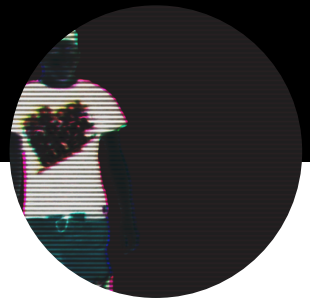
ANÁLISIS DINÁMICO DE APLICACIONES EN iOS Y watchOS

SIN JAILBREAK



amnesia

Deloitte.



RootedCON:~$ whoami

Arán Lora @4r4nL

Senior Security Analyst en Deloitte.

Miembro de amnesia

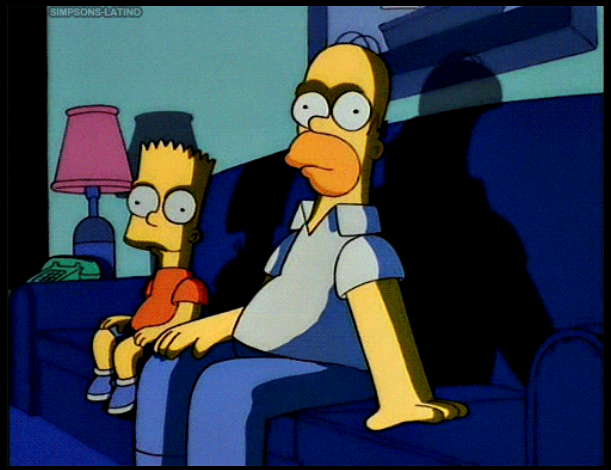
Objetivos

Como poder **manipular** proceso “**on-the-fly**” en **iOS** y **watchOS sin JB**



1.
La teoría

un poco de base nunca viene mal...



IPA FILE FORMAT

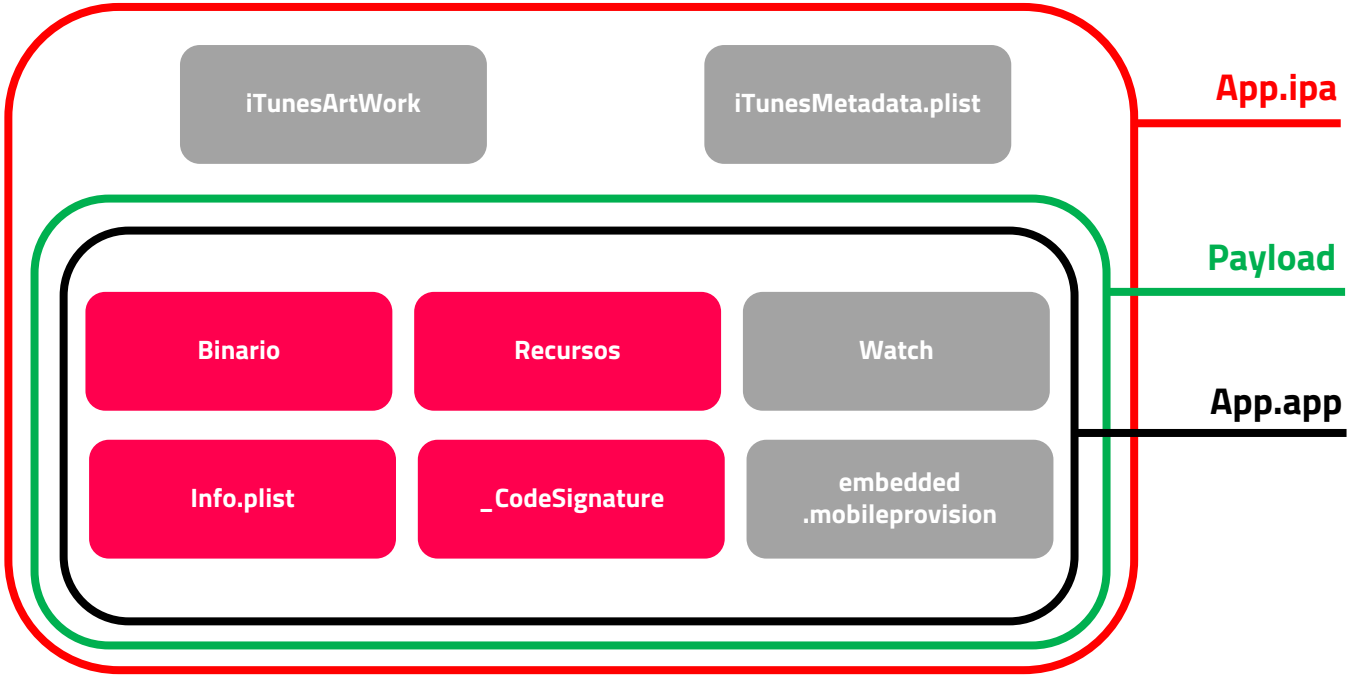


- Fichero ZIP o LZFSE
- Contiene la APP para iOS y watchOS (.app)
- Instalable mediante USB y Over-The-Air
- App Store -> Cifrados (FairPlay)
- Firmado por Apple
- Se ejecutan en SandBox



ANATOMÍA


IPA


iTunesArtWork
iTunesMetadata.plist
App.ipa
Payload
Binario
Recursos
Watch
App.app
Info.plist
_CodeSignature
embedded
.mobileprovision
Requerido
Opcional

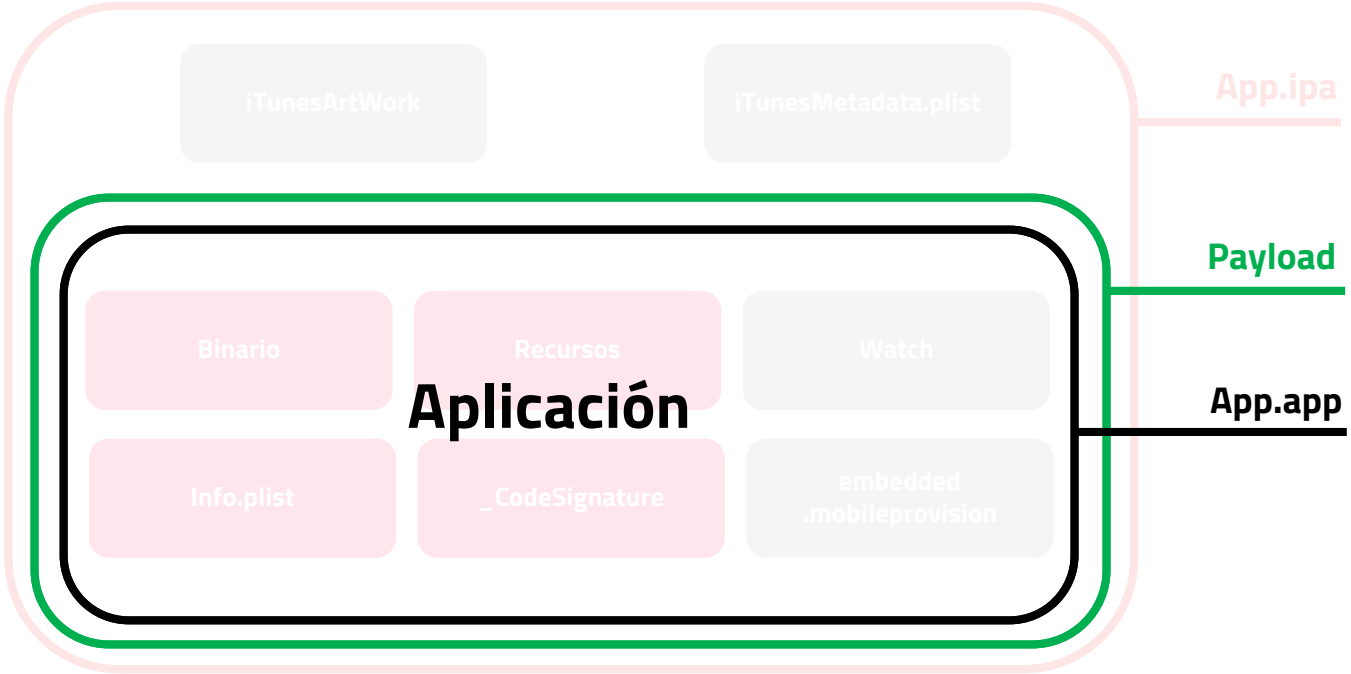
ANATOMÍA


IPA


iTunesArtWork
iTunesMetadata.plist
App.ipa
Payload
Icono de la app para la App Store e iTunes
Metadatos de la app para la App Store e iTunes
App.app
Info.plist
_CodeSignature
embedded .mobileprovision
Requerido
Opcional

ANATOMÍA


IPA


iTunesArtWork
iTunesMetadata.plist
App.ipa
Payload
Binario
Recursos
Watch
Aplicación
App.app
Info.plist
_CodeSignature
embedded
.mobileprovision
Requerido
Opcional

ANATOMÍA



iTunesArtWork

iTunesMetadata.plist

App.ipa

Payload

App.app

Binario

Binario de la app

Info.plist

Configuración de la app

Requerido

Opcional

* Info.plist Keys and Values:
https://developer.apple.com/library/archive/documentation/General/Reference/InfoPlistKeyReference/Introduction/Introduction.html

ANATOMÍA



Diferentes tipos de ficheros:

- Imágenes (PNG,JPG,..)
- Strings
- Assets.car
- ...

- Librerías (Frameworks, dylib)
- UI (nib, storyboardc)
- plist
- ...



