System Extension

A system extension implements features that require kernel-level cooperation, such as custom security and network behaviors.

Three Types os System Extensions you can build;

1. Network Extensions
2. Driver Extensions
3. Endpoint Security Extensions

1. Network Extensions;

- Replacement of network kernel extensions
- **Capabilities include;**
  - Content filter
  - DNS Proxy
  - VPN client

2. Driver Extensions;

- A replacement for device-driver kexts that used **IOKit**
- **Control**
  - **USB**
  - **Serial**
  - **NIC (Network Interface Controller)**
  - **HID (Human Interface Device)**
- Must be written in C or C++ (default is C++ 17)

DriverKit;

- **Driver extensions** are built with **DriverKit**
- New SDK w/All New frameworks as of Catalina
- Based on **IOKit** but “Updated” and “Modernized”
- Designed for building Driver extensions in user space
- **DriverKit interfaces use a new file type with a .iig extension**

NetworkingDriverKit;

- **Used for creating network interfaces**

HIDDriverKit;

- Used for creating HID devices

USBSerialDriverKit

- **Used to make USB serial device available to the OS**

USB DriverKit

- Used to make use of USB device providers in your driverd.

Driver Extension Security

Entitlements needed

- **com.apple.developer.driverkit:** For all driver extensions entitlement.
- **com.apple.developer.driverkit.transport.usb**
  - Attach to a device (transport eneitlement, specific to device type)
  - **Transport entitlement;** to take control of a device
- **com.apple.developer.driverkit.family.hid.device**
  - Provide service to the OS (Family entitlement)

Driver Extension Compatibility

MacOS 10.15 will be the last release to fully support **kexts** without compromises

“Install a Kernel Extension only on Mojave or earlier”

DriverKit extension was introduced as of **MacOS Catalina**

3. Endpoint Security Extensions;

- Replacement for kexts (**kauth**) to monitor security-related events.
- Apps you can build
  - **Endpoint Detection & Response**
  - **Anti-virus**
  - Data Loss Prevention

After completion you can distribute your System Extension directly to your users using your Developer ID *(Or on the Mac App Store)*

Can’t do App store w/kernel extensions. ^

- Give **descriptive name** with **CFBundleDisplayName** key
  - In the extensions **info.plist** to give it a good localized name
    - Also include **usage description string** (what it does, why a user would run it)
- Give **custom icon** that **relates to your app’s icon**

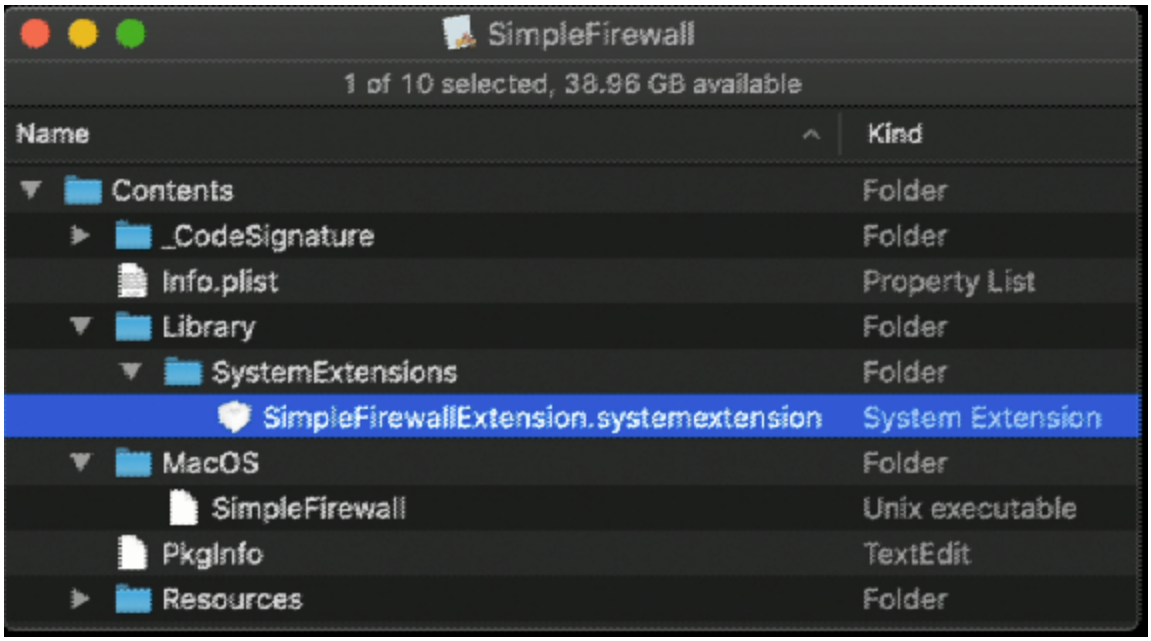
For **driver extensions** use the key **OSBundleUsageDescription** in the extensions **info.plist**

