



IOS STATIC ANALYSIS REPORT



🍏 swipewipe (3.0.2)

File Name: com.aoklab.fewerphotos_1583884012_3.0.2.ipa

Identifier: com.aoklab.fewerphotos

Scan Date: Jan. 11, 2025, 12:27 p.m.

App Security Score: **52/100 (MEDIUM RISK)**

Grade:



FINDINGS SEVERITY

 HIGH	 MEDIUM	 INFO	 SECURE	 HOTSPOT
1	6	1	1	1

FILE INFORMATION

File Name: com.aoklab.fewerphotos_1583884012_3.0.2.ipa
Size: 124.41MB
MD5: bf3225537289e0b34e5ff9278bc11f96
SHA1: 3eb912299ad5d4a82acc33ca1604cf71175059ae
SHA256: 358aa25681641e34f62d28e90b8a7c88a53fdad36b25c80894be6dee84445c30

APP INFORMATION

App Name: swipewipe
App Type: Swift
Identifier: com.aoklab.fewerphotos
SDK Name: iphoneos17.5

Version: 3.0.2
Build: 3
Platform Version: 17.5
Min OS Version: 16.4
Supported Platforms: iPhoneOS,

BINARY INFORMATION

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

#CUSTOM URL SCHEMES

URL NAME	SCHEMES
None Editor	fb392801433038523
None Editor	swipewipeapp
None Editor	com.googleusercontent.apps.845813118025-sfhrq3jojgvn5ivo9d235lp68fb2p7g

APPLICATION PERMISSIONS

PERMISSIONS	STATUS	INFO	REASON IN MANIFEST
NSPhotoLibraryUsageDescription	dangerous	Access the user's photo library.	To help you select photos to remove

APP TRANSPORT SECURITY (ATS)

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

NO	ISSUE	SEVERITY	DESCRIPTION
1	App Transport Security AllowsArbitraryLoads is allowed	high	App Transport Security restrictions are disabled for all network connections. Disabling ATS means that unsecured HTTP connections are allowed. HTTPS connections are also allowed, and are still subject to default server trust evaluation. However, extended security checks like requiring a minimum Transport Layer Security (TLS) protocol version—are disabled. This setting is not applicable to domains listed in NSExceptionDomains.

IPA BINARY CODE ANALYSIS

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

NO	ISSUE	SEVERITY	STANDARDS	DESCRIPTION
1	Binary makes use of insecure API(s)	warning	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) <code>_fopen</code> , <code>_memcpy</code> , <code>_printf</code> , <code>_sprintf</code> , <code>_sscanf</code> , <code>_stat</code> , <code>_strcat</code> , <code>_strcpy</code> , <code>_strlen</code> , <code>_strncpy</code> , <code>_vsprintf</code>
2	Binary makes use of the insecure Random function(s)	warning	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) <code>_random</code> , <code>_srand</code>
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 <code>_NSLog</code> function for logging.
4	Binary makes use of malloc function	warning	CWE: CWE-789: Uncontrolled Memory Allocation OWASP Top 10: M7: Client Code Quality OWASP MASVS: MSTG-CODE-8	The binary may use <code>_malloc</code> function instead of <code>calloc</code>

🚩 IPA BINARY ANALYSIS

PROTECTION	STATUS	SEVERITY	DESCRIPTION
NX	False	info	The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. However iOS never allows an app to execute from writeable memory. You do not need to specifically enable the 'NX bit' because it's always enabled for all third-party code.
PIE	True	info	The binary is build with <code>-fPIC</code> flag which enables Position independent code. This makes Return Oriented Programming (ROP) attacks much more difficult to execute reliably.
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 (<code>@rpath</code>) set. In certain cases an attacker can abuse this feature to run arbitrary executable for code execution and privilege escalation. Remove the compiler option <code>-rpath</code> to remove <code>@rpath</code> .
CODE SIGNATURE	True	info	This binary has a code signature.
ENCRYPTED	True	info	This binary is encrypted.
SYMBOLS STRIPPED	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.

🚩 DYNAMIC LIBRARY & FRAMEWORK BINARY ANALYSIS

NO	DYLIB/Framework	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
1	Payload/fewerphoto.app/Frameworks/InMobiSDK.framework/InMobiSDK	<p>False info</p> <p>The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. However iOS never allows an app to execute from writeable memory. You do not need to specifically enable the 'NX bit' because it's always enabled for all third-party code.</p>	<p>True info</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>True info</p> <p>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.</p>	<p>True warning</p> <p>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.</p>	<p>True info</p> <p>This binary has a code signature.</p>	<p>True info</p> <p>This binary is encrypted.</p>	<p>False warning</p> <p>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.</p>

NO	DYLIB/Framework	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
2	Payload/fewerphoto.app/Frameworks/FBSDKLoginKit.framework/FBSDKLoginKit	<p>False info</p> <p>The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. However iOS never allows an app to execute from writeable memory. You do not need to specifically enable the 'NX bit' because it's always enabled for all third-party code.</p>	<p>True info</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>True info</p> <p>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.</p>	<p>True warning</p> <p>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.</p>	<p>True info</p> <p>This binary has a code signature.</p>	<p>True info</p> <p>This binary is encrypted.</p>	<p>False warning</p> <p>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.</p>

NO	DYLIB/Framework	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
3	Payload/fewerphoto.app/Frameworks/VungleAdsSDK.framework/VungleAdsSDK	<p>False info</p> <p>The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. However iOS never allows an app to execute from writeable memory. You do not need to specifically enable the 'NX bit' because it's always enabled for all third-party code.</p>	<p>False high</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.</p>	<p>False high</p> <p>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.</p>	<p>False info</p> <p>The binary does not have Runpath Search Path (@rpath) set.</p>	<p>True info</p> <p>This binary has a code signature.</p>	<p>False warning</p> <p>This binary is not encrypted.</p>	<p>False warning</p> <p>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.</p>

NO	DYLIB/Framework	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
4	Payload/fewerphoto.app/Frameworks/FBAEMKit.framework/FBAEMKit	<p>False info</p> <p>The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. However iOS never allows an app to execute from writeable memory. You do not need to specifically enable the 'NX bit' because it's always enabled for all third-party code.</p>	<p>True info</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>True info</p> <p>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.</p>	<p>True warning</p> <p>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.</p>	<p>True info</p> <p>This binary has a code signature.</p>	<p>True info</p> <p>This binary is encrypted.</p>	<p>False warning</p> <p>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.</p>

NO	DYLIB/Framework	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
5	Payload/fewerphoto.app/Frameworks/MTGSDKSplash.framework/MTGSDKSplash	<p>False info</p> <p>The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. However iOS never allows an app to execute from writeable memory. You do not need to specifically enable the 'NX bit' because it's always enabled for all third-party code.</p>	<p>False high</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.</p>	<p>False high</p> <p>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.</p>	<p>False info</p> <p>The binary does not have Runpath Search Path (@rpath) set.</p>	<p>True info</p> <p>This binary has a code signature.</p>	<p>False warning</p> <p>This binary is not encrypted.</p>	<p>False warning</p> <p>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.</p>

NO	DYLIB/Framework	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
6	Payload/fewerphoto.app/Frameworks/IASDKCore.framework/IASDKCore	<p>False info</p> <p>The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. However iOS never allows an app to execute from writeable memory. You do not need to specifically enable the 'NX bit' because it's always enabled for all third-party code.</p>	<p>False high</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.</p>	<p>False high</p> <p>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.</p>	<p>False info</p> <p>The binary does not have Runpath Search Path (@rpath) set.</p>	<p>True info</p> <p>This binary has a code signature.</p>	<p>False warning</p> <p>This binary is not encrypted.</p>	<p>False warning</p> <p>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.</p>

