Playing In The App Sandbox

Boisy G. Pitre

About Me

- Senior Software Engineer in the Mac Products Group at Nuance Communications
- Owner of Tee-Boy, developer of *WeatherSnoop*
- Author of *Developer to Developer* monthly column in MacTech Magazine


Presentation Goals

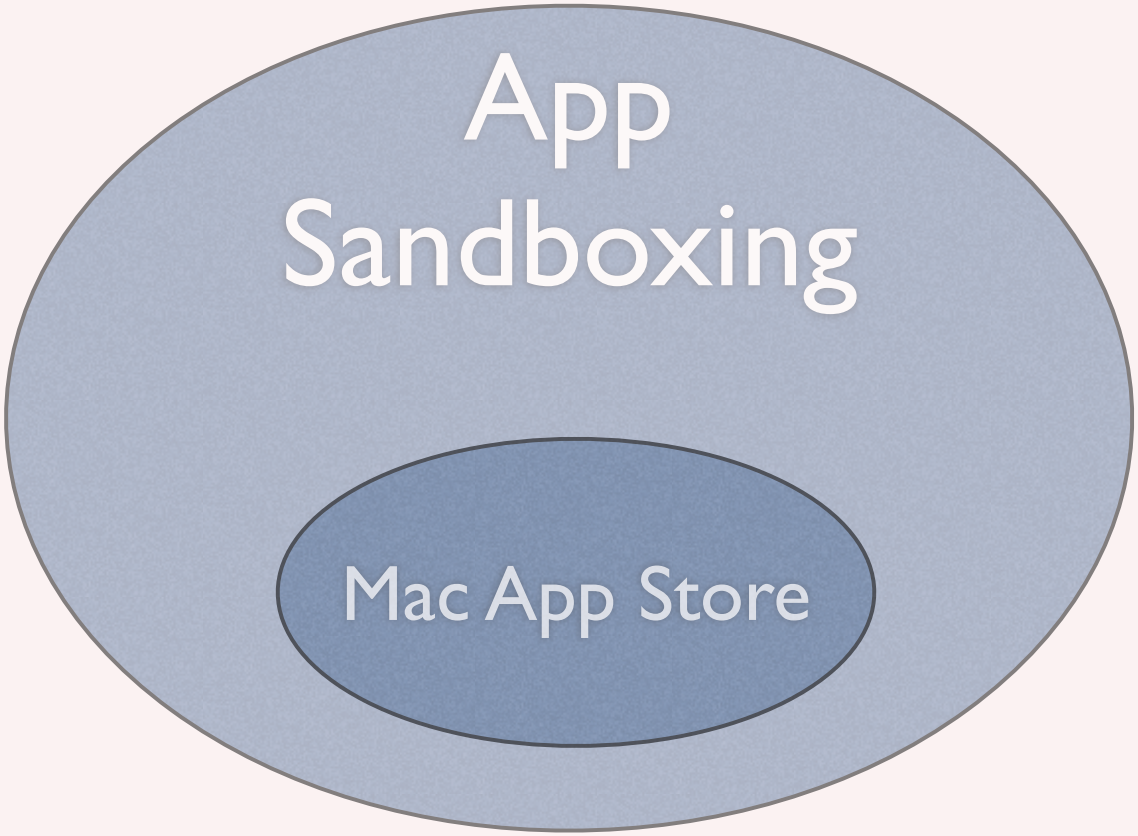
- Examine App Sandboxing's "Raison d'Etre"
- Peek at the Technology Behind Sandboxing
- Explore Entitlements and XPC
- Demo
- Caveats

What Is App Sandboxing?