ANATOMÍA



iTunesArtWork

iTunesMetadata.plist

App.ipa

ID desarrollador

UUID Dispositivos

ID App

Payload

Binario

Recursos

Entitlements

App.app

Firma de todos los ficheros

_CodeSignature

embedded
.mobileprovision

Requerido

Opcional

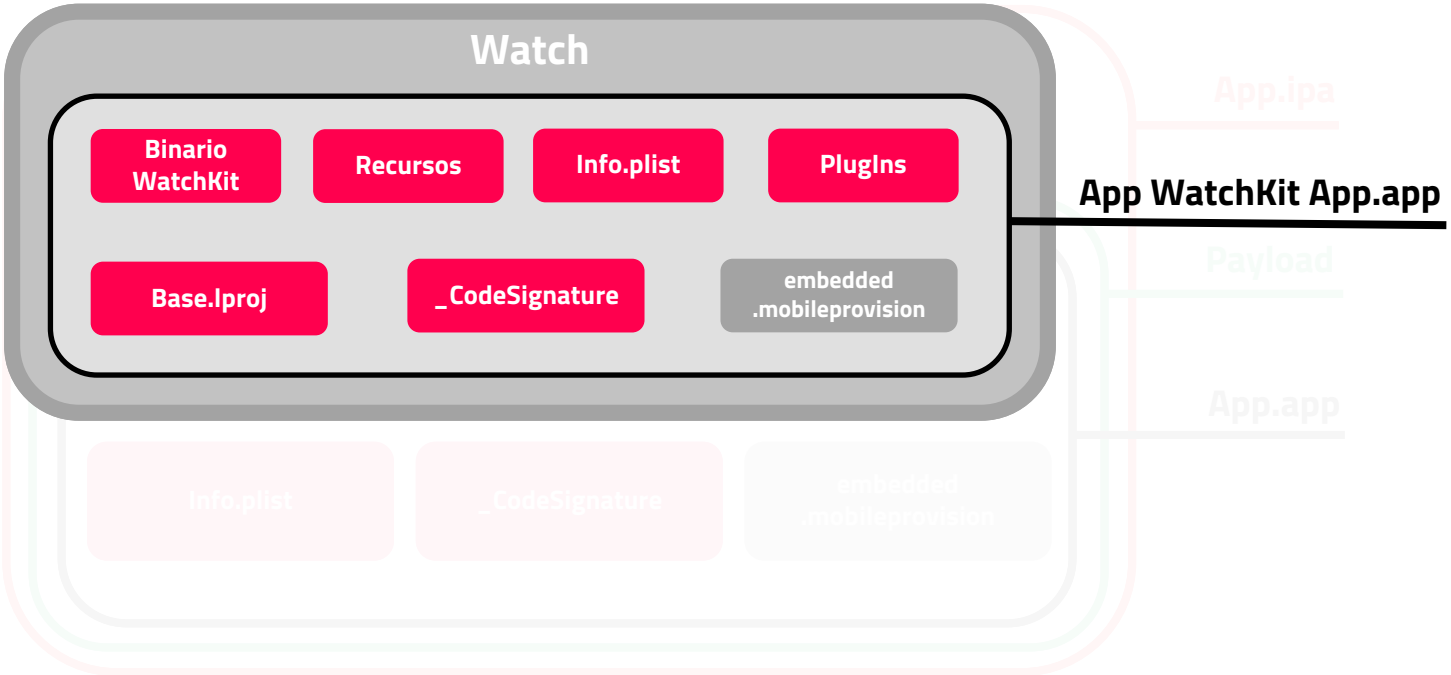
ANATOMÍA


IPA


iTunesArtWork
iTunesMetadata.plist
App.ipa
Carpeta que contiene
la app del Watch
Payload
Binario
Recursos
Watch
App.app
Info.plist
_CodeSignature
embedded
.mobileprovision
Requerido
Opcional

ANATOMÍA


IPA


Watch
Binario WatchKit
Recursos
Info.plist
PlugIns
Base.lproj
_CodeSignature
embedded .mobileprovision
App WatchKit App.app
Requerido
Opcional

ANATOMÍA



Watch

Binario WatchKit

Recursos

Info.plist

PlugIns

Base.lproj

_CodeSignature

embedded .mobileprovision

App WatchKit App.app

Igual que antes

El binario solo carga la UI

Requerido

Opcional

ANATOMÍA



Watch

Binario WatchKit

Recursos

Info.plist

PlugIns

Base.lproj

_CodeSignature

embedded .mobileprovision

App WatchKit App.app

UI de la app (storyboardc)

Requerido

Opcional

ANATOMÍA



Watch

Interface.storyboardc

Interface.plist

Interface-notification.plist

Base.lproj

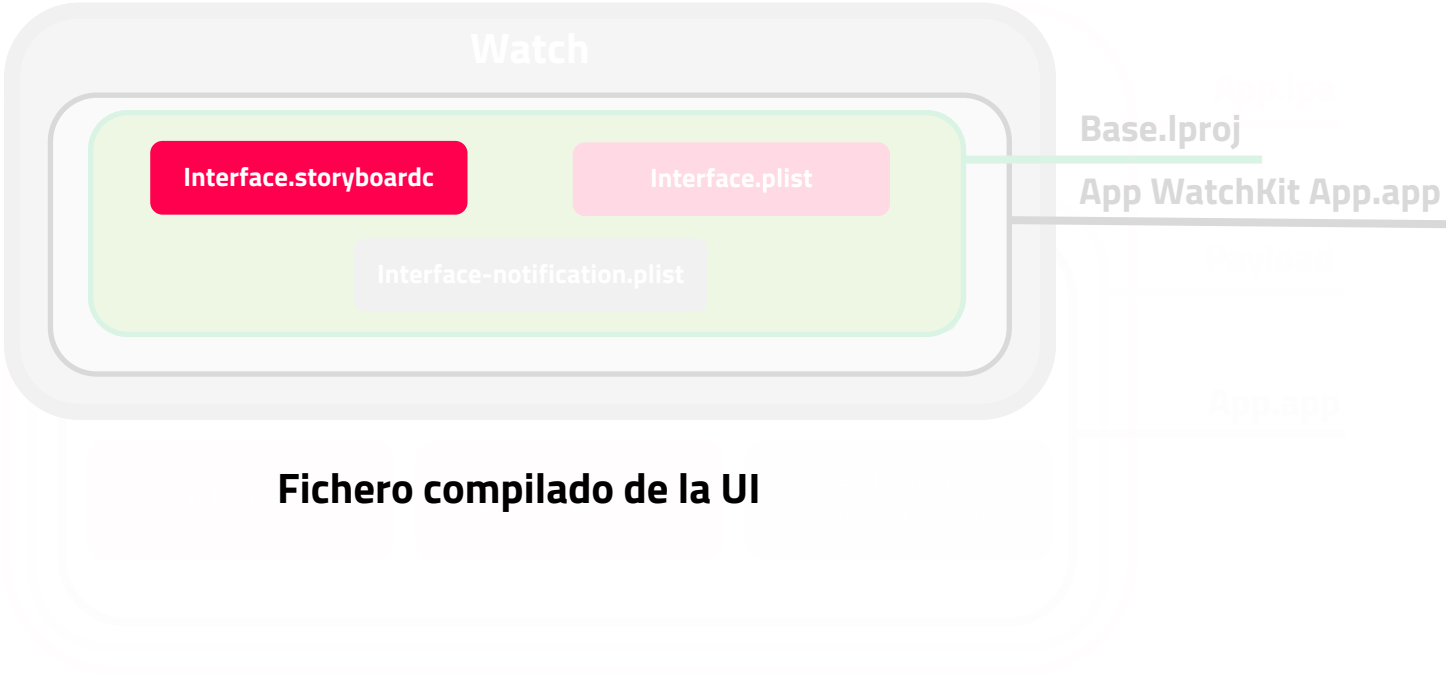
App WatchKit App.app

Requerido

Opcional

ANATOMÍA





Fichero compilado de la UI

Requerido

Opcional

ANATOMÍA



Watch

Interface.storyboardc

Interface.plist

Interface-notification.plist

Base.lproj

App WatchKit App.app

Configuración de las vistas

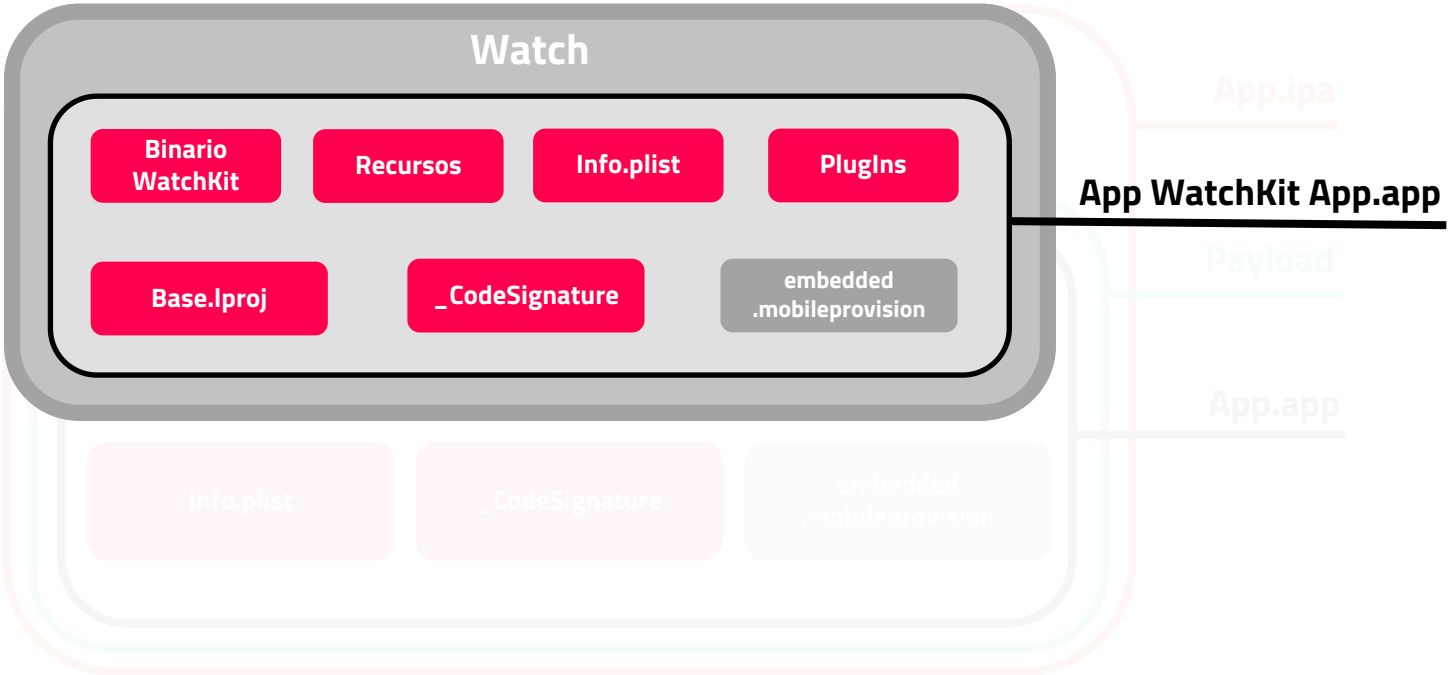
Permiten ver las acciones

Requerido

Opcional

ANATOMÍA


IPA


Watch
Binario WatchKit
Recursos
Info.plist
PlugIns
Base.lproj
_CodeSignature
embedded .mobileprovision
App WatchKit App.app
Requerido
Opcional

