

command-line warm-up

Utilities

Navigation: < > [Grid] [List] [Compare] [Info] [Settings] [Share] [Link] [Refresh] [Search]

Search: Search

Name	Date Modified	Size	Kind
DeployStudio Runtime	May 27, 2014, 3:26 PM	93 bytes	Alias
Digital Color Meter	May 16, 2016, 11:55 AM	1.3 MB	Appl
Disk Utility	May 16, 2016, 11:55 AM	4.7 MB	Appl
Disney Animation Setup Assistant	Apr 30, 2015, 12:26 PM	46.9 MB	Appl
Grab	May 16, 2016, 11:55 AM	2 MB	Appl
Grapher	Mar 21, 2016, 4:05 PM	36.6 MB	Appl
HP Utility	Sep 13, 2013, 2:18 PM	46 bytes	Alias
Keychain Access	May 16, 2016, 11:55 AM	14.2 MB	Appl
Managed Software Update	Jun 13, 2016, 9:04 AM	30 bytes	Alias
Migration Assistant	May 16, 2016, 11:55 AM	2.7 MB	Appl
QuickTime Player 7	Oct 22, 2013, 4:56 PM	22.8 MB	Appl
Script Editor	Mar 21, 2016, 4:05 PM	9.3 MB	Appl
System Information	Mar 21, 2016, 4:05 PM	262.7 MB	Appl
Terminal	May 16, 2016, 11:55 AM	8.6 MB	Appl
Uninstall Adobe IT tools	Oct 30, 2014, 11:39 AM	88 bytes	Alias
VoiceOver Utility	May 16, 2016, 11:55 AM	9.3 MB	Appl
XQuartz	May 18, 2016, 9:14 AM	7.8 MB	Appl
yo	May 23, 2015, 6:28 AM	4.6 MB	Appl

1 of 35 selected, 23.35 GB available



gneagle@:~ — -tcsh — 80x24

Last login: Thu Jun 16 15:49:42 on ttys003
gneagle@humantorch:~ % █

where am i?

pwd: print working directory

```
Last login: Mon Jun 27 07:49:42 on ttys003
gneagle@humantorch:~ % pwd
/Users/gneagle
gneagle@humantorch:~ %
```

what's here?

ls: list directory contents

```
gneagle@humantorch:~ % ls
```

```
Applications          Movies  
Creative Cloud Files  Music  
Desktop               Pictures  
Documents             Public  
Downloads             Sites  
Dropbox               Virtual Machines.localized  
Library  
gneagle@humantorch:~ %
```

ls: list directory contents

```
gneagle@humantorch:~ % ls -al
```

```
total 488
```

```
drwxr-xr-x@ 92 gneagle  staff   3128 Jun 17 09:32 .
drwxr-xr-x  9 root      admin   306 Oct  8 2015 ..
-r-----  1 gneagle  staff    7 Apr 26 08:24 .CFUserTextEncoding
-rw-----  1 gneagle  staff 45060 Jun 17 10:37 .DS_Store
drwxr-xr-x  4 gneagle  staff   136 Nov  6 2012 .DownloadManager
drwx-----  3 gneagle  staff   102 Jun 22 2015 .FileSync
drwxr-xr-x  3 gneagle  staff   102 Nov  6 2012 .GalleryRemote
-rw-----  1 gneagle  staff  1322 Nov  6 2012 .ICEauthority
drwxrwxr-x  2 gneagle  staff    68 Mar 25 10:14 .InstallAnywhere
-rw-----  1 gneagle  staff    32 Nov  6 2012 .MCOP-random-seed
drwxrwxr-x  3 gneagle  staff   102 Apr 26 2013 .RSA
drwxr-xr-t  4 gneagle  staff   136 Nov  6 2012 .TemporaryItems
drwx-----  4 gneagle  staff   136 Jun 17 09:56 .Trash
-rw-----  1 gneagle  staff   996 Apr 21 14:39 .Xauthority
drwxr-xr-x  2 gneagle  staff    68 Nov  6 2012 .Xcode
drwxr-xr-x  5 gneagle  staff   170 Nov  6 2012 .adobe
-rw-r--r--  1 gneagle  staff   190 Nov  6 2012 .alias
drwxrwxr-x  3 gneagle  staff   102 Sep 11 2014 .aspera
```


drwxr-xr-x

6 gneagle

staff

204 Feb 23 15:20 Public

mode

owner

group

size

modification date

name

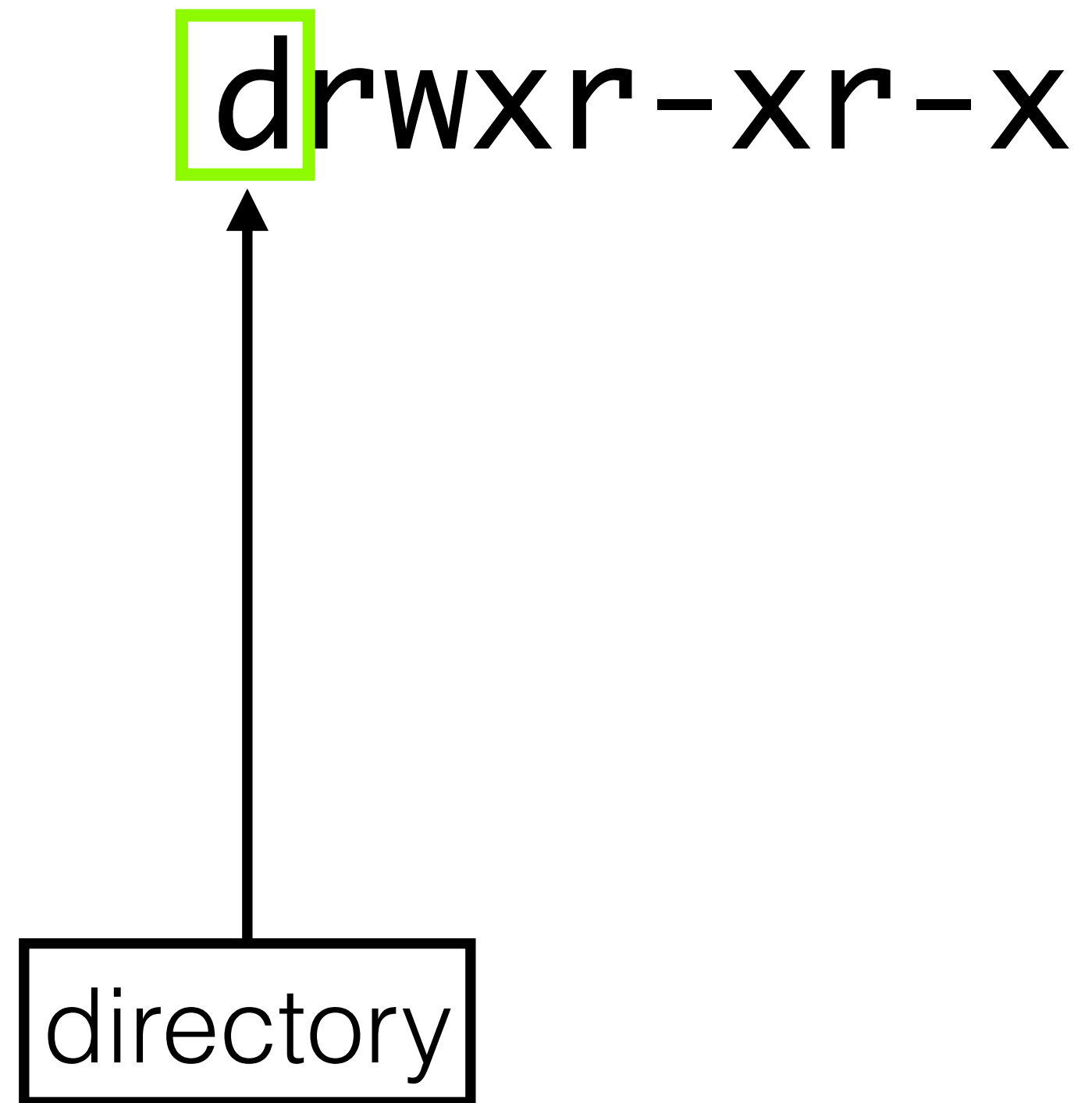
number of links

drwxr-xr-x

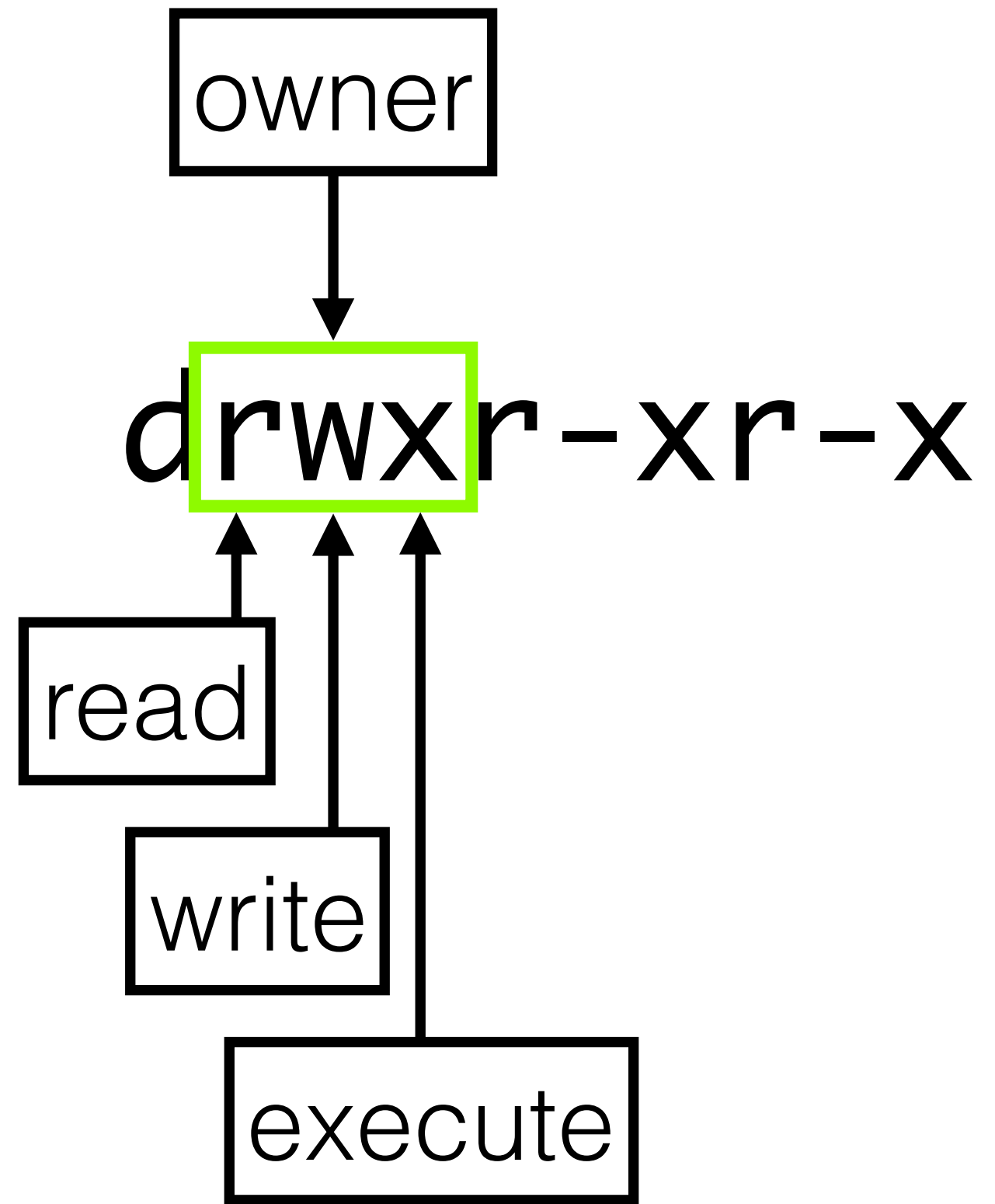
6 gneagle staff

mode

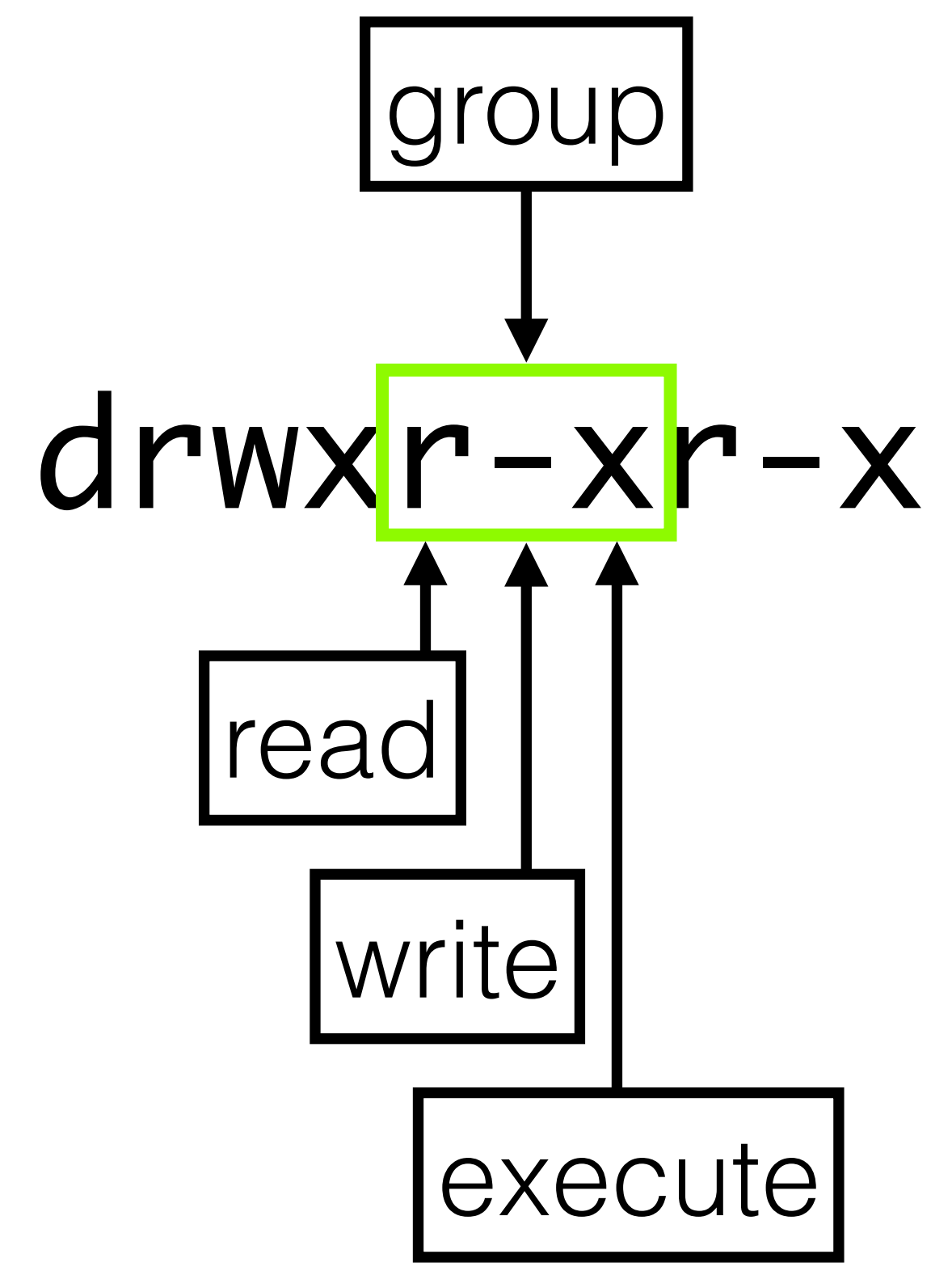




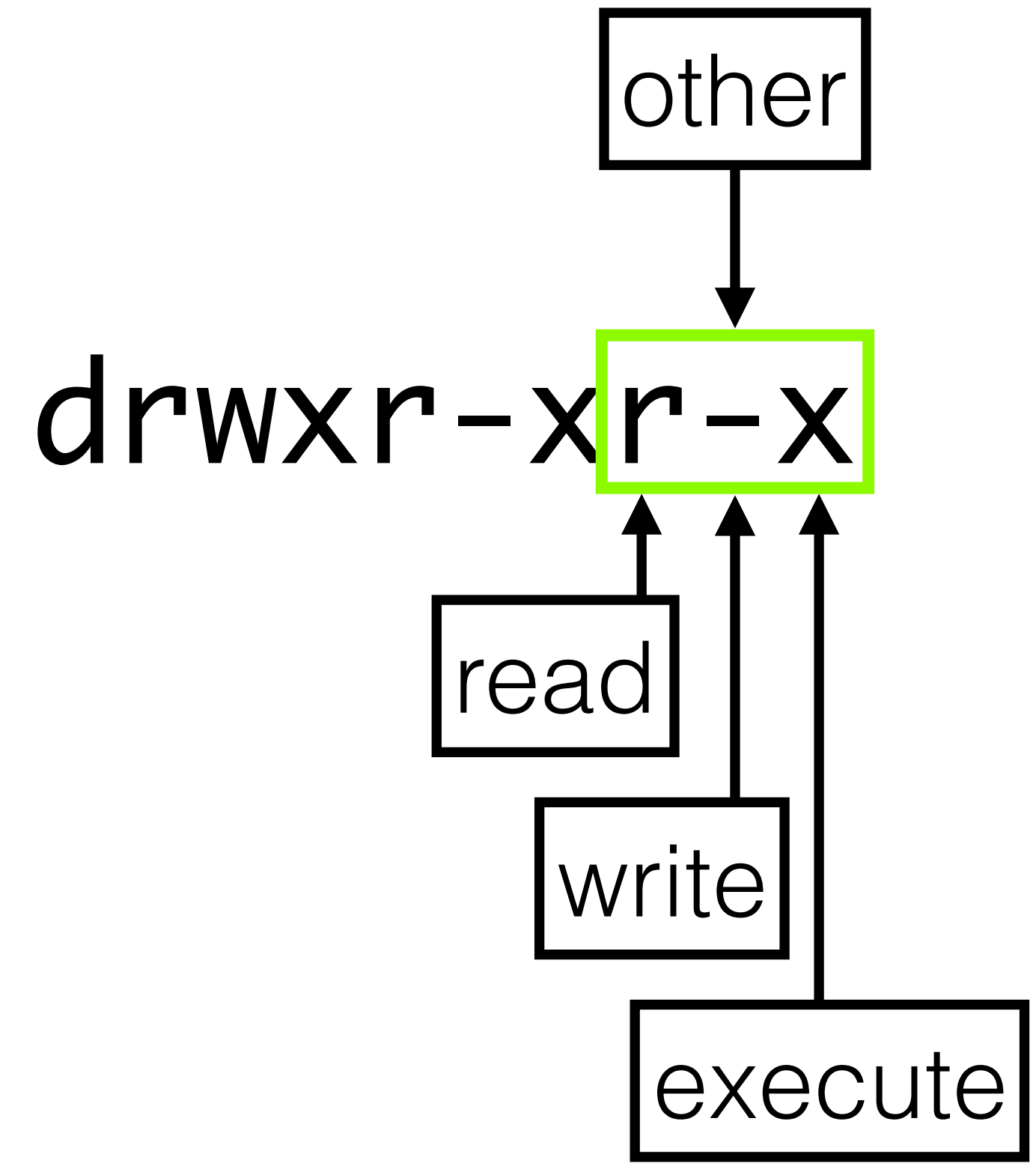
6 gneagle staff



6 gneagle staff



6 gneagle staff



6 gneagle staff

moving around

cd: change directory

```
Last login: Mon Jun 27 07:49:42 on ttys003
gneagle@humantorch:~ % pwd
/Users/gneagle
gneagle@humantorch:~ % cd /Applications
gneagle@humantorch:/Applications % pwd
/Applications
gneagle@humantorch:/Applications %
```


cd: change directory

```
gneagle@humantorch:~ % cd /Applications/  
gneagle@humantorch:/Applications % cd ~  
gneagle@humantorch:~ % pwd  
/Users/gneagle
```