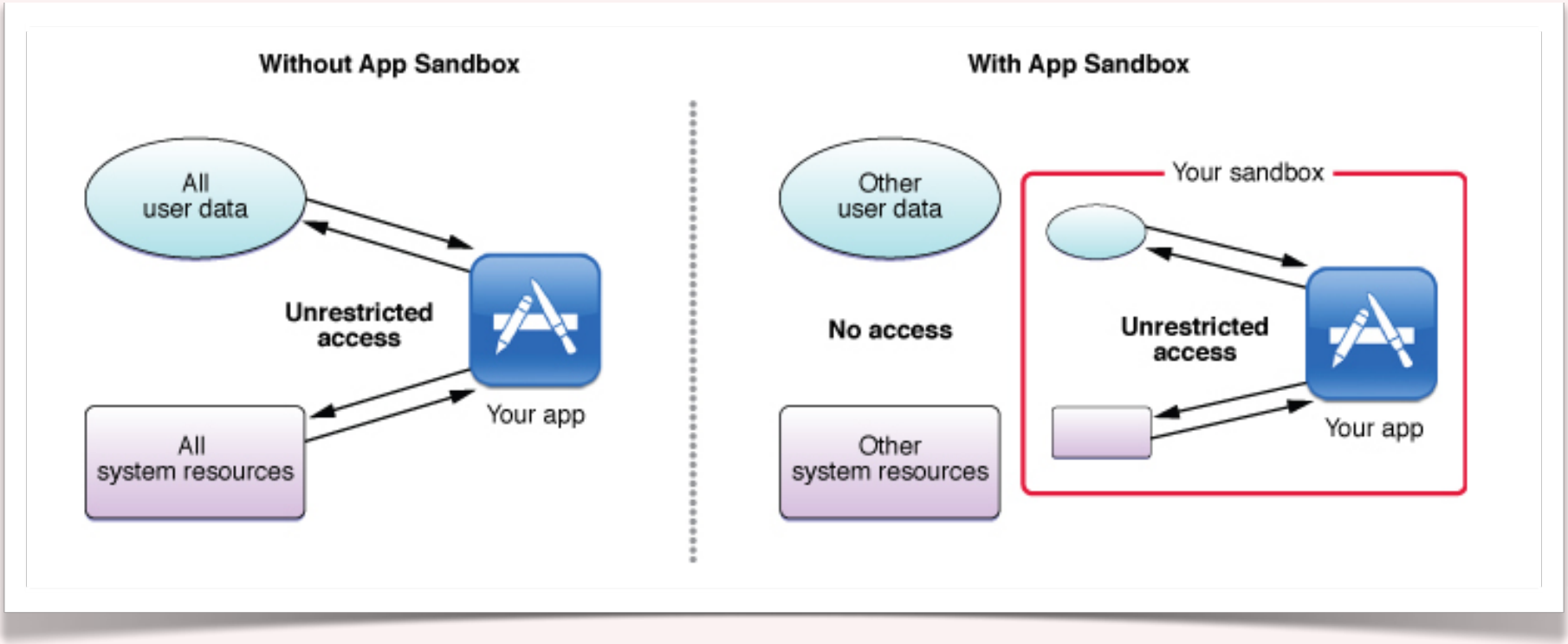
- An app's "Personal Space"
- A wall of protection surrounding your app
- Keeps the app from doing bad things
- A preemptive strike by Apple to head off malware, viruses, and to strengthen platform integrity

Keeps “Bad Actors” Out


It's The Future!



How It Works


Without App Sandbox
All user data
Unrestricted access
Your app
All system resources
With App Sandbox
Other user data
No access
Other system resources
Your sandbox
Unrestricted access
Your app

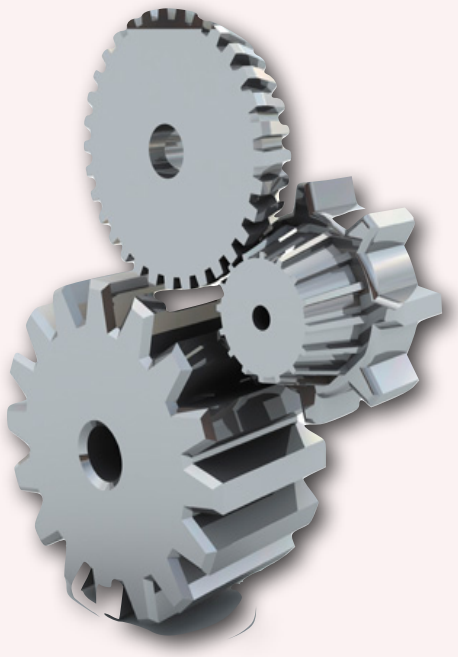

To Sandbox... Or Not?

