IOS STATIC ANALYSIS REPORT

app_icon

MACKAudit (2.3)

File Name:	MACKAudit_v2.3_wo_verify.ipa
Identifier:	com.mack.audit
Scan Date:	March 22, 2024, 7:09 a.m.
App Security Score:	34/100 (HIGH RISK)
Grade:	C

FINDINGS SEVERITY

HIGH	MEDIUM	INFO	SECURE	HOTSPOT
3	1	1	1	2

FILE INFORMATION

File Name: MACKAudit_v2.3_wo_verify.ipa
Size: 14.51MB
MD5: c6a01b7519b36ca62260a5375b0346b4
SHA1: 87b8677163f59eb16d899eba04ebf4173fe39429
SHA256: f8dc3e32fe96dac0a390cd908eba5f0aae66269454077925ed9eb2b7cd9394cf

APP INFORMATION

App Name: MACKAudit
App Type: Swift
Identifier: com.mack.audit
SDK Name: iphoneos16.4
Version: 2.3
Build: 21
Platform Version: 16.4
Min OS Version: 11.0
Supported Platforms: iPhoneOS,

BINARY INFORMATION

Arch: ARM64
Sub Arch: CPU_SUBTYPE_ARM64_ALL
Bit: 64-bit
Endian: <

APPLICATION PERMISSIONS

PERMISSIONS	STATUS	INFO	REASON IN MANIFEST
NSAppleMusicUsageDescription	dangerous	Access Apple Media Library.	MACK Audit need to access the Media library to upload the files in app.
NSCameraUsageDescription	dangerous	Access the Camera.	MACK Audit need to access your camera to configure the url through qrscanning
NSPhotoLibraryUsageDescription	dangerous	Access the user's photo library.	MACK Audit need to access your Photo library to store the image in directory

APP TRANSPORT SECURITY (ATS)

NO	ISSUE	SEVERITY	DESCRIPTION

IPA BINARY CODE ANALYSIS

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

NO	ISSUE	SEVERITY	STANDARDS	DESCRIPTION

NO	ISSUE	SEVERITY	STANDARDS	DESCRIPTION
1	Binary makes use of insecure API(s)	high	**CWE:** CWE-676: Use of Potentially Dangerous Function **OWASP Top 10:** M7: Client Code Quality **OWASP MASVS:** MSTG-CODE-8	The binary may contain the following insecure API(s) _fopen , _memcpy , _printf , _sscanf , _stat , _strcpy , _strlen , _strncat , _strncpy , _vsnprintf
2	Binary makes use of the insecure Random function(s)	high	**CWE:** CWE-330: Use of Insufficiently Random Values **OWASP Top 10:** M5: Insufficient Cryptography **OWASP MASVS:** MSTG-CRYPTO-6	The binary may use the following insecure Random function(s) _random
3	Binary makes use of Logging function	info	**CWE:** CWE-532: Insertion of Sensitive Information into Log File **OWASP MASVS:** MSTG-STORAGE-3	The binary may use _NSLog function for logging.
4	Binary makes use of malloc function	high	**CWE:** CWE-789: Uncontrolled Memory Allocation **OWASP Top 10:** M7: Client Code Quality **OWASP MASVS:** MSTG-CODE-8	The binary may use _malloc function instead of calloc

IPA BINARY ANALYSIS

PROTECTION	STATUS	SEVERITY	DESCRIPTION
NX	True	info	The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.
PIE	True	info	The binary is build with -fPIC flag which enables Position independent code. This makes Return Oriented Programming (ROP) attacks much more difficult to execute reliably.

PROTECTION	STATUS	SEVERITY	DESCRIPTION
STACK CANARY	True	info	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.
ARC	True	info	The binary is compiled with Automatic Reference Counting (ARC) flag. ARC is a compiler feature that provides automatic memory management of Objective-C objects and is an exploit mitigation mechanism against memory corruption vulnerabilities.
RPATH	True	warning	The binary has Runpath Search Path (@rpath) set. In certain cases an attacker can abuse this feature to run arbitrary executable for code execution and privilege escalation. Remove the compiler option -rpath to remove @rpath.
CODE SIGNATURE	True	info	This binary has a code signature.
ENCRYPTED	False	warning	This binary is not encrypted.
SYMBOLS STRIPPED	True	info	Debug Symbols are stripped

DYNAMIC LIBRARY & FRAMEWORK BINARY ANALYSIS

NO	DYLIB/FRAMEWORK	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED

NO	DYLIB/FRAMEWORK	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
1	Frameworks/libswiftDarwin.dylib	True info The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.	False warning 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. This might be okey for pure Swift dylibs.	True info The binary is compiled with Automatic Reference Counting (ARC) flag. ARC is a compiler feature that provides automatic memory management of Objective-C objects and is an exploit mitigation mechanism against memory corruption vulnerabilities.	False info The binary does not have Runpath Search Path (@rpath) set.	True info This binary has a code signature.	False warning This binary is not encrypted.	False warning Debug Symbols are available. To strip debugging symbols, set Strip Debug Symbols During Copy to YES, Deployment Postprocessing to YES, and Strip Linked Product to YES in project's build settings.