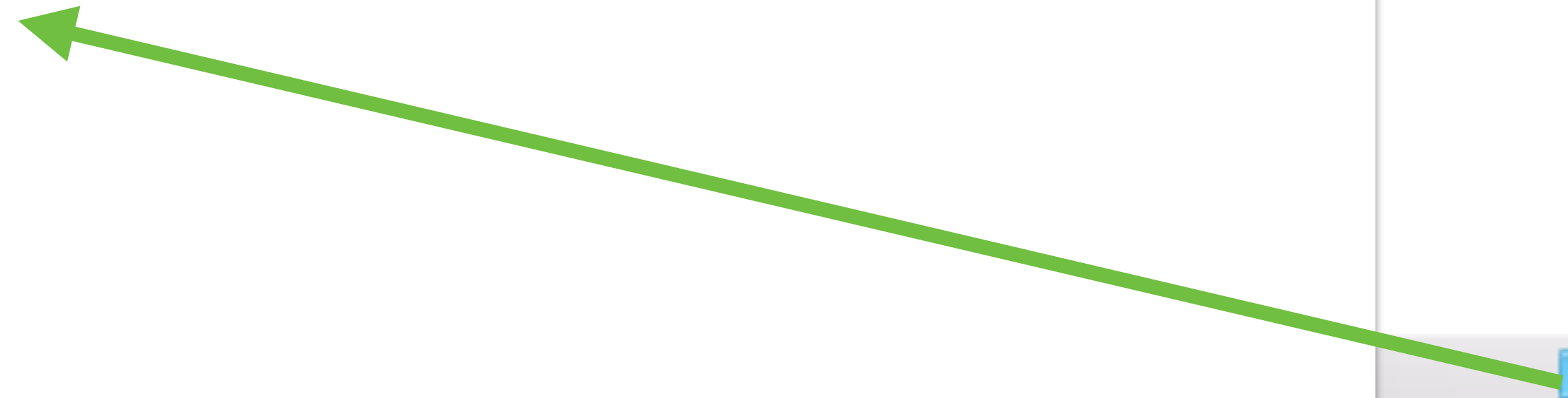
tab completion

```
gneagle@humantorch:~ % ls /Applications/Utilities/
```

```
gneagle@humantorch:~ % ls /usr/share/dict/words
```

dragging into Terminal

```
gneagle@:~ — -tcsh — 80x24
Last login: Mon Jun 20 15:41:24 on ttys002
gneagle@humantorch:~ % ls █
```



Library

^	Date Modified	Size	K
	Today, 4:30 PM	--	F
	Oct 7, 2015, 1:11 PM	--	F
	Jun 14, 2013, 4:05 PM	--	F
	Apr 13, 2015, 1:14 PM	--	F
	Oct 7, 2015, 1:11 PM	--	F
	Nov 6, 2012, 4:22 PM	--	F
s	Nov 7, 2012, 2:18 PM	--	F
s	Apr 20, 2016, 10:31 AM	--	F
	Nov 6, 2012, 4:22 PM	--	F
	Nov 7, 2012, 2:17 PM	--	F
	Nov 6, 2012, 4:22 PM	--	F
	Nov 17, 2013, 7:02 AM	--	F
	Nov 7, 2012, 2:17 PM	--	F
	Apr 7, 2015, 8:32 AM	--	F
	Nov 6, 2012, 4:22 PM	--	F
	Jun 15, 2016, 7:04 PM	--	F

- ▶ NDSPCShowServer
- ▶ ODBC
- ▶ PDF Services
- ▶ Phones
- ▶ PhotoshopCrashes

- ▶ Movies
- ▶ Music
- ▶ Pictures
- ▶ gneagle

```
gneagle@:~ — -tcsh — 80x24
Last login: Mon Jun 20 15:41:24 on ttys002
gneagle@humantorch:~ % ls /Users/gneagle/Library █
```

Library

Settings Share Lock iCloud Search

^	Date Modified	Size	K
	Today, 4:30 PM	--	F
	Oct 7, 2015, 1:11 PM	--	F
	Jun 14, 2013, 4:05 PM	--	F
	Apr 13, 2015, 1:14 PM	--	F
	Oct 7, 2015, 1:11 PM	--	F
	Nov 6, 2012, 4:22 PM	--	F
s	Nov 7, 2012, 2:18 PM	--	F
s	Apr 20, 2016, 10:31 AM	--	F
	Nov 6, 2012, 4:22 PM	--	F
	Nov 7, 2012, 2:17 PM	--	F
	Nov 6, 2012, 4:22 PM	--	F
	Nov 17, 2013, 7:02 AM	--	F
	Nov 7, 2012, 2:17 PM	--	F
	Apr 7, 2015, 8:32 AM	--	F
	Nov 6, 2012, 4:22 PM	--	F
	Jun 15, 2016, 7:04 PM	--	F

▶ NDSPCShowServer
 ▶ ODBC
 ▶ PDF Services
 ▶ Phones
 ▶ PhotoshopCrashes

- ▶ Movies
- ▶ Music
- ▶ Pictures
- ▶ gneagle

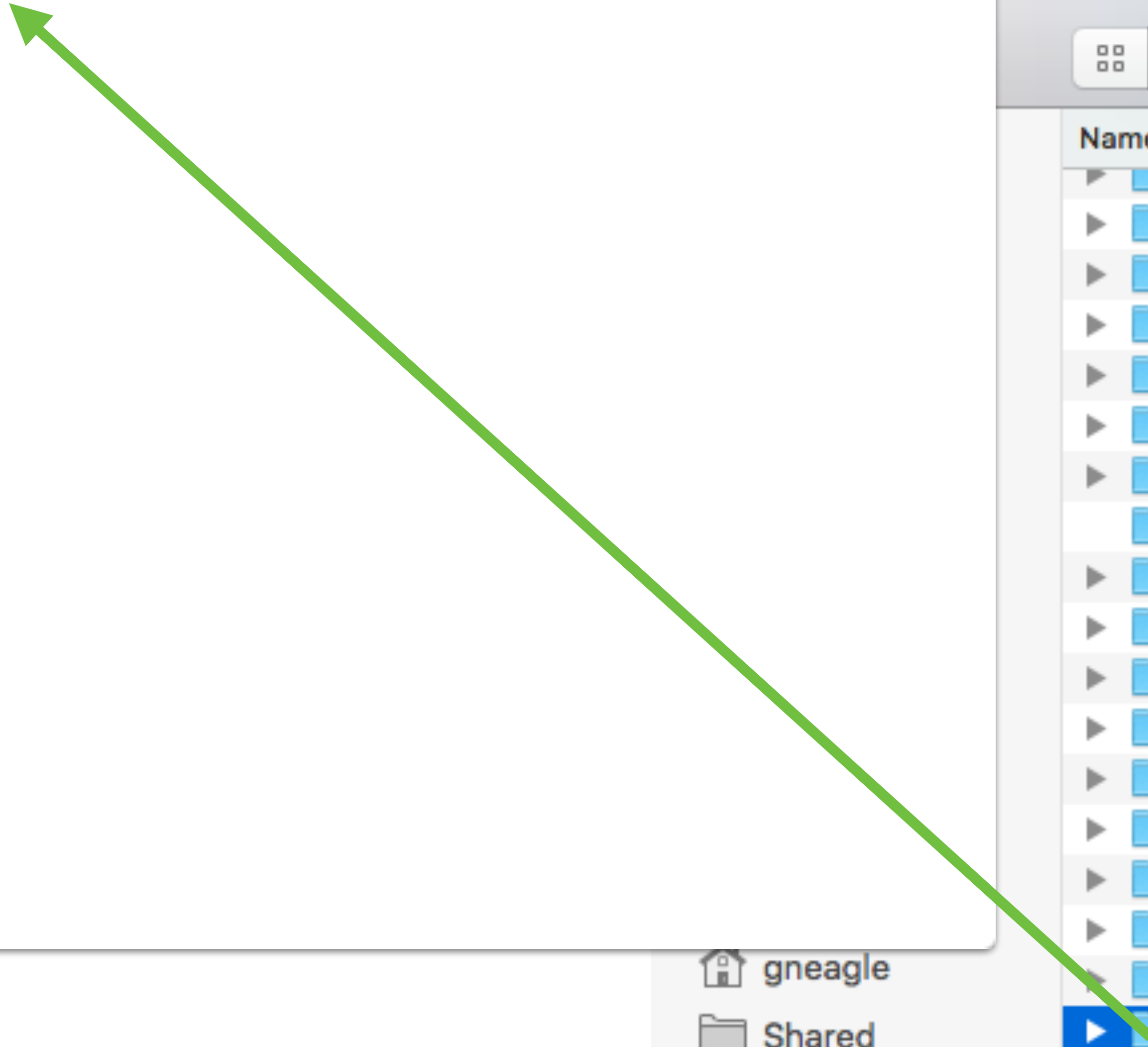
```
gneagle@:~ — -tcsh — 80x24
Last login: Mon Jun 20 15:41:24 on ttys002
gneagle@humantorch:~ % ls █
```

Library

Name	Date Modified
Logs	Today, 4:36 PM
Mail	Oct 7, 2015, 1:11 PM
Mail Downloads	Jun 14, 2013, 4:05 P
Messages	Apr 13, 2015, 1:14 P
Metadata	Oct 7, 2015, 1:11 PM
Microsoft	Nov 6, 2012, 4:22 P
Microsoft Windows	Nov 7, 2012, 2:18 P
Mobile Documents	Apr 20, 2016, 10:31
MobileDevice	Nov 6, 2012, 4:22 P
MobileSafari	Nov 7, 2012, 2:17 P
Mozilla	Nov 6, 2012, 4:22 P
NDSPCShowServer	Nov 17, 2013, 7:02 A
ODBC	Nov 7, 2012, 2:17 P
PDF Services	Apr 7, 2015, 8:32 AM
Phones	Nov 6, 2012, 4:22 P
PhotoshopCrashes	Jun 15, 2016, 7:04 P
PreferencePanels	Oct 26, 2013, 9:48 .
Preferences	Today, 4:51 PM

gneagle
Shared

1 of 97 selected, 32.21 GB available




```
gneagle@:~ — -tcsh — 80x24
Last login: Mon Jun 20 15:41:24 on ttys002
gneagle@humantorch:~ % ls /Users/gneagle/Library/Preferences
```

gneagle
Shared

Library

Name	Date Modified
Logs	Today, 4:36 PM
Mail	Oct 7, 2015, 1:11 PM
Mail Downloads	Jun 14, 2013, 4:05 P
Messages	Apr 13, 2015, 1:14 P
Metadata	Oct 7, 2015, 1:11 PM
Microsoft	Nov 6, 2012, 4:22 P
Microsoft Windows	Nov 7, 2012, 2:18 PM
Mobile Documents	Apr 20, 2016, 10:31
MobileDevice	Nov 6, 2012, 4:22 P
MobileSafari	Nov 7, 2012, 2:17 PM
Mozilla	Nov 6, 2012, 4:22 P
NDSPCShowServer	Nov 17, 2013, 7:02 A
ODBC	Nov 7, 2012, 2:17 PM
PDF Services	Apr 7, 2015, 8:32 AM
Phones	Nov 6, 2012, 4:22 P
PhotoshopCrashes	Jun 15, 2016, 7:04 P
PreferencePanes	Oct 26, 2013, 9:48 .
Preferences	Today, 4:51 PM

1 of 97 selected, 32.21 GB available

```
gneagle@:~ — -tcsh — 80x24
widget-com.apple.widget.stocks.plist
widget-com.apple.widget.stocks.plist.lockfile
widget-com.apple.widget.unitconverter.plist
widget-com.apple.widget.weather.plist
widget-com.apple.widget.weather.plist.5VEKw4I
widget-com.apple.widget.weather.plist.7Ai3MSw
widget-com.apple.widget.weather.plist.7MXSEiJ
widget-com.apple.widget.weather.plist.IW0aew0
widget-com.apple.widget.weather.plist.Kr0iMTl
widget-com.apple.widget.weather.plist.NvBdpcN
widget-com.apple.widget.weather.plist.UtG9hiP
widget-com.apple.widget.weather.plist.YaMHv9P
widget-com.apple.widget.weather.plist.lockfile
widget-com.apple.widget.weather.plist.nbmfoIa
widget-com.apple.widget.web-clip.plist
widget-com.apple.widget.web-clip.plist.lockfile
widget-com.apple.widget.worldclock.plist
widget-com.apple.widget.worldclock.plist.lockfile
widget-com.disneyanimation.widget.worldclock.plist
widget-com.trafficgauge.mac.widget.plist
widget-com.utsire.iframe.plist
widget-org.unobtrusive.widget.GoogleMaps.plist
xcodebuild.plist
gneagle@humantorch:~ %
```

gneagle
Shared

Library

Name	Date Modified
Logs	Today, 4:36 PM
Mail	Oct 7, 2015, 1:11 PM
Mail Downloads	Jun 14, 2013, 4:05 P
Messages	Apr 13, 2015, 1:14 P
Metadata	Oct 7, 2015, 1:11 PM
Microsoft	Nov 6, 2012, 4:22 P
Microsoft Windows	Nov 7, 2012, 2:18 P
Mobile Documents	Apr 20, 2016, 10:31
MobileDevice	Nov 6, 2012, 4:22 P
MobileSafari	Nov 7, 2012, 2:17 P
Mozilla	Nov 6, 2012, 4:22 P
NDSPCShowServer	Nov 17, 2013, 7:02 A
ODBC	Nov 7, 2012, 2:17 P
PDF Services	Apr 7, 2015, 8:32 AM
Phones	Nov 6, 2012, 4:22 P
PhotoshopCrashes	Jun 15, 2016, 7:04 P
PreferencePanels	Oct 26, 2013, 9:48 .
Preferences	Today, 4:51 PM

1 of 97 selected, 32.21 GB available

display a (text) file

cat -- concatenate and print files

```
gneagle@humantorch:~ % cat /usr/share/dict/words
```

A

a

aa

aal

aalii

aam

Aani

aardvark

aardwolf

Aaron

Aaronic

Aaronical

Aaronite

Aaronitic

Aaru

Ab

aba

Ababdeh

Ababua

abac

cat -- concatenate and print files

zymosthenic
zymotechnic
zymotechnical
zymotechnics
zymotechny
zymotic
zymotically
zymotize
zymotoxic
zymurgy
Zyrenian
Zyrian
Zyryan
zythem
Zythia
zythum
Zyzomys
Zyzzogeton
gneagle@humantorch:~ %

head -- print the start of a file

```
gneagle@humantorch:~ % head -10 /usr/share/dict/words
```

A

a

aa

aal

aalii

aam

Aani

aardvark

aardwolf

Aaron

tail -- print the end of a file

```
gneagle@humantorch:~ % tail -12 /usr/share/dict/words
zymotically
zymotize
zymotoxic
zymurgy
Zyrenian
Zyrian
Zyryan
zythem
Zythia
zythum
Zyzomys
Zyzzogeton
```