- What cannot be sandboxed
  - kernel extensions
  - apps that set preferences of other apps
  - apps that send Apple events to other apps

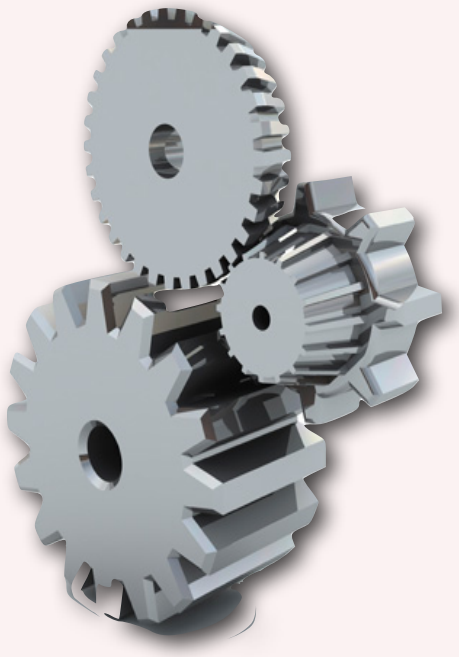
Sandbox? or Quicksand?


Delays In Enforcement

- Was coming November 2011
- Then Postponed to March 2012
- Now Postponed to June 2012