ANATOMÍA



Watch

Binario | Recursos | Info.plist | PlugIns

Base.lproj | _CodeSignature | embedded .mobileprovision

App WatchKit App.app

Contiene WatchKit Extension

Requerido

Opcional

ANATOMÍA



Watch

Binario WatchKit Extension

Recursos

Info.plist

embedded .mobileprovision

_CodeSignature

App WatchKit App.app

PlugIns

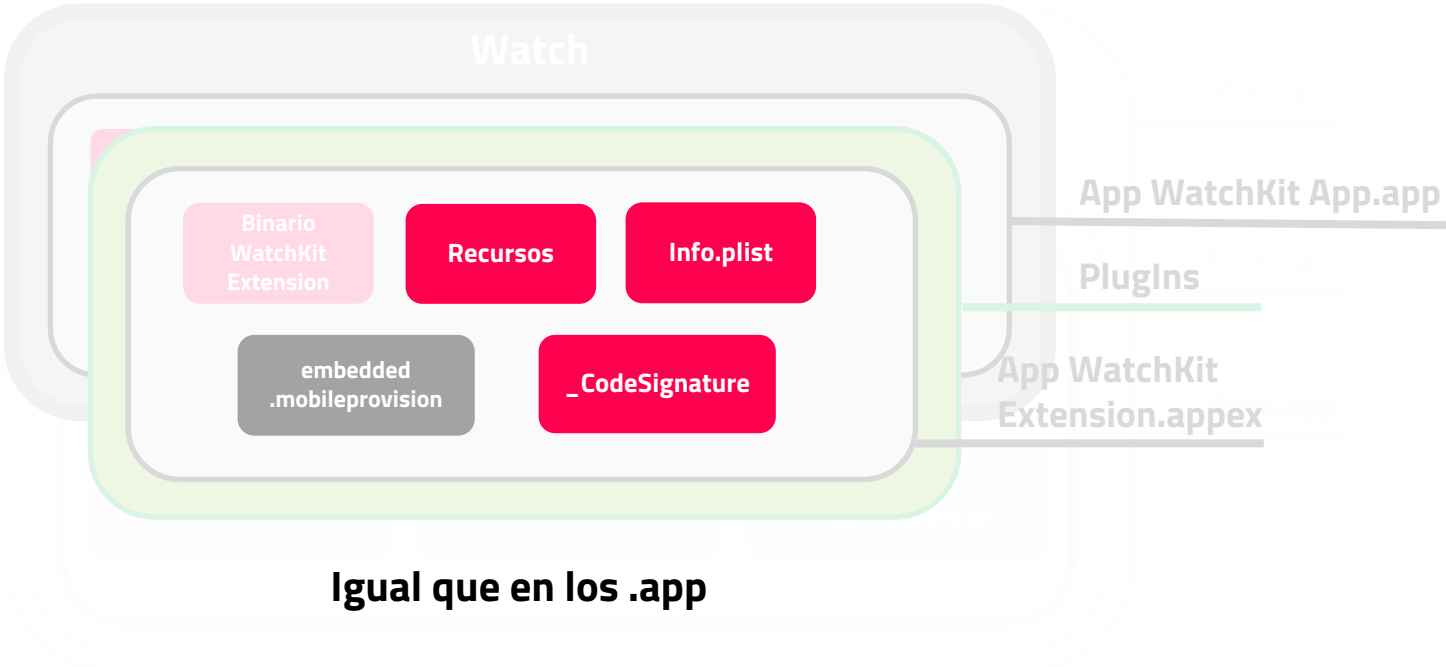
App WatchKit Extension.appex

Requerido

Opcional

ANATOMÍA

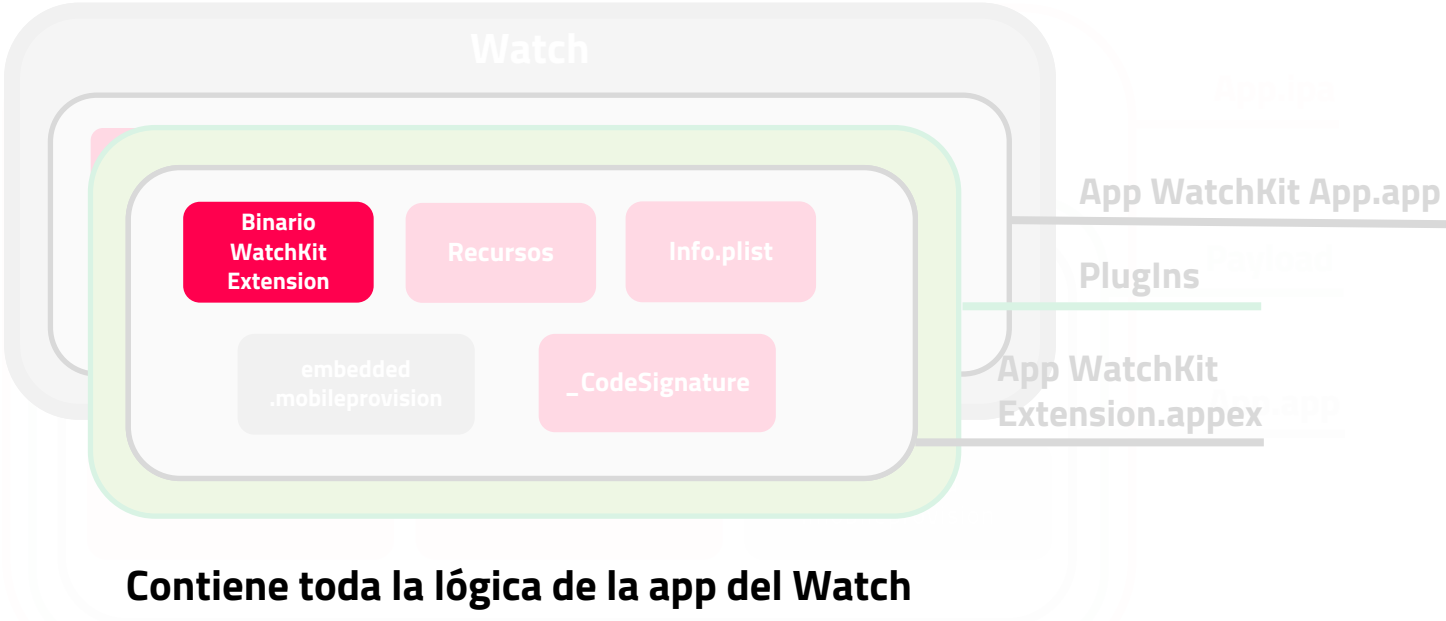




Igual que en los .app

ANATOMÍA

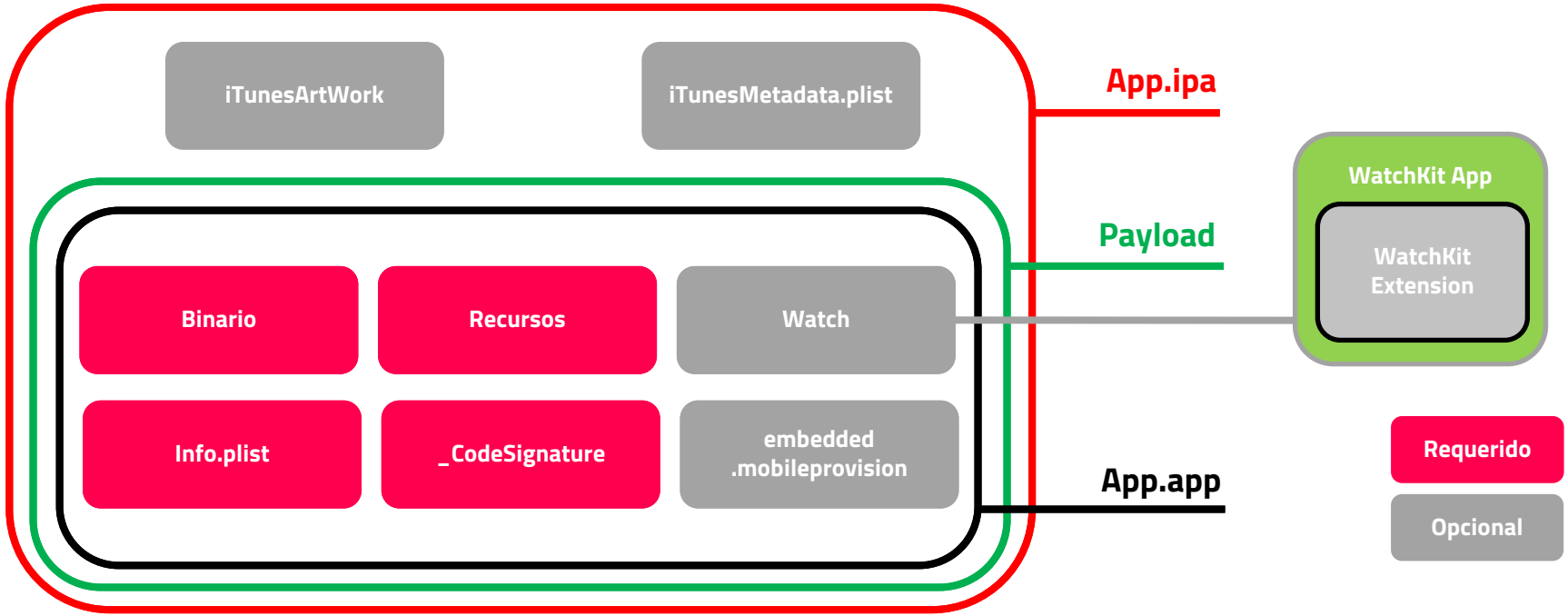




Contiene toda la lógica de la app del Watch

ANATOMÍA


IPA

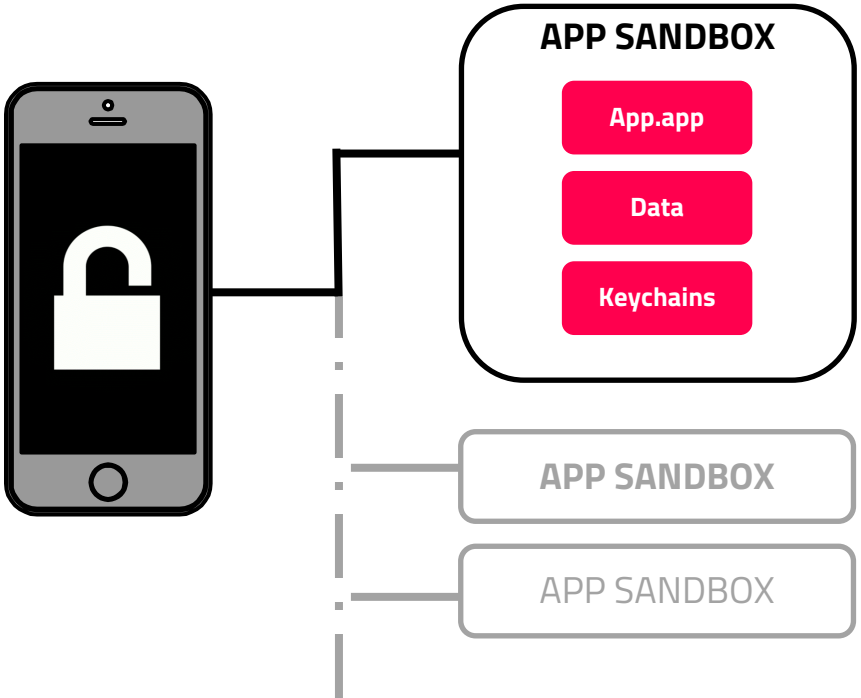

iTunesArtWork
iTunesMetadata.plist
App.ipa
WatchKit App
Payload
WatchKit Extension
Binario
Recursos
Watch
Info.plist
_CodeSignature
embedded .mobileprovision
App.app
Requerido
Opcional

al final es....