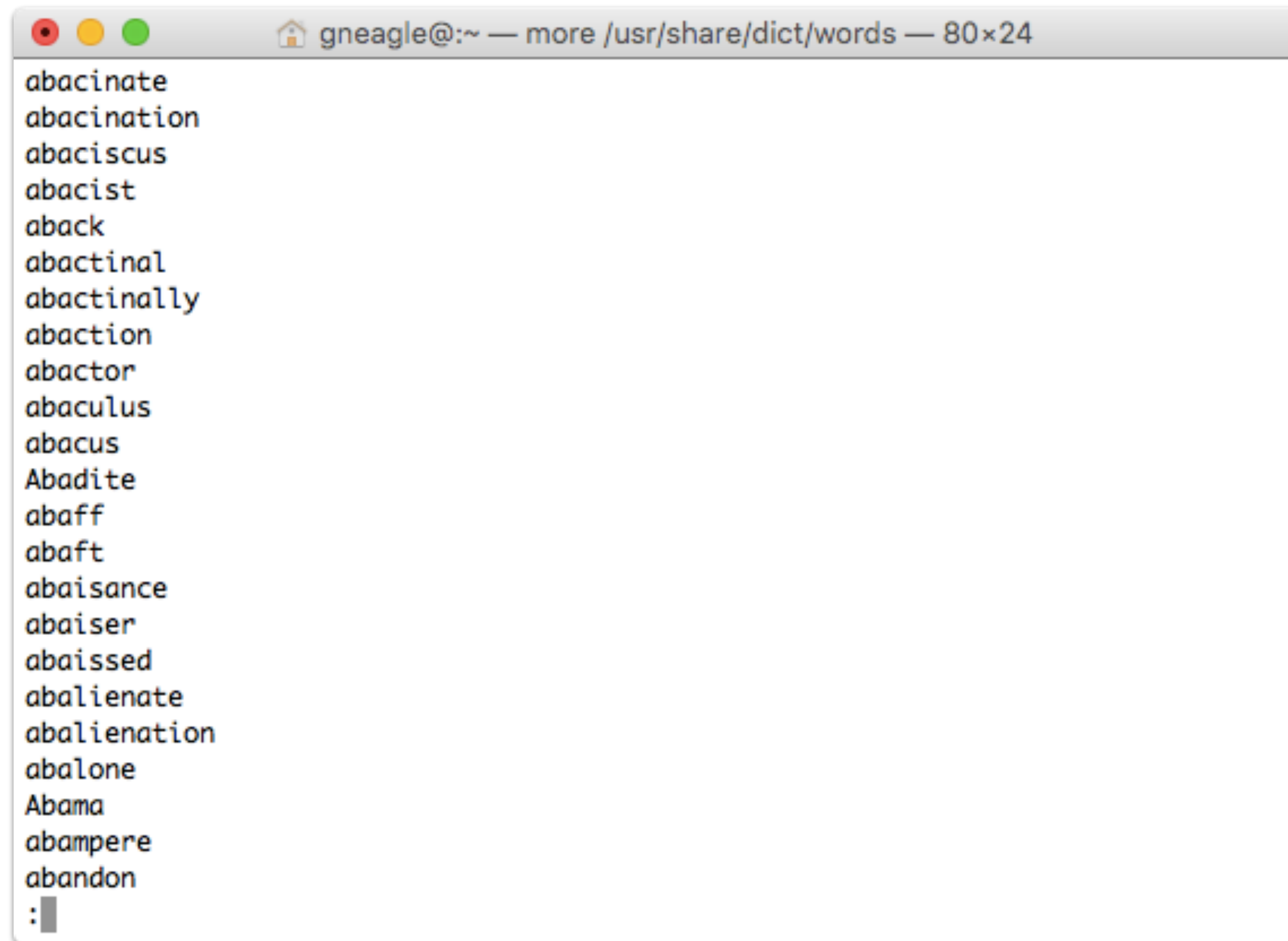
more – view a file a page at a time



A terminal window with a title bar that reads "gneagle@:~ — more /usr/share/dict/words — 80x24". The window contains a list of words from a dictionary, starting with 'A' and 'a'. The words listed are: A, a, aa, aal, aalii, aam, Aani, aardvark, aardwolf, Aaron, Aaronic, Aaronical, Aaronite, Aaronitic, Aaru, Ab, aba, Ababdeh, Ababua, abac, abaca, abacate, abacay. At the bottom of the window, the path "/usr/share/dict/words" is highlighted with a black background and white text.

```
A
a
aa
aal
aalii
aam
Aani
aardvark
aardwolf
Aaron
Aaronic
Aaronical
Aaronite
Aaronitic
Aaru
Ab
aba
Ababdeh
Ababua
abac
abaca
abacate
abacay
/usr/share/dict/words
```


more – view a file a page at a time



The image shows a terminal window with a title bar that reads "gneagle@:~ — more /usr/share/dict/words — 80x24". The window contains a list of words from a dictionary, displayed one per line. The words are: abacinate, abacination, abaciscus, abacist, aback, abactinal, abactinally, abaction, abactor, abaculus, abacus, Abadite, abaff, abaft, abaisance, abaiser, abaissed, abalienate, abalienation, abalone, Abama, abampere, and abandon. A cursor is visible at the end of the last line.

```
abacinate
abacination
abaciscus
abacist
aback
abactinal
abactinally
abaction
abactor
abaculus
abacus
Abadite
abaff
abaft
abaisance
abaiser
abaissed
abalienate
abalienation
abalone
Abama
abampere
abandon
:
```

finding text in a file

grep – search for text patterns

```
grep <text> <file>
```

grep – search for text patterns

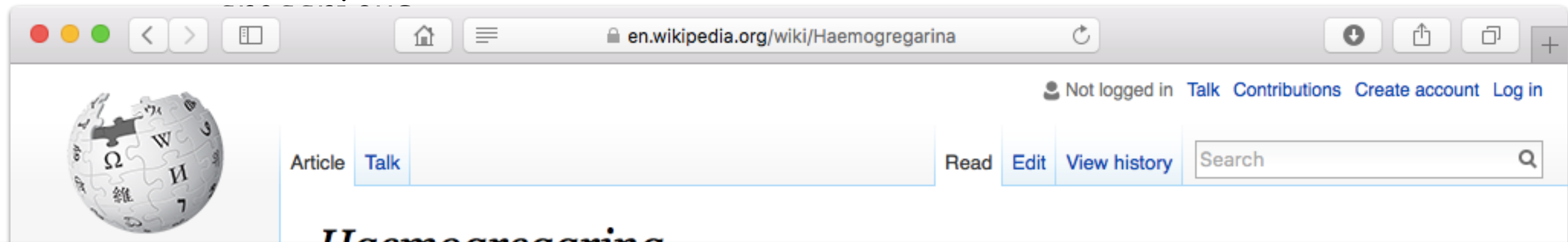
```
gneagle@humantorch:~ % grep grep /usr/share/dict/words  
gneagle@humantorch:~ %
```

grep – search for text patterns

```
gneagle@humantorch:~ % grep greg /usr/share/dict/words
aggregable
aggregant
Aggregata
Aggregatae
aggregate
aggregately
aggregateness
aggregation
aggregative
aggregator
aggregatory
agregation
agrege
coaggregate
coaggregated
coaggregation
congregable
```

reaggregate
reaggregation
Schizogregarine – search for text patterns
schizogregarine
Schizogregarinida
segregable
segregant
segregate
segregateness
segregation
segregational
segregationist
segregative
segregator
subaggregate
unaggregated
uncongregated
uncongregational
unegregious
unregarious
unsegregable
unsegregated
unsegregatedness
gneagle@humantorch:~ %

gregarinous



Haemogregarina

From Wikipedia, the free encyclopedia

Haemogregarina is a genus of haemoprotozoans, parasitic mainly on cold-blooded vertebrates. They are unicellular organisms which are parasitic in the red blood cells. *Haemogregarina* infects lower vertebrates (fish and reptiles) as intermediate hosts and leeches (as definitive hosts).

- [Related changes](#)
- [Upload file](#)
- [Special pages](#)
- [Permanent link](#)
- [Page information](#)
- [Wikidata item](#)
- [Cite this page](#)

General description [\[edit \]](#)

These parasites are generally considered non-pathogenic, and have been described in the red blood cells of desert tortoises. They are elongate to fusiform oval organisms found in the red blood cells. Although the size varies, they are larger than the cell's nucleus. The organism stains a basophilic colour and has a surrounding clear zone.

segregationist

wc – word, line, character, and byte count

```
gneagle@humantorch:~ % wc /usr/share/dict/words  
235886  235886 2493109 /usr/share/dict/words
```


wc – word, line, character, and byte count

```
gneagle@humantorch:~ % wc /usr/share/dict/words  
235886  235886 2493109 /usr/share/dict/words
```

```
gneagle@humantorch:~ % wc -l /usr/share/dict/words  
235886 /usr/share/dict/words
```

wc – word, line, character, and byte count

```
gneagle@humantorch:~ % wc /usr/share/dict/words  
235886  235886 2493109 /usr/share/dict/words
```

```
gneagle@humantorch:~ % wc -l /usr/share/dict/words  
235886 /usr/share/dict/words
```

```
gneagle@humantorch:~ % wc -w /usr/share/dict/words  
235886 /usr/share/dict/words
```

ps – process status

```
gneagle@humantorch:~ % ps axwww
```

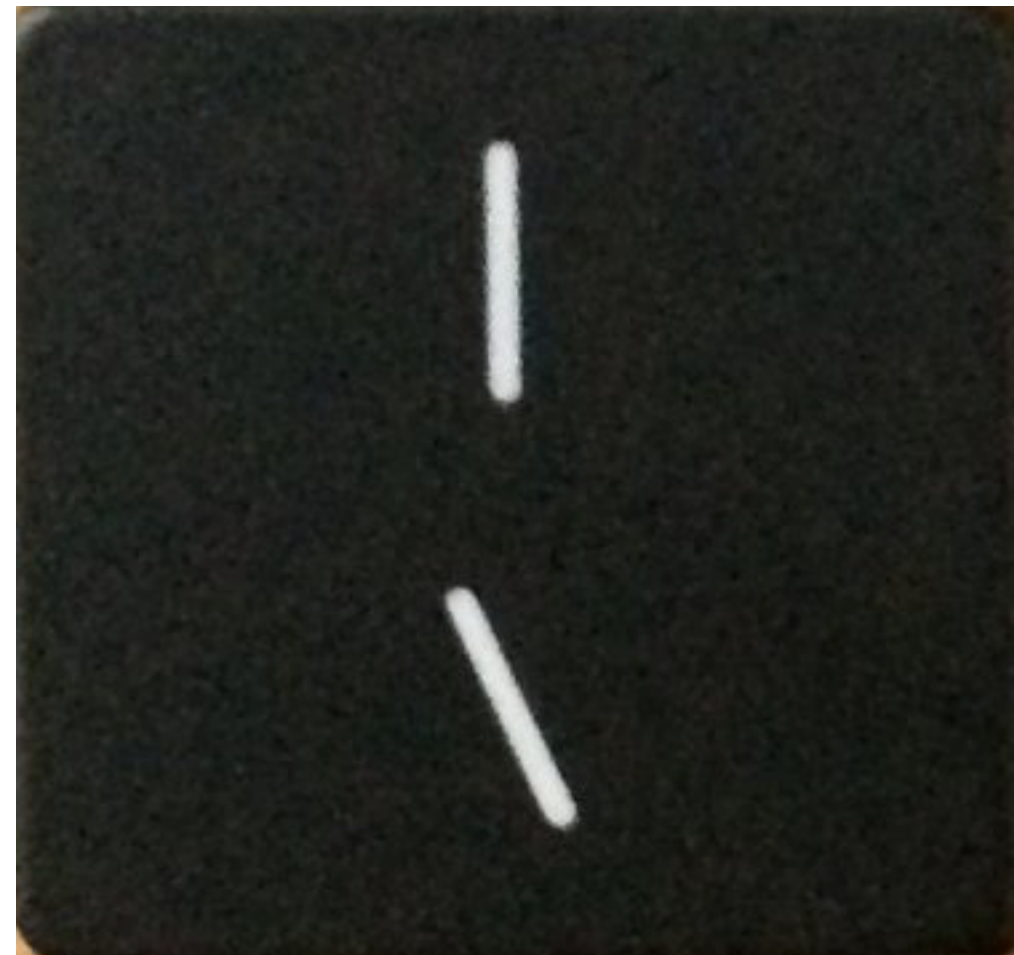
PID	TT	STAT	TIME	COMMAND
1	??	Ss	56:59.98	/sbin/launchd
59	??	Ss	10:43.80	/usr/sbin/syslogd
60	??	Ss	2:09.71	/usr/libexec/UserEventAgent (System)
62	??	SNs	990:11.34	/Applications/CrashPlan.app/Contents/MacOS/CrashPlanService - Dapp=CrashPlanService -Xmn10m -Xms15m -Xmx1024m -DappBaseName=CrashPlan -Djava.awt.headless=true Dfile.encoding=UTF-8 -Djava.net.preferIPv4Stack=true -Dsun.net.inetaddr.ttl=300 - Dnetworkaddress.cache.ttl=300 -Dsun.net.inetaddr.negative.ttl=0 - Dnetworkaddress.cache.negative.ttl=0 -DCP_USER_NAME= -DCP_USER_HOME= -cp lib/ com.backup42.desktop.jar:lang com.backup42.service.CPService CP_USER_NAME= CP_USER_HOME=
63	??	Ss	3:09.34	/usr/libexec/kextd
64	??	Ss	7:25.89	/System/Library/Frameworks/CoreServices.framework/Versions/A/Frameworks/ FSEvents.framework/Versions/A/Support/fseventsd
68	??	Ss	0:08.63	/System/Library/CoreServices/appleeventsd --server
69	??	Ss	3:56.02	/usr/libexec/configd
70	??	Ss	1:50.96	/System/Library/CoreServices/powerd.bundle/powerd
76	??	Ss	8:56.45	/usr/libexec/airportd
79	??	SNs	0:05.52	/usr/libexec/warmd
80	??	Ss	71:17.05	/System/Library/Frameworks/CoreServices.framework/Frameworks/

```
24831  ??  Ss      0:00.43 /System/Library/PrivateFrameworks/CommerceKit.framework/Versions/A/
XPCServices/com.apple.CommerceKit.TransactionService.xpc/Contents/MacOS/
com.apple.CommerceKit.TransactionService
24853  ??  Ss      0:03.98 /System/Library/PrivateFrameworks/CloudDocsDaemon.framework/XPCService
ContainerMetadataExtractor.xpc/Contents/MacOS/ContainerMetadataExtractor
24954  ??  Ss      3:43.44 /usr/sbin/cfprefsd daemon
24957  ??  S       2:44.34 /usr/sbin/cfprefsd agent
25099  ??  S       0:01.92 /usr/sbin/cfprefsd agent
25105  ??  S       0:01.99 /usr/sbin/cfprefsd agent
30468  ??  S       0:00.09 /System/Library/PrivateFrameworks/CallHistory.framework/Support/
CallHistoryPluginHelper
31015  ??  S       0:00.07 /System/Library/Frameworks/CoreServices.framework/Frameworks/
Metadata.framework/Versions/A/Support/mdworker -s mdworker -c MDSImporterWorker -m
com.apple.mdworker.shared
31539  ??  S       0:00.06 /System/Library/Frameworks/CoreServices.framework/Frameworks/
Metadata.framework/Versions/A/Support/mdworker -s mdworker -c MDSImporterWorker -m
com.apple.mdworker.shared
31644  ??  Ss      0:00.34 /usr/sbin/ocspd
31756  ??  S       0:00.02 /System/Library/CoreServices/AirPort Base Station Agent.app/Contents/
MacOS/AirPort Base Station Agent --launchd
31768  ??  SNs     0:00.04 /usr/sbin/netbiosd
32372  ??  Ss      0:08.53 /System/Library/Frameworks/WebKit.framework/Versions/A/XPCServices/
com.apple.WebKit.WebContent.xpc/Contents/MacOS/com.apple.WebKit.WebContent
32561  ??  S       0:00.12 /System/Library/PrivateFrameworks/AOSKit.framework/Versions/A/
XPCServices/com.apple.iCloudHelper.xpc/Contents/MacOS/com.apple.iCloudHelper
```

pipes

pipes – connecting the output of one process
to the input of another

```
gneagle@humantorch:~ % ps axwww | grep Terminal
98045  ??  U   17:04.29 /Applications/Utilities/Terminal.app/Contents/MacOS/Terminal
33534 s001 S+   0:00.00 grep Terminal
```



pipes – connecting the output of one process
to the input of another

```
gneagle@humantorch:~ % ps axwww | wc -l  
296
```

Tearing open packages

What is a package?

A package is an archive of files and directories.

The archive contains information about location, kind, owner, group, and mode of each file and directory.

The archive contains information about **location**, **kind**, **owner**, **group**, and **mode** of each file and directory.

A package may also
optionally run some scripts.