Technology



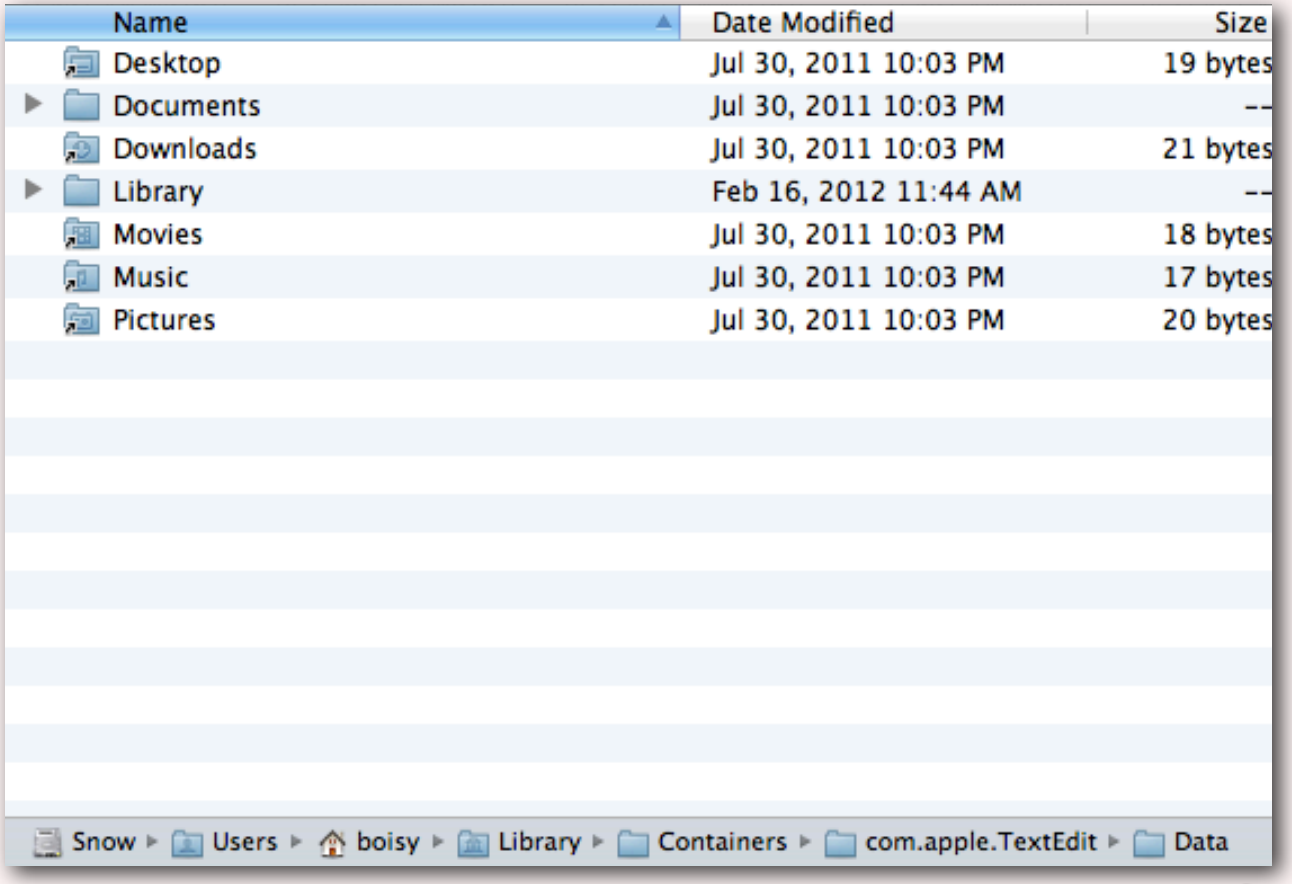

Lowest Level

- sandbox-exec
  - launches apps in a sandboxed environment
  - uses a profile to determine restrictions
  - /usr/share/sandbox
- sandboxd
  - daemon that performs sandbox services


Containers

- Sandboxed apps get their own private file system
  - ${Home}/Library/Containers/<app_bundle_id>/Data/
- Access is unhampered in that folder


Container Folder




Powerbox

- Allows access to files outside of the container
- Supports NSOpenPanel, NSSavePanel, dragging to Dock icon
- User-directed, automatically makes selected files accessible for the duration of the application's run


Entitlements


Entitlements Are...

- "Grants" from Apple which allow your app to do certain things
- Specifically named (com.apple.security....)
- Enumerated in YourApp.entitlements plist file, stored in your application bundle


Enforcement

- Enforced at the system level
- Violations are flagged and prevented from executing
- Violations also show up in Console.app


Entitlements

Read-only access to the user's Movies folder and iTunes movies	`com.apple.security.assets.movies.read-only`
Read/write access to the user's Movies folder and iTunes movies	`com.apple.security.assets.movies.read-write`
Read-only access to the user's Music folder	`com.apple.security.assets.music.read-write`
Read/write access to the user's Music folder	`com.apple.security.assets.pictures.read-only`
Read-only access to the user's Pictures folder	`com.apple.security.assets.pictures.read-only`
Read/write access to the user's Pictures folder	`com.apple.security.assets.pictures.read-write`
Interaction with Bluetooth devices	`com.apple.security.device.bluetooth`
Capture of movies and still images using the built-in camera, if available	`com.apple.security.device.camera`


