IOS STATIC ANALYSIS REPORT

app_icon

iGoat-Swift (1.0)

File Name: iGoat-Swift.ipa

Identifier:	OWASP.iGoat-Swift
Scan Date:	Aug. 18, 2024, 11:28 a.m.
App Security Score:	16/100 (CRITICAL RISK)
Grade:	
Trackers Detection:	1/432

FINDINGS SEVERITY

HIGH	MEDIUM	INFO	SECURE	HOTSPOT
4	2	2	0	0

FILE INFORMATION

File Name: iGoat-Swift.ipa
Size: 15.93MB
MD5: e73a7bf48e090a445febc06253a2ae60
SHA1: e560f00633d96a40f1d0f949ff3a854830e3af50
SHA256: 364273106c7fdb7b627bf7821a1539af4044025bf7190ebb760afb4b85c15a47

APP INFORMATION

App Name: iGoat-Swift
App Type: Swift
Identifier: OWASP.iGoat-Swift
SDK Name: iphoneos13.2
Version: 1.0
Build: 1
Platform Version: 13.2
Min OS Version: 10.0
Supported Platforms: iPhoneOS,

BINARY INFORMATION

Arch: ARM

Sub Arch: CPU_SUBTYPE_ARM_V7
Bit: 32-bit
Endian: <

#CUSTOM URL SCHEMES

URL NAME	SCHEMES
com.iGoat.myCompany Editor	iGoat

APPLICATION PERMISSIONS

PERMISSIONS	STATUS	INFO	REASON IN MANIFEST
NSFaceIDUsageDescription	normal	Access the ability to authenticate with Face ID.	iGoat would like to use FaceID to authenticate you.

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

NO	ISSUE	SEVERITY	STANDARDS	DESCRIPTION
1	Binary makes use of insecure API(s)	high	**CWE:** CWE-676: Use of Potentially Dangerous Function **OWASP Top 10:** M7: Client Code Quality **OWASP MASVS:** MSTG-CODE-8	The binary may contain the following insecure API(s) _fopen , _memcpy , _strcpy , _strlen , _strncpy
2	Binary makes use of the insecure Random function(s)	high	**CWE:** CWE-330: Use of Insufficiently Random Values **OWASP Top 10:** M5: Insufficient Cryptography **OWASP MASVS:** MSTG-CRYPTO-6	The binary may use the following insecure Random function(s) _random
3	Binary makes use of Logging function	info	**CWE:** CWE-532: Insertion of Sensitive Information into Log File **OWASP MASVS:** MSTG-STORAGE-3	The binary may use _NSLog function for logging.
4	Binary makes use of malloc function	high	**CWE:** CWE-789: Uncontrolled Memory Allocation **OWASP Top 10:** M7: Client Code Quality **OWASP MASVS:** MSTG-CODE-8	The binary may use _malloc function instead of calloc
5	Binary uses WebView Component.	info	**OWASP MASVS:** MSTG-CODE-9	The binary may use UIWebView Component.

IPA BINARY ANALYSIS

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

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

DYNAMIC LIBRARY & FRAMEWORK BINARY ANALYSIS

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

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

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

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

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

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

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

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

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

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

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

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

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

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

NO	DYLIB/FRAMEWORK	NX	STACK CANARY	ARC	RPATH	CODE SIGNATURE	ENCRYPTED	SYMBOLS STRIPPED
1	Payload/iGoat-Swift.app/Frameworks/Realm.framework/Realm	True info The binary has NX bit set. This marks a memory page non-executable making attacker injected shellcode non-executable.	True info This binary has a stack canary value added to the stack so that it will be overwritten by a stack buffer that overflows the return address. This allows detection of overflows by verifying the integrity of the canary before function return.	True info The binary is compiled with Automatic Reference Counting (ARC) flag. ARC is a compiler feature that provides automatic memory management of Objective-C objects and is an exploit mitigation mechanism against memory corruption vulnerabilities.	True warning The binary has Runpath Search Path (@rpath) set. In certain cases an attacker can abuse this feature to run arbitrary executable for code execution and privilege escalation. Remove the compiler option -rpath to remove @rpath.	True info This binary has a code signature.	False warning This binary is not encrypted.	False warning Debug Symbols are available. To strip debugging symbols, set Strip Debug Symbols During Copy to YES, Deployment Postprocessing to YES, and Strip Linked Product to YES in project's build settings.