NO	DYLIB/Framework	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
7	Payload/fewerphoto.app/Frameworks/grpc.framework/grpc	<p>False info</p> <p>The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. However iOS never allows an app to execute from writeable memory. You do not need to specifically enable the 'NX bit' because it's always enabled for all third-party code.</p>	<p>False high</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.</p>	<p>False high</p> <p>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.</p>	<p>False info</p> <p>The binary does not have Runpath Search Path (@rpath) set.</p>	<p>True info</p> <p>This binary has a code signature.</p>	<p>False warning</p> <p>This binary is not encrypted.</p>	<p>False warning</p> <p>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.</p>

NO	DYLIB/Framework	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
8	Payload/fewerphoto.app/Frameworks/StackProductPresentation.framework/StackProductPresentation	<p>False info</p> <p>The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. However iOS never allows an app to execute from writeable memory. You do not need to specifically enable the 'NX bit' because it's always enabled for all third-party code.</p>	<p>False high</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.</p>	<p>False high</p> <p>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.</p>	<p>False info</p> <p>The binary does not have Runpath Search Path (@rpath) set.</p>	<p>True info</p> <p>This binary has a code signature.</p>	<p>False warning</p> <p>This binary is not encrypted.</p>	<p>False warning</p> <p>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.</p>

NO	DYLIB/Framework	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
9	Payload/fewerphoto.app/Frameworks/MWMPublishingSDK.framework/MWMPublishingSDK	<p>False info</p> <p>The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. However iOS never allows an app to execute from writeable memory. You do not need to specifically enable the 'NX bit' because it's always enabled for all third-party code.</p>	<p>True info</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>True info</p> <p>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.</p>	<p>True warning</p> <p>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.</p>	<p>True info</p> <p>This binary has a code signature.</p>	<p>True info</p> <p>This binary is encrypted.</p>	<p>False warning</p> <p>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.</p>

NO	DYLIB/Framework	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
10	Payload/fewerphoto.app/Frameworks/PAGAdSDK.framework/PAGAdSDK	<p>False info</p> <p>The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. However iOS never allows an app to execute from writeable memory. You do not need to specifically enable the 'NX bit' because it's always enabled for all third-party code.</p>	<p>False high</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.</p>	<p>False high</p> <p>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.</p>	<p>False info</p> <p>The binary does not have Runpath Search Path (@rpath) set.</p>	<p>True info</p> <p>This binary has a code signature.</p>	<p>False warning</p> <p>This binary is not encrypted.</p>	<p>False warning</p> <p>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.</p>

NO	DYLIB/Framework	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
11	Payload/fewerphoto.app/Frameworks/openssl_grpc.framework/openssl_grpc	<p>False info</p> <p>The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. However iOS never allows an app to execute from writeable memory. You do not need to specifically enable the 'NX bit' because it's always enabled for all third-party code.</p>	<p>False high</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.</p>	<p>False high</p> <p>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.</p>	<p>False info</p> <p>The binary does not have Runpath Search Path (@rpath) set.</p>	<p>True info</p> <p>This binary has a code signature.</p>	<p>False warning</p> <p>This binary is not encrypted.</p>	<p>False warning</p> <p>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.</p>

NO	DYLIB/Framework	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
12	Payload/fewerphoto.app/Frameworks/DotLottiePlayer.framework/DotLottiePlayer	<p>False info</p> <p>The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. However iOS never allows an app to execute from writeable memory. You do not need to specifically enable the 'NX bit' because it's always enabled for all third-party code.</p>	<p>True info</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>False high</p> <p>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.</p>	<p>False info</p> <p>The binary does not have Runpath Search Path (@rpath) set.</p>	<p>True info</p> <p>This binary has a code signature.</p>	<p>True info</p> <p>This binary is encrypted.</p>	<p>False warning</p> <p>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.</p>

NO	DYLIB/Framework	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
13	Payload/fewerphoto.app/Frameworks/SCSDKLoginKit.framework/SCSDKLoginKit	<p>False info</p> <p>The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. However iOS never allows an app to execute from writeable memory. You do not need to specifically enable the 'NX bit' because it's always enabled for all third-party code.</p>	<p>True info</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>True info</p> <p>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.</p>	<p>True warning</p> <p>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.</p>	<p>True info</p> <p>This binary has a code signature.</p>	<p>True info</p> <p>This binary is encrypted.</p>	<p>False warning</p> <p>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.</p>

NO	DYLIB/Framework	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
14	Payload/fewerphoto.app/Frameworks/MTGSDKReward.framework/MTGSDKReward	<p>False info</p> <p>The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. However iOS never allows an app to execute from writeable memory. You do not need to specifically enable the 'NX bit' because it's always enabled for all third-party code.</p>	<p>False high</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.</p>	<p>False high</p> <p>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.</p>	<p>False info</p> <p>The binary does not have Runpath Search Path (@rpath) set.</p>	<p>True info</p> <p>This binary has a code signature.</p>	<p>False warning</p> <p>This binary is not encrypted.</p>	<p>False warning</p> <p>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.</p>

NO	DYLIB/Framework	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
15	Payload/fewerphoto.app/Frameworks/DTBIOSSDK.framework/DTBIOSSDK	<p>False info</p> <p>The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. However iOS never allows an app to execute from writeable memory. You do not need to specifically enable the 'NX bit' because it's always enabled for all third-party code.</p>	<p>True info</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>True info</p> <p>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.</p>	<p>True warning</p> <p>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.</p>	<p>True info</p> <p>This binary has a code signature.</p>	<p>True info</p> <p>This binary is encrypted.</p>	<p>False warning</p> <p>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.</p>

NO	DYLIB/Framework	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
16	Payload/fewerphoto.app/Frameworks/PaywallTemplates.framework/PaywallTemplates	<p>False info</p> <p>The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. However iOS never allows an app to execute from writeable memory. You do not need to specifically enable the 'NX bit' because it's always enabled for all third-party code.</p>	<p>True info</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>True info</p> <p>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.</p>	<p>True warning</p> <p>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.</p>	<p>True info</p> <p>This binary has a code signature.</p>	<p>True info</p> <p>This binary is encrypted.</p>	<p>False warning</p> <p>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.</p>

NO	DYLIB/Framework	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
17	Payload/fewerphoto.app/Frameworks/FirebaseFirestoreInternal.framework/FirebaseFirestoreInternal	<p>False info</p> <p>The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. However iOS never allows an app to execute from writeable memory. You do not need to specifically enable the 'NX bit' because it's always enabled for all third-party code.</p>	<p>False high</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.</p>	<p>False high</p> <p>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.</p>	<p>False info</p> <p>The binary does not have Runpath Search Path (@rpath) set.</p>	<p>True info</p> <p>This binary has a code signature.</p>	<p>False warning</p> <p>This binary is not encrypted.</p>	<p>False warning</p> <p>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.</p>

NO	DYLIB/Framework	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
18	Payload/fewerphoto.app/Frameworks/MolocoSDK.framework/MolocoSDK	<p>False info</p> <p>The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. However iOS never allows an app to execute from writeable memory. You do not need to specifically enable the 'NX bit' because it's always enabled for all third-party code.</p>	<p>True info</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>True info</p> <p>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.</p>	<p>True warning</p> <p>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.</p>	<p>True info</p> <p>This binary has a code signature.</p>	<p>True info</p> <p>This binary is encrypted.</p>	<p>False warning</p> <p>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.</p>

NO	DYLIB/Framework	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
19	Payload/fewerphoto.app/Frameworks/UnityAds.framework/UnityAds	<p>False info</p> <p>The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. However iOS never allows an app to execute from writeable memory. You do not need to specifically enable the 'NX bit' because it's always enabled for all third-party code.</p>	<p>False high</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.</p>	<p>False high</p> <p>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.</p>	<p>False info</p> <p>The binary does not have Runpath Search Path (@rpath) set.</p>	<p>True info</p> <p>This binary has a code signature.</p>	<p>False warning</p> <p>This binary is not encrypted.</p>	<p>False warning</p> <p>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.</p>