NO	DYLIB/FRAMEWORK	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
2	Frameworks/libswiftMetal.dylib	True info The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.	True info 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.	True info The binary is compiled with Automatic Reference Counting (ARC) flag. ARC is a compiler feature that provides automatic memory management of Objective-C objects and is an exploit mitigation mechanism against memory corruption vulnerabilities.	False info The binary does not have Runpath Search Path (@rpath) set.	True info This binary has a code signature.	False warning This binary is not encrypted.	False warning Debug Symbols are available. To strip debugging symbols, set Strip Debug Symbols During Copy to YES, Deployment Postprocessing to YES, and Strip Linked Product to YES in project's build settings.

NO	DYLIB/FRAMEWORK	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
3	Frameworks/libswiftUIKit.dylib	True info The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.	False warning 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. This might be okey for pure Swift dylibs.	True info The binary is compiled with Automatic Reference Counting (ARC) flag. ARC is a compiler feature that provides automatic memory management of Objective-C objects and is an exploit mitigation mechanism against memory corruption vulnerabilities.	False info The binary does not have Runpath Search Path (@rpath) set.	True info This binary has a code signature.	False warning This binary is not encrypted.	False warning Debug Symbols are available. To strip debugging symbols, set Strip Debug Symbols During Copy to YES, Deployment Postprocessing to YES, and Strip Linked Product to YES in project's build settings.

NO	DYLIB/FRAMEWORK	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
4	Frameworks/libswiftCoreFoundation.dylib	True info The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.	False warning 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. This might be okey for pure Swift dylibs.	False high The binary is not compiled with Automatic Reference Counting (ARC) flag. ARC is a compiler feature that provides automatic memory management of Objective-C objects and protects from memory corruption vulnerabilities. Use compiler option -fobjc-arc to enable ARC or set Objective-C Automatic Reference Counting to YES in project configuration.	False info The binary does not have Runpath Search Path (@rpath) set.	True info This binary has a code signature.	False warning This binary is not encrypted.	False warning Debug Symbols are available. To strip debugging symbols, set Strip Debug Symbols During Copy to YES, Deployment Postprocessing to YES, and Strip Linked Product to YES in project's build settings.

NO	DYLIB/FRAMEWORK	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
5	Frameworks/libswiftDispatch.dylib	True info The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.	True info 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.	True info The binary is compiled with Automatic Reference Counting (ARC) flag. ARC is a compiler feature that provides automatic memory management of Objective-C objects and is an exploit mitigation mechanism against memory corruption vulnerabilities.	False info The binary does not have Runpath Search Path (@rpath) set.	True info This binary has a code signature.	False warning This binary is not encrypted.	False warning Debug Symbols are available. To strip debugging symbols, set Strip Debug Symbols During Copy to YES, Deployment Postprocessing to YES, and Strip Linked Product to YES in project's build settings.

NO	DYLIB/FRAMEWORK	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
6	Frameworks/libswiftCoreGraphics.dylib	True info The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.	False warning 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. This might be okey for pure Swift dylibs.	True info The binary is compiled with Automatic Reference Counting (ARC) flag. ARC is a compiler feature that provides automatic memory management of Objective-C objects and is an exploit mitigation mechanism against memory corruption vulnerabilities.	False info The binary does not have Runpath Search Path (@rpath) set.	True info This binary has a code signature.	False warning This binary is not encrypted.	False warning Debug Symbols are available. To strip debugging symbols, set Strip Debug Symbols During Copy to YES, Deployment Postprocessing to YES, and Strip Linked Product to YES in project's build settings.