For **other types of system extensions** use the key **NSSystemExtensionUsage** description in the extensions **info.plist** file.

The **System Extension** itself is a **seperate sub-bundle** of your app

- Has it’s **own** executable **info.plist**
  - Embedded within the application.

Example of system extension embedded in an application;



Driver Extensions

- Use .**dext** suffix
- CFBundlePackageType = DEXT
- Uses **OSBundle*** Info.plist keys
  - **Similar** to **kernel extension bundles**
- Flat structure: no Contents folder

System extension bundles of **other extension types**;

- Use the .**systemextension** bundle
- Use CFBundlePackageType **SYSX** (System extension)

In XCode your **system extension** is a **seperate target**

XCode has built in **templates** for;

- **Network Extensions**
- **DriverKit Driver**

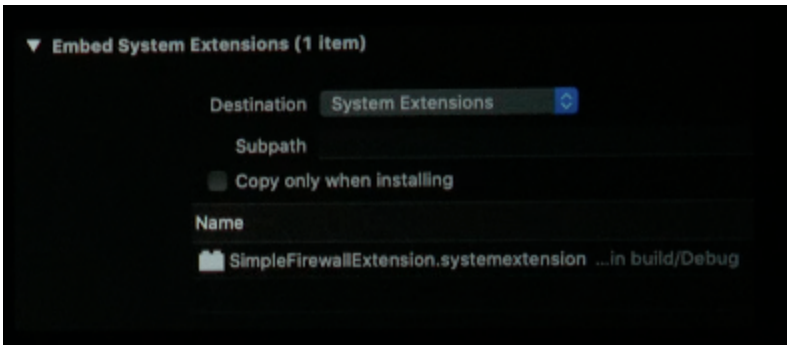
When you **create** such a **target** XCode will ask if you wanna **embed it in an application that's already part of your project**.

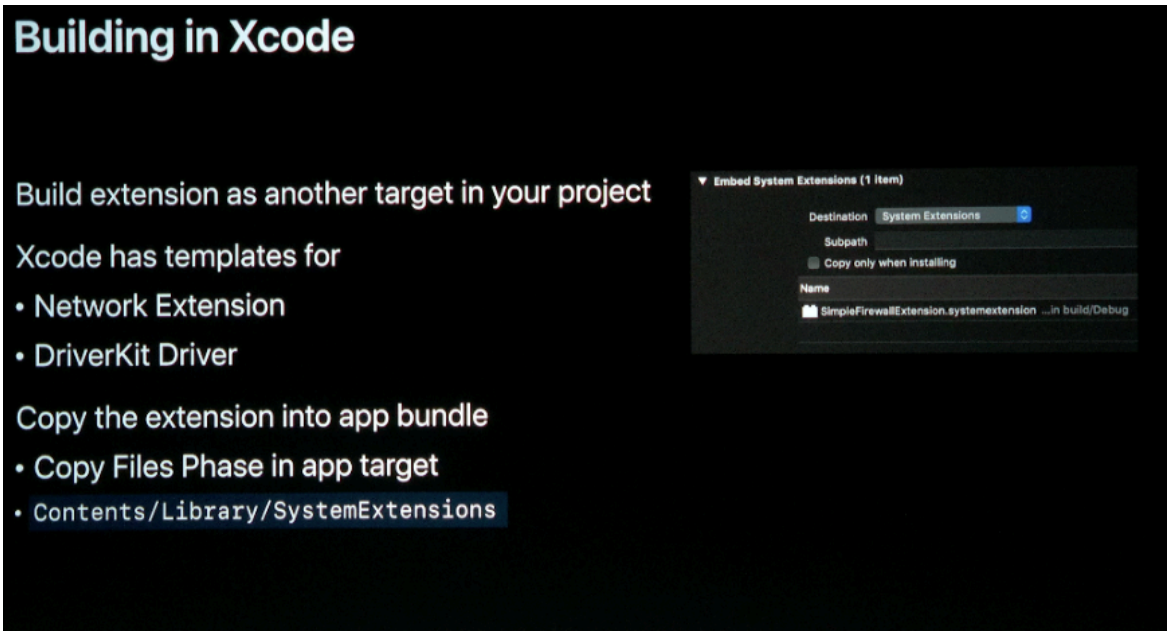
Answer yes to **commence** the **following sequence**;