NO	DYLIB/Framework	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
20	Payload/fewerphoto.app/Frameworks/MTGSDKBidding.framework/MTGSDKBidding	<p>False info</p> <p>The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. However iOS never allows an app to execute from writeable memory. You do not need to specifically enable the 'NX bit' because it's always enabled for all third-party code.</p>	<p>False high</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.</p>	<p>False high</p> <p>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.</p>	<p>False info</p> <p>The binary does not have Runpath Search Path (@rpath) set.</p>	<p>True info</p> <p>This binary has a code signature.</p>	<p>False warning</p> <p>This binary is not encrypted.</p>	<p>False warning</p> <p>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.</p>

NO	DYLIB/Framework	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
21	Payload/fewerphoto.app/Frameworks/AppLovinSDK.framework/AppLovinSDK	<p>False info</p> <p>The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. However iOS never allows an app to execute from writeable memory. You do not need to specifically enable the 'NX bit' because it's always enabled for all third-party code.</p>	<p>True info</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>True info</p> <p>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.</p>	<p>True warning</p> <p>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.</p>	<p>True info</p> <p>This binary has a code signature.</p>	<p>True info</p> <p>This binary is encrypted.</p>	<p>False warning</p> <p>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.</p>

NO	DYLIB/Framework	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
22	Payload/fewerphoto.app/Frameworks/StackModules.framework/StackModules	<p>False info</p> <p>The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. However iOS never allows an app to execute from writeable memory. You do not need to specifically enable the 'NX bit' because it's always enabled for all third-party code.</p>	<p>False high</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.</p>	<p>False high</p> <p>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.</p>	<p>False info</p> <p>The binary does not have Runpath Search Path (@rpath) set.</p>	<p>True info</p> <p>This binary has a code signature.</p>	<p>False warning</p> <p>This binary is not encrypted.</p>	<p>False warning</p> <p>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.</p>

NO	DYLIB/Framework	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
23	Payload/fewerphoto.app/Frameworks/AppLovinQualityService.framework/AppLovinQualityService	<p>False info</p> <p>The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. However iOS never allows an app to execute from writeable memory. You do not need to specifically enable the 'NX bit' because it's always enabled for all third-party code.</p>	<p>True info</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>True info</p> <p>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.</p>	<p>True warning</p> <p>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.</p>	<p>True info</p> <p>This binary has a code signature.</p>	<p>True info</p> <p>This binary is encrypted.</p>	<p>False warning</p> <p>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.</p>

NO	DYLIB/Framework	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
24	Payload/fewerphoto.app/Frameworks/UserMessagingPlatform.framework/UserMessagingPlatform	<p>False info</p> <p>The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. However iOS never allows an app to execute from writeable memory. You do not need to specifically enable the 'NX bit' because it's always enabled for all third-party code.</p>	<p>False high</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.</p>	<p>False high</p> <p>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.</p>	<p>False info</p> <p>The binary does not have Runpath Search Path (@rpath) set.</p>	<p>True info</p> <p>This binary has a code signature.</p>	<p>False warning</p> <p>This binary is not encrypted.</p>	<p>False warning</p> <p>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.</p>

NO	DYLIB/Framework	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
25	Payload/fewerphoto.app/Frameworks/SCSDKCoreKit.framework/SCSDKCoreKit	<p>False info</p> <p>The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. However iOS never allows an app to execute from writeable memory. You do not need to specifically enable the 'NX bit' because it's always enabled for all third-party code.</p>	<p>True info</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>True info</p> <p>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.</p>	<p>True warning</p> <p>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.</p>	<p>True info</p> <p>This binary has a code signature.</p>	<p>True info</p> <p>This binary is encrypted.</p>	<p>False warning</p> <p>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.</p>

NO	DYLIB/Framework	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
26	Payload/fewerphoto.app/Frameworks/MTGSDK.framework/MTGSDK	<p>False info</p> <p>The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. However iOS never allows an app to execute from writeable memory. You do not need to specifically enable the 'NX bit' because it's always enabled for all third-party code.</p>	<p>False high</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.</p>	<p>False high</p> <p>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.</p>	<p>False info</p> <p>The binary does not have Runpath Search Path (@rpath) set.</p>	<p>True info</p> <p>This binary has a code signature.</p>	<p>False warning</p> <p>This binary is not encrypted.</p>	<p>False warning</p> <p>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.</p>

NO	DYLIB/Framework	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
27	Payload/fewerphoto.app/Frameworks/BUReyFoundation_Global.framework/BUReyFoundation_Global	<p>False info</p> <p>The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. However iOS never allows an app to execute from writeable memory. You do not need to specifically enable the 'NX bit' because it's always enabled for all third-party code.</p>	<p>False high</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.</p>	<p>False high</p> <p>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.</p>	<p>False info</p> <p>The binary does not have Runpath Search Path (@rpath) set.</p>	<p>True info</p> <p>This binary has a code signature.</p>	<p>False warning</p> <p>This binary is not encrypted.</p>	<p>False warning</p> <p>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.</p>

NO	DYLIB/Framework	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
28	Payload/fewerphoto.app/Frameworks/MTGSDKBanner.framework/MTGSDKBanner	<p>False info</p> <p>The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. However iOS never allows an app to execute from writeable memory. You do not need to specifically enable the 'NX bit' because it's always enabled for all third-party code.</p>	<p>False high</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.</p>	<p>False high</p> <p>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.</p>	<p>False info</p> <p>The binary does not have Runpath Search Path (@rpath) set.</p>	<p>True info</p> <p>This binary has a code signature.</p>	<p>False warning</p> <p>This binary is not encrypted.</p>	<p>False warning</p> <p>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.</p>

NO	DYLIB/Framework	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
29	Payload/fewerphoto.app/Frameworks/GoogleMobileAds.framework/GoogleMobileAds	<p>False info</p> <p>The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. However iOS never allows an app to execute from writeable memory. You do not need to specifically enable the 'NX bit' because it's always enabled for all third-party code.</p>	<p>False high</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.</p>	<p>False high</p> <p>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.</p>	<p>False info</p> <p>The binary does not have Runpath Search Path (@rpath) set.</p>	<p>True info</p> <p>This binary has a code signature.</p>	<p>False warning</p> <p>This binary is not encrypted.</p>	<p>False warning</p> <p>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.</p>

NO	DYLIB/Framework	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
30	Payload/fewerphoto.app/Frameworks/grpcpp.framework/grpcpp	<p>False info</p> <p>The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. However iOS never allows an app to execute from writeable memory. You do not need to specifically enable the 'NX bit' because it's always enabled for all third-party code.</p>	<p>False high</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.</p>	<p>False high</p> <p>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.</p>	<p>False info</p> <p>The binary does not have Runpath Search Path (@rpath) set.</p>	<p>True info</p> <p>This binary has a code signature.</p>	<p>False warning</p> <p>This binary is not encrypted.</p>	<p>False warning</p> <p>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.</p>

NO	DYLIB/Framework	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
31	Payload/fewerphoto.app/Frameworks/FirebaseAnalytics.framework/FirebaseAnalytics	<p>False info</p> <p>The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. However iOS never allows an app to execute from writeable memory. You do not need to specifically enable the 'NX bit' because it's always enabled for all third-party code.</p>	<p>False high</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.</p>	<p>False high</p> <p>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.</p>	<p>False info</p> <p>The binary does not have Runpath Search Path (@rpath) set.</p>	<p>True info</p> <p>This binary has a code signature.</p>	<p>False warning</p> <p>This binary is not encrypted.</p>	<p>False warning</p> <p>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.</p>