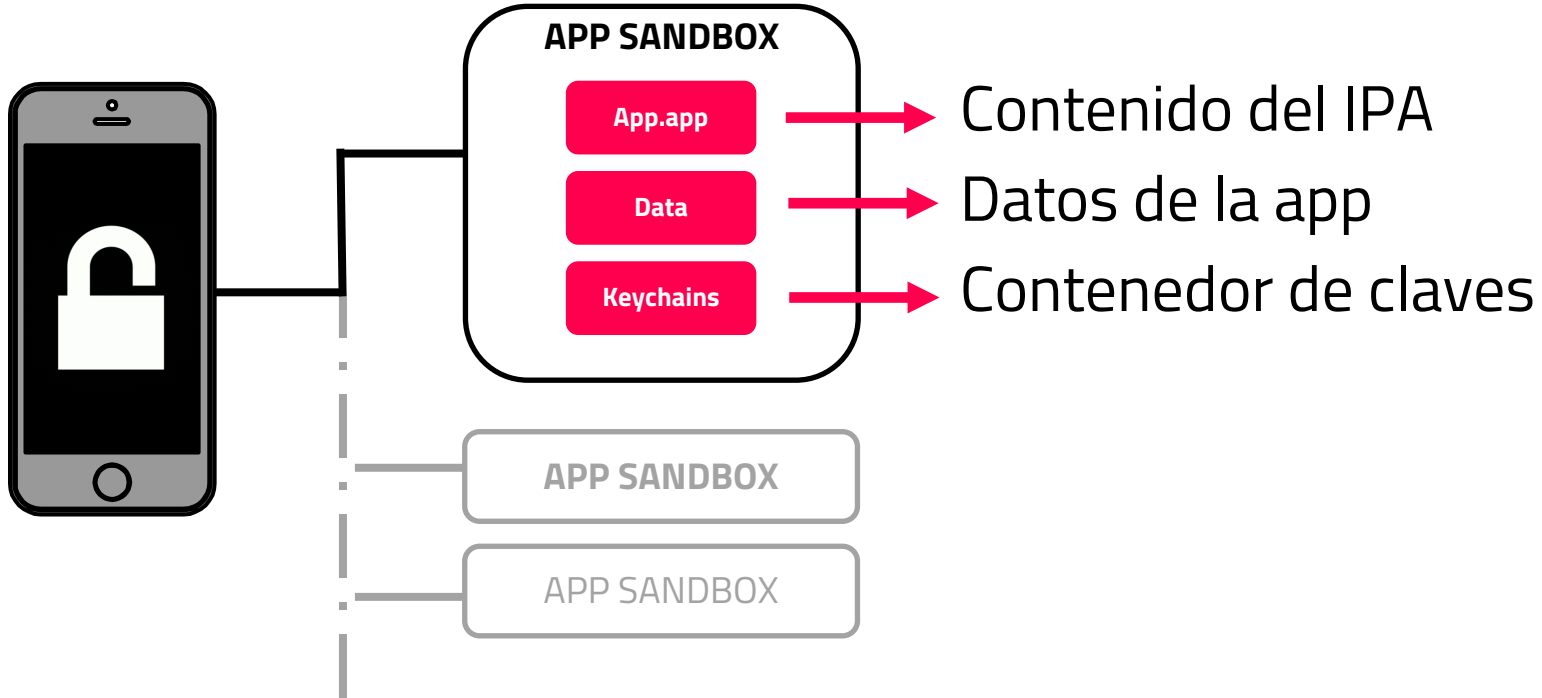
como una matrioska



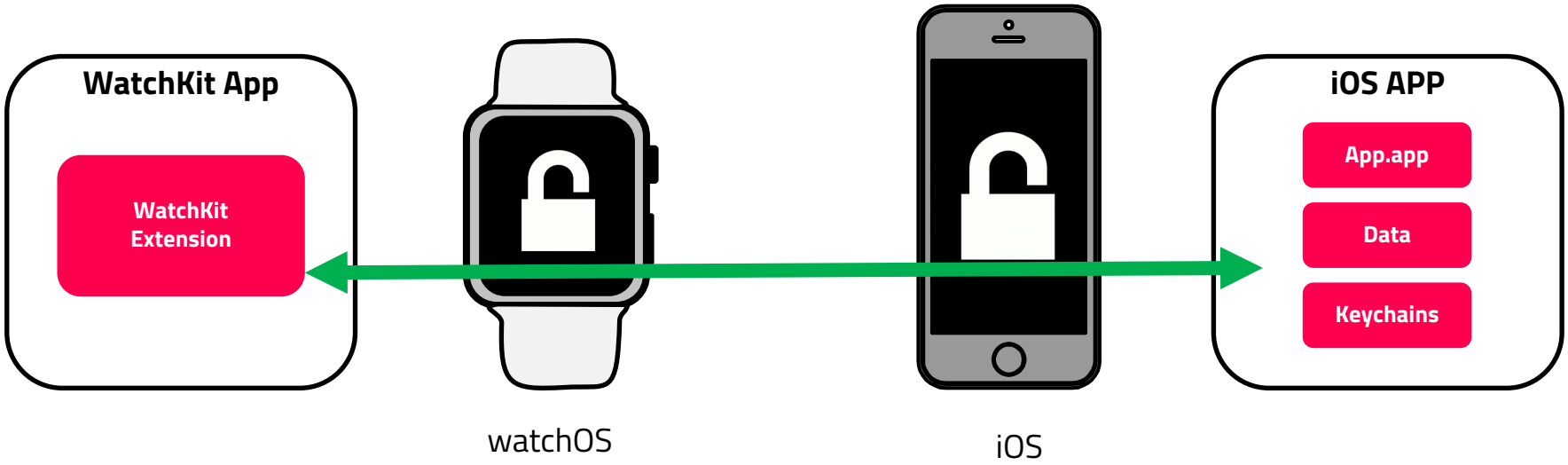
SANDBOX



SANDBOX


APP SANDBOX
App.app
Data
Keychains
Contenido del IPA
Datos de la app
Contenedor de claves
APP SANDBOX
APP SANDBOX

SANDBOX



The Watch App Architecture:
https://developer.apple.com/library/archive/documentation/General/Conceptual/WatchKitProgrammingGuide/DesigningaWatchKitApp.html

WATCH



Watch

- Versión reducida de iOS
- Librerías de sistema especificas
- **Arquitectura arm_v7k y arm64_32_v8**

¿COMO ENTRAR?



Jailbreak



Inyectar y firmar

¿COMO ENTRAR?



Jailbreak

- Mayor versatilidad y facilidad
- Muchas herramientas disponibles
- **No disponible en últimas versiones**

¿COMO ENTRAR?



Watch

- Jailbreak en watchOS 4.1
- Posibilidad de downgrade con iBUS Tool

¿COMO ENTRAR?



Watch



https://www.mfcbox.com

¿COMO ENTRAR?



Watch

- Jailbreak en watchOS 4.1
- Posibilidad de downgrade con iBUS Tool
- No hay Cydia, ni Frida, ni cycript,....

¡NI NÁ!

- Comunidad muy limitada

¿COMO ENTRAR?



Watch

...entonces ¿Jailbreak para qué?

por ahora, solo es útil para

dumpear, desarrollar e investigar

¿COMO ENTRAR?



Jailbreak



Inyectar y firmar

¿COMO ENTRAR?



Inyectar y firmar

Frida

Frida es un set de herramientas de instrumentación dinámica creada por Ole André (@oleavr).

Inyecta en un proceso un motor JavaScript y nos permite interactuar con él, hookear, tracear,... (entre otras cosas)

¿COMO ENTRAR?



Inyectar y firmar

- Nos da acceso directo al proceso
- Se puede hacer lo mismo que con JB
- Válido para cualquier versión de iOS
- Proceso inicial más complejo
- **¡Requiere del IPA sin cifrar!**

¿COMO ENTRAR?



Watch

¿Y con watchOS?

¿COMO ENTRAR?

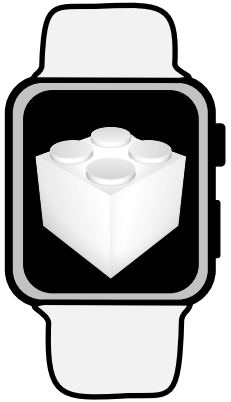


Watch

¿Y con watchOS?

También se puede

¿COMO ENTRAR?



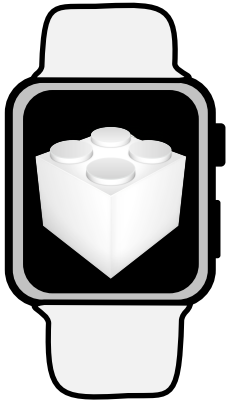
Watch

¿Y con watchOS?

También se puede

pero, **por ahora**, **Frida no**

¿COMO ENTRAR?



Watch

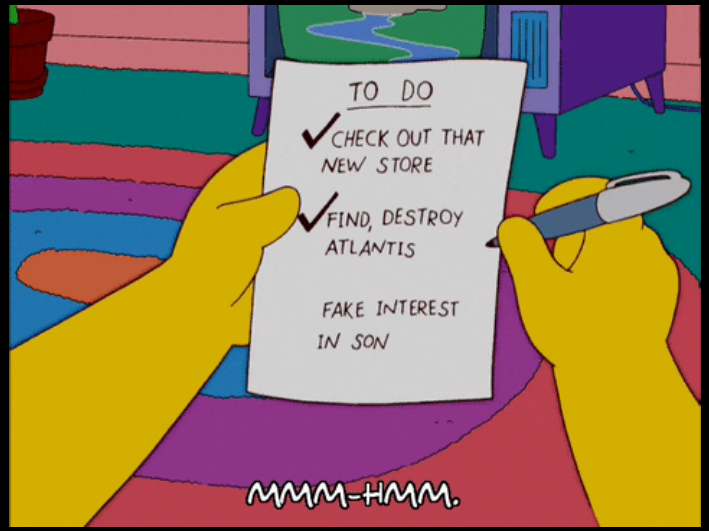
Hooks a la vieja usanza

inyectando nuestra

custom dylib

2.
Cooking recipe

ya era hora....