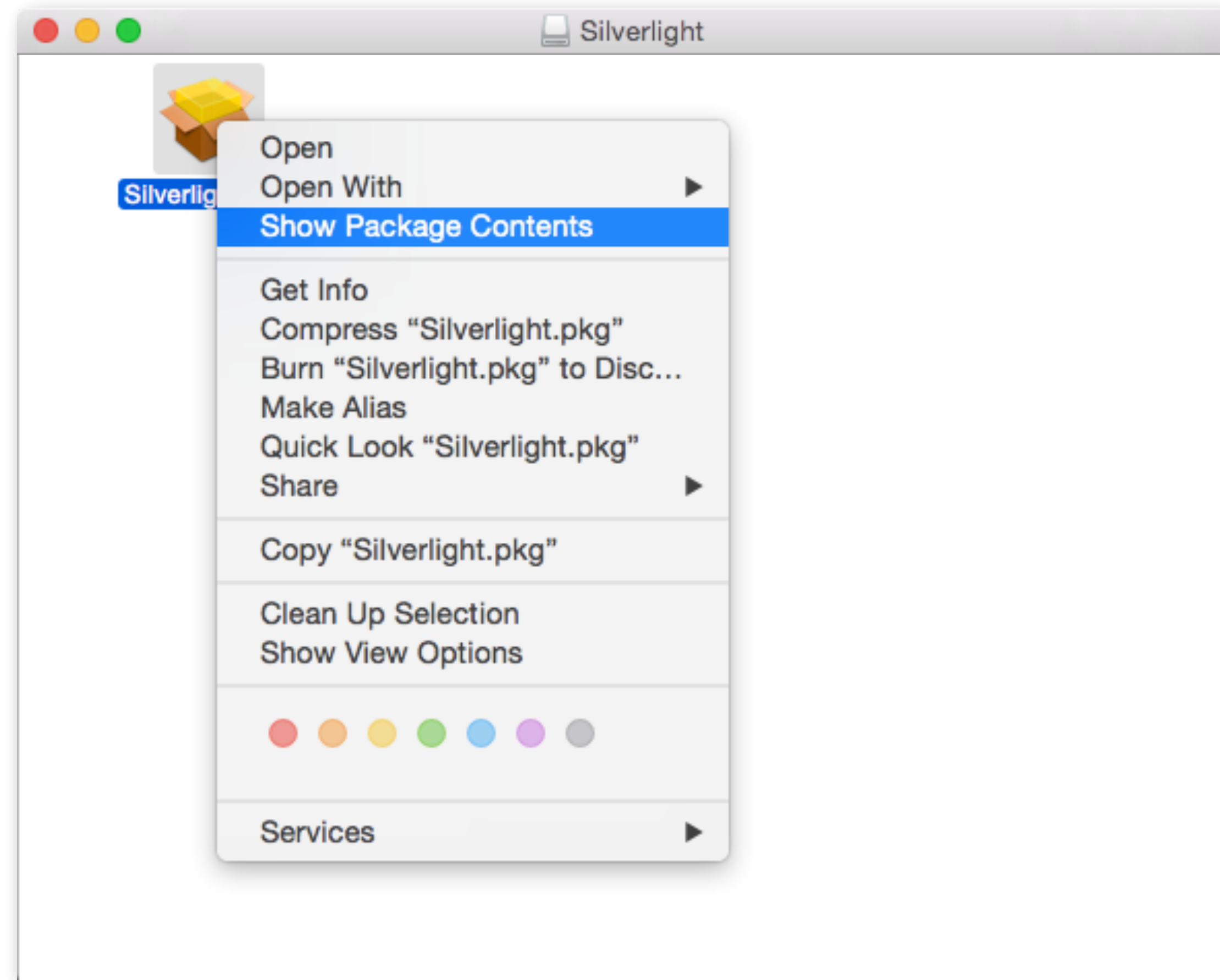
A package **may** also
optionally run some scripts.

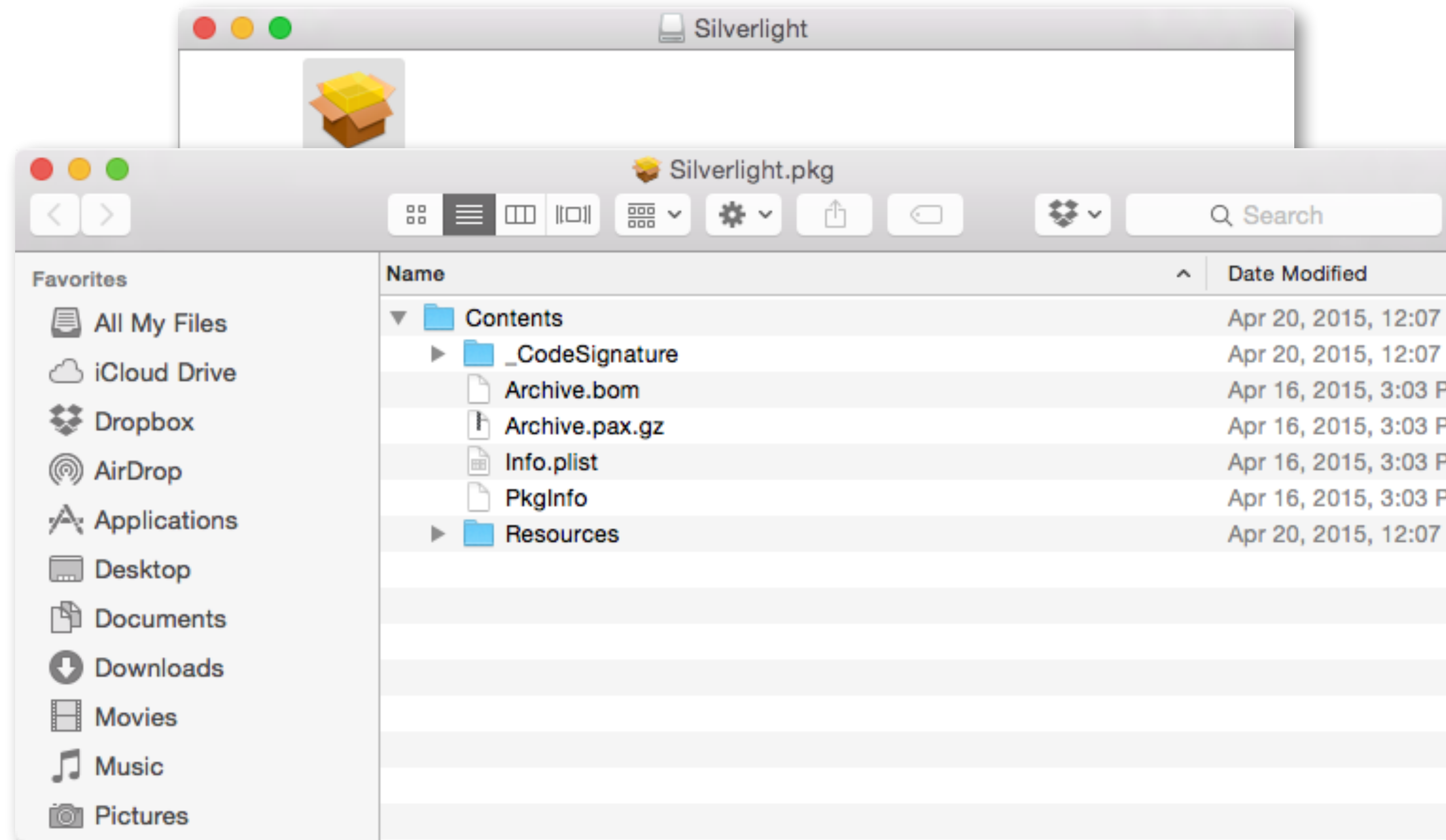
Package Types

- Bundle
- Flat

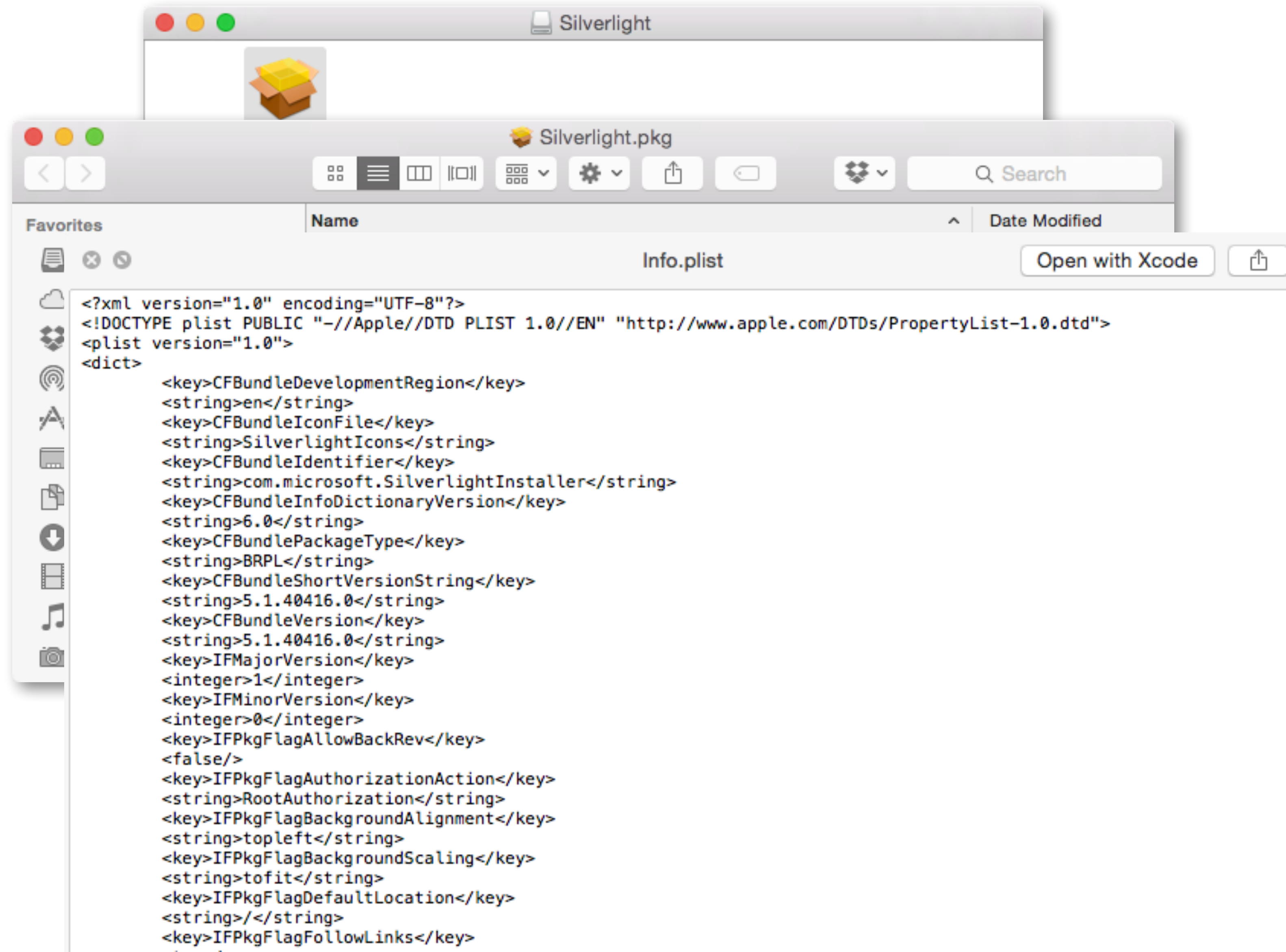
Bundle packages

- Like other OS X bundles (apps, preference panes, etc): a special type of directory





Info.plist: package metadata



```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
  <key>CFBundleDevelopmentRegion</key>
  <string>en</string>
  <key>CFBundleIconFile</key>
  <string>SilverlightIcons</string>
  <key>CFBundleIdentifier</key>
  <string>com.microsoft.SilverlightInstaller</string>
  <key>CFBundleInfoDictionaryVersion</key>
  <string>6.0</string>
  <key>CFBundlePackageType</key>
  <string>BRPL</string>
  <key>CFBundleShortVersionString</key>
  <string>5.1.40416.0</string>
  <key>CFBundleVersion</key>
  <string>5.1.40416.0</string>
  <key>IFMajorVersion</key>
  <integer>1</integer>
  <key>IFMinorVersion</key>
  <integer>0</integer>
  <key>IFPkgFlagAllowBackRev</key>
  <false/>
  <key>IFPkgFlagAuthorizationAction</key>
  <string>RootAuthorization</string>
  <key>IFPkgFlagBackgroundAlignment</key>
  <string>topleft</string>
  <key>IFPkgFlagBackgroundScaling</key>
  <string>tofit</string>
  <key>IFPkgFlagDefaultLocation</key>
  <string>/</string>
  <key>IFPkgFlagFollowLinks</key>
  <true/>
```

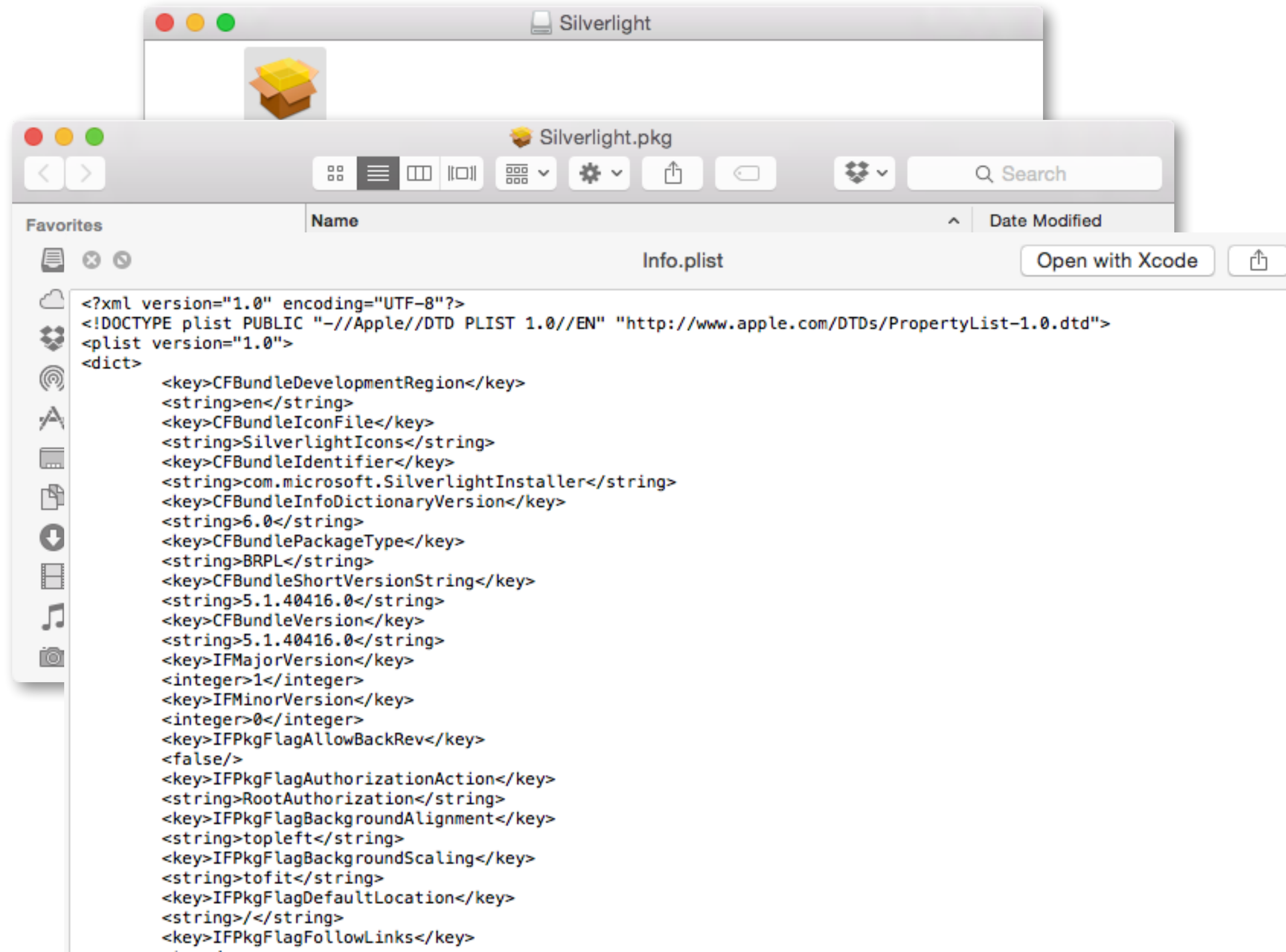
```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
  <key>CFBundleDevelopmentRegion</key>
  <string>en</string>
  <key>CFBundleIconFile</key>
  <string>SilverlightIcons</string>
  <key>CFBundleIdentifier</key>
  <string>com.microsoft.SilverlightInstaller</string>
  <key>CFBundleInfoDictionaryVersion</key>
  <string>6.0</string>
  <key>CFBundlePackageType</key>
  <string>RRPI</string>
  <key>CFBundleShortVersionString</key>
  <string>5.1.40416.0</string>
  <key>CFBundleVersion</key>
  <string>5.1.40416.0</string>
  <key>IFMajorVersion</key>
  <integer>1</integer>
  <key>IFMinorVersion</key>
  <integer>0</integer>
  <key>IFPkgFlagAllowBackRev</key>
  <false/>
  <key>IFPkgFlagAuthorizationAction</key>
  <string>RootAuthorization</string>
  <key>IFPkgFlagBackgroundAlignment</key>
  <string>topleft</string>
  <key>IFPkgFlagBackgroundScaling</key>
  <string>tofit</string>
  <key>IFPkgFlagDefaultLocation</key>
  <string>/</string>
  <key>IFPkgFlagFollowLinks</key>
  <true/>
```