NO	DYLIB/Framework	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
32	Payload/fewerphoto.app/Frameworks/IronSource.framework/IronSource	<p>False info</p> <p>The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. However iOS never allows an app to execute from writeable memory. You do not need to specifically enable the 'NX bit' because it's always enabled for all third-party code.</p>	<p>False high</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.</p>	<p>False high</p> <p>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.</p>	<p>False info</p> <p>The binary does not have Runpath Search Path (@rpath) set.</p>	<p>True info</p> <p>This binary has a code signature.</p>	<p>False warning</p> <p>This binary is not encrypted.</p>	<p>False warning</p> <p>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.</p>

NO	DYLIB/Framework	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
33	Payload/fewerphoto.app/Frameworks/OMSDK_Appodeal.framework/OMSDK_Appodeal	<p>False info</p> <p>The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. However iOS never allows an app to execute from writeable memory. You do not need to specifically enable the 'NX bit' because it's always enabled for all third-party code.</p>	<p>True info</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>True info</p> <p>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.</p>	<p>True warning</p> <p>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.</p>	<p>True info</p> <p>This binary has a code signature.</p>	<p>True info</p> <p>This binary is encrypted.</p>	<p>False warning</p> <p>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.</p>

NO	DYLIB/Framework	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
34	Payload/fewerphoto.app/Frameworks/MTGSDKNewInterstitial.framework/MTGSDKNewInterstitial	<p>False info</p> <p>The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. However iOS never allows an app to execute from writeable memory. You do not need to specifically enable the 'NX bit' because it's always enabled for all third-party code.</p>	<p>False high</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.</p>	<p>False high</p> <p>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.</p>	<p>False info</p> <p>The binary does not have Runpath Search Path (@rpath) set.</p>	<p>True info</p> <p>This binary has a code signature.</p>	<p>False warning</p> <p>This binary is not encrypted.</p>	<p>False warning</p> <p>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.</p>

NO	DYLIB/Framework	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
35	Payload/fewerphoto.app/Frameworks/FBSDKCoreKit.framework/FBSDKCoreKit	<p>False info</p> <p>The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. However iOS never allows an app to execute from writeable memory. You do not need to specifically enable the 'NX bit' because it's always enabled for all third-party code.</p>	<p>True info</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>True info</p> <p>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.</p>	<p>True warning</p> <p>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.</p>	<p>True info</p> <p>This binary has a code signature.</p>	<p>True info</p> <p>This binary is encrypted.</p>	<p>False warning</p> <p>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.</p>

NO	DYLIB/Framework	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
36	Payload/fewerphoto.app/Frameworks/BidMachine.framework/BidMachine	<p>False info</p> <p>The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. However iOS never allows an app to execute from writeable memory. You do not need to specifically enable the 'NX bit' because it's always enabled for all third-party code.</p>	<p>False high</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.</p>	<p>False high</p> <p>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.</p>	<p>False info</p> <p>The binary does not have Runpath Search Path (@rpath) set.</p>	<p>True info</p> <p>This binary has a code signature.</p>	<p>False warning</p> <p>This binary is not encrypted.</p>	<p>False warning</p> <p>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.</p>

NO	DYLIB/Framework	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
37	Payload/fewerphoto.app/Frameworks/GoogleAppMeasurement.framework/GoogleAppMeasurement	<p>False info</p> <p>The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. However iOS never allows an app to execute from writeable memory. You do not need to specifically enable the 'NX bit' because it's always enabled for all third-party code.</p>	<p>False high</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.</p>	<p>False high</p> <p>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.</p>	<p>False info</p> <p>The binary does not have Runpath Search Path (@rpath) set.</p>	<p>True info</p> <p>This binary has a code signature.</p>	<p>False warning</p> <p>This binary is not encrypted.</p>	<p>False warning</p> <p>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.</p>

NO	DYLIB/Framework	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
38	Payload/fewerphoto.app/Frameworks/FBAudienceNetwork.framework/FBAudienceNetwork	<p>False info</p> <p>The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. However iOS never allows an app to execute from writeable memory. You do not need to specifically enable the 'NX bit' because it's always enabled for all third-party code.</p>	<p>False high</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.</p>	<p>False high</p> <p>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.</p>	<p>False info</p> <p>The binary does not have Runpath Search Path (@rpath) set.</p>	<p>True info</p> <p>This binary has a code signature.</p>	<p>False warning</p> <p>This binary is not encrypted.</p>	<p>False warning</p> <p>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.</p>

NO	DYLIB/Framework	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
39	Payload/fewerphoto.app/Frameworks/abs.framework/abs	<p>False info</p> <p>The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. However iOS never allows an app to execute from writeable memory. You do not need to specifically enable the 'NX bit' because it's always enabled for all third-party code.</p>	<p>False high</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.</p>	<p>False high</p> <p>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.</p>	<p>False info</p> <p>The binary does not have Runpath Search Path (@rpath) set.</p>	<p>True info</p> <p>This binary has a code signature.</p>	<p>False warning</p> <p>This binary is not encrypted.</p>	<p>False warning</p> <p>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.</p>

NO	DYLIB/Framework	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
40	Payload/fewerphoto.app/Frameworks/StackRendering.framework/StackRendering	<p>False info</p> <p>The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. However iOS never allows an app to execute from writeable memory. You do not need to specifically enable the 'NX bit' because it's always enabled for all third-party code.</p>	<p>False high</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.</p>	<p>False high</p> <p>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.</p>	<p>False info</p> <p>The binary does not have Runpath Search Path (@rpath) set.</p>	<p>True info</p> <p>This binary has a code signature.</p>	<p>False warning</p> <p>This binary is not encrypted.</p>	<p>False warning</p> <p>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.</p>

NO	DYLIB/Framework	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
41	Payload/fewerphoto.app/Frameworks/FBSDKShareKit.framework/FBSDKShareKit	<p>False info</p> <p>The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. However iOS never allows an app to execute from writeable memory. You do not need to specifically enable the 'NX bit' because it's always enabled for all third-party code.</p>	<p>True info</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>True info</p> <p>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.</p>	<p>True warning</p> <p>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.</p>	<p>True info</p> <p>This binary has a code signature.</p>	<p>True info</p> <p>This binary is encrypted.</p>	<p>False warning</p> <p>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.</p>

NO	DYLIB/Framework	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
42	Payload/fewerphoto.app/Frameworks/GoogleAppMeasurementIdentitySupport.framework/GoogleAppMeasurementIdentitySupport	<p>False info</p> <p>The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. However iOS never allows an app to execute from writeable memory. You do not need to specifically enable the 'NX bit' because it's always enabled for all third-party code.</p>	<p>False high</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.</p>	<p>False high</p> <p>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.</p>	<p>False info</p> <p>The binary does not have Runpath Search Path (@rpath) set.</p>	<p>True info</p> <p>This binary has a code signature.</p>	<p>False warning</p> <p>This binary is not encrypted.</p>	<p>False warning</p> <p>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.</p>

NO	DYLIB/Framework	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
43	Payload/fewerphoto.app/Frameworks/SCSDKCreativeKit.framework/SCSDKCreativeKit	<p>False info</p> <p>The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. However iOS never allows an app to execute from writeable memory. You do not need to specifically enable the 'NX bit' because it's always enabled for all third-party code.</p>	<p>True info</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>True info</p> <p>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.</p>	<p>True warning</p> <p>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.</p>	<p>True info</p> <p>This binary has a code signature.</p>	<p>True info</p> <p>This binary is encrypted.</p>	<p>False warning</p> <p>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.</p>