LISTA DE LA COMPRA



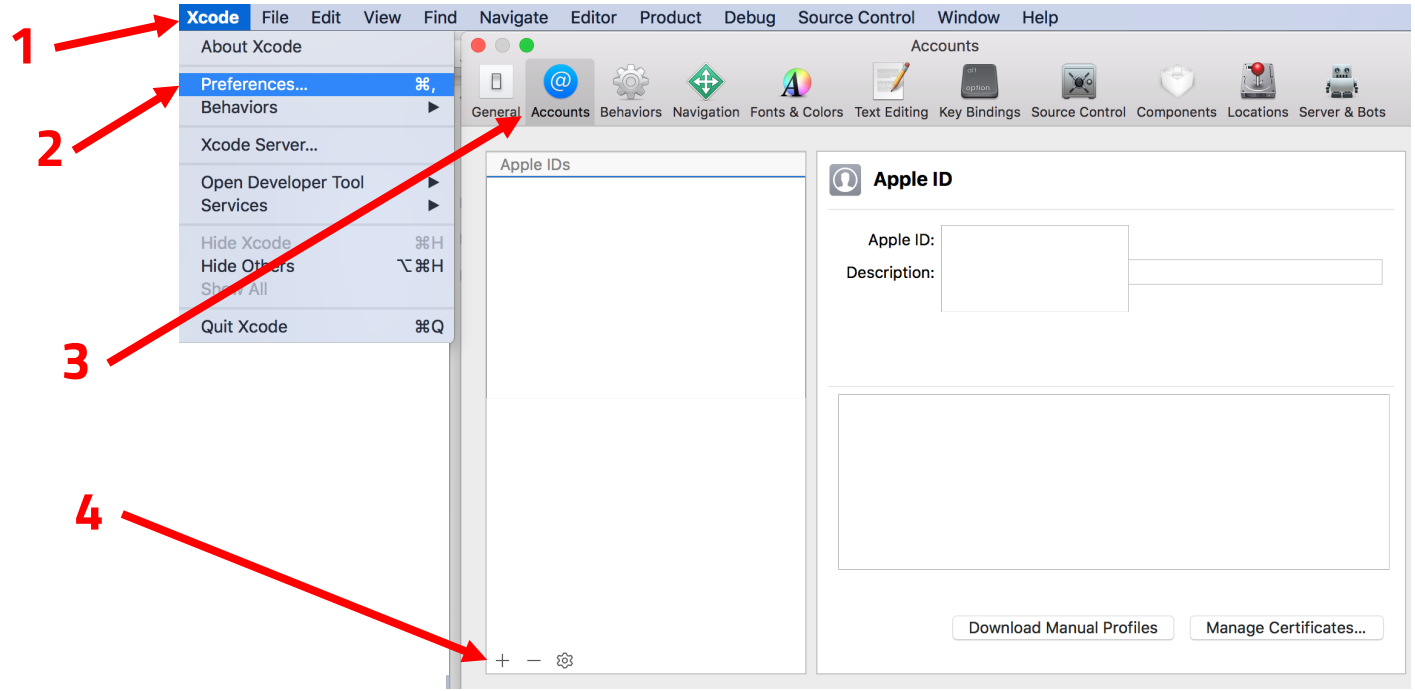
- IPA sin cifrar
- MacOS (VM, Hackintosh) con Xcode
- Cuenta de Apple (gratuita)
- Dylib (Frida y custom para watchOS)
- Insert_dylib
- Fastlane sign (próximamente applesign)
- ios-deploy
- Paciencia

PASOS A SEGUIR

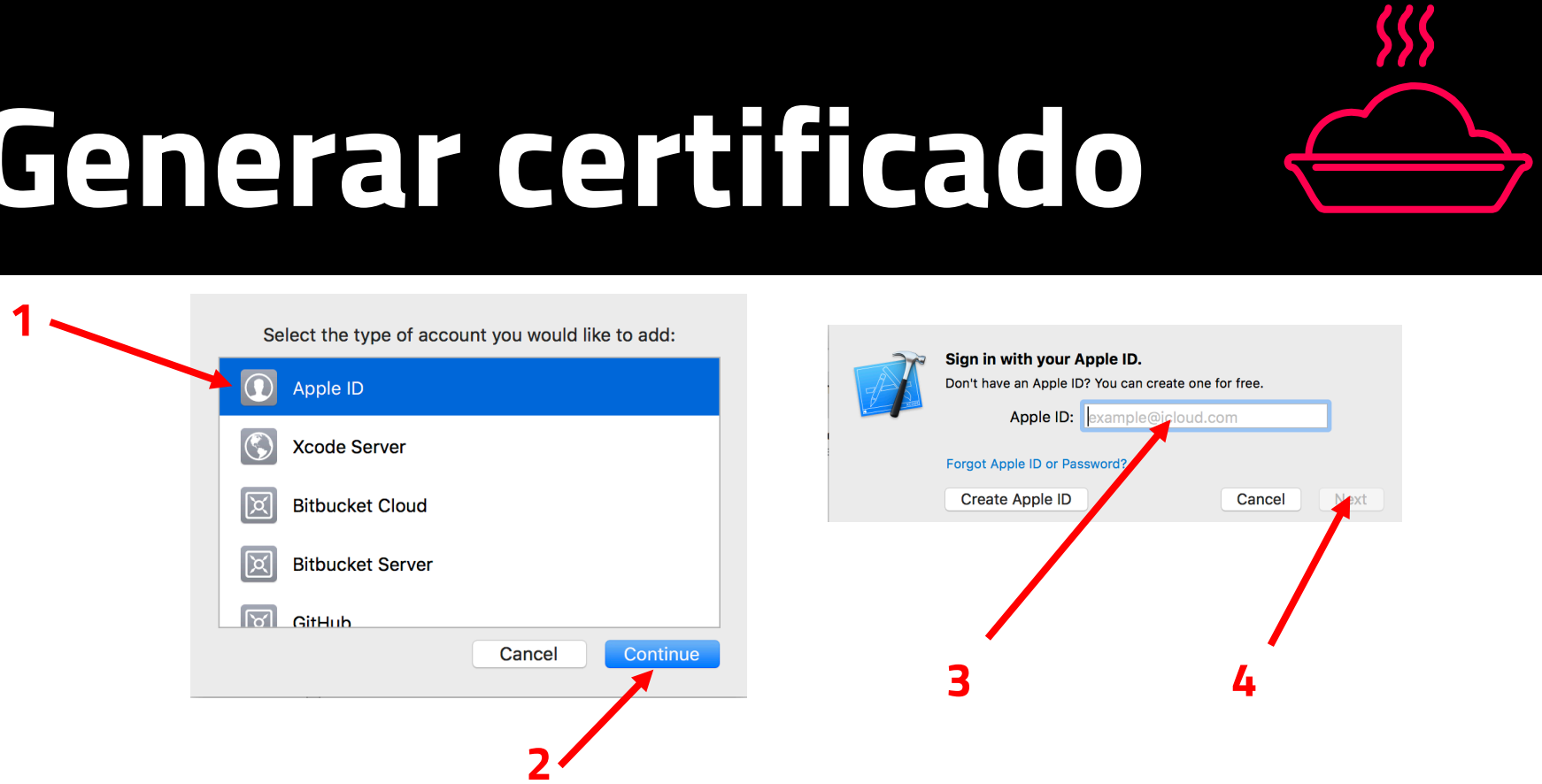


1. Generar certificado de desarrollador
2. Generar proyecto de prueba
3. Obtener los embedded.mobileprovision
4. Obtener y Firmar las librerías
5. Inyectar, Firmar e Instalar