NO	DYLIB/FRAMEWORK	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
7	Frameworks/libswiftCoreImage.dylib	True info The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.	False warning 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. This might be okey for pure Swift dylibs.	False high The binary is not compiled with Automatic Reference Counting (ARC) flag. ARC is a compiler feature that provides automatic memory management of Objective-C objects and protects from memory corruption vulnerabilities. Use compiler option -fobjc-arc to enable ARC or set Objective-C Automatic Reference Counting to YES in project configuration.	False info The binary does not have Runpath Search Path (@rpath) set.	True info This binary has a code signature.	False warning This binary is not encrypted.	False warning Debug Symbols are available. To strip debugging symbols, set Strip Debug Symbols During Copy to YES, Deployment Postprocessing to YES, and Strip Linked Product to YES in project's build settings.

NO	DYLIB/FRAMEWORK	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
8	Frameworks/libswiftFoundation.dylib	True info The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.	True info 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.	True info The binary is compiled with Automatic Reference Counting (ARC) flag. ARC is a compiler feature that provides automatic memory management of Objective-C objects and is an exploit mitigation mechanism against memory corruption vulnerabilities.	False info The binary does not have Runpath Search Path (@rpath) set.	True info This binary has a code signature.	False warning This binary is not encrypted.	False warning Debug Symbols are available. To strip debugging symbols, set Strip Debug Symbols During Copy to YES, Deployment Postprocessing to YES, and Strip Linked Product to YES in project's build settings.

NO	DYLIB/FRAMEWORK	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
9	Frameworks/libswiftObjectiveC.dylib	True info The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.	False warning 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. This might be okey for pure Swift dylibs.	False high The binary is not compiled with Automatic Reference Counting (ARC) flag. ARC is a compiler feature that provides automatic memory management of Objective-C objects and protects from memory corruption vulnerabilities. Use compiler option -fobjc-arc to enable ARC or set Objective-C Automatic Reference Counting to YES in project configuration.	False info The binary does not have Runpath Search Path (@rpath) set.	True info This binary has a code signature.	False warning This binary is not encrypted.	False warning Debug Symbols are available. To strip debugging symbols, set Strip Debug Symbols During Copy to YES, Deployment Postprocessing to YES, and Strip Linked Product to YES in project's build settings.

NO	DYLIB/FRAMEWORK	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
10	Frameworks/libswiftCore.dylib	True info The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.	True info 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.	True info The binary is compiled with Automatic Reference Counting (ARC) flag. ARC is a compiler feature that provides automatic memory management of Objective-C objects and is an exploit mitigation mechanism against memory corruption vulnerabilities.	False info The binary does not have Runpath Search Path (@rpath) set.	True info This binary has a code signature.	False warning This binary is not encrypted.	False warning Debug Symbols are available. To strip debugging symbols, set Strip Debug Symbols During Copy to YES, Deployment Postprocessing to YES, and Strip Linked Product to YES in project's build settings.

NO	DYLIB/FRAMEWORK	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
11	Frameworks/libswiftQuartzCore.dylib	True info The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.	False warning 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. This might be okey for pure Swift dylibs.	False high The binary is not compiled with Automatic Reference Counting (ARC) flag. ARC is a compiler feature that provides automatic memory management of Objective-C objects and protects from memory corruption vulnerabilities. Use compiler option -fobjc-arc to enable ARC or set Objective-C Automatic Reference Counting to YES in project configuration.	False info The binary does not have Runpath Search Path (@rpath) set.	True info This binary has a code signature.	False warning This binary is not encrypted.	False warning Debug Symbols are available. To strip debugging symbols, set Strip Debug Symbols During Copy to YES, Deployment Postprocessing to YES, and Strip Linked Product to YES in project's build settings.

NO	DYLIB/FRAMEWORK	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
12	Frameworks/libswiftos.dylib	True info The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.	False warning 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. This might be okey for pure Swift dylibs.	True info The binary is compiled with Automatic Reference Counting (ARC) flag. ARC is a compiler feature that provides automatic memory management of Objective-C objects and is an exploit mitigation mechanism against memory corruption vulnerabilities.	False info The binary does not have Runpath Search Path (@rpath) set.	True info This binary has a code signature.	False warning This binary is not encrypted.	False warning Debug Symbols are available. To strip debugging symbols, set Strip Debug Symbols During Copy to YES, Deployment Postprocessing to YES, and Strip Linked Product to YES in project's build settings.