Entitlements

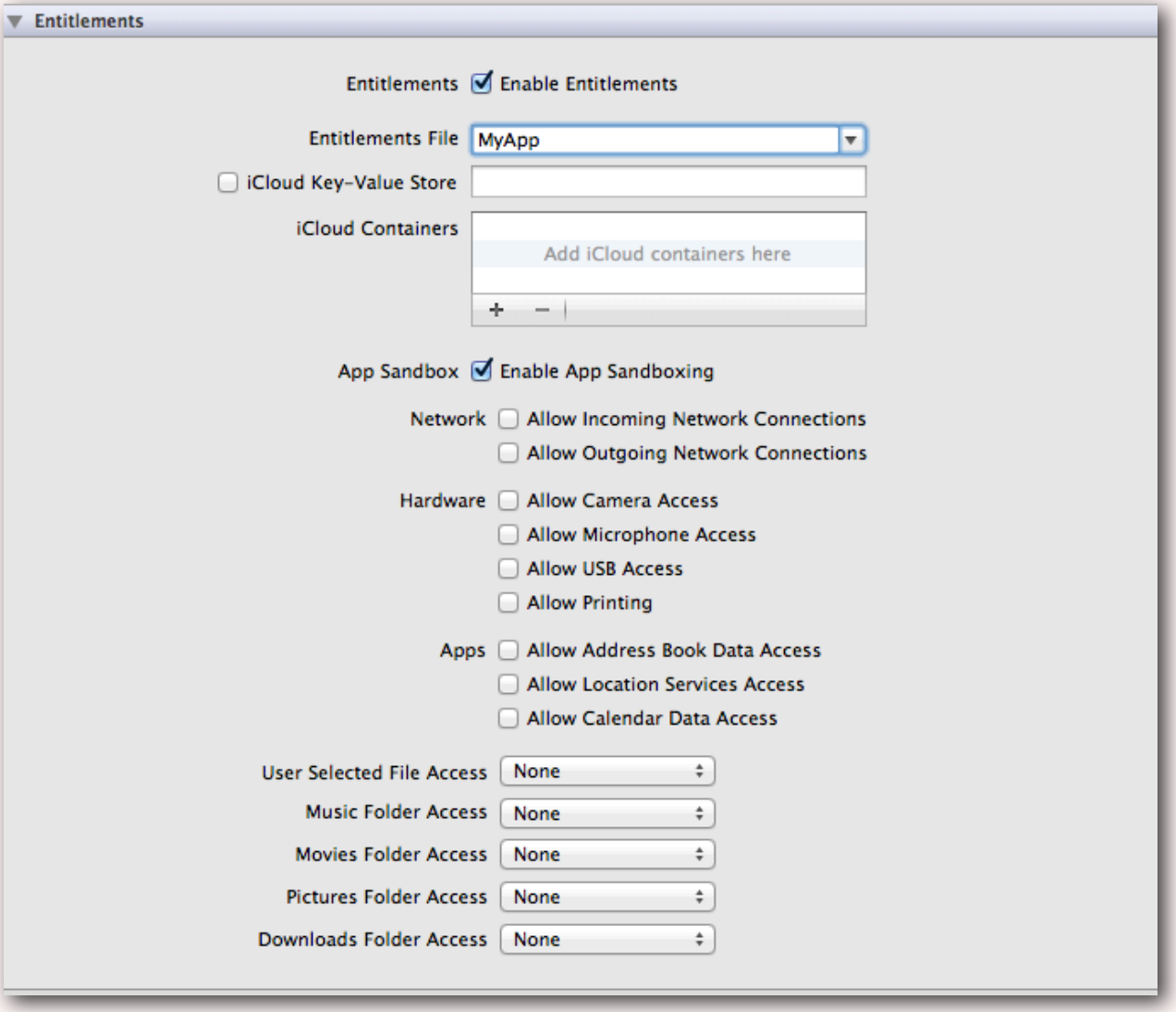
Recording of audio using the built-in microphone, if available, along with access to audio input using any Core Audio API that supports audio input	`com.apple.security.device.microphone`
Interaction with serial devices	`com.apple.security.device.serial`
Interaction with USB devices, including HID devices such as joysticks	`com.apple.security.device.usb`
Read/write access to the user's Downloads folder	`com.apple.security.files.downloads.read-write`
Use of app-scoped bookmarks and URLs	`com.apple.security.files.bookmarks.app-scope`
Use of document-scoped bookmarks and URLs	`com.apple.security.files.bookmarks.collection-scope`
Read-only access to files the user has selected using an Open or Save dialog	`com.apple.security.files.user-selected.read-only`
Read-only access to files the user has selected using an Open or Save dialog	`com.apple.security.files.user-selected.read-only`
Read/write access to files the user has selected using an Open or Save dialog	`com.apple.security.files.user-selected.read-write`
Child process inheritance of the parent's sandbox	`com.apple.security.inherit`


Entitlements

Outgoing network socket for connecting to other machines	`com.apple.security.network.client`
Incoming network socket for listening for requests from other machines	`com.apple.security.network.server`
Read/write access to contacts in the user's address book; allows apps to infer the default address book if more than one is present on a system	`com.apple.security.personal-information.addressbook`
Use of the Core Location framework for determining the computer's geographical location	`com.apple.security.personal-information.location`
Printing	`com.apple.security.print`


Exception Entitlements

- Temporary grants by Apple for tasks that haven't been permanently dealt with
- May be altered or completely disappear in subsequent revisions of the OS
- Ex: Sending Apple Events, Home and Absolute path relative access


Entitlements
Entitlements
Enable Entitlements
Entitlements File
MyApp
iCloud Key-Value Store
iCloud Containers
Add iCloud containers here
App Sandbox
Enable App Sandboxing
Network
Allow Incoming Network Connections
Allow Outgoing Network Connections
Hardware
Allow Camera Access
Allow Microphone Access
Allow USB Access
Allow Printing
Apps
Allow Address Book Data Access
Allow Location Services Access
Allow Calendar Data Access
User Selected File Access
None
Music Folder Access
None
Movies Folder Access
None
Pictures Folder Access
None
Downloads Folder Access
None