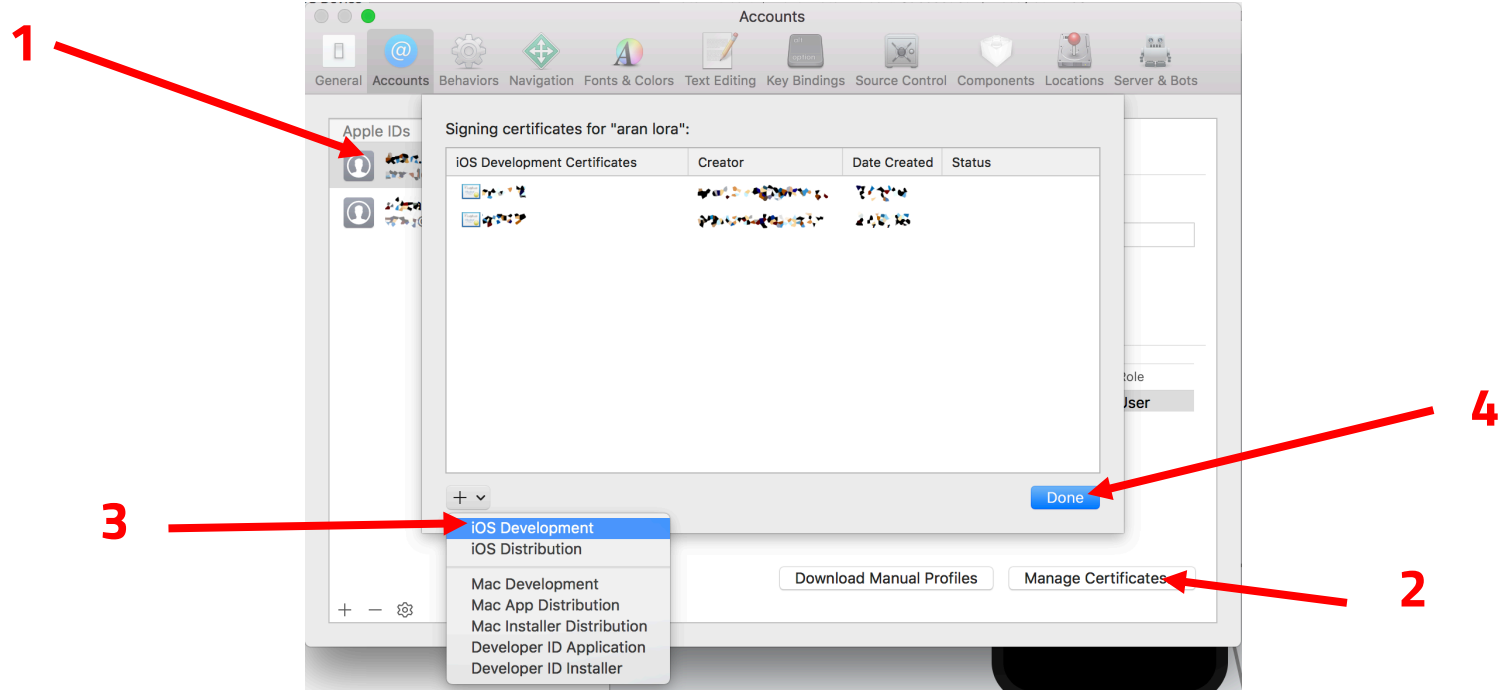
Generar certificado



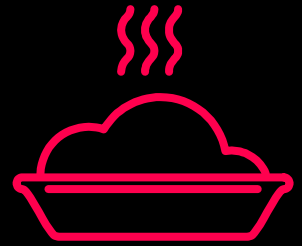
Generar certificado

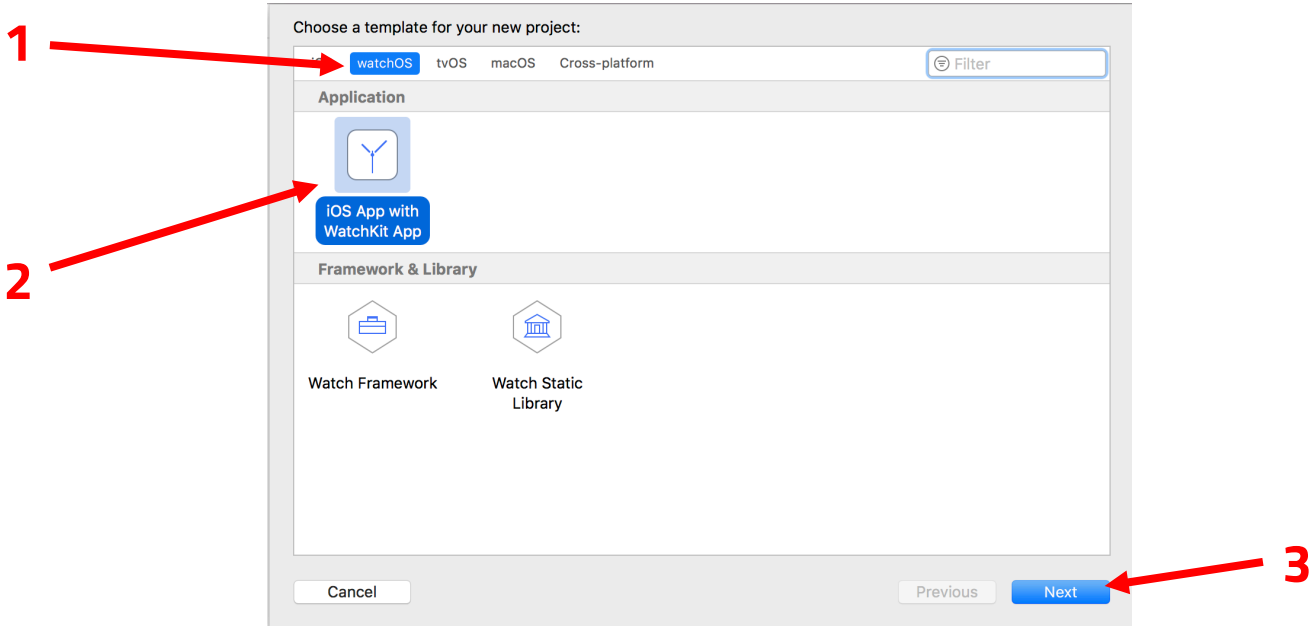


Generar certificado

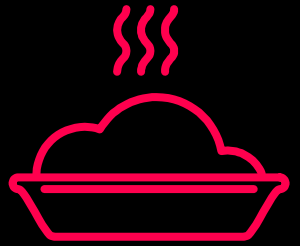


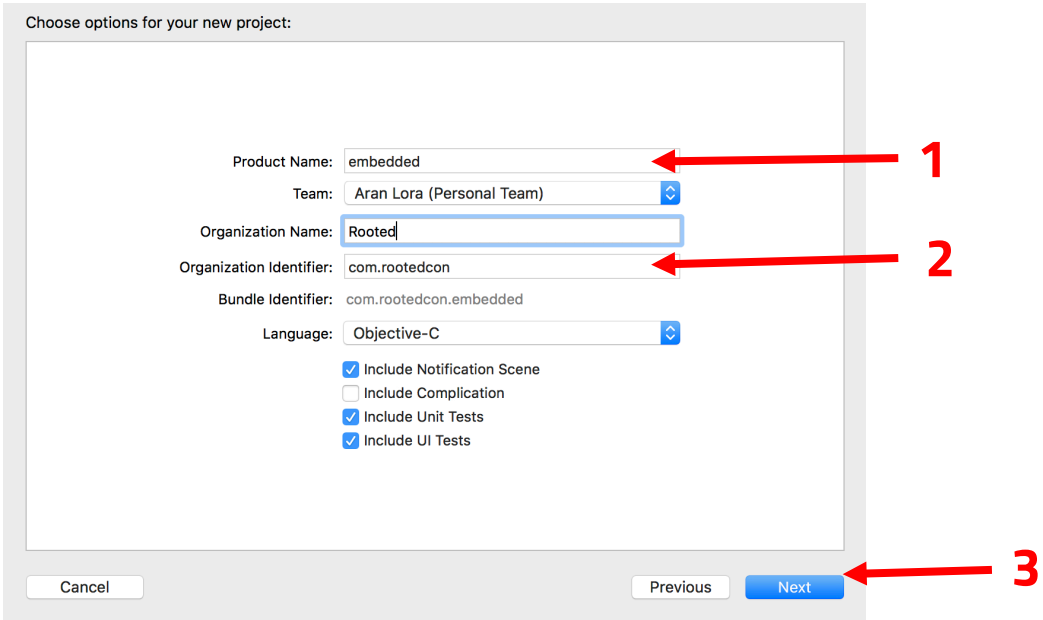
Generar proyecto



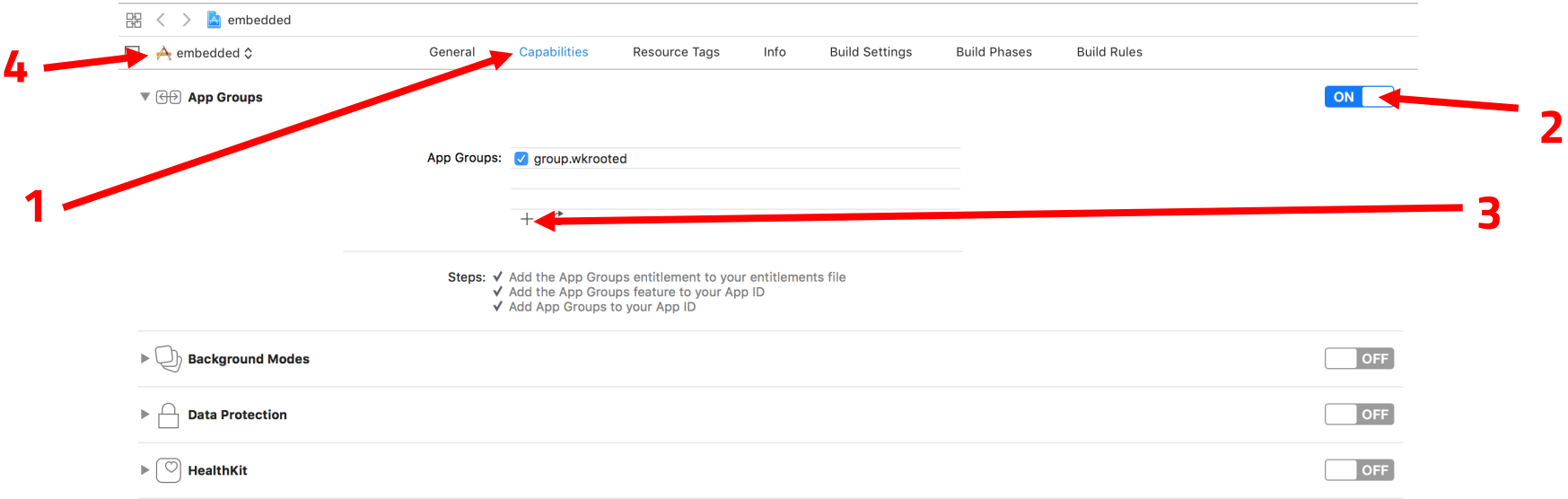


Generar proyecto

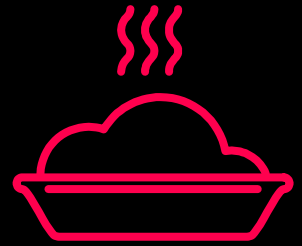


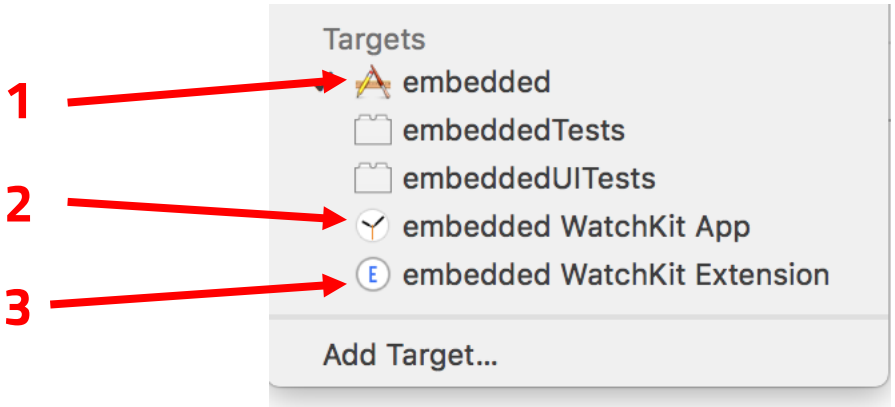


Generar proyecto

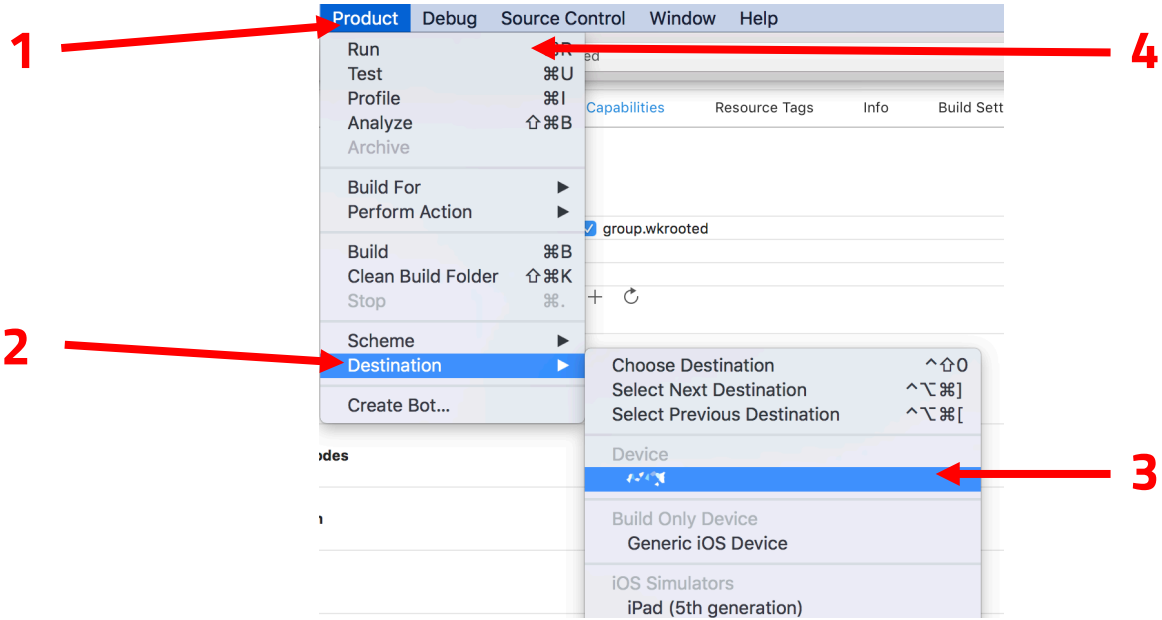


Generar proyecto

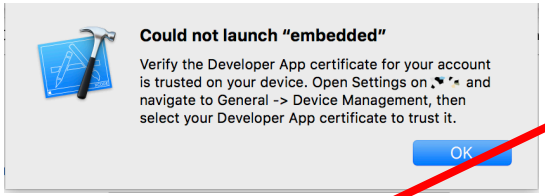




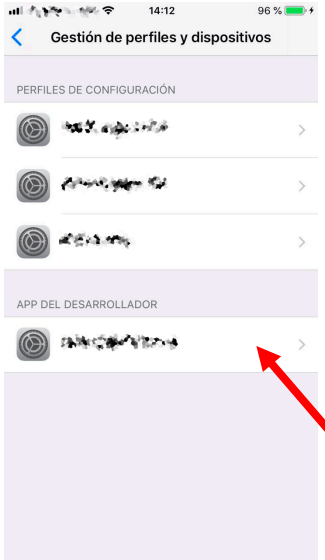
Generar proyecto

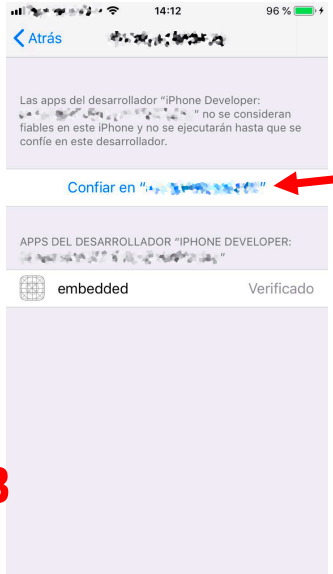


Generar proyecto









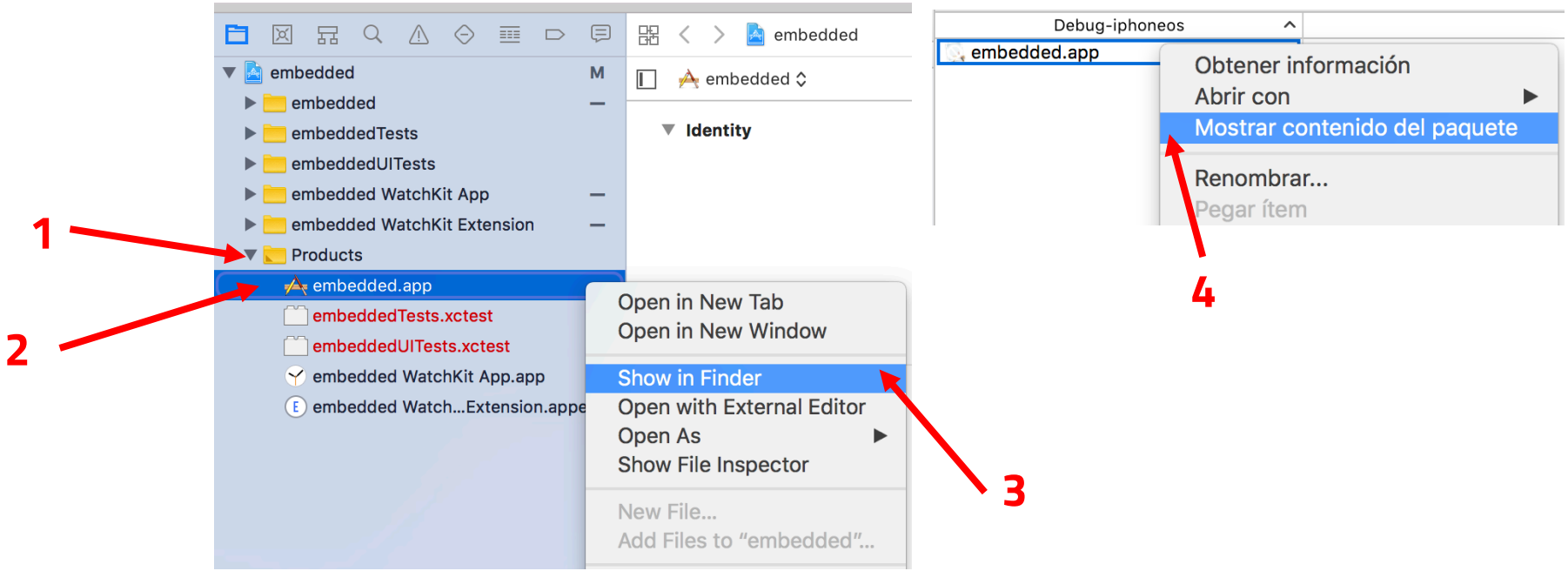
1

2

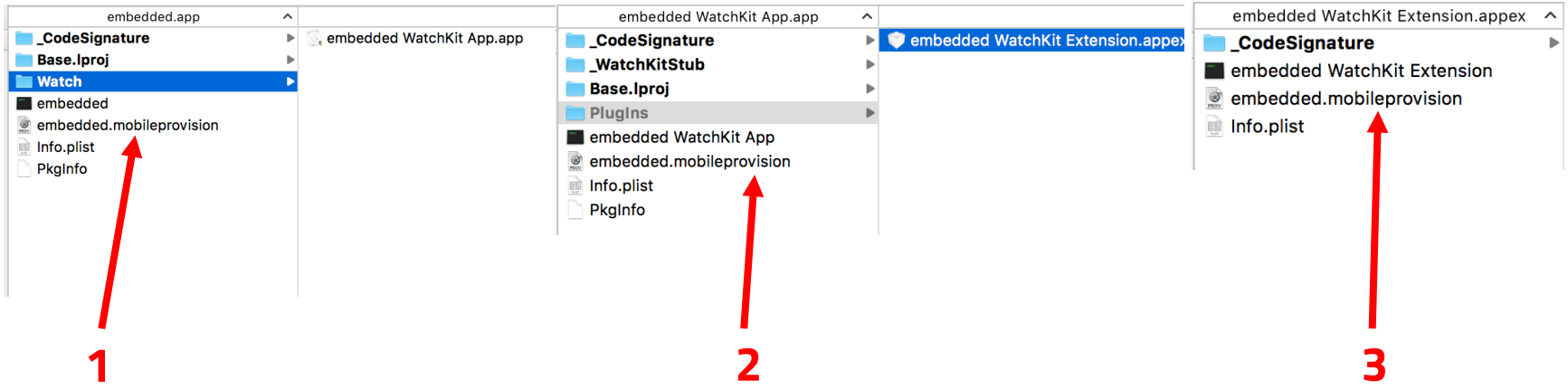
3

4

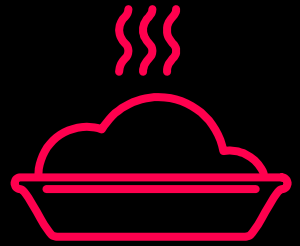
Obtener emp


embedded
embedded
embeddedTests
embeddedUITests
embedded WatchKit App
embedded WatchKit Extension
Products
embedded.app
embeddedTests.xctest
embeddedUITests.xctest
embedded WatchKit App.app
embedded Watch...Extension.appe
Identity
Open in New Tab
Open in New Window
Show in Finder
Open with External Editor
Open As
Show File Inspector
New File...
Add Files to "embedded"...
1
2
3
Debug-iphoneos
embedded.app
Obtener información
Abrir con
Mostrar contenido del paquete
Renombrar...
Pegar ítem
4

Obtener emp


embedded.app
_CodeSignature
Base.lproj
Watch
embedded
embedded.mobileprovision
Info.plist
PkgInfo
embedded WatchKit App.app
1
