoga.so	OK
2024-11-07 01:36:29	Analyzing apktool_out/lib/arm64-v8a/libtaz_full.4.9.0.5.so	OK
2024-11-07 01:36:29	Analyzing apktool_out/lib/arm64-v8a/libhermes-executor-release.so	OK
2024-11-07 01:36:29	Analyzing apktool_out/lib/arm64-v8a/libsecure_core.so	OK
2024-11-07 01:36:29	Analyzing apktool_out/lib/arm64-v8a/libtaz_native.so	OK
2024-11-07 01:36:29	Analyzing apktool_out/lib/arm64-v8a/libglog_init.so	OK
2024-11-07 01:36:29	Analyzing apktool_out/lib/arm64-v8a/libhermes-inspector.so	OK
2024-11-07 01:36:29	Analyzing apktool_out/lib/arm64-v8a/libfolly_futures.so	OK
2024-11-07 01:36:29	Analyzing apktool_out/lib/arm64-v8a/libglog.so	OK
2024-11-07 01:36:29	Analyzing apktool_out/lib/arm64-v8a/libc++_shared.so	OK
2024-11-07 01:36:29	Analyzing apktool_out/lib/arm64-v8a/libfb.so	OK

2024-11-07 01:36:29	Analyzing apktool_out/lib/arm64-v8a/libTMXProfiling-6.0-138-jni.so	OK
2024-11-07 01:36:29	Analyzing apktool_out/lib/arm64-v8a/libjsinspector.so	OK
2024-11-07 01:36:29	Analyzing apktool_out/lib/arm64-v8a/libnative-image-transcoder.so	OK
2024-11-07 01:36:29	Analyzing apktool_out/lib/arm64-v8a/libmobilecheck.so	OK
2024-11-07 01:36:29	Analyzing apktool_out/lib/arm64-v8a/libjsc.so	OK
2024-11-07 01:36:29	Analyzing apktool_out/lib/arm64-v8a/libopencv_java3.so	OK
2024-11-07 01:36:30	Reading Code Signing Certificate	OK
2024-11-07 01:36:32	Running APKiD 2.1.5	OK
2024-11-07 01:36:41	Updating Trackers Database....	OK
2024-11-07 01:36:41	Detecting Trackers	OK
2024-11-07 01:36:47	Decompiling APK to Java with jadx	OK

2024-11-07 01:37:29	Converting DEX to Smali	OK
2024-11-07 01:37:29	Code Analysis Started on - java_source	OK
2024-11-07 01:38:16	Android SAST Completed	OK
2024-11-07 01:38:16	Android API Analysis Started	OK
2024-11-07 01:39:04	Android Permission Mapping Started	OK
2024-11-07 01:39:54	Android Permission Mapping Completed	OK
2024-11-07 01:40:06	Finished Code Analysis, Email and URL Extraction	OK
2024-11-07 01:40:06	Extracting String data from APK	OK
2024-11-07 01:40:07	Extracting String data from SO	OK
2024-11-07 01:40:08	Extracting String data from Code	OK
2024-11-07 01:40:08	Extracting String values and entropies from Code	OK



2024-11-07 01:40:13	Performing Malware check on extracted domains	OK
2024-11-07 01:40:16	Saving to Database	OK

---

**Report Generated by - MobSF v4.1.4**

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.