NO	DYLIB/Framework	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
44	Payload/fewerphoto.app/Frameworks/MTGSDKNativeAdvanced.framework/MTGSDKNativeAdvanced	<p>False info</p> <p>The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. However iOS never allows an app to execute from writeable memory. You do not need to specifically enable the 'NX bit' because it's always enabled for all third-party code.</p>	<p>False high</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.</p>	<p>False high</p> <p>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.</p>	<p>False info</p> <p>The binary does not have Runpath Search Path (@rpath) set.</p>	<p>True info</p> <p>This binary has a code signature.</p>	<p>False warning</p> <p>This binary is not encrypted.</p>	<p>False warning</p> <p>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.</p>

NO	DYLIB/Framework	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
45	Payload/fewerphoto.app/Frameworks/FBSDKCoreKit_Basics.framework/FBSDKCoreKit_Basics	<p>False info</p> <p>The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. However iOS never allows an app to execute from writeable memory. You do not need to specifically enable the 'NX bit' because it's always enabled for all third-party code.</p>	<p>True info</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>True info</p> <p>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.</p>	<p>True warning</p> <p>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.</p>	<p>True info</p> <p>This binary has a code signature.</p>	<p>True info</p> <p>This binary is encrypted.</p>	<p>False warning</p> <p>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.</p>

NO	DYLIB/Framework	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
46	Payload/fewerphoto.app/Frameworks/MTGSDKInterstitialVideo.framework/MTGSDKInterstitialVideo	<p>False info</p> <p>The binary does not have NX bit set. NX bit offer protection against exploitation of memory corruption vulnerabilities by marking memory page as non-executable. However iOS never allows an app to execute from writeable memory. You do not need to specifically enable the 'NX bit' because it's always enabled for all third-party code.</p>	<p>False high</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.</p>	<p>False high</p> <p>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.</p>	<p>False info</p> <p>The binary does not have Runpath Search Path (@rpath) set.</p>	<p>True info</p> <p>This binary has a code signature.</p>	<p>False warning</p> <p>This binary is not encrypted.</p>	<p>False warning</p> <p>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.</p>

</> 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
--------	----------------

🔍 DOMAIN MALWARE CHECK

DOMAIN

tizen.org

www.videolan.org

purl.org

www.nullnonmarkingreturnadieresismacronaringacuteabrevegraveadotmacronaringbelowacircumflexdotbelowacircumflexacuteacircumflexildeacircumflexgraveacircumflexhookaboveabreveabreveacuteabrevetildeabrevehookaboveabrevedotbelowinvertedbreveacaron

www.w3.org

www.mwm.ai

DOMAIN

crl.apple.com

appsflyer-skadnetwork.com

ns.adobe.com

ocsp.apple.com

www.apple.com

developer.apple.com

EMAIL

FILE

EMAIL
 f@z.ec

FILE

m2@jo.j3
 1@qyb.ukl
 kəsw@kku.wdjzkb
 re0@s.y5
 tap@gcw.ybq
 hiq@-v.i5
 p@mkm.xf
 2c_@l.c7ybf_
 hi@n.0piv
 ɛ@p.qb
 8@t.cvj
 p█@2wvm_1!
 rm@tkf.qz0nn
 9l@b.roz3gæ
 0g.w@r.fof
 bú@6q9nv.ihag
 m@nuxç.h_ r
 a@1.k31a
 dmě@a.q†
 r@_c_
 n@pəsz.ga
 v-f@no.l_
 wz5e@c.ke
 r@f.jgruxjj
 2@j.fecim
 nyo@j77q.361al
 ĩ@d.fg
 bm6@m.5b
 o@4pwp.ujx
 bf@bth.hm
 _b@o.jq
 n@1.zn
 k@b.ci
 cpioz1@□rt.ma
 qt@s.uzz
 8t@ät.yie
 ɔ@p.egg
 iv@d.0l
 █i@x.lv
 su@cs.8ty
 zl@2.5l
 a@rnztq.tu
 b-@í_6
 kp@h.5k
 -@p.jjv9
 q@g.it
 .@b.fk
 f4□@d.ıawπ
 wj@s.p□jk7
 .s@□.gk
 f@q.mm
 ovj-8μz@4c.eəlǰv
 pq69mxi@a.kmoũ
 y@-.5f□
 fio_j@hp.nwz
 t7@3d.hm
 gh@b.6u
 jf@6zw.ff
 ø@pj.4bƒw

<p>q@ss1zv.pt e@rs.vt</p>	
<p>x@r.rtf EMAIL 0@ul.5e 0@tvw.cq</p>	<p>FILE</p>
<p>b4s@d.5c i@hd.8jxx x@a.3g7hbl 8@n.pn j@jj.gy p etu4nb6@8.to 7@dml.l16 ax@yt.xiuy_dü lprx@z.xgbh 2@8.yt xz@c.sje m.h@vr.uğ sow@ofk.fr z@qf.3b ñpii@xcl.xv e1ax@uhls.vt w+@k.hj e@4ca.gyzgfyue 5@s.nhxç q@u.g8 b@u.uñn o@0pe.heh f@2.qi f@zp.v6 i4@x.wjfeñ c@-xxf9.rl qs@m.eç w@0p.zf oi@h.8p □@c.1l w9@g.lld4 jn@jn._xd3jfc d@2k.8□ q@3.ohdu xy@[_m3 alt@4n.6azl1 oj@opa.zs crn@o.sx bu glbl@f7rx.y.ia s@5c.ot s@s.67 8@vc.ud +kx!j@s.tp l@tp.!!! d@x.sx 7@glmr.λdf mfamein@og.pep fo@kx.r□ z@i.vk z2u@y.job +q021@5e0.if4 xu@6f.tn f@6.to b0zi6g@_j' fom@1g8.v1 rx□@dgbx.gek _k@v.ög 6@9.fnx w@v.bφ t@7.nnu</p>	<p>fewerphoto.app/fewerphoto</p>

0@g.ln c@w.lg	
p@dzi.n_d E-mail uj@x.2ihhluye	FILE
bu@s.mf s@d.c_0gz9 x█@u4f.9kbb7 6jq@8a0.im 8Δrp@vcdwr.cqiy 3ā@æ5txdavo.qn 7@h2d□.yv e@b.r22 s5@z8.pq k@nwu.gpl r@s.ae n@lmx.ay8y f@c.fx al@o.qa 0@tā.vk4 y@85.lb q@axsx.mjfj řc@-g.5s w@t78g3w6z.0qju 2@xmcwf.1s4 q@azz.lnw 0@i.x03w dvfh@h.n5k s@w.5n 6@yb.hw6_ pljp@6.5qz qi@x.cyq□ 8j8@p.2.lp 2as@n.gf4j□ c7@5._ v@j.dq8 ea@jv2f.ckl9 ed@w.bqx tfbA@ys.bk8fci+pp h@1pix.kfk gok@-orl 2@v.lgf e@_dσle óând@h.u.uo p@-tyw.va eq10@5.[H8 d@p.q9 er@ziujuc.a_ 0@zin.kjw█a_ycdf □@ó.ndkit □7@ku.zz q@zh.rd d@gy.zhъf vn@w._dzt zr@wqz.rgy b@eza.g2 zh@4.pwp9c1 ag@t.kv q+@g.□z zq@or.uqw z@rw.3q wy@in.qof ie@ilx.my t@b.cthi j@yfr.1□	