embedded WatchKit App.app
_CodeSignature
_WatchKitStub
Base.lproj
PlugIns
embedded WatchKit App
embedded.mobileprovision
Info.plist
PkgInfo
embedded WatchKit Extension.appex
2
embedded WatchKit Extension.appex
_CodeSignature
embedded WatchKit Extension
embedded.mobileprovision
Info.plist
3

Dylibs



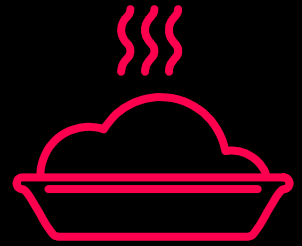


iOS



watchOS

Dylibs





iOS

Frida

https://frida.re/

Flex

https://github.com/Flipboard/FLEX

HttpServerDebug

https://github.com/rob2468/HttpServerDebug

passionfruit

https://github.com/chaitin/passionfruit

objection

https://github.com/sensepost/objection

brida

https://github.com/federicodotta/Brida

* https://github.com/dweinstein/awesome-frida



Passionfruit / iPhone / DVIA com.highaltitudehacks.dvia

Manage Hooks

General | Files | Modules | Classes | Output 4 | UIDump | Storage

Binary

Encrypted NO | PIE ENABLED | ARC ENABLED | Canary ENABLED

Identifier

com.highaltitudehacks.dvia

Bundle

/var/containers/Bundle/Application/AA692D04-1B08-4A11-B69F-91AFD0050AC6/DamnVulnerableIOSApp.app

Executable

/var/containers/Bundle/Application/AA692D04-1B08-4A11-B69F-91AFD0050AC6/DamnVulnerableIOSApp.app/DamnVulnerableIOSApp

Data Directory

/private/var/mobile/Containers/Data/Application/18FBF57B-AE44-4F86-AFD3-EE1F97D77153

Version

2.0

URL Scheme

(empty name)

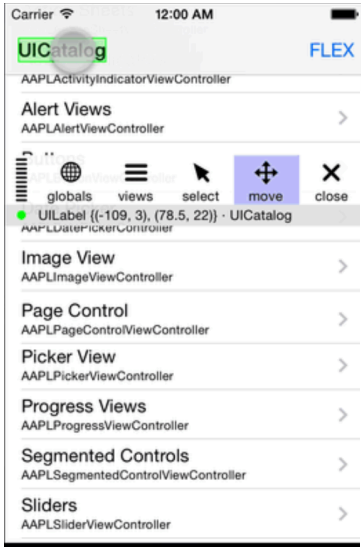
- dvia://

Metainfo

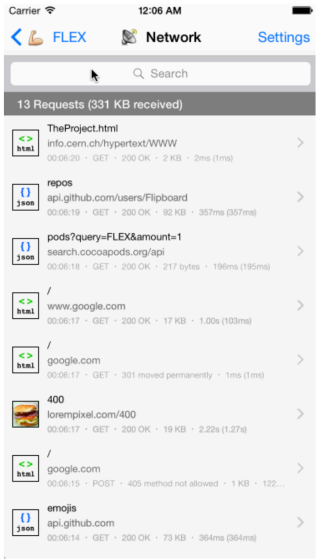
+ Expand All | − Collapse All | Search keys and values...