Copy Files Phase;

Copy extension into app bundle

- Copy Files Phase in app target
- Contents/Library/SystemExtensions

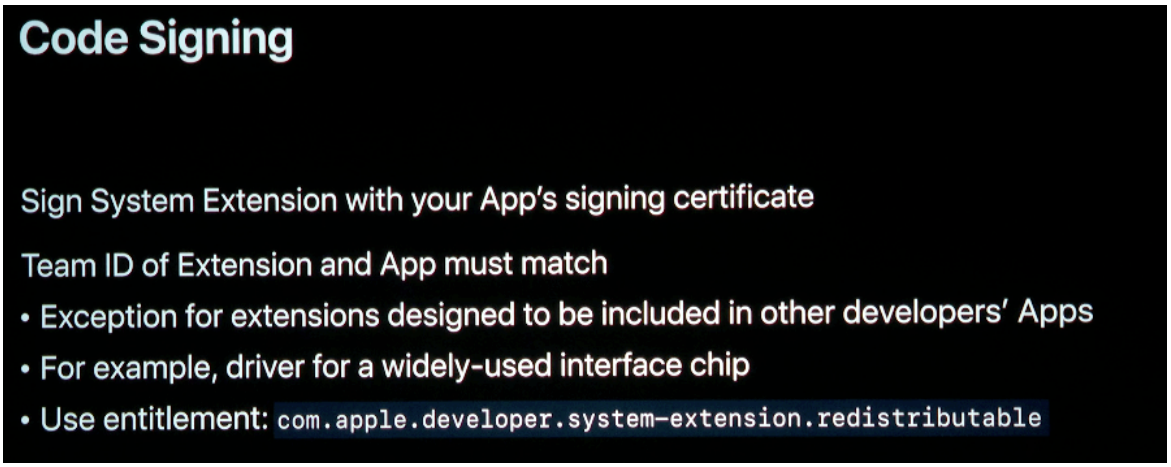




Code Signing;

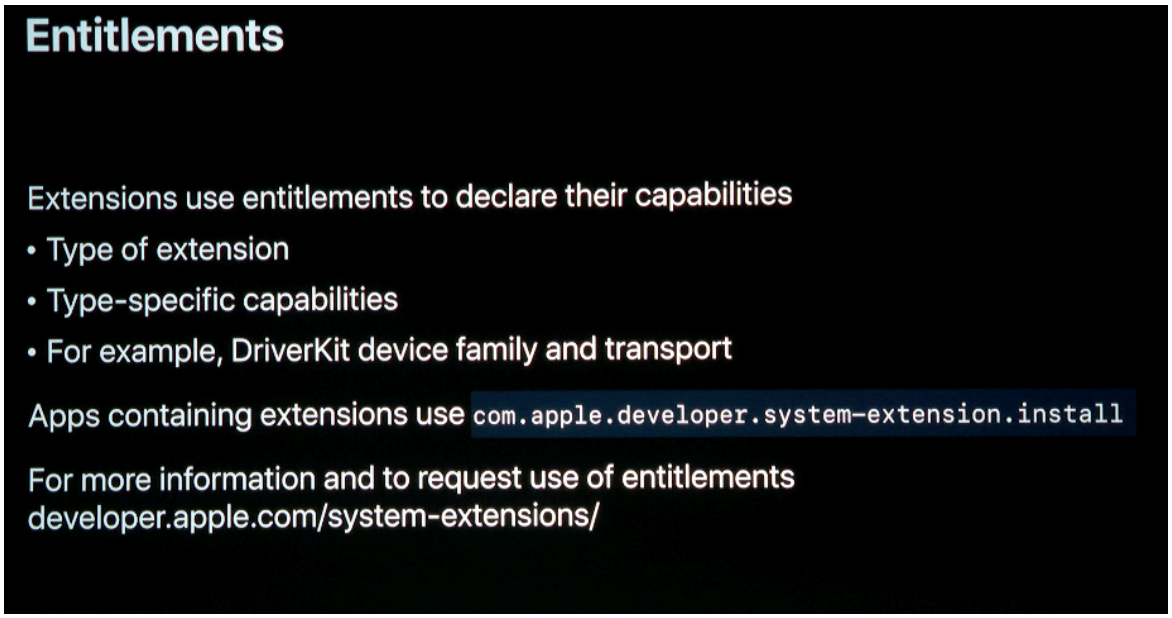
After building your **system extension;**

- **Sign it** w/**same cert** you that you **sign your app** with
  - **(Kext signing** required a special **Kext specific signing certificate)**
  - **Team ID** of **Extension** and **App must match**
    - **Exception: extensions** designed **to be used** in **other developers Apps**
      - **Example: Driver** for **widely-used** interface chip
      - Use entitlement: **com.apple.developer.system-extension.redistributable**



System extensions signed w/Developer ID **must be Notarized**

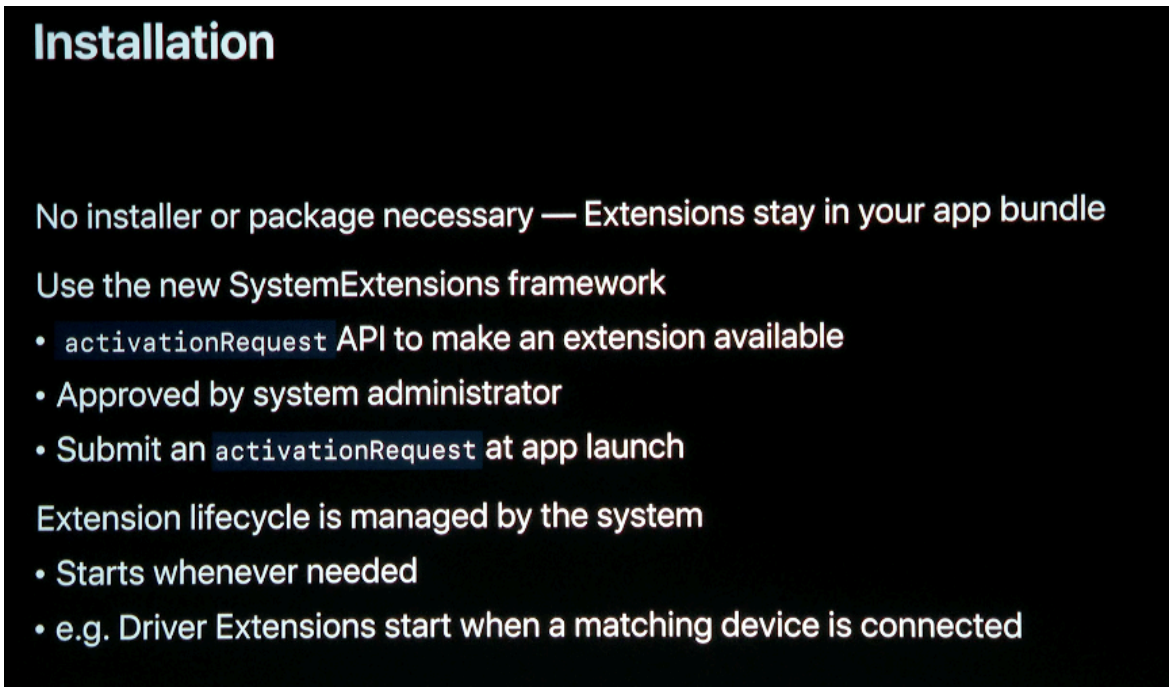
Entitlements Slide;



You can **turn sip off** to **disable** some **checks** for **code signing** and **entitlements**

- *(While testing)*

System Extension Installation on a Users System;



Updating your System Extension;

- To **update** your system extension **update** your **app bundle**
  - User may install new version they download
  - Your **auto-updater** may **update** the **app bundle** in place
  - **Release new version** on **app store**
    - Updated for user

Moving app to Trash deactivates all its extensions

- There is also a **deactivationRequest** API

WWDC19

https://developer.apple.com/videos/play/wwdc2019/702/