dar@65.hcyv gm@0a.4q e@fjrh.sl	
EMAIL@u.qc 9u@raxh9v30.3dj	FILE
1@x.nj p@cj.kiqw +@ai-h.1vDe 5wptw6@x.kbfuzyf as1p@u.r6	
u@_exs7 knav@mn.rvucd1 brp@e.odv	fewerphoto.app/.AppLovinQualityService/AppLovinQualityService.json
i@o.eb	fewerphoto.app/SC_Info/fewerphoto.supf
i@o.eb	fewerphoto.app/SC_Info/fewerphoto.supp
yi@j.co7 wi@rv.wqluo z@d.dgfn _@1xw1.nz aa8lx@nm.5v i@trz.ku .1@r.dzhw8 7@z5.sn +2@cf4o.l uw@ns8s.os9wxer _rcm@3.hoherjomc	fewerphoto.app/MWMPaywalls_MWMPaywalls.bundle/swp_ios_onboarding_intern_nat.bundle/customOnboardingSwipewipe-slide4.mp4
4nq@k.ts pl-@s.s9 h@gw.h4 8@mw.2.9 k.ej@x.v54 6@lk5.wea wrtwpsth@f.hg 6@r lib.trd f@fk19c.ym ptb@zf6.g3 s0@2.erm k@ejtu.4dfx ods@h.oh.j1 i@kv.cz u@m.8d_jrer p-l@t7l.5zx try@n.gz i.cn@za.xo b@2z.vimh ox@y.w5 i@a.md n@kfm.om	fewerphoto.app/MWMPaywalls_MWMPaywalls.bundle/swp_ios_onboarding_intern_nat.bundle/customOnboardingSwipewipe-slide3.mp4
biA@h@s.drc z@mgken.d9 uf@o.oerkXx mld@m.s_ vf@9j.6a 8q@z.a1 t@vsyf.tvydth	fewerphoto.app/MWMPaywalls_MWMPaywalls.bundle/swp_ios_onboarding_intern_nat.bundle/customOnboardingSwipewipe-slide2.mp4
git@code.musicworld	fewerphoto.app/MWMPaywalls_MWMPaywalls.bundle/swp_ios_onboarding_intern_nat.bundle/Readme.md

EMAIL git@code.musicworld	FILE fewerphoto.app/MWMPaywalls_MWMPaywalls.bundle/Swp_ios_store_control_1024_w030999_y0010999.bundle/Readme.md
l@تép.06l c@t3.06o9ci z7ulqi@wha.■ u	fewerphoto.app/MWMPaywalls_MWMPaywalls.bundle/swp_ios_onboarding_ri_2steps_HIW_next_default.bundle/PaywallRi2stepHIWnext-backgroundVideo.mp4
git@code.musicworld	fewerphoto.app/MWMPaywalls_MWMPaywalls.bundle/swp_ios_onboarding_ri_2steps_HIW_next_default.bundle/Readme.md
sset_template_2steps@1g.png	fewerphoto.app/MWMPaywalls_MWMPaywalls.bundle/swp_ios_onboarding_ri_mkt_questions_v1.bundle/880B90F9-58F6-43EC-BDA4-61970585D1D2.mp4
git@code.musicworld	fewerphoto.app/MWMPaywalls_MWMPaywalls.bundle/Swp_ios_onboarding_control_1024_w030999_y0010999.bundle/Readme.md
-@xjw3j.o2vy p@b.q7 dv@j.jk du@ye.dccâh bme@5.jff ahxn@7r.Δg z@t5.0u 1n@4k.y97 07@5g1ds.km02g 4@degcisk.vA sgî@wz2j.az29 m@f.dbrqzruhâ 0@m.d0ut r@đ6.1w'sn g@qp2.1cy 4@-.dr 8udj@t.tu tc@vz47.[]sw di@p.fe i@z.gzhbkc t@fftz.ch 1h0z@j.x.ă 8k■@[]b.t2l c@m.4v[]řd 3@k.42k j@0.xj09 p@08phjo.bbza un@h.0g wzu@69■rrb x@e.bu kdys@ojq.lwxdn tn@ip.gw6a m@d.dy hut@tbgq.2ir lx@igw.sn	fewerphoto.app/MWMPaywalls_MWMPaywalls.bundle/swp_ios_onboarding_ri_2steps_HIW_next_black.bundle/PaywallRi2stepHIWnext-backgroundVideo.mp4
git@code.musicworld	fewerphoto.app/MWMPaywalls_MWMPaywalls.bundle/swp_ios_onboarding_ri_2steps_HIW_next_black.bundle/Readme.md

EMAIL	FILE
<p>6@q.i0 q@-zf.vf s@o.wµ ozhf@m.fi owe@afs92ws.nq ú@rmj.ug x@n.6_lr v-@m.naw7uψ a@_kp1.i0 z@qa.5x z@n.4a kpwqd@-4.ls ni@5u.8i5</p>	<p>fewerphoto.app/MWMPaywalls_MWMPaywalls.bundle/swp_ios_onboarding_gu_mkt_slide.bundle/74FDEBDD-32D1-463F-94CE-C9788F458240.mp4</p>
<p>uajkl@bu.rk h@oe.ps6a oc@yfd-d.yw4 9hv@v.esk y@p.vē bz@ikkk.iq a@lm.bf</p>	<p>fewerphoto.app/MWMPaywalls_MWMPaywalls.bundle/swp_ios_onboarding_gu_mkt_slide.bundle/9813D14E-7C36-4381-8915-F0A237E7818D.mp4</p>
<p>z@f.0xl9 j@ae.xkxuiχ k@d.gyl o@n7h.ngōvs0 ihnq@u.przkm ftssjq@3eecrrgh.zrkr3 d@p.hxqz avm@e1.cm a@sc.pjka pv@d.ff ac.@c-zmgck.sea 9@swq.mi 2ke@u9ib.zθ r.@cα.xhv</p>	<p>fewerphoto.app/MWMPaywalls_MWMPaywalls.bundle/swp_ios_onboarding_gu_mkt_slide.bundle/7F24047B-7AA1-4F18-911C-79F032F44DF6.mp4</p>
<p>d@e.by x@x.cg q@t5n.bnig x@4f.7u hw@p.tv pcvg@runo.8m q@d.9w 2@bm.v7 x@hb.lbr qe@fqas.l pt@■■_c 9@qlm.mgbagá</p>	<p>fewerphoto.app/MWMPaywalls_MWMPaywalls.bundle/swp_ios_onboarding_gu_mkt_slide.bundle/FED6F491-D408-4DAA-B782-34C493725E7F.mp4</p>
<p>l@تép.o6l c@t3.o6o9ci z7ulqi@wha.■u</p>	<p>fewerphoto.app/MWMPaywalls_MWMPaywalls.bundle/swp_ios_onboarding_ri_2steps_HIW_next_default_v2.bundle/PaywallRi2stepHIWnext-backgroundVideo.mp4</p>
<p>git@code.musicworld</p>	<p>fewerphoto.app/MWMPaywalls_MWMPaywalls.bundle/swp_ios_onboarding_ri_2steps_HIW_next_default_v2.bundle/Readme.md</p>

EMAIL e@1.d.3	FILE fewerphoto.app/PlugIns/InteractiveWidgetExtension.appex/InteractiveWidgetExtension
r@s.ae 5@k.dv	IPA Strings Dump
a@q.zq	Payload/fewerphoto.app/Frameworks/InMobiSDK.framework/InMobiSDK

HARDCODED SECRETS

POSSIBLE SECRETS
FacebookClientToken : 6d2c99bdd66b7e4a5a67f7d65cf9f26d
gcplD : swipewipe-backend
API_KEY : AlzaSyCXQ_ZuctPJOgosfwyhw9O_l7wLJgPinnU
mwmAppToken : e854ce04ae0cfe9d7de65e60a6f2e5f3e7bdf171525e857bd39f8f0be43ca831

APP STORE INFORMATION

Title: Photo Cleaner: Swipewipe

Score: 4.69783 **Features:** iosUniversal, **Price:** 0.0 **Category:** Utilities, Photo & Video,
App Store URL: [com.aoklab.fewerphotos](https://apps.apple.com/us/app/photo-cleaner-swipewipe/id1452269707?uo=4)