</> CODE ANALYSIS

NO	ISSUE	SEVERITY	STANDARDS	FILES

! OFAC SANCTIONED COUNTRIES

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

DOMAIN	COUNTRY/REGION

DOMAIN MALWARE CHECK

DOMAIN	STATUS	GEOLOCATION
www.linkedin.com	ok	**IP:** 13.107.42.14 **Country:** United Kingdom of Great Britain and Northern Ireland **Region:** England **City:** London **Latitude:** 51.508530 **Longitude:** -0.125740 **View:** Google Map
s3.us-east-2.amazonaws.com	ok	**IP:** 52.219.228.65 **Country:** United States of America **Region:** Washington **City:** Seattle **Latitude:** 47.627499 **Longitude:** -122.346199 **View:** Google Map

DOMAIN	STATUS	GEOLOCATION
ocsp.apple.com	ok	**IP:** 184.50.250.161 **Country:** United States of America **Region:** Colorado **City:** Greenwood Village **Latitude:** 39.617210 **Longitude:** -104.950813 **View:** Google Map
m.youtube.com	ok	**IP:** 142.250.79.14 **Country:** United States of America **Region:** California **City:** Mountain View **Latitude:** 37.405991 **Longitude:** -122.078514 **View:** Google Map
www.igoatapp.com	ok	**IP:** 185.199.109.153 **Country:** United States of America **Region:** Pennsylvania **City:** California **Latitude:** 40.065632 **Longitude:** -79.891708 **View:** Google Map
www.owasp.org	ok	**IP:** 172.67.10.39 **Country:** United States of America **Region:** California **City:** San Francisco **Latitude:** 37.775700 **Longitude:** -122.395203 **View:** Google Map

DOMAIN	STATUS	GEOLOCATION
www.arxan.com	ok	**IP:** 3.224.123.132 **Country:** United States of America **Region:** Virginia **City:** Ashburn **Latitude:** 39.043720 **Longitude:** -77.487488 **View:** Google Map
www.w3.org	ok	**IP:** 104.18.22.19 **Country:** United States of America **Region:** California **City:** San Francisco **Latitude:** 37.775700 **Longitude:** -122.395203 **View:** Google Map
www.apple.com	ok	**IP:** 23.37.12.246 **Country:** Philippines **Region:** Cebu **City:** Cebu City **Latitude:** 10.316720 **Longitude:** 123.890709 **View:** Google Map
realm.io	ok	**IP:** 108.139.113.113 **Country:** United States of America **Region:** Washington **City:** Seattle **Latitude:** 47.606209 **Longitude:** -122.332069 **View:** Google Map

DOMAIN	STATUS	GEOLOCATION
www.paypalobjects.com	ok	**IP:** 151.101.3.1 **Country:** United States of America **Region:** California **City:** San Francisco **Latitude:** 37.775700 **Longitude:** -122.395203 **View:** Google Map
api.mixpanel.com	ok	**IP:** 35.186.241.51 **Country:** United States of America **Region:** Missouri **City:** Kansas City **Latitude:** 39.099731 **Longitude:** -94.578568 **View:** Google Map
twitter.com	ok	**IP:** 104.244.42.129 **Country:** United States of America **Region:** California **City:** San Francisco **Latitude:** 37.773968 **Longitude:** -122.410446 **View:** Google Map
crl.apple.com	ok	**IP:** 17.253.10.204 **Country:** United States of America **Region:** Massachusetts **City:** Boston **Latitude:** 42.358429 **Longitude:** -71.059769 **View:** Google Map

DOMAIN	STATUS	GEOLOCATION
www.krvw.com	ok	**IP:** 66.207.131.17 **Country:** United States of America **Region:** Pennsylvania **City:** Pittsburgh **Latitude:** 40.464062 **Longitude:** -79.947060 **View:** Google Map
platform.twitter.com	ok	**IP:** 151.101.176.157 **Country:** United States of America **Region:** California **City:** San Francisco **Latitude:** 37.775700 **Longitude:** -122.395203 **View:** Google Map
www.github.com	ok	**IP:** 20.201.28.151 **Country:** United States of America **Region:** Washington **City:** Redmond **Latitude:** 47.682899 **Longitude:** -122.120903 **View:** Google Map
github.com	ok	**IP:** 20.201.28.151 **Country:** United States of America **Region:** Washington **City:** Redmond **Latitude:** 47.682899 **Longitude:** -122.120903 **View:** Google Map

DOMAIN	STATUS	GEOLOCATION
www.paypal.com	ok	**IP:** 151.101.65.21 **Country:** United States of America **Region:** California **City:** San Francisco **Latitude:** 37.775700 **Longitude:** -122.395203 **View:** Google Map