- **Info.plist:**
  - CFBundleName: `DamnVulnerableIOSApp`
  - DTXcode: `0721`
  - DTSDKName: `iphoneos9.2`
  - CFBundleNumericVersion: `33587200`
  - DTSDKBuild: `13C75`
  - CFBundleDevelopmentRegion: `en`
  - CFBundleVersion: `2.0`
  - BuildMachineOSBuild: `15B42`
  - DTPlatformName: `iphoneos`
  - CFBundlePackageType: `APPL`
  - UIMainStoryboardFile: `Main`
  - **CFBundleSupportedPlatforms:**
  - CFBundleShortVersionString: `2.0`
  - **NSAppTransportSecurity:**
  - CFBundleInfoDictionaryVersion: `6.0`
  - **UIRequiredDeviceCapabilities:**
  - CFBundleExecutable: `DamnVulnerableIOSApp`
  - **UILaunchImages:**
  - **CFBundleURLTypes:**
  - CFBundleIdentifier: `com.highaltitudehacks.dvia`
  - DTCompiler: `com.apple.compilers.llvm.clang.1_0`
  - CFBundleSignature: `????`

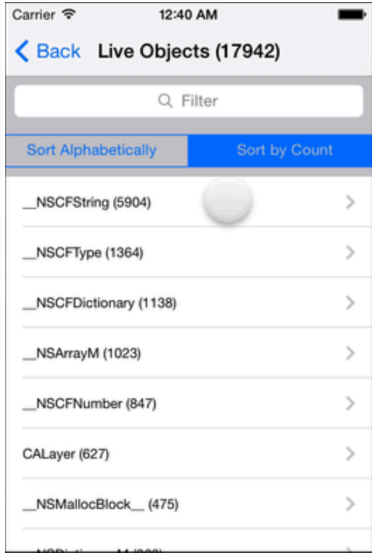
FLEX



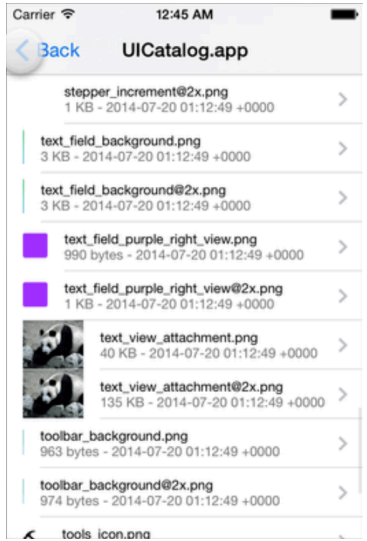
Editor de UI



Visor Red

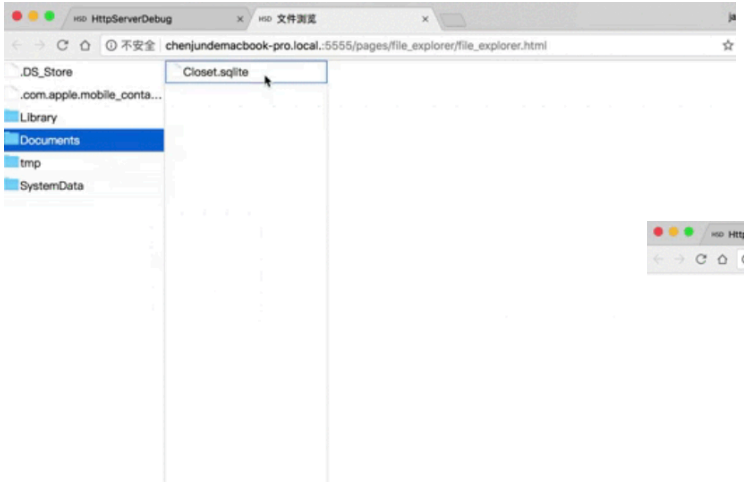


Objetos

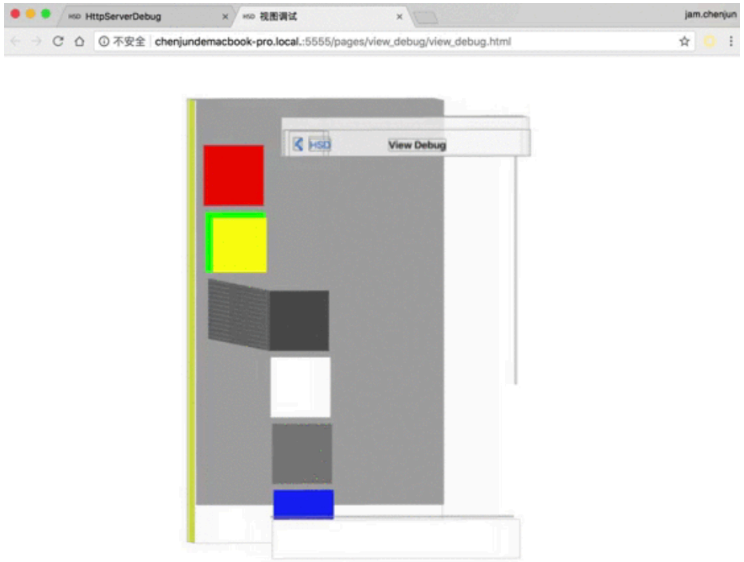


Ficheros

HttpServerDebug



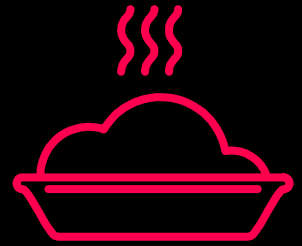
Visor de ficherso y sqlite



Visor de elementos UI

* Sí, interfaz en chino

Dylibs





watchOS

1 - Crear Proyecto **Watch Framework**

2 - **John Coates's autohook**
https://gist.github.com/JohnCoates/c0d77f130d033b206367db480f7c18ae

3 - Sazonar al gusto

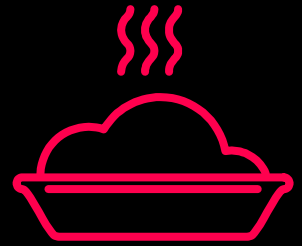
4 - Compilar para watchOS

AUTOHOOK



```
@import UIKit;
@import WatchKit;
#import <Foundation/Foundation.h>
#import "AutoHook.h"

@interface HOOK : NSObject <AutoHook>
@end

@implementation HOOK

+ (NSArray *)targetClasses {
    return @[@"InterfaceController"];
}

- (void)hook_showAlertAction {
    //[self original_showAlertAction];
    WKAlertAction *act = [WKAlertAction actionWithTitle:@"OK" style:WKAlertActionStyleCancel handler:^(void){
        NSLog(@"ALERT YES ");
    }];

    NSArray *testing = @[act];

    [[[WKExtension sharedExtension] rootInterfaceController] presentAlertControllerWithTitle:@"Voila"
        message:@"This is Rooted !" preferredStyle:WKAlertControllerStyleAlert actions:testing];
}

- (void)original_showAlertAction { }

@end
```

Dylibs





watchOS

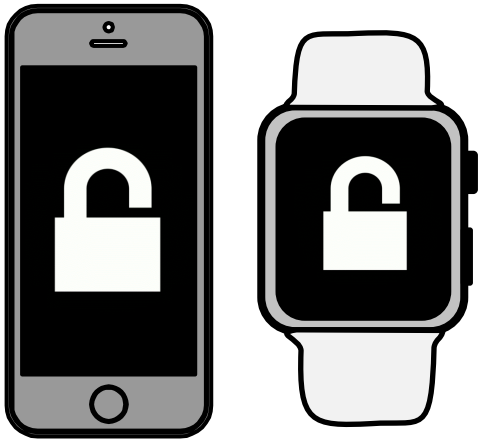
1 - Crear Proyecto **Watch Framework**

2 - **John Coates's autohook**
https://gist.github.com/JohnCoates/c0d77f130d033b206367db480f7c18ae

3 - **Sazonar** al gusto

4 - **Compilar** para watchOS

Dylibs



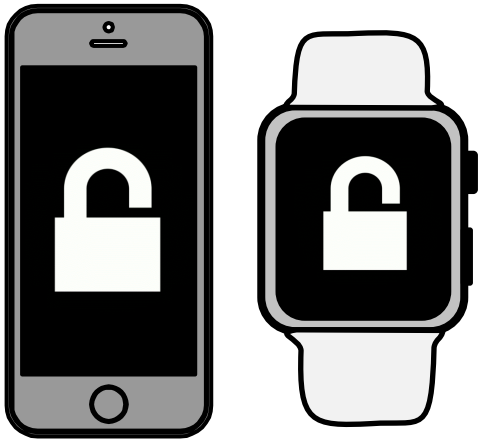
iOS watchOS

Si la dylib no está firmada:

1. Ver ID identidad firmante:
   security find-identity -v -p codesigning

2. Firmar:
   codesign -f -s *Hash_identidad* FridaGadget.dylib

Inyectar



iOS watchOS

1. Descomprimir IPA
2. Crear (si no existe) la carpeta Frameworks
3. Copiar dylib o framework
4. Inyectar en el binario con insert_dylib
5. Firmar con Fastlane sign
6. Instalar con ios-deploy

Inyectar



iOS

Peculiaridades de Frida en modo Gadget (dylib):

- **Inyectar:**

insert_dylib "@executable_path/Frameworks/FridaGadget.dylib" Payload/MyApp.app/MyApp –inplace

- **Ejecutar en modo debug:**

ios-deploy --bundle Payload/*.app -m -L

- **Justo después ejecutar frida:**

frida -U “gadget”

DEMO

¡¡Gracias!!

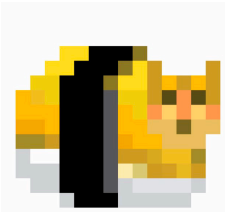




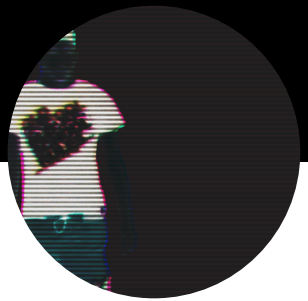
amnesia

Deloitte.

/RootedCON



@ajpallares



¿Preguntas?

@4r4nL