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
  <key>CFBundleDevelopmentRegion</key>
  <string>en</string>
  <key>CFBundleIconFile</key>
  <string>SilverlightIcons</string>
  <key>CFBundleIdentifier</key>
  <string>com.microsoft.SilverlightInstaller</string>
  <key>CFBundleInfoDictionaryVersion</key>
  <string>6.0</string>
  <key>CFBundlePackageType</key>
  <string>BRPL</string>
  <key>CFBundleShortVersionString</key>
  <string>5.1.40416.0</string>
  <key>CFBundleVersion</key>
  <string>5.1.40416.0</string>
  <key>IFMajorVersion</key>
  <integer>1</integer>
  <key>IFMinorVersion</key>
  <integer>0</integer>
  <key>IFPkgFlagAllowBackRev</key>
  <false/>
  <key>IFPkgFlagAuthorizationAction</key>
  <string>RootAuthorization</string>
  <key>IFPkgFlagBackgroundAlignment</key>
  <string>topleft</string>
  <key>IFPkgFlagBackgroundScaling</key>
  <string>tofit</string>
  <key>IFPkgFlagDefaultLocation</key>
  <string>/</string>
  <key>IFPkgFlagFollowLinks</key>
  <true/>
```

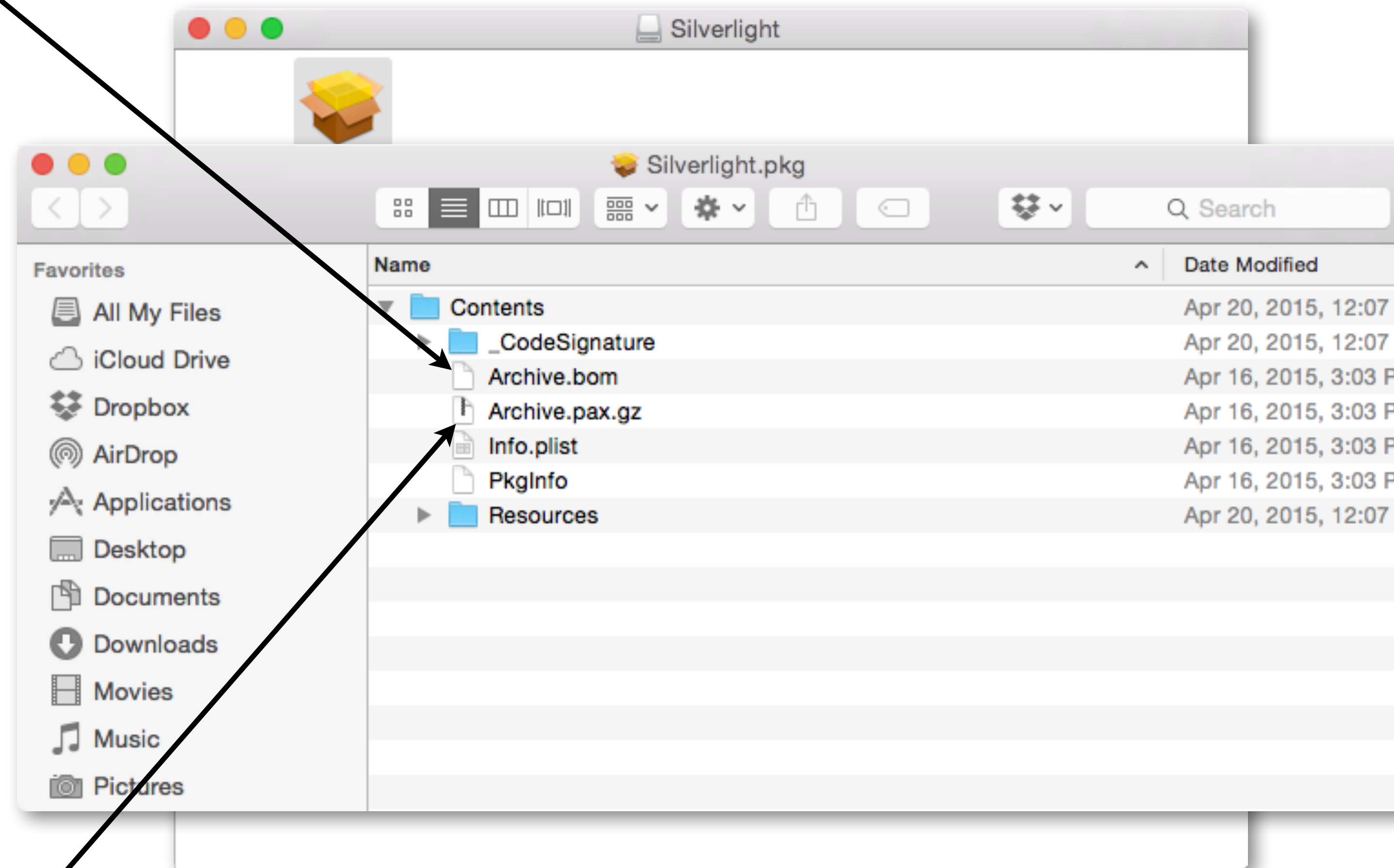
```
<key>CFBundleIconFile</key>
<string>SilverlightIcons</string>
<key>CFBundleIdentifier</key>
<string>com.microsoft.SilverlightInstaller</string>
<key>CFBundleInfoDictionaryVersion</key>
<string>6.0</string>
<key>CFBundlePackageType</key>
<string>BRPL</string>
<key>CFBundleShortVersionString</key>
<string>5.1.40416.0</string>
<key>CFBundleVersion</key>
<string>5.1.40416.0</string>
<key>IFMajorVersion</key>
<integer>1</integer>
<key>IFMinorVersion</key>
<integer>0</integer>
<key>IFPkgFlagAllowBackRev</key>
<false/>
<key>IFPkgFlagAuthorizationAction</key>
<string>RootAuthorization</string>
<key>IFPkgFlagBackgroundAlignment</key>
<string>topleft</string>
<key>IFPkgFlagBackgroundScaling</key>
<string>tofit</string>
<key>IFPkgFlagDefaultLocation</key>
<string>/</string>
<key>IFPkgFlagFollowLinks</key>
<true/>
<key>IFPkgFlagInstallFat</key>
<false/>
<key>IFPkgFlagInstalledSize</key>
<integer>99420</integer>
<key>IFPkgFlagIsRequired</key>
<false/>
<key>IFPkgFlagOverwritePermissions</key>
<false/>
```

```
<key>CFBundleVersion</key>
<string>5.1.40416.0</string>
<key>IFMajorVersion</key>
<integer>1</integer>
<key>IFMinorVersion</key>
<integer>0</integer>
<key>IFPkgFlagAllowBackRev</key>
<false/>
<key>IFPkgFlagAuthorizationAction</key>
<string>RootAuthorization</string>
<key>IFPkgFlagBackgroundAlignment</key>
<string>topleft</string>
<key>IFPkgFlagBackgroundScaling</key>
<string>tofit</string>
<key>IFPkgFlagDefaultLocation</key>
<string>/</string>
<key>IFPkgFlagFollowLinks</key>
<true/>
<key>IFPkgFlagInstallFat</key>
<false/>
<key>IFPkgFlagInstalledSize</key>
<integer>99420</integer>
<key>IFPkgFlagIsRequired</key>
<false/>
<key>IFPkgFlagOverwritePermissions</key>
<false/>
<key>IFPkgFlagRelocatable</key>
<false/>
<key>IFPkgFlagRestartAction</key>
<string>None</string>
<key>IFPkgFlagRootVolumeOnly</key>
<true/>
<key>IFPkgFlagUpdateInstalledLanguages</key>
<false/>
<key>IFPkgFormatVersion</key>
```

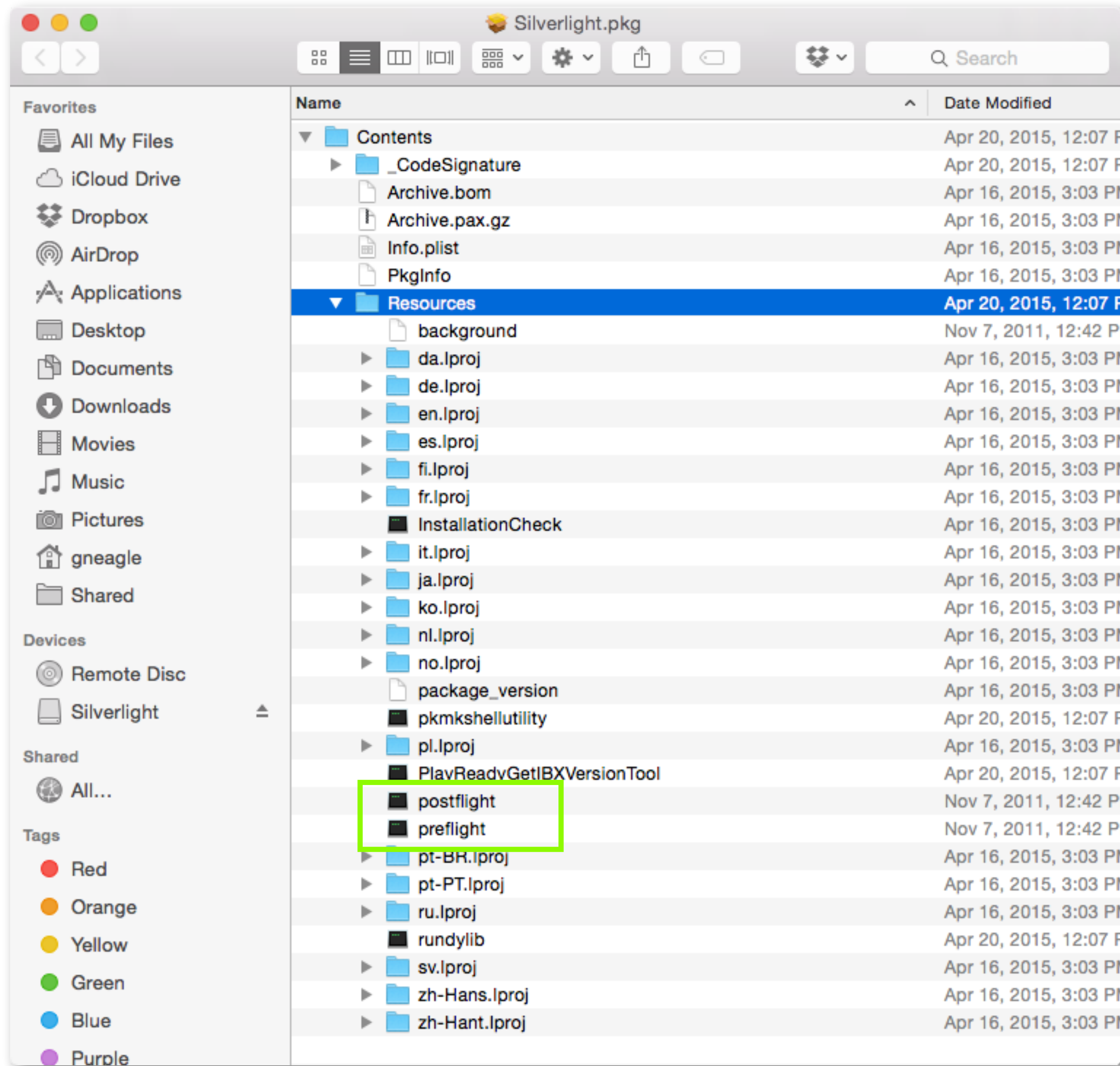

Info.plist: package metadata



Archive.bom: list of filesystem items and metadata



Archive.pax.gz: compressed archive of files

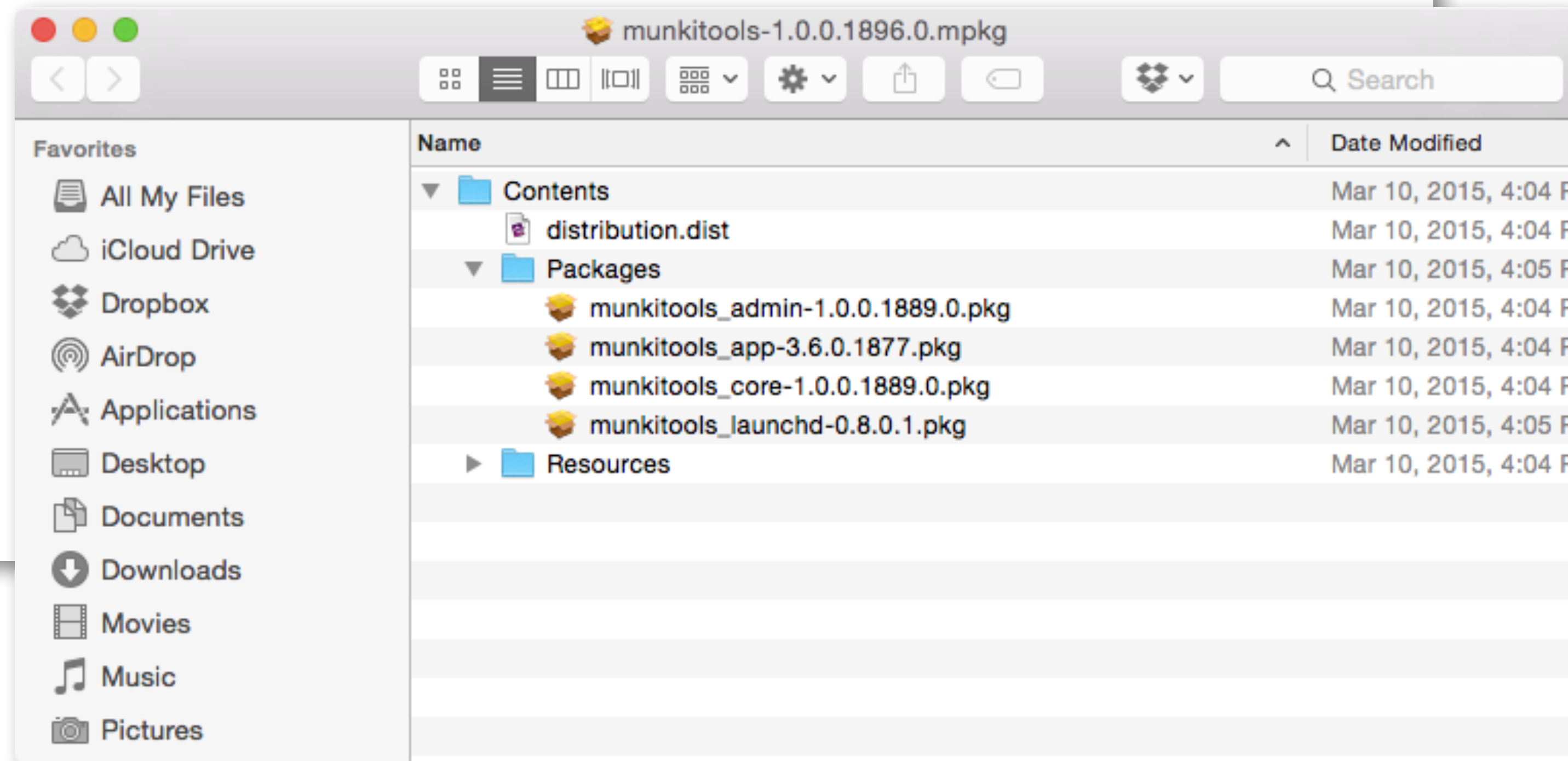
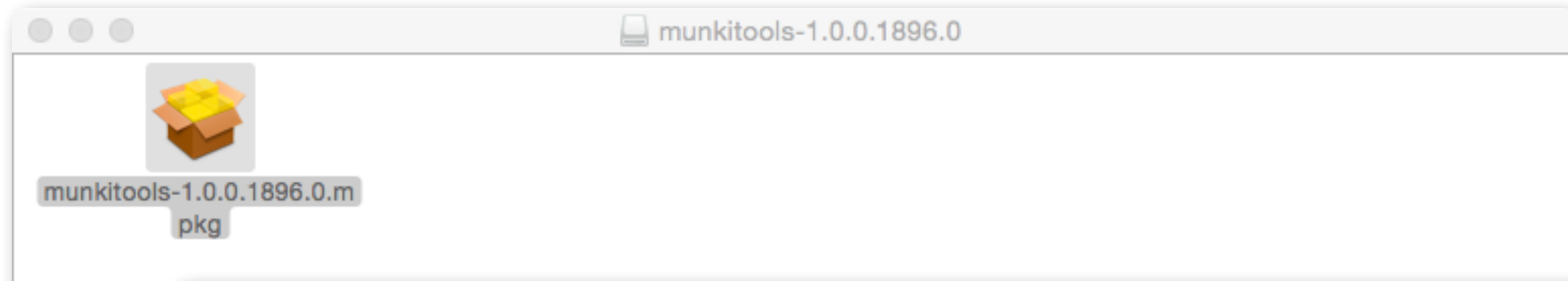


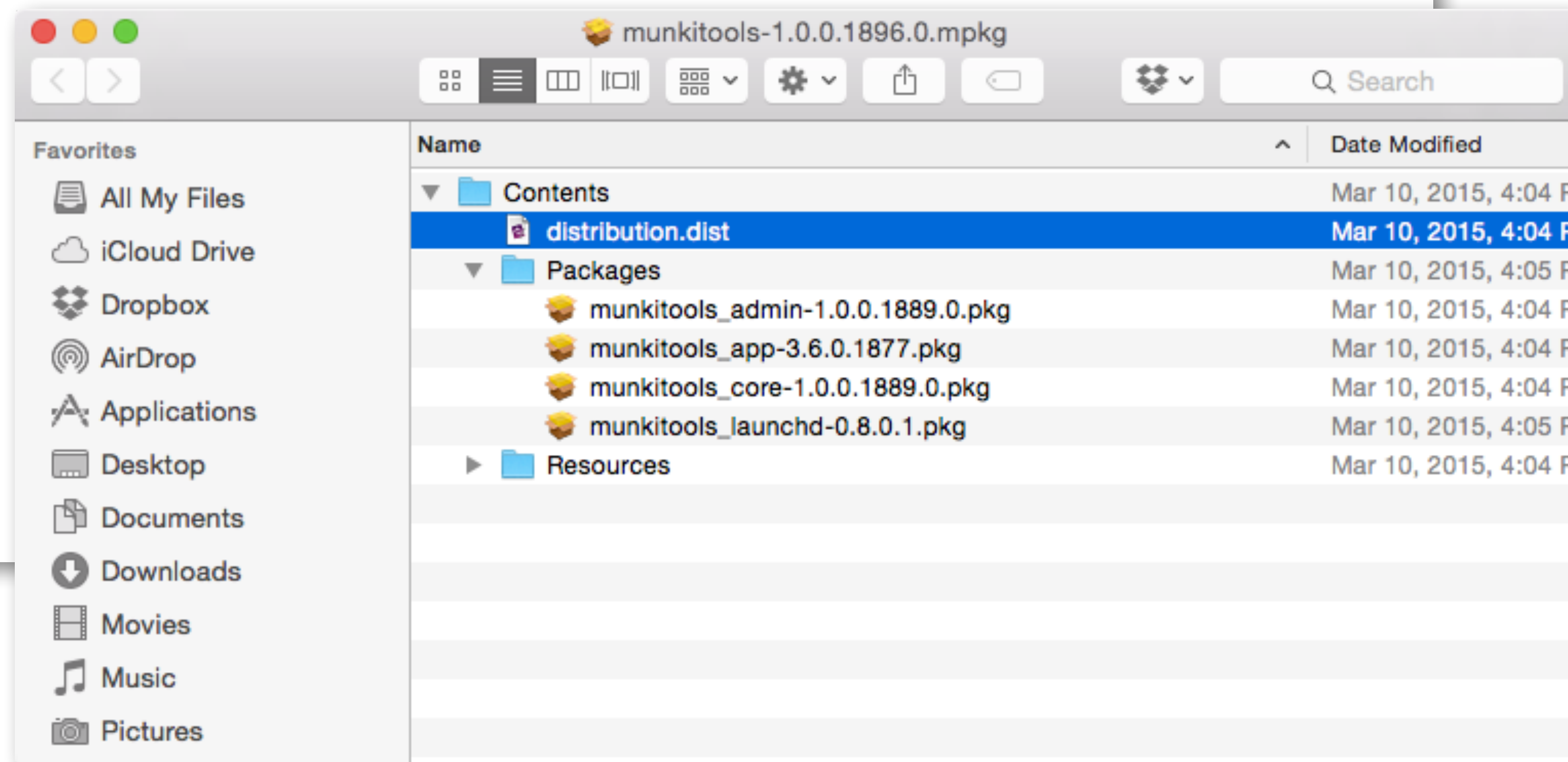
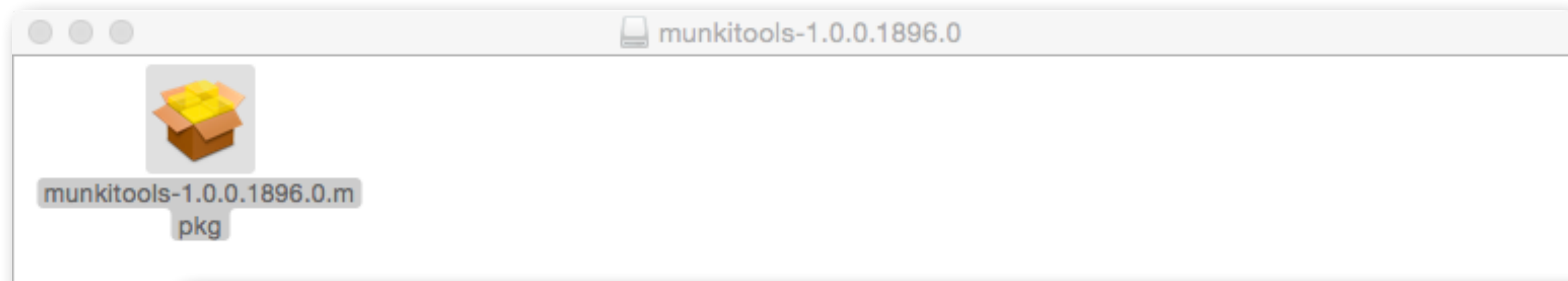
Bundle pkg scripts

- preflight
- preinstall
- preupgrade
- postupgrade
- postinstall
- postflight

Bundle metapackage
and distribution pkgs








```
<?xml version="1.0" encoding="utf-8"?>
<installer-script minSpecVersion="1.000000" authoringTool="com.apple.PackageMaker"
authoringToolVersion="3.0.4" authoringToolBuild="179">
  <title>Munki - Managed software installation for OS X</title>
  <options customize="allow" allow-external-scripts="no"/>
  <domains enable_anywhere="true"/>
  <choices-outline>
    <line choice="core"/>
    <line choice="admin"/>
    <line choice="app"/>
    <line choice="launchd"/>
  </choices-outline>
  <choice id="core" title="Munki core tools" description="Core command-line tools used by
munki.">
    <pkg-ref id="com.googlecode.munki.core"/>
  </choice>
  <choice id="admin" title="Munki admin tools" description="Command-line munki admin tools.">
    <pkg-ref id="com.googlecode.munki.admin"/>
  </choice>
  <choice id="app" title="Managed Software Update" description="Managed Software Update
application.">
    <pkg-ref id="com.googlecode.munki.app"/>
  </choice>
  <choice id="launchd" title="Munki launchd agents" description="Munki launch daemons and launch
agents.">
    <pkg-ref id="com.googlecode.munki.launchd"/>
  </choice>

  <pkg-ref id="com.googlecode.munki.core" installKBytes="588" version="1.0.0.1889.0"
auth="Root">file:./Contents/Packages/munkitools_core-1.0.0.1889.0.pkg</pkg-ref>
  <pkg-ref id="com.googlecode.munki.admin" installKBytes="132" version="1.0.0.1889.0"
```

```
</choice>
  <choice id="app" title="Managed Software Update" description="Managed Software Update
application.">
    <pkg-ref id="com.googlecode.munki.app"/>
  </choice>
  <choice id="launchd" title="Munki launchd agents" description="Munki launch daemons and launch
agents.">
    <pkg-ref id="com.googlecode.munki.launchd"/>
  </choice>

  <pkg-ref id="com.googlecode.munki.core" installKBytes="588" version="1.0.0.1889.0"
auth="Root">file:./Contents/Packages/munkitools_core-1.0.0.1889.0.pkg</pkg-ref>
  <pkg-ref id="com.googlecode.munki.admin" installKBytes="132" version="1.0.0.1889.0"
auth="Root">file:./Contents/Packages/munkitools_admin-1.0.0.1889.0.pkg</pkg-ref>
  <pkg-ref id="com.googlecode.munki.app" installKBytes="1520" version="3.6.0.1877"
auth="Root">file:./Contents/Packages/munkitools_app-3.6.0.1877.pkg</pkg-ref>
  <pkg-ref id="com.googlecode.munki.launchd" installKBytes="32" version="1.0.0.1889.0"
auth="Root" onConclusion="RequireRestart">file:./Contents/Packages/munkitools_launchd-0.8.0.1.pkg</
pkg-ref>

</installer-script>
```

```
</choice>
  <choice id="app" title="Managed Software Update" description="Managed Software Update
application.">
    <pkg-ref id="com.googlecode.munki.app"/>
  </choice>
  <choice id="launchd" title="Munki launchd agents" description="Munki launch daemons and launch
agents.">
    <pkg-ref id="com.googlecode.munki.launchd"/>
  </choice>

  <pkg-ref id="com.googlecode.munki.core" installKBytes="588" version="1.0.0.1889.0"
auth="Root">file:./Contents/Packages/munkitools_core-1.0.0.1889.0.pkg</pkg-ref>
  <pkg-ref id="com.googlecode.munki.admin" installKBytes="132" version="1.0.0.1889.0"
auth="Root">file:./Contents/Packages/munkitools_admin-1.0.0.1889.0.pkg</pkg-ref>
  <pkg-ref id="com.googlecode.munki.app" installKBytes="1520" version="3.6.0.1877"
auth="Root">file:./Contents/Packages/munkitools_app-3.6.0.1877.pkg</pkg-ref>
  <pkg-ref id="com.googlecode.munki.launchd" installKBytes="32" version="1.0.0.1889.0"
auth="Root" onConclusion="RequireRestart">file:./Contents/Packages/munkitools_launchd-0.8.0.1.pkg</
pkg-ref>

</installer-script>
```

```
</choice>
  <choice id="app" title="Managed Software Update" description="Managed Software Update
application.">
    <pkg-ref id="com.googlecode.munki.app"/>
  </choice>
  <choice id="launchd" title="Munki launchd agents" description="Munki launch daemons and launch
agents.">
    <pkg-ref id="com.googlecode.munki.launchd"/>
  </choice>

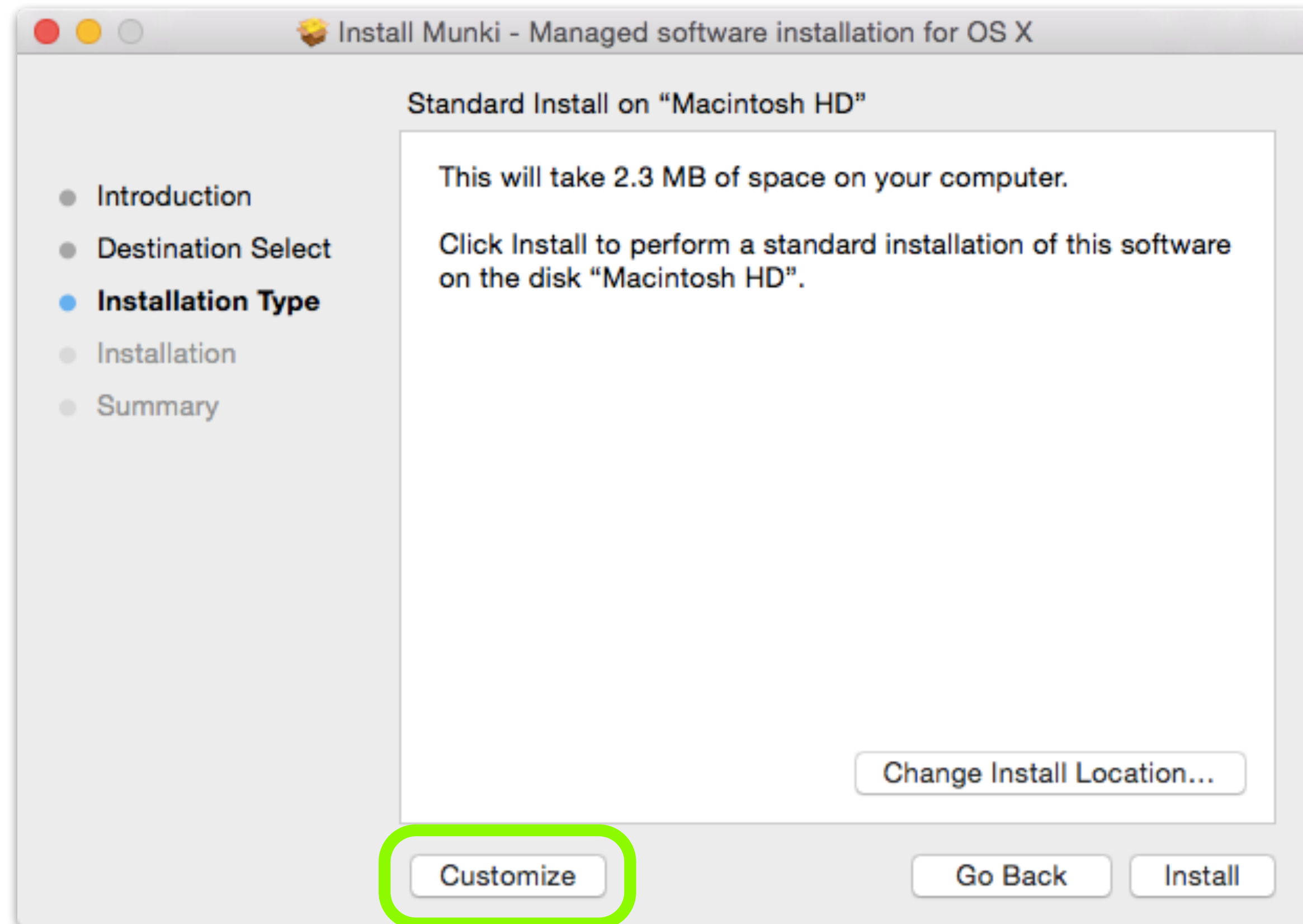
  <pkg-ref id="com.googlecode.munki.core" installKBytes="588" version="1.0.0.1889.0"
auth="Root">file:./Contents/Packages/munkitools_core-1.0.0.1889.0.pkg</pkg-ref>
  <pkg-ref id="com.googlecode.munki.admin" installKBytes="132" version="1.0.0.1889.0"
auth="Root">file:./Contents/Packages/munkitools_admin-1.0.0.1889.0.pkg</pkg-ref>
  <pkg-ref id="com.googlecode.munki.app" installKBytes="1520" version="3.6.0.1877"
auth="Root">file:./Contents/Packages/munkitools_app-3.6.0.1877.pkg</pkg-ref>
  <pkg-ref id="com.googlecode.munki.launchd" installKBytes="32" version="1.0.0.1889.0"
auth="Root" onConclusion="RequireRestart">file:./Contents/Packages/munkitools_launchd-0.8.0.1.pkg</
pkg-ref>

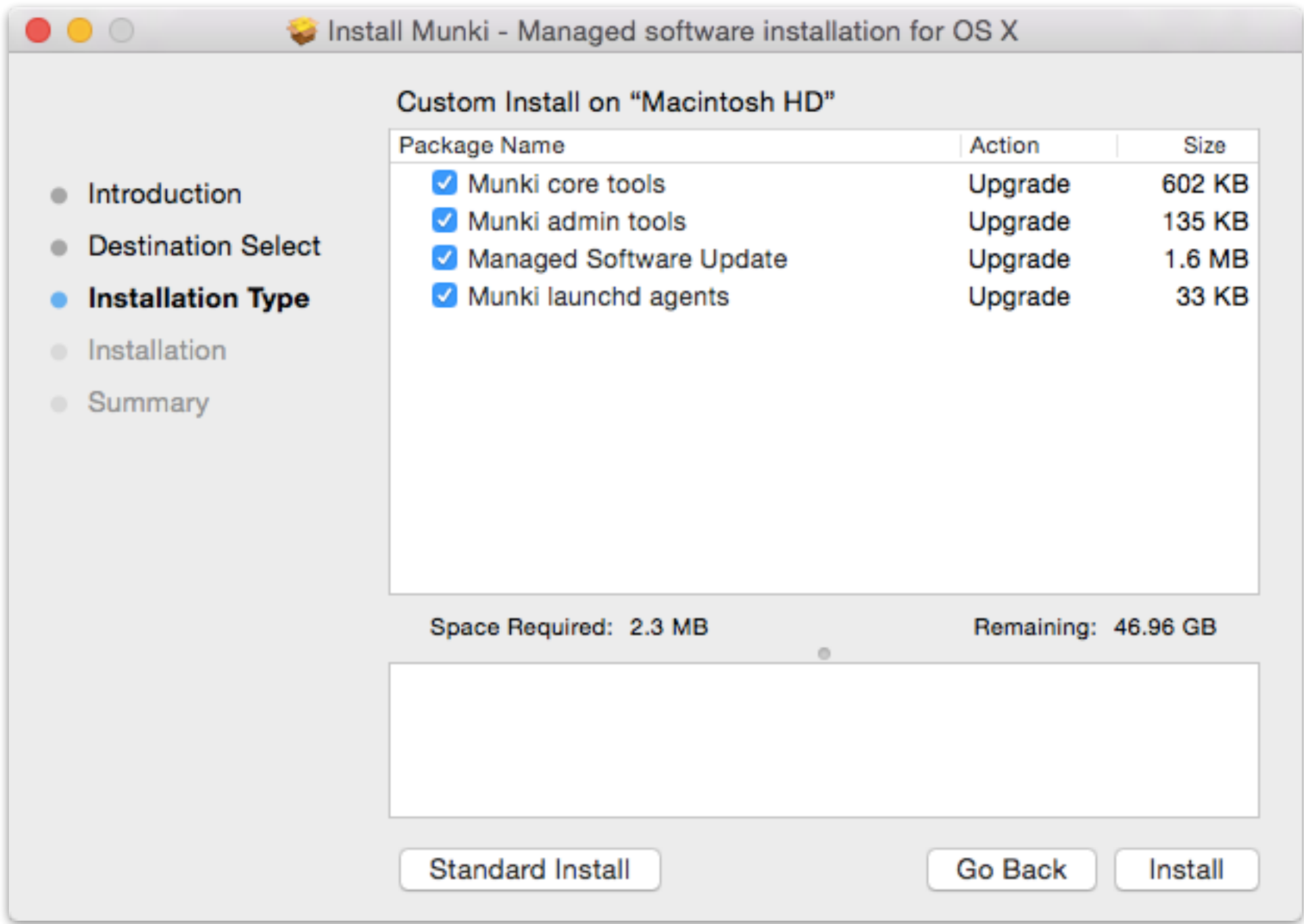
</installer-script>
```

```
</choice>
  <choice id="app" title="Managed Software Update" description="Managed Software Update
application.">
    <pkg-ref id="com.googlecode.munki.app"/>
  </choice>
  <choice id="launchd" title="Munki launchd agents" description="Munki launch daemons and launch
agents.">
    <pkg-ref id="com.googlecode.munki.launchd"/>
  </choice>

  <pkg-ref id="com.googlecode.munki.core" installKBytes="588" version="1.0.0.1889.0"
auth="Root">file:./Contents/Packages/munkitools_core-1.0.0.1889.0.pkg</pkg-ref>
  <pkg-ref id="com.googlecode.munki.admin" installKBytes="132" version="1.0.0.1889.0"
auth="Root">file:./Contents/Packages/munkitools_admin-1.0.0.1889.0.pkg</pkg-ref>
  <pkg-ref id="com.googlecode.munki.app" installKBytes="1520" version="3.6.0.1877"
auth="Root">file:./Contents/Packages/munkitools_app-3.6.0.1877.pkg</pkg-ref>
  <pkg-ref id="com.googlecode.munki.launchd" installKBytes="32" version="1.0.0.1889.0"
auth="Root" onConclusion="RequireRestart">file:./Contents/Packages/munkitools_launchd-0.8.0.1.pkg</
pkg-ref>

</installer-script>
```





Review

Examine the contents of the ne-3.0.1 package* and answer these questions:

- What is the package id?
(Hint: look for CFBundleIdentifier)
- What is the version of the package?
(Hint: look for CFBundleShortVersionString or CFBundleVersion)

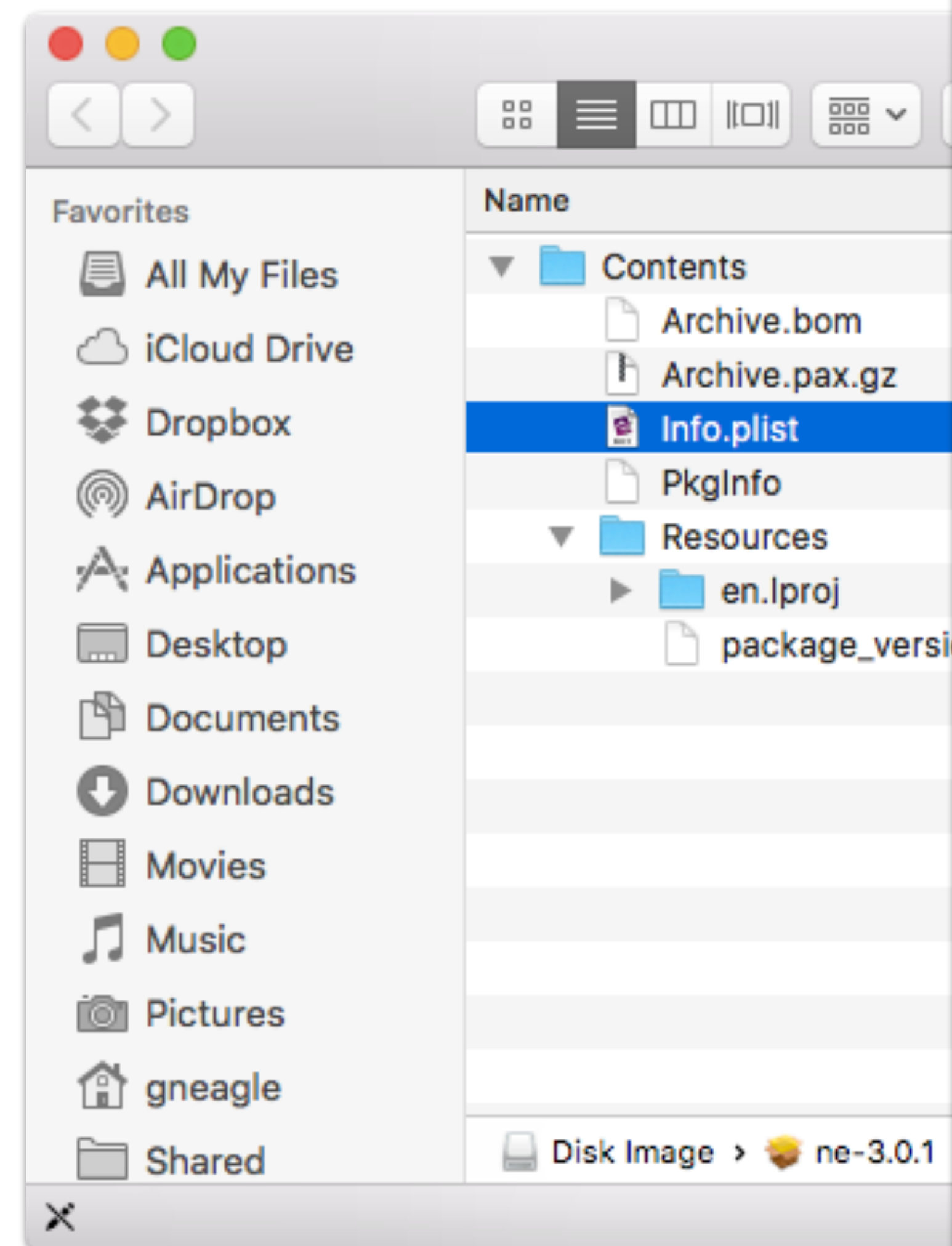
*ne-3.0.1.pkg, found on ne-3.0.1.dmg, available from <http://ne.di.unimi.it/ne-3.0.1.dmg>

Examine the contents of the ne-3.0.1 package* and answer these questions:

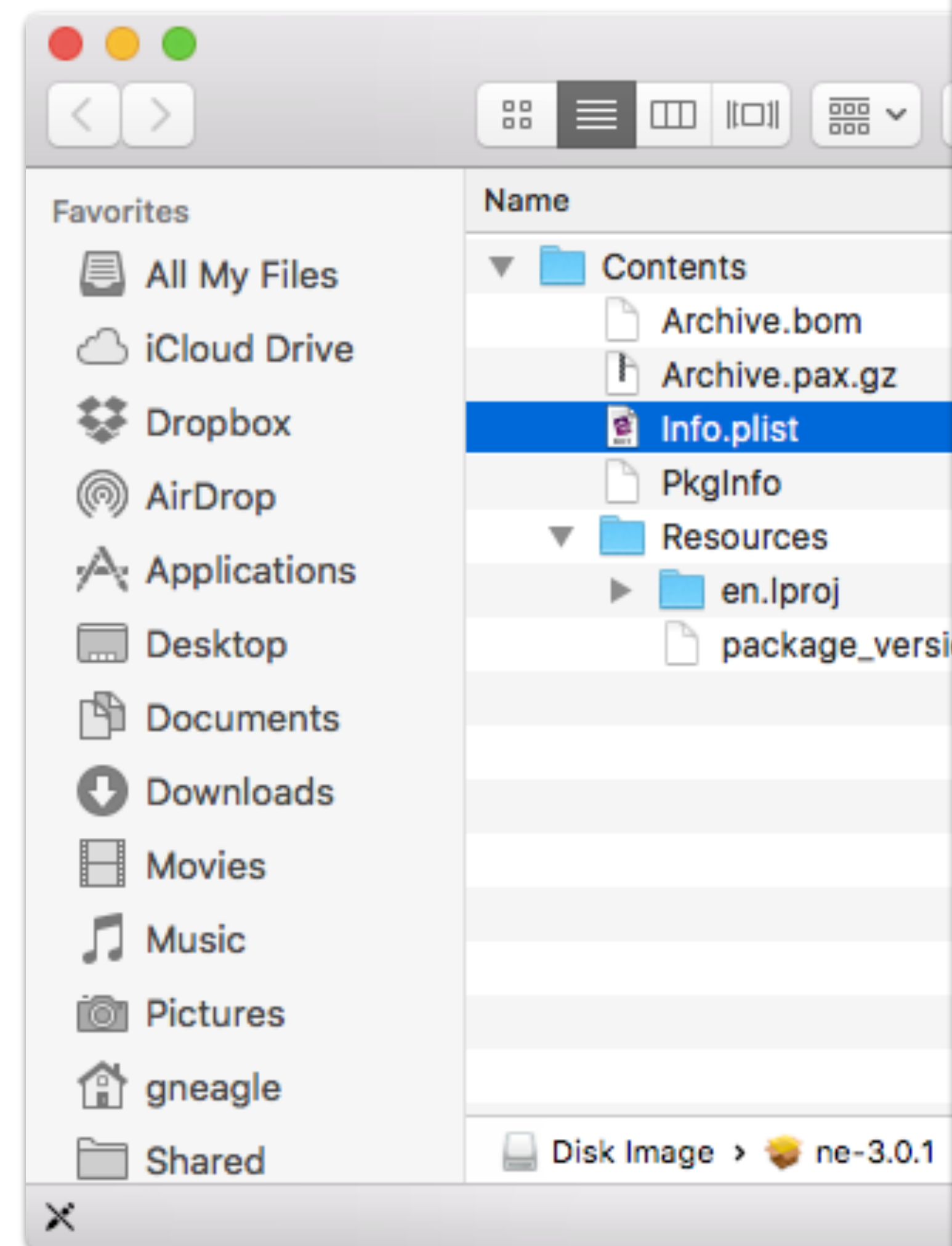
- What is the package id?
- Answer: “di.unimi.it.package.pkg”

Examine the contents of the ne-3.0.1 package* and answer these questions:

- What is the version of the package?
- Answer: 1.0



```
Info.plist Open with TextMate ⌵
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
  <key>CFBundleIdentifier</key>
  <string>di.unimi.it.package.pkg</string>
  <key>CFBundleShortVersionString</key>
  <string>1.0</string>
  <key>IFMajorVersion</key>
  <integer>1</integer>
  <key>IFMinorVersion</key>
  <integer>0</integer>
  <key>IFPkgFlagAllowBackRev</key>
  <false/>
  <key>IFPkgFlagAuthorizationAction</key>
  <string>RootAuthorization</string>
  <key>IFPkgFlagDefaultLocation</key>
  <string>/</string>
  <key>IFPkgFlagFollowLinks</key>
  <true/>
  <key>IFPkgFlagInstallFat</key>
  <false/>
  <key>IFPkgFlagInstalledSize</key>
  <integer>3636</integer>
  <key>IFPkgFlagIsRequired</key>
  <false/>
  <key>IFPkgFlagOverwritePermissions</key>
  <false/>
  <key>IFPkgFlagRelocatable</key>
  <false/>
  <key>IFPkgFlagRestartAction</key>
  <string>None</string>
  <key>IFPkgFlagRootVolumeOnly</key>
  <false/>
  <key>IFPkgFlagUpdateInstalledLanguages</key>
  <false/>
  <key>IFPkgFormatVersion</key>
  <real>0.10000000149011612</real>
</dict>
</plist>
```



```
Info.plist Open with TextMate ⏏ ⏏
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
  <key>CFBundleIdentifier</key>
  <string>di.unimi.it.package.pka</string>
  <key>CFBundleShortVersionString</key>
  <string>1.0</string>
  <key>Irmajorversion</key>
  <integer>1</integer>
  <key>IFMinorVersion</key>
  <integer>0</integer>
  <key>IFPkgFlagAllowBackRev</key>
  <false/>
  <key>IFPkgFlagAuthorizationAction</key>
  <string>RootAuthorization</string>
  <key>IFPkgFlagDefaultLocation</key>
  <string>/</string>
  <key>IFPkgFlagFollowLinks</key>
  <true/>
  <key>IFPkgFlagInstallFat</key>
  <false/>
  <key>IFPkgFlagInstalledSize</key>
  <integer>3636</integer>
  <key>IFPkgFlagIsRequired</key>
  <false/>
  <key>IFPkgFlagOverwritePermissions</key>
  <false/>
  <key>IFPkgFlagRelocatable</key>
  <false/>
  <key>IFPkgFlagRestartAction</key>
  <string>None</string>
  <key>IFPkgFlagRootVolumeOnly</key>
  <false/>
  <key>IFPkgFlagUpdateInstalledLanguages</key>
  <false/>
  <key>IFPkgFormatVersion</key>
  <real>0.10000000149011612</real>
</dict>
</plist>
```

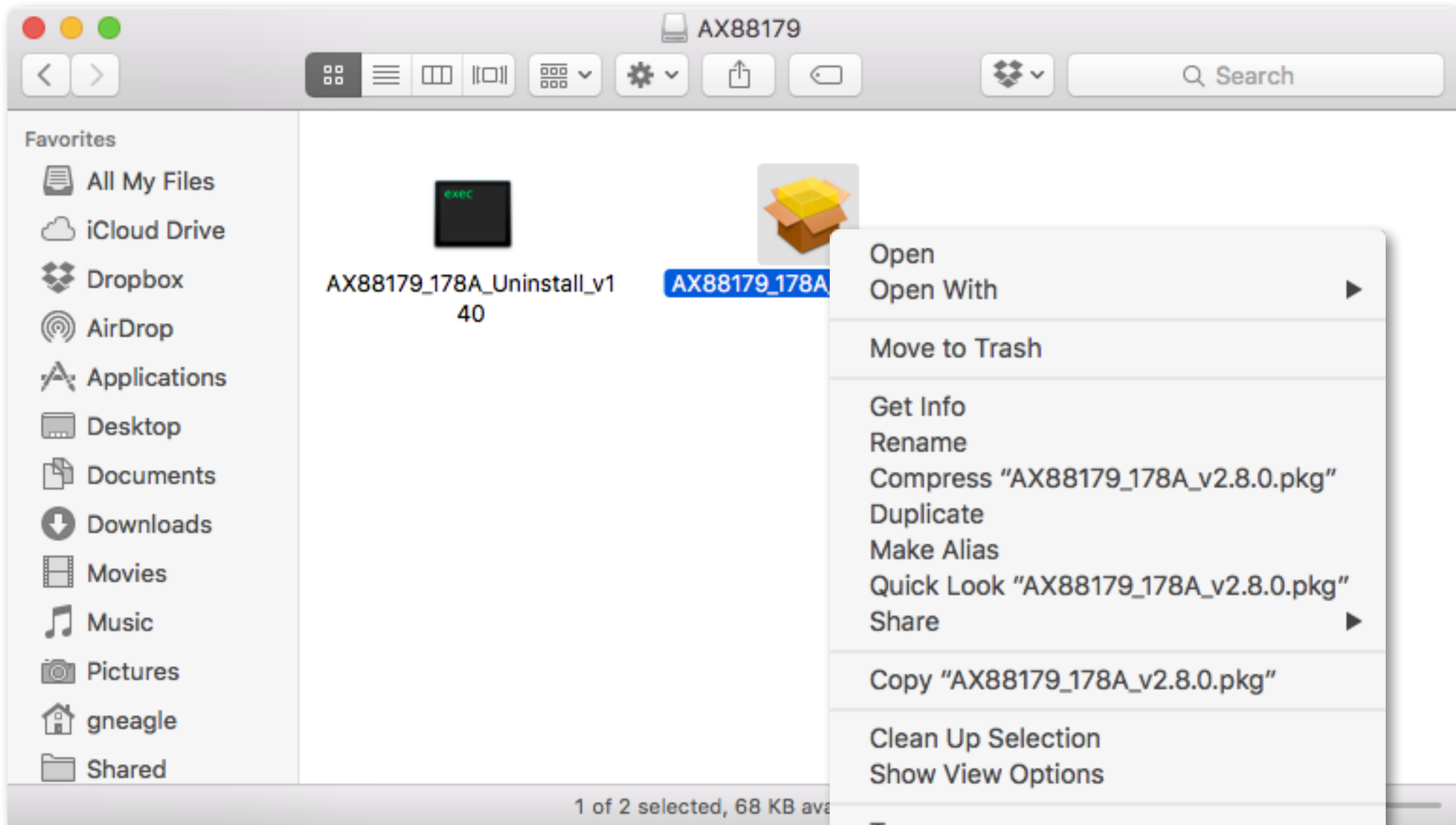
Flat packages

Flat packages

- Introduced in Mac OS X 10.5 Leopard
- "Flat" because they are a single file

Flat packages

- Component pkgs
- Distribution pkgs



AX88179



Search

Favorites

- All My Files
- iCloud Drive
- Dropbox
- AirDrop
- Applications
- Desktop
- Documents
- Downloads
- Movies
- Music
- Pictures
- gneagle
- Shared

AX88179_178A_Uninstall_v1
40

AX88179_178A

- Open
- Open With
- Move to Trash
- Get Info
- Rename
- Compress "AX88179_178A_v2.8.0.pkg"
- Duplicate
- Make Alias
- Quick Look "AX88179_178A_v2.8.0.pkg"
- Share
- Copy "AX88179_178A_v2.8.0.pkg"
- Clean Up Selection
- Show View Options
- Tags...

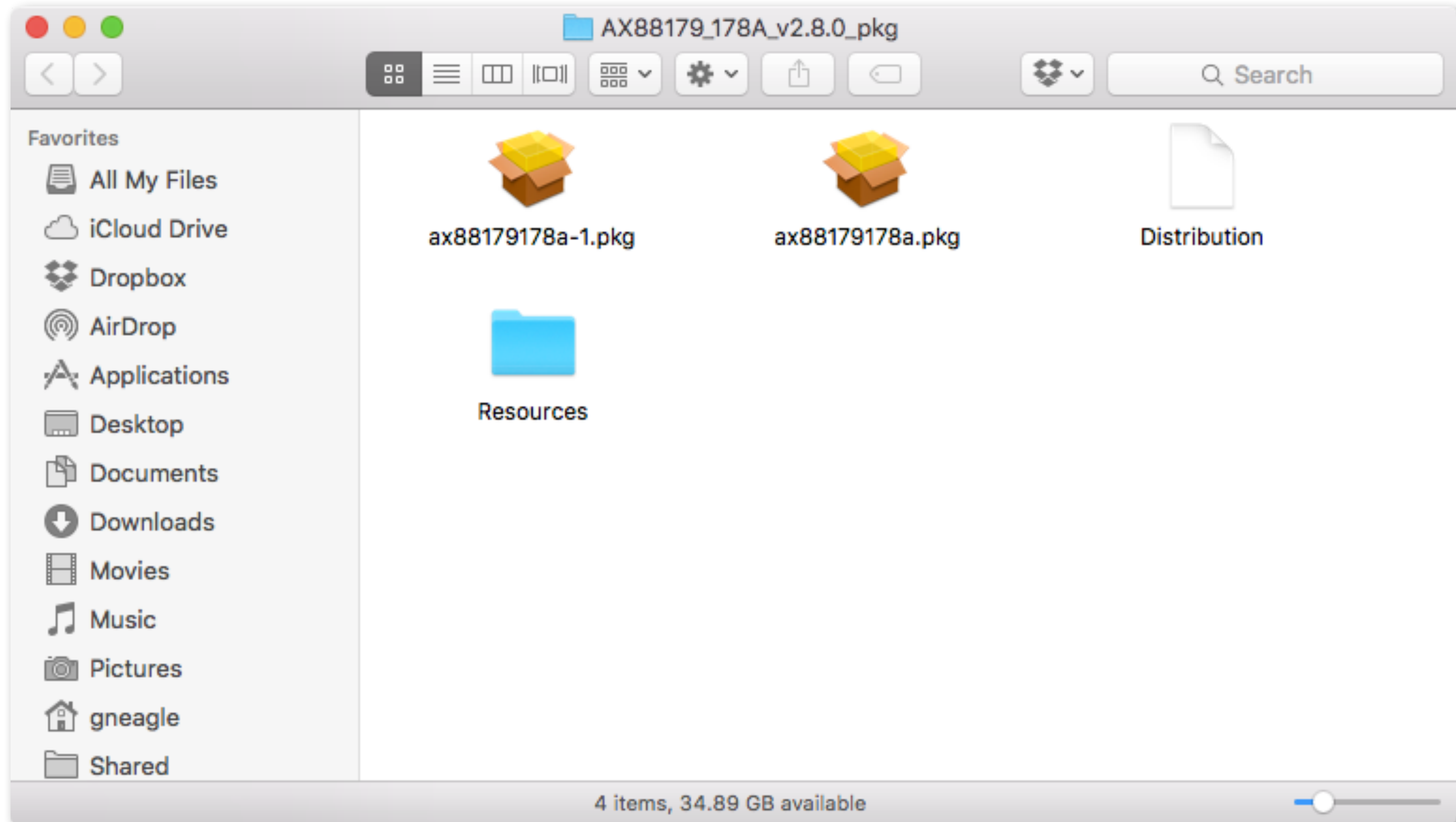
1 of 2 selected, 68 KB available



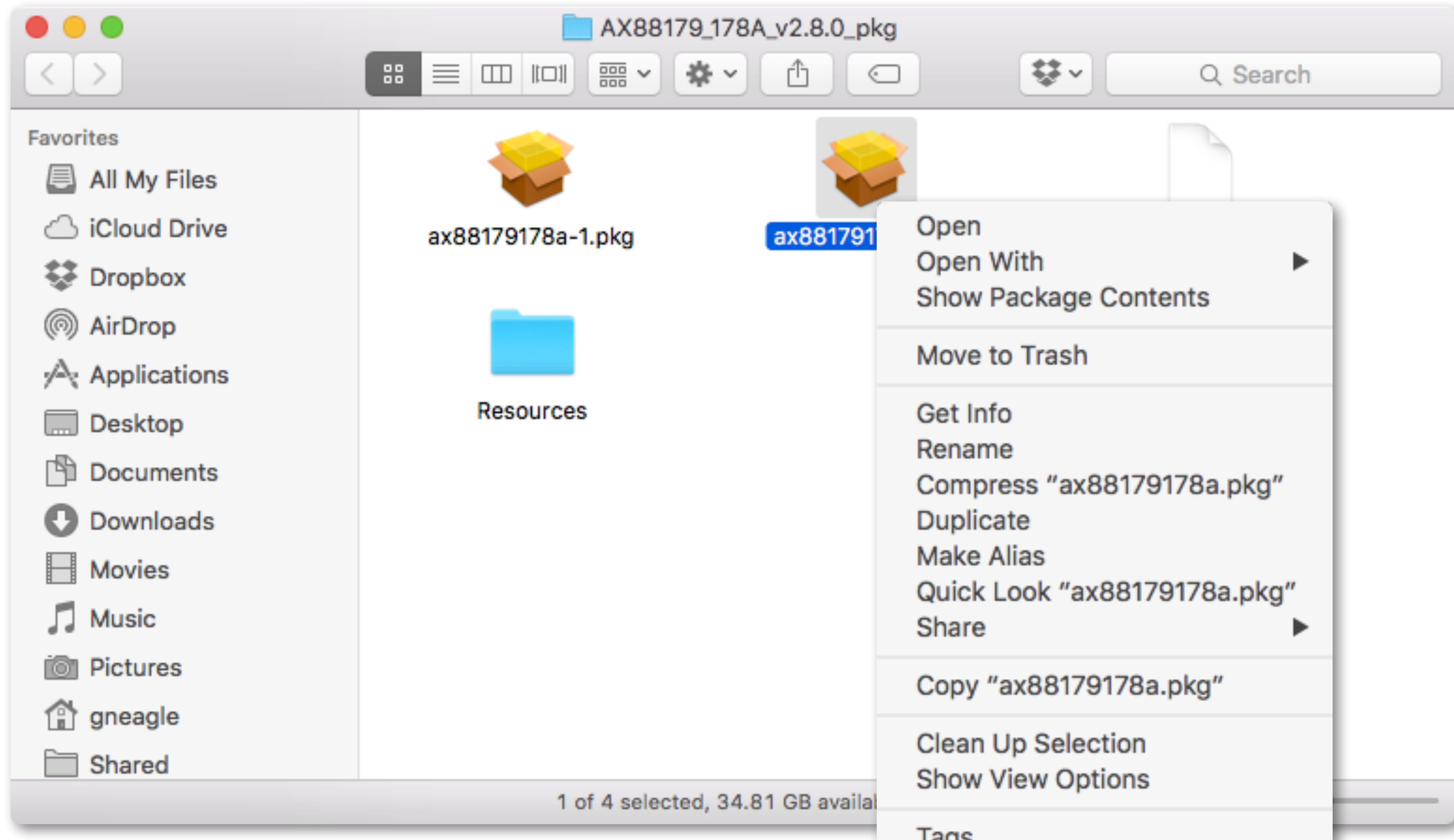
pkgutil --expand

```
% pkgutil --expand /Volumes/AX88179/AX88179_178A_v2.8.0.pkg /tmp/AX88179_178A_v2.8.0_pkg  
% open /tmp/AX88179_178A_v2.8.0_pkg
```

Flat distribution package



Flat distribution package



ax88179178a.pkg

Name	Date Modified
Bom	Apr 19, 2016, 12:01 AM
PackageInfo	Apr 19, 2016, 12:01 AM
Payload	Apr 19, 2016, 12:01 AM
Scripts	Today, 11:03 AM
postinstall	Sep 29, 2014, 4:15 AM
preinstall	Feb 18, 2014, 11:14 PM

6 items, 34.81 GB available

Flat

Silverlight.pkg

Name	Date Modified
Contents	Apr 20, 2015, 12:07 PM
_CodeSignature	Apr 20, 2015, 12:07 PM
Archive.bom	Apr 16, 2015, 3:03 PM
Archive.pax.gz	Apr 16, 2015, 3:03 PM
Info.plist	Apr 16, 2015, 3:03 PM
PkgInfo	Apr 16, 2015, 3:03 PM
Resources	Apr 20, 2015, 12:07 PM
background	Nov 7, 2011, 12:42 PM
da.lproj	Apr 16, 2015, 3:03 PM
de.lproj	Apr 16, 2015, 3:03 PM
en.lproj	Apr 16, 2015, 3:03 PM
es.lproj	Apr 16, 2015, 3:03 PM
fi.lproj	Apr 16, 2015, 3:03 PM
fr.lproj	Apr 16, 2015, 3:03 PM
InstallationCheck	Apr 16, 2015, 3:03 PM
it.lproj	Apr 16, 2015, 3:03 PM
ja.lproj	Apr 16, 2015, 3:03 PM
ko.lproj	Apr 16, 2015, 3:03 PM
nl.lproj	Apr 16, 2015, 3:03 PM
no.lproj	Apr 16, 2015, 3:03 PM
package_version	Apr 16, 2015, 3:03 PM
pkmkshellutility	Apr 20, 2015, 12:07 PM
pl.lproj	Apr 16, 2015, 3:03 PM
PlayReadyGetIBXVersionTool	Apr 20, 2015, 12:07 PM
postflight	Nov 7, 2011, 12:42 PM
preflight	Nov 7, 2011, 12:42 PM
pt-BR.lproj	Apr 16, 2015, 3:03 PM
pt-PT.lproj	Apr 16, 2015, 3:03 PM
ru.lproj	Apr 16, 2015, 3:03 PM
rundylib	Apr 20, 2015, 12:07 PM
sv.lproj	Apr 16, 2015, 3:03 PM
zh-Hans.lproj	Apr 16, 2015, 3:03 PM
zh-Hant.lproj	Apr 16, 2015, 3:03 PM

Bundle

- Red
- Orange
- Yellow
- Green
- Blue
- Purple

```
% cat /tmp/AX88179_178A_v2.8.0_pkg/ax88179178a.pkg/PackageInfo
```

```
<pkg-info format-version="2" identifier="com.asix.pkg.ax88179-178a-10.9"
version="2.8.0" install-location="/Library/Extensions" auth="root" postinstall-
action="restart">
  <payload installKBytes="104" numberOfFiles="11"/>
  <scripts>
    <preinstall file="./preinstall"/>
    <postinstall file="./postinstall"/>
  </scripts>
  <bundle id="com.asix.driver.ax88179-178a"
CFBundleIdentifier="com.asix.driver.ax88179-178a" path="./AX88179_178A.kext"
CFBundleVersion="1.8.0"/>
  <bundle-version>
    <bundle id="com.asix.driver.ax88179-178a"/>
  </bundle-version>
</pkg-info>
```

```
% cat /tmp/AX88179_178A_v2.8.0_pkg/ax88179178a.pkg/PackageInfo
```

```
<pkg-info format-version="2" identifier="com.asix.pkg.ax88179-178a-10.9"  
version="2.8.0" install-location="/Library/Extensions" auth="root" postinstall-  
action="restart">  
  <payload installKBytes="104" numberOfFiles="11"/>  
  <scripts>  
    <preinstall file="./preinstall"/>  
    <postinstall file="./postinstall"/>  
  </scripts>  
  <bundle id="com.asix.driver.ax88179-178a"  
CFBundleIdentifier="com.asix.driver.ax88179-178a" path="./AX88179_178A.kext"  
CFBundleVersion="1.8.0"/>  
  <bundle-version>  
    <bundle id="com.asix.driver.ax88179-178a"/>  
  </bundle-version>  
</pkg-info>
```

```
% cat /tmp/AX88179_178A_v2.8.0_pkg/ax88179178a.pkg/PackageInfo
```

```
<pkg-info format-version="2" identifier="com.asix.pkg.ax88179-178a-10.9"  
version="2.8.0" install-location="/Library/Extensions" auth="root" postinstall-  
action="restart">  
  <payload installKBytes="104" numberOfFiles="11"/>  
  <scripts>  
    <preinstall file="./preinstall"/>  
    <postinstall file="./postinstall"/>  
  </scripts>  
  <bundle id="com.asix.driver.ax88179-178a"  
CFBundleIdentifier="com.asix.driver.ax88179-178a" path="./AX88179_178A.kext"  
CFBundleVersion="1.8.0"/>  
  <bundle-version>  
    <bundle id="com.asix.driver.ax88179-178a"/>  
  </bundle-version>  
</pkg-info>
```



```
% cat /tmp/AX88179_178A_v2.8.0_pkg/ax88179178a.pkg/PackageInfo
```

```
<pkg-info format-version="2" identifier="com.asix.pkg.ax88179-178a-10.9"  
version="2.8.0" install-location="/Library/Extensions" auth="root" postinstall-  
action="restart">  
  <payload installKBytes="104" numberOfFiles="11"/>  
  <scripts>  
    <preinstall file="./preinstall"/>  
    <postinstall file="./postinstall"/>  
  </scripts>  
  <bundle id="com.asix.driver.ax88179-178a"  
CFBundleIdentifier="com.asix.driver.ax88179-178a" path="./AX88179_178A.kext"  
CFBundleVersion="1.8.0"/>  
  <bundle-version>  
    <bundle id="com.asix.driver.ax88179-178a"/>  
  </bundle-version>  
</pkg-info>
```

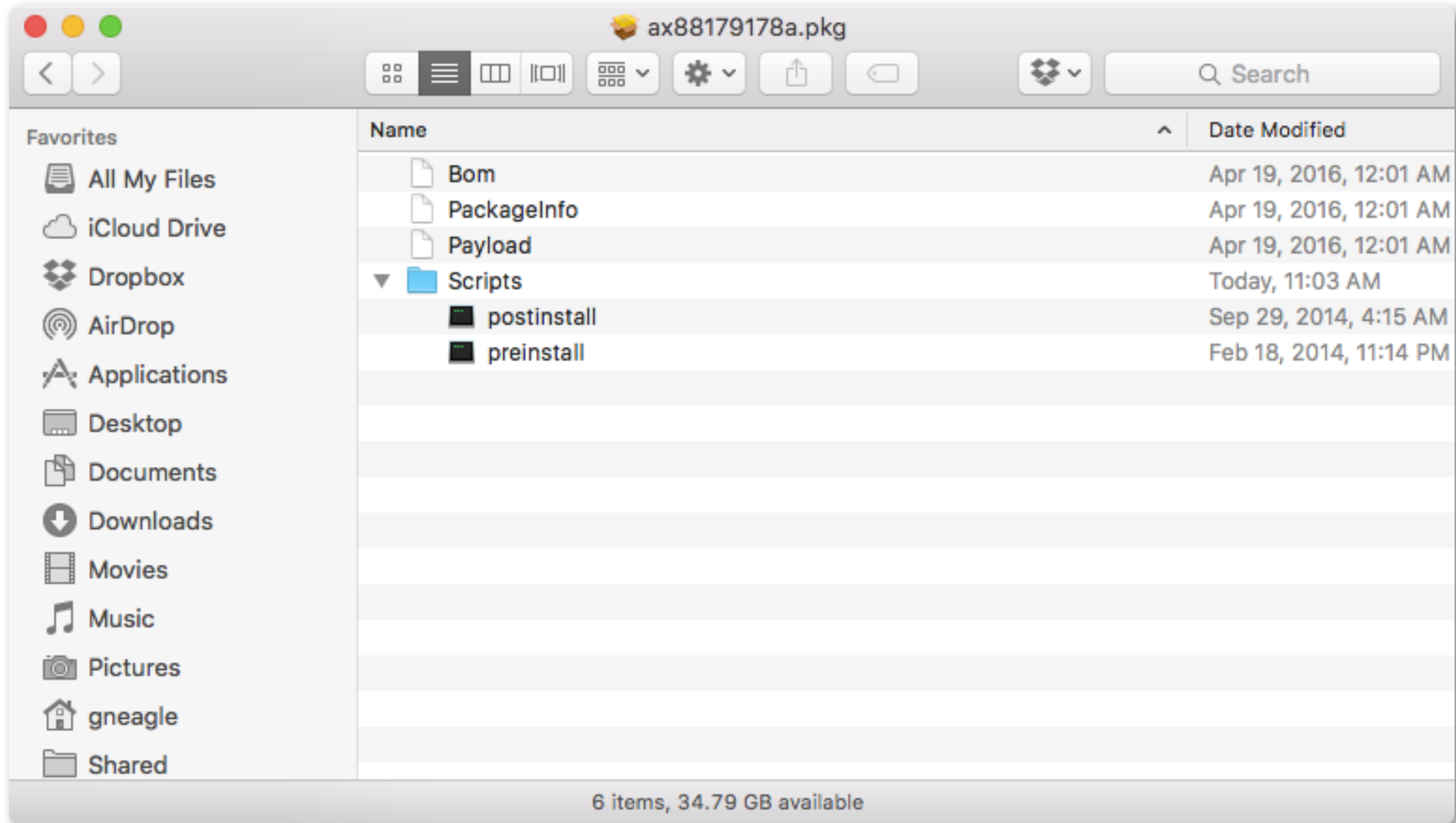
```
% cat /tmp/AX88179_178A_v2.8.0_pkg/ax88179178a.pkg/PackageInfo
```

```
<pkg-info format-version="2" identifier="com.asix.pkg.ax88179-178a-10.9"  
version="2.8.0" install-location="/Library/Extensions" auth="root" postinstall-  
action="restart">  
  <payload installKBytes="104" numberOfFiles="11"/>  
  <scripts>  
    <preinstall file="./preinstall"/>  
    <postinstall file="./postinstall"/>  
  </scripts>  
  <bundle id="com.asix.driver.ax88179-178a"  
CFBundleIdentifier="com.asix.driver.ax88179-178a" path="./AX88179_178A.kext"  
CFBundleVersion="1.8.0"/>  
  <bundle-version>  
    <bundle id="com.asix.driver.ax88179-178a"/>  
  </bundle-version>  
</pkg-info>
```

```
% cat /tmp/AX88179_178A_v2.8.0_pkg/ax88179178a.pkg/PackageInfo
```

```
<pkg-info format-version="2" identifier="com.asix.pkg.ax88179-178a-10.9"  
version="2.8.0" install-location="/Library/Extensions" auth="root" postinstall-  
action="restart">  
  <payload installKBytes="104" numberOfFiles="11"/>  
  <scripts>  
    <preinstall file="./preinstall"/>  
    <postinstall file="./postinstall"/>  
  </scripts>  
  <bundle id="com.asix.driver.ax88179-178a"  
CFBundleIdentifier="com.asix.driver.ax88179-178a" path="./AX88179_178A.kext"  
CFBundleVersion="1.8.0"/>  
  <bundle-version>  
    <bundle id="com.asix.driver.ax88179-178a"/>  
  </bundle-version>  
</pkg-info>
```