Developer: MWM
Developer ID: 452269707

Developer Website: <https://musicworldmedia.com>

Developer URL: <https://apps.apple.com/us/developer/mwm/id452269707?uo=4>

Supported Devices iPhone5s-iPhone5s, iPadAir-iPadAir, iPadAirCellular-iPadAirCellular, iPadMiniRetina-iPadMiniRetina, iPadMiniRetinaCellular-iPadMiniRetinaCellular, iPhone6-iPhone6, iPhone6Plus-iPhone6Plus, iPadAir2-iPadAir2, iPadAir2Cellular-iPadAir2Cellular, iPadMini3-iPadMini3, iPadMini3Cellular-iPadMini3Cellular, iPodTouchSixthGen-iPodTouchSixthGen, iPhone6s-iPhone6s, iPhone6sPlus-iPhone6sPlus, iPadMini4-iPadMini4, iPadMini4Cellular-iPadMini4Cellular, iPadPro-iPadPro, iPadProCellular-iPadProCellular, iPadPro97-iPadPro97, iPadPro97Cellular-iPadPro97Cellular, iPhoneSE-iPhoneSE, iPhone7-iPhone7, iPhone7Plus-iPhone7Plus, iPad611-iPad611, iPad612-iPad612, iPad71-iPad71, iPad72-iPad72, iPad73-iPad73, iPad74-iPad74, iPhone8-iPhone8, iPhone8Plus-iPhone8Plus, iPhoneX-iPhoneX, iPad75-iPad75, iPad76-iPad76, iPhoneXS-iPhoneXS, iPhoneXSMax-iPhoneXSMax, iPhoneXR-iPhoneXR, iPad812-iPad812, iPad834-iPad834, iPad856-iPad856, iPad878-iPad878, iPadMini5-iPadMini5, iPadMini5Cellular-iPadMini5Cellular, iPadAir3-iPadAir3, iPadAir3Cellular-iPadAir3Cellular, iPodTouchSeventhGen-iPodTouchSeventhGen, iPhone11-iPhone11, iPhone11Pro-iPhone11Pro, iPadSeventhGen-iPadSeventhGen, iPadSeventhGenCellular-iPadSeventhGenCellular, iPhone11ProMax-iPhone11ProMax, iPhoneSESecondGen-iPhoneSESecondGen, iPadProSecondGen-iPadProSecondGen, iPadProSecondGenCellular-iPadProSecondGenCellular, iPadProFourthGen-iPadProFourthGen, iPadProFourthGenCellular-iPadProFourthGenCellular, iPhone12Mini-iPhone12Mini, iPhone12-iPhone12, iPhone12Pro-iPhone12Pro, iPhone12ProMax-iPhone12ProMax, iPadAir4-iPadAir4, iPadAir4Cellular-iPadAir4Cellular, iPadEighthGen-iPadEighthGen, iPadEighthGenCellular-iPadEighthGenCellular, iPadProThirdGen-iPadProThirdGen, iPadProThirdGenCellular-iPadProThirdGenCellular, iPadProFifthGen-iPadProFifthGen, iPadProFifthGenCellular-iPadProFifthGenCellular, iPhone13Pro-iPhone13Pro, iPhone13ProMax-iPhone13ProMax, iPhone13Mini-iPhone13Mini, iPhone13-iPhone13, iPadMiniSixthGen-iPadMiniSixthGen, iPadMiniSixthGenCellular-iPadMiniSixthGenCellular, iPadNinthGen-iPadNinthGen, iPadNinthGenCellular-iPadNinthGenCellular, iPhoneSEThirdGen-iPhoneSEThirdGen, iPadAirFifthGen-iPadAirFifthGen, iPadAirFifthGenCellular-iPadAirFifthGenCellular, iPhone14-iPhone14, iPhone14Plus-iPhone14Plus, iPhone14Pro-iPhone14Pro, iPhone14ProMax-iPhone14ProMax, iPadTenthGen-iPadTenthGen, iPadTenthGenCellular-iPadTenthGenCellular, iPadPro11FourthGen-iPadPro11FourthGen, iPadPro11FourthGenCellular-iPadPro11FourthGenCellular, iPadProSixthGen-iPadProSixthGen, iPadProSixthGenCellular-iPadProSixthGenCellular, iPhone15-iPhone15, iPhone15Plus-iPhone15Plus, iPhone15Pro-iPhone15Pro, iPhone15ProMax-iPhone15ProMax, iPadAir11M2-iPadAir11M2, iPadAir11M2Cellular-iPadAir11M2Cellular, iPadAir13M2-iPadAir13M2, iPadAir13M2Cellular-iPadAir13M2Cellular, iPadPro11M4-iPadPro11M4, iPadPro11M4Cellular-iPadPro11M4Cellular, iPadPro13M4-iPadPro13M4, iPadPro13M4Cellular-iPadPro13M4Cellular, iPhone16-iPhone16, iPhone16Plus-iPhone16Plus, iPhone16Pro-iPhone16Pro, iPhone16ProMax-iPhone16ProMax, iPadMiniA17Pro-iPadMiniA17Pro, iPadMiniA17ProCellular-iPadMiniA17ProCellular,

Description:

Swipewipe is the app that will (finally) help you clean up your camera roll. And you'll enjoy reminiscing while you do it. We'll save you the time: Yes, there are other apps that can help you quickly delete photos on your phone. But none of them worked for us! We wanted a simple, fun, elegant solution that let us go month-by-month, work our way through all our photos, videos, screenshots, and everything else in our camera roll, and decide - one by one - what to keep and what to get rid of. That's Swipewipe. Here's how it works: swipe right to keep a photo, and left to delete it. If you make a mistake or change your mind, just tap the current photo to go back. Hold down on a picture to see its metadata. After you're done reviewing that month's photos, take one last look at the photos you chose to keep and the ones you chose to delete, make any tweaks you need to, and then...you're done! Each time you finish a month, it'll be crossed out. (You can always revisit that month, though.) If you get part way through a month and want to take a break, you can quit the app - a progress wheel will appear next to that month on the main screen, showing you how much further you have to go. If you'd rather not go month by month (or even if you do!) we think you'll like our new On This Day feature. It sticks to the top of your Swipewipe home screen, and each day, it updates with the photos that you took on this date one year ago, two years ago, and so on. Revisit your memories on their anniversary, and swipe through to determine what you'd like to keep and what you'd like to delete. (It's pretty fun.) We also have: - Bookmarks

(for any pictures you want to set aside) - A widget (and streaks!) for On This Day - Stats that show you how many photos you've reviewed, how much memory you've saved, and more ...and we're always adding cool new stuff! Our camera rolls shouldn't be such a mess. You should be able to look back at the memories you've made without getting interrupted by blurry duplicates, irrelevant screenshots, and other clutter that keeps you from the good stuff. That's why we're making Swipewipe. Hope you like it, and happy swiping! (Oh, and if you have any feedback, ideas, complaints, life advice, or anything else — email us at hey@swipewipe.app!) Apple Terms of Use Agreement: <https://www.apple.com/legal/internet-services/itunes/dev/stdeula/>

☰ SCAN LOGS

Timestamp	Event	Error
2025-01-11 12:27:39	iOS Binary (IPA) Analysis Started	OK
2025-01-11 12:27:39	Generating Hashes	OK
2025-01-11 12:27:39	Extracting IPA	OK
2025-01-11 12:27:39	Unzipping	OK
2025-01-11 12:27:40	iOS File Analysis and Normalization	OK
2025-01-11 12:27:40	iOS Info.plist Analysis Started	OK
2025-01-11 12:27:40	Finding Info.plist in iOS Binary	OK
2025-01-11 12:27:40	Fetching Details from App Store: com.aoklab.fewerphotos	OK
2025-01-11 12:27:40	Searching for secrets in plist files	OK
2025-01-11 12:27:40	Starting Binary Analysis	OK
2025-01-11 12:27:41	Dumping Classes from the binary	OK
2025-01-11 12:27:41	Running jtool against the binary for dumping classes	OK
2025-01-11 12:27:42	Library Binary Analysis Started	OK
2025-01-11 12:27:42	Framework Binary Analysis Started	OK