</> CODE ANALYSIS

NO	ISSUE	SEVERITY	STANDARDS	FILES

! OFAC SANCTIONED COUNTRIES

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

DOMAIN	COUNTRY/REGION
ruslanskorb.com	IP: 154.216.11.52 Country: Hong Kong Region: Hong Kong City: Hong Kong

DOMAIN MALWARE CHECK

DOMAIN	STATUS	GEOLOCATION
reactjs.org	ok	**IP:** 76.76.21.21 **Country:** United States of America **Region:** California **City:** Walnut **Latitude:** 34.015400 **Longitude:** -117.858223 **View:** Google Map

DOMAIN	STATUS	GEOLOCATION
clients3.google.com	ok	**IP:** 142.250.196.46 **Country:** United States of America **Region:** California **City:** Mountain View **Latitude:** 37.405991 **Longitude:** -122.078514 **View:** Google Map
www.w3.org	ok	**IP:** 104.18.23.19 **Country:** United States of America **Region:** California **City:** San Francisco **Latitude:** 37.775700 **Longitude:** -122.395203 **View:** Google Map
macktesting.solverminds.net	ok	**IP:** 52.187.68.122 **Country:** Singapore **Region:** Singapore **City:** Singapore **Latitude:** 1.289670 **Longitude:** 103.850067 **View:** Google Map
oe.defaultu534	ok	No Geolocation information available.
npms.io	ok	**IP:** 104.21.49.221 **Country:** United States of America **Region:** California **City:** San Francisco **Latitude:** 37.775700 **Longitude:** -122.395203 **View:** Google Map
t.mainbundledir	ok	No Geolocation information available.

DOMAIN	STATUS	GEOLOCATION
ocsp.apple.com	ok	**IP:** 23.207.74.93 **Country:** United Arab Emirates **Region:** Al Fujayrah **City:** Al Fujayrah **Latitude:** 25.116409 **Longitude:** 56.341412 **View:** Google Map
y.verbosecreate	ok	No Geolocation information available.
www.apple.com	ok	**IP:** 104.121.229.31 **Country:** India **Region:** Karnataka **City:** Bengaluru **Latitude:** 12.976230 **Longitude:** 77.603287 **View:** Google Map
reactnavigation.org	ok	**IP:** 104.21.34.251 **Country:** United States of America **Region:** California **City:** San Francisco **Latitude:** 37.775700 **Longitude:** -122.395203 **View:** Google Map
fb.me	ok	**IP:** 163.70.138.35 **Country:** France **Region:** Ile-de-France **City:** Nanterre **Latitude:** 48.891979 **Longitude:** 2.206750 **View:** Google Map

DOMAIN	STATUS	GEOLOCATION
github.com	ok	**IP:** 20.207.73.82 **Country:** United States of America **Region:** Washington **City:** Redmond **Latitude:** 47.682899 **Longitude:** -122.120903 **View:** Google Map
momentjs.com	ok	**IP:** 104.17.93.38 **Country:** United States of America **Region:** California **City:** San Francisco **Latitude:** 37.775700 **Longitude:** -122.395203 **View:** Google Map
crl.apple.com	ok	**IP:** 49.206.251.226 **Country:** India **Region:** Tamil Nadu **City:** Coimbatore **Latitude:** 11.000000 **Longitude:** 76.966667 **View:** Google Map
react-native-community.github.io	ok	**IP:** 185.199.108.153 **Country:** United States of America **Region:** Pennsylvania **City:** California **Latitude:** 40.065632 **Longitude:** -79.891708 **View:** Google Map

DOMAIN	STATUS	GEOLOCATION
stackoverflow.com	ok	**IP:** 104.18.32.7 **Country:** United States of America **Region:** Texas **City:** Dallas **Latitude:** 32.783058 **Longitude:** -96.806671 **View:** Google Map
certs.apple.com	ok	**IP:** 17.253.18.200 **Country:** Brazil **Region:** Sao Paulo **City:** Sao Paulo **Latitude:** -23.547501 **Longitude:** -46.636108 **View:** Google Map
ruslanskorb.com	ok	**IP:** 154.216.11.52 **Country:** Hong Kong **Region:** Hong Kong **City:** Hong Kong **Latitude:** 22.285521 **Longitude:** 114.157692 **View:** Google Map
static.hsappstatic.net	ok	**IP:** 104.18.79.253 **Country:** United States of America **Region:** California **City:** San Francisco **Latitude:** 37.775700 **Longitude:** -122.395203 **View:** Google Map

DOMAIN	STATUS	GEOLOCATION
encrypted-tbn0.gstatic.com	ok	**IP:** 142.250.195.206 **Country:** United States of America **Region:** California **City:** Mountain View **Latitude:** 37.405991 **Longitude:** -122.078514 **View:** Google Map

EMAILS

EMAIL	FILE
git@github.com	auditInspection.app/main.jsbundle

Report Generated by - MobSF v3.9.5 Beta

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.