EMAILS

EMAIL	FILE
h@f.0f f@5.o1 f@f.lxf 0f@f.jl0 fx@ftpfyf.5iqfzf johndoe@yap.com john@test.com f@fin.qjf fs4pfs0@fs.0f x@f.n1 h@f.eva fj4@fj20fj.xfj h@f-.0f h@f.a0 frf3fg@f20f.pf	iGoat-Swift.app/iGoat-Swift
paypal@owasp.org swaroop.yermalkar@owasp.org	iGoat-Swift.app/splash.html
ophychius@gmail.com	iGoat-Swift.app/rutger.html
swaroop.yermalkar@owasp.org	iGoat-Swift.app/Swaroop.html

EMAIL	FILE
swaroop.yermalkar@owasp.org	iGoat-Swift.app/Swaroop_anthony.html
swaroop.yermalkar@owasp.org	iGoat-Swift.app/Swaroop_Junard.html
swaroop.yermalkar@owasp.org	iGoat-Swift.app/Swaroop_Heefan.html
mansi.sheth@gmail.com	iGoat-Swift.app/mansi.html
jcarter@arxan.com ken@krvw.com	iGoat-Swift.app/KRvWAssociates.html
john@test.com johndoe@yap.com	IPA Strings Dump
help@realm.io	Payload/iGoat-Swift.app/Frameworks/Realm.framework/Realm

TRACKERS

TRACKER	CATEGORIES	URL
MixPanel	Analytics	https://reports.exodus-privacy.eu.org/trackers/118

SCAN LOGS

Timestamp	Event	Error
2024-08-18 11:27:51	iOS Binary (IPA) Analysis Started	OK

2024-08-18 11:27:51	Generating Hashes	OK
2024-08-18 11:27:51	Extracting IPA	OK
2024-08-18 11:27:51	Unzipping	OK
2024-08-18 11:27:51	iOS File Analysis and Normalization	OK
2024-08-18 11:27:51	iOS Info.plist Analysis Started	OK
2024-08-18 11:27:51	Finding Info.plist in iOS Binary	OK
2024-08-18 11:27:51	Fetching Details from App Store: OWASP.iGoat-Swift	OK
2024-08-18 11:27:51	Searching for secrets in plist files	OK
2024-08-18 11:27:51	Starting Binary Analysis	OK
2024-08-18 11:27:52	Dumping Classes from the binary	OK
2024-08-18 11:27:52	Running jtool against the binary for dumping classes	OK
2024-08-18 11:27:53	Library Binary Analysis Started	OK

2024-08-18 11:27:53	Analyzing Payload/iGoat-Swift.app/Frameworks/libswiftos.dylib	OK
2024-08-18 11:27:53	Analyzing Payload/iGoat-Swift.app/Frameworks/libswiftFoundation.dylib	OK
2024-08-18 11:27:54	Analyzing Payload/iGoat-Swift.app/Frameworks/libswiftUIKit.dylib	OK
2024-08-18 11:27:55	Analyzing Payload/iGoat-Swift.app/Frameworks/libswiftObjectiveC.dylib	OK
2024-08-18 11:27:55	Analyzing Payload/iGoat-Swift.app/Frameworks/libswiftCoreData.dylib	OK
2024-08-18 11:27:55	Analyzing Payload/iGoat-Swift.app/Frameworks/libswiftCoreImage.dylib	OK
2024-08-18 11:27:55	Analyzing Payload/iGoat-Swift.app/Frameworks/libswiftMetal.dylib	OK
2024-08-18 11:27:55	Analyzing Payload/iGoat-Swift.app/Frameworks/libswiftDispatch.dylib	OK
2024-08-18 11:27:55	Analyzing Payload/iGoat-Swift.app/Frameworks/libswiftDarwin.dylib	OK
2024-08-18 11:27:55	Analyzing Payload/iGoat-Swift.app/Frameworks/libswiftCore.dylib	OK
2024-08-18 11:27:59	Analyzing Payload/iGoat-Swift.app/Frameworks/libswiftCoreGraphics.dylib	OK
2024-08-18 11:28:00	Analyzing Payload/iGoat-Swift.app/Frameworks/libswiftQuartzCore.dylib	OK

2024-08-18 11:28:00	Analyzing Payload/iGoat-Swift.app/Frameworks/libswiftCoreFoundation.dylib	OK
2024-08-18 11:28:00	Framework Binary Analysis Started	OK
2024-08-18 11:28:00	Analyzing Payload/iGoat-Swift.app/Frameworks/Realm.framework/Realm	OK
2024-08-18 11:28:05	Fetching IPA icon path	OK
2024-08-18 11:28:05	Extracting String Metadata	OK
2024-08-18 11:28:05	Extracting URL and Email from IPA	OK
2024-08-18 11:28:06	Performing Malware check on extracted domains	OK
2024-08-18 11:28:13	Detecting Trackers from Domains	OK
2024-08-18 11:28:13	Updating Database...	OK

Report Generated by - MobSF v4.0.6

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.