2025-01-11 12:27:42	Analyzing Payload/fewerphoto.app/Frameworks/InMobiSDK.framework/InMobiSDK	OK
2025-01-11 12:27:42	Analyzing Payload/fewerphoto.app/Frameworks/FBSDKLoginKit.framework/FBSDKLoginKit	OK
2025-01-11 12:27:42	Analyzing Payload/fewerphoto.app/Frameworks/VungleAdsSDK.framework/VungleAdsSDK	OK
2025-01-11 12:27:42	Analyzing Payload/fewerphoto.app/Frameworks/FBAEMKit.framework/FBAEMKit	OK
2025-01-11 12:27:43	Analyzing Payload/fewerphoto.app/Frameworks/MTGSDKSplash.framework/MTGSDKSplash	OK
2025-01-11 12:27:43	Analyzing Payload/fewerphoto.app/Frameworks/IASDKCore.framework/IASDKCore	OK
2025-01-11 12:27:43	Analyzing Payload/fewerphoto.app/Frameworks/grpc.framework/grpc	OK
2025-01-11 12:27:43	Analyzing Payload/fewerphoto.app/Frameworks/StackProductPresentation.framework/StackProductPresentation	OK
2025-01-11 12:27:43	Analyzing Payload/fewerphoto.app/Frameworks/MWMPublishingSDK.framework/MWMPublishingSDK	OK
2025-01-11 12:27:43	Analyzing Payload/fewerphoto.app/Frameworks/PAGAdSDK.framework/PAGAdSDK	OK
2025-01-11 12:27:43	Analyzing Payload/fewerphoto.app/Frameworks/openssl_grpc.framework/openssl_grpc	OK
2025-01-11 12:27:43	Analyzing Payload/fewerphoto.app/Frameworks/DotLottiePlayer.framework/DotLottiePlayer	OK
2025-01-11 12:27:43	Analyzing Payload/fewerphoto.app/Frameworks/SCSDKLoginKit.framework/SCSDKLoginKit	OK
2025-01-11 12:27:43	Analyzing Payload/fewerphoto.app/Frameworks/MTGSDKReward.framework/MTGSDKReward	OK
2025-01-11 12:27:43	Analyzing Payload/fewerphoto.app/Frameworks/DTBiOSSDK.framework/DTBiOSSDK	OK
2025-01-11 12:27:43	Analyzing Payload/fewerphoto.app/Frameworks/PaywallTemplates.framework/PaywallTemplates	OK
2025-01-11 12:27:43	Analyzing Payload/fewerphoto.app/Frameworks/FirebaseFirestoreInternal.framework/FirebaseFirestoreInternal	OK

2025-01-11 12:27:43	Analyzing Payload/fewerphoto.app/Frameworks/MolocoSDK.framework/MolocoSDK	OK
2025-01-11 12:27:43	Analyzing Payload/fewerphoto.app/Frameworks/UnityAds.framework/UnityAds	OK
2025-01-11 12:27:43	Analyzing Payload/fewerphoto.app/Frameworks/MTGSDKBidding.framework/MTGSDKBidding	OK
2025-01-11 12:27:43	Analyzing Payload/fewerphoto.app/Frameworks/AppLovinSDK.framework/AppLovinSDK	OK
2025-01-11 12:27:43	Analyzing Payload/fewerphoto.app/Frameworks/StackModules.framework/StackModules	OK
2025-01-11 12:27:43	Analyzing Payload/fewerphoto.app/Frameworks/AppLovinQualityService.framework/AppLovinQualityService	OK
2025-01-11 12:27:43	Analyzing Payload/fewerphoto.app/Frameworks/UserMessagingPlatform.framework/UserMessagingPlatform	OK
2025-01-11 12:27:43	Analyzing Payload/fewerphoto.app/Frameworks/SCSDKCoreKit.framework/SCSDKCoreKit	OK
2025-01-11 12:27:43	Analyzing Payload/fewerphoto.app/Frameworks/MTGSDK.framework/MTGSDK	OK
2025-01-11 12:27:43	Analyzing Payload/fewerphoto.app/Frameworks/BUReLyFoundation_Global.framework/BUReLyFoundation_Global	OK
2025-01-11 12:27:43	Analyzing Payload/fewerphoto.app/Frameworks/MTGSDKBanner.framework/MTGSDKBanner	OK
2025-01-11 12:27:43	Analyzing Payload/fewerphoto.app/Frameworks/GoogleMobileAds.framework/GoogleMobileAds	OK
2025-01-11 12:27:43	Analyzing Payload/fewerphoto.app/Frameworks/grpcpp.framework/grpcpp	OK
2025-01-11 12:27:43	Analyzing Payload/fewerphoto.app/Frameworks/FirebaseAnalytics.framework/FirebaseAnalytics	OK
2025-01-11 12:27:43	Analyzing Payload/fewerphoto.app/Frameworks/IronSource.framework/IronSource	OK
2025-01-11 12:27:43	Analyzing Payload/fewerphoto.app/Frameworks/OMSDK_Appodeal.framework/OMSDK_Appodeal	OK
2025-01-11 12:27:43	Analyzing Payload/fewerphoto.app/Frameworks/MTGSDKNewInterstitial.framework/MTGSDKNewInterstitial	OK

2025-01-11 12:27:43	Analyzing Payload/fewerphoto.app/Frameworks/FBSDKCoreKit.framework/FBSDKCoreKit	OK
2025-01-11 12:27:43	Analyzing Payload/fewerphoto.app/Frameworks/BidMachine.framework/BidMachine	OK
2025-01-11 12:27:43	Analyzing Payload/fewerphoto.app/Frameworks/GoogleAppMeasurement.framework/GoogleAppMeasurement	OK
2025-01-11 12:27:43	Analyzing Payload/fewerphoto.app/Frameworks/FBAudienceNetwork.framework/FBAudienceNetwork	OK
2025-01-11 12:27:43	Analyzing Payload/fewerphoto.app/Frameworks/absl.framework/absl	OK
2025-01-11 12:27:43	Analyzing Payload/fewerphoto.app/Frameworks/StackRendering.framework/StackRendering	OK
2025-01-11 12:27:43	Analyzing Payload/fewerphoto.app/Frameworks/FBSDKShareKit.framework/FBSDKShareKit	OK
2025-01-11 12:27:44	Analyzing Payload/fewerphoto.app/Frameworks/GoogleAppMeasurementIdentitySupport.framework/GoogleAppMeasurementIdentitySupport	OK
2025-01-11 12:27:44	Analyzing Payload/fewerphoto.app/Frameworks/SCSDKCreativeKit.framework/SCSDKCreativeKit	OK
2025-01-11 12:27:44	Analyzing Payload/fewerphoto.app/Frameworks/MTGSDKNativeAdvanced.framework/MTGSDKNativeAdvanced	OK
2025-01-11 12:27:44	Analyzing Payload/fewerphoto.app/Frameworks/FBSDKCoreKit_Basics.framework/FBSDKCoreKit_Basics	OK
2025-01-11 12:27:44	Analyzing Payload/fewerphoto.app/Frameworks/MTGSDKInterstitialVideo.framework/MTGSDKInterstitialVideo	OK
2025-01-11 12:27:44	Extracting String Metadata	OK
2025-01-11 12:27:44	Extracting URL and Email from IPA	OK
2025-01-11 12:27:50	Performing Malware check on extracted domains	OK
2025-01-11 12:27:52	Fetching IPA icon path	OK
2025-01-11 12:27:53	Detecting Trackers from Domains	OK

2025-01-11 12:27:53	Saving to Database	OK
---------------------	--------------------	----

Report Generated by - MobSF v4.2.9

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

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