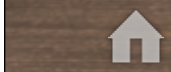

XPC


Privilege Separation

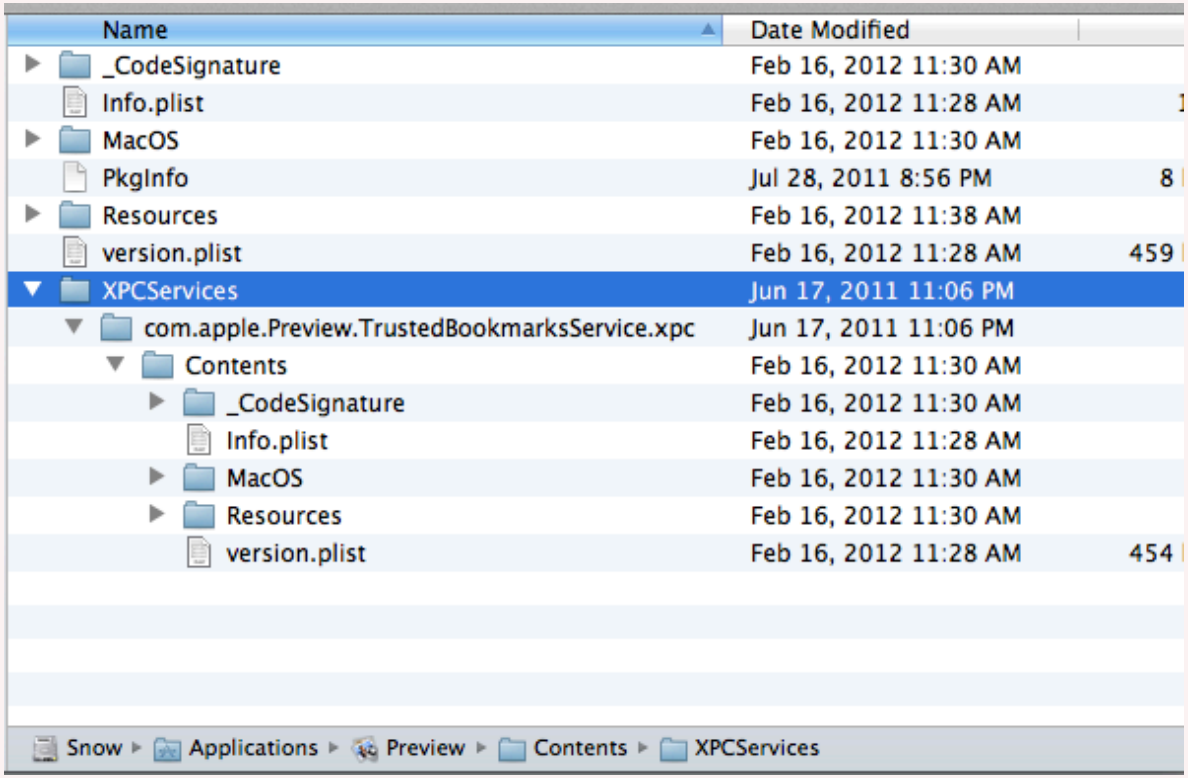
- Functionality is split into smaller executable components on a process level
- Each component has its own entitlements
- Use of Interprocess Communication to establish connection between main app and XPC components


The Benefits

- Functionality is partitioned
- Attackers can only affect a subset of your application
- Code reuse between apps




XPC In Preview




Demo


Migrating Existing Apps


Migration Manifest

- A plist file embedded in your app
- Facilitates moving files from old locations to new sandboxed folder


CAUTION


App Sandboxing Caveats

- Sandbox violations can cause errors in places that you have not expected them
- Check for, and handle, error values on methods that you call
- Not doing so can cause unexpected app failures


A Case Study

- Sandboxed App "A" starts up and loads in 3rd party QuickTime plug-ins
- One plug-in attempts to open a shared memory object, but never checks for failure.
- Sandbox violation prevents opening, but plug-in attempts to use a NULL pointer
- Guess what happened to App "A"??


The Bottom Line

- 3rd party frameworks, libraries or plug-ins are susceptible points of failure
- Test your app thoroughly to find and eliminate sandbox violations


Summary

- App Sandboxing protects your apps
- Entitlements guide your app's behavior
- XPC mitigates security issues by partitioning behavior into small processes
- Testing is critical to surface issues raised by sandbox violations


More Information

- Apple's Developer Forums
- App Sandbox Design Guide
- Daemons and Services Programming Guide

Boisy G. Pitre
boisy@tee-boy.com