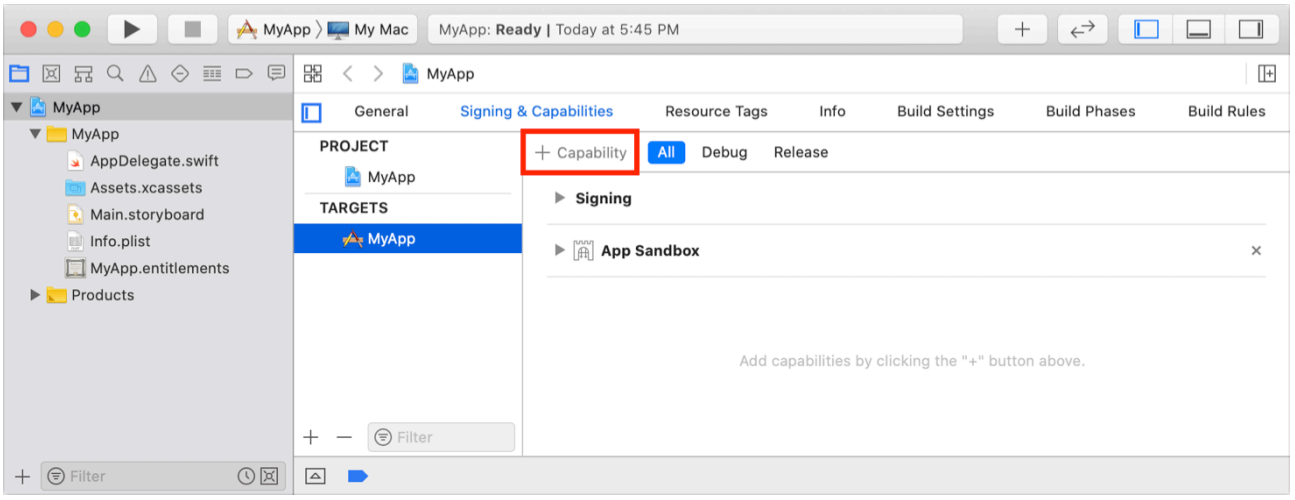
Hardened Runtime | Apple Developer Documentation

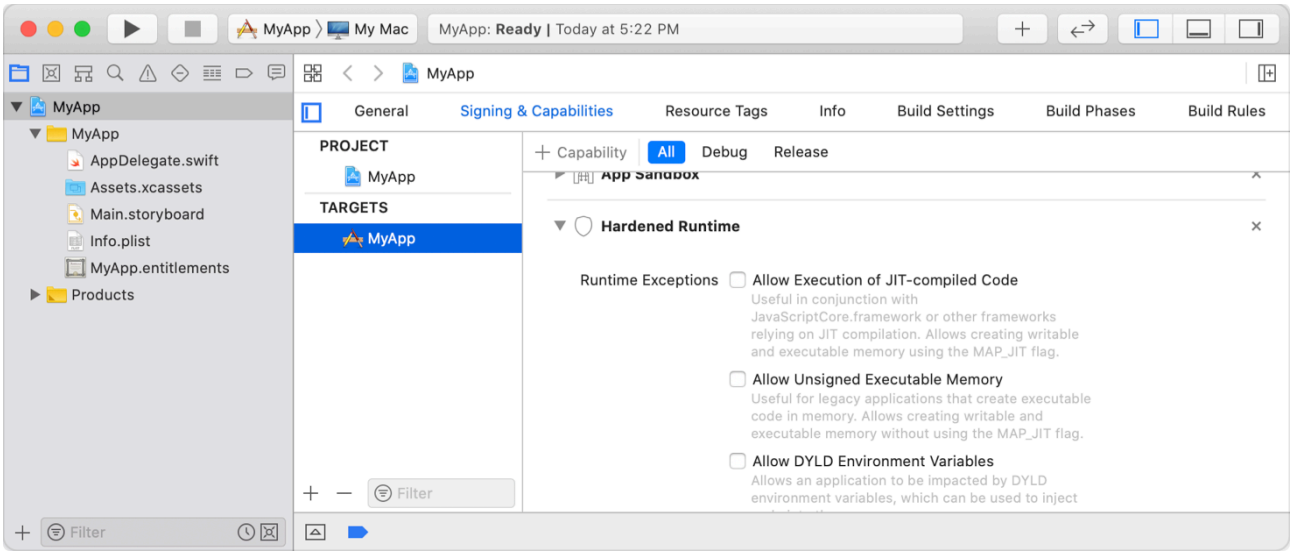
Archived: 2026-04-05 21:18:26 UTC

Overview

The Hardened Runtime, along with System Integrity Protection (SIP), protects the runtime integrity of your software by preventing certain classes of exploits, like code injection, dynamically linked library (DLL) hijacking, and process memory space tampering. To enable the Hardened Runtime for your app, navigate in Xcode to your target's Signing & Capabilities information and click the + button. In the window that appears, choose Hardened Runtime.



The Hardened Runtime doesn't affect the operation of most apps, but it does disallow certain less common capabilities, like just-in-time (JIT) compilation. If your app relies on a capability that the Hardened Runtime restricts, add an entitlement to disable an individual protection. You add an entitlement by enabling one of the runtime exceptions or access permissions listed in Xcode. Make sure to use only the entitlements that are absolutely necessary for your app's functionality.



You add entitlements only to executables. Shared libraries, frameworks, and in-process plug-ins inherit the entitlements of their host executable.

Due to their privileged position in the system, macOS refuses to load system extensions that use Hardened Runtime exception entitlements. There's one exception to this general rule: macOS allows the `Allow execution of JIT-compiled code entitlement` in non-DEXT system extensions.

The default value of these Boolean entitlements is false. When Xcode signs your code, it includes an entitlement only if the value is true. If you're manually signing code, follow this convention to ensure maximum compatibility. Don't include an entitlement if the value is false.

Source: https://developer.apple.com/documentation/security/hardened_runtime