`--deep` Considered Harmful

This thread has been locked by a moderator.

2.9k

Many of the trusted execution problems I see are caused by folks signing their product using the `--deep` option. While that can work in some circumstances, I generally recommend against it. There are two issues with `--deep`:

- It applies the same code signing options to every code item that it signs, something that's not appropriate in general. For example, you might have an app containing a nested command-line tool, where the app and the tool need different entitlements. The `--deep` option will apply the same entitlements to both, which is a serious mistake.
- It only signs code that it can find, and it only finds code in nested code sites. If you put code in a place where the system is expecting to find data, `--deep` won't sign it.

The first issue is fundamental to how `--deep` works, and is the main reason you should not use it. Indeed, on macOS it may cause the trusted execution system to block your program from running. For the details, see the *Check for Entitlements on Library Code* section of Resolving Library Loading Problems.

The second issue is only a problem if you don't follow the rules for nesting code and data within a bundle, as documented in Placing Content in a Bundle. However, my experience is that the products that don't follow those rules are exactly the same sort of products that try to use `--deep`.

The alternative to `--deep` is to sign each code item separately, from the inside out. If your product has lots of nested code, automate this using a script.

Note One exception to the prohibition on `--deep` is Automator apps. If you're signing an Automator app, see this DevForums post.

For detailed information on how to correctly sign and package Mac software, see Creating Distribution-Signed Code for Mac and Packaging Mac Software for Distribution.

Share and Enjoy

—
Quinn "The Eskimo!" @ Developer Technical Support @ Apple
`let myEmail = "eskimo" + "1" + "@" + "apple.com"`

Revision history:

- **2022-06-14** Added a link to Resolving Library Loading Problems. Replaced the link to *Signing a Mac Product For Distribution* with a link to Creating Distribution-Signed Code for Mac. Made other minor editorial changes.
- **2021-10-21** Replaced the nested code reference with one to Placing Content in a Bundle. Minor editorial changes.
- **2020-09-15** Adopted the correct terminology for Automator apps.
- **2020-03-09** First version.

Code Signing

Reply

Posted 2 years ago by eskimo

Add a Comment

This site contains user submitted content, comments and opinions and is for informational purposes only. Apple disclaims any and all liability for the acts, omissions and conduct of any third parties in connection with or related to your use of the site. All postings and use of the content on this site are subject to the Apple Developer Forums Participation Agreement.

Developer › Forums

Platforms
iOS
iPadOS
macOS
tvOS
watchOS

Tools
Swift
SwiftUI
SF Symbols
Swift Playgrounds
TestFlight
Xcode
Xcode Cloud

Topics & Technologies
Accessibility
Accessories
App Extensions
App Store
Audio & Video
Augmented Reality
Business
Design
Distribution
Education
Fonts
Games
Health & Fitness
In-App Purchase
Localization
Maps & Location
Machine Learning
Security
Safari & Web

Resources
Documentation
Curriculum
Downloads
Forums
Videos

Support
Support Articles
Contact Us
Bug Reporting
System Status

Account
Apple Developer
App Store Connect
Certificates, IDs, & Profiles
Feedback Assistant

Programs
Apple Developer Program
Apple Developer Enterprise Program
App Store Small Business Program
MFi Program
News Partner Program
Video Partner Program
Security Bounty Program
Security Research Device Program

Events
App Accelerators
App Store Awards
Apple Design Awards
Apple Developer Academies
Entrepreneur Camp
Tech Talks
WWDC

To view the latest developer news, visit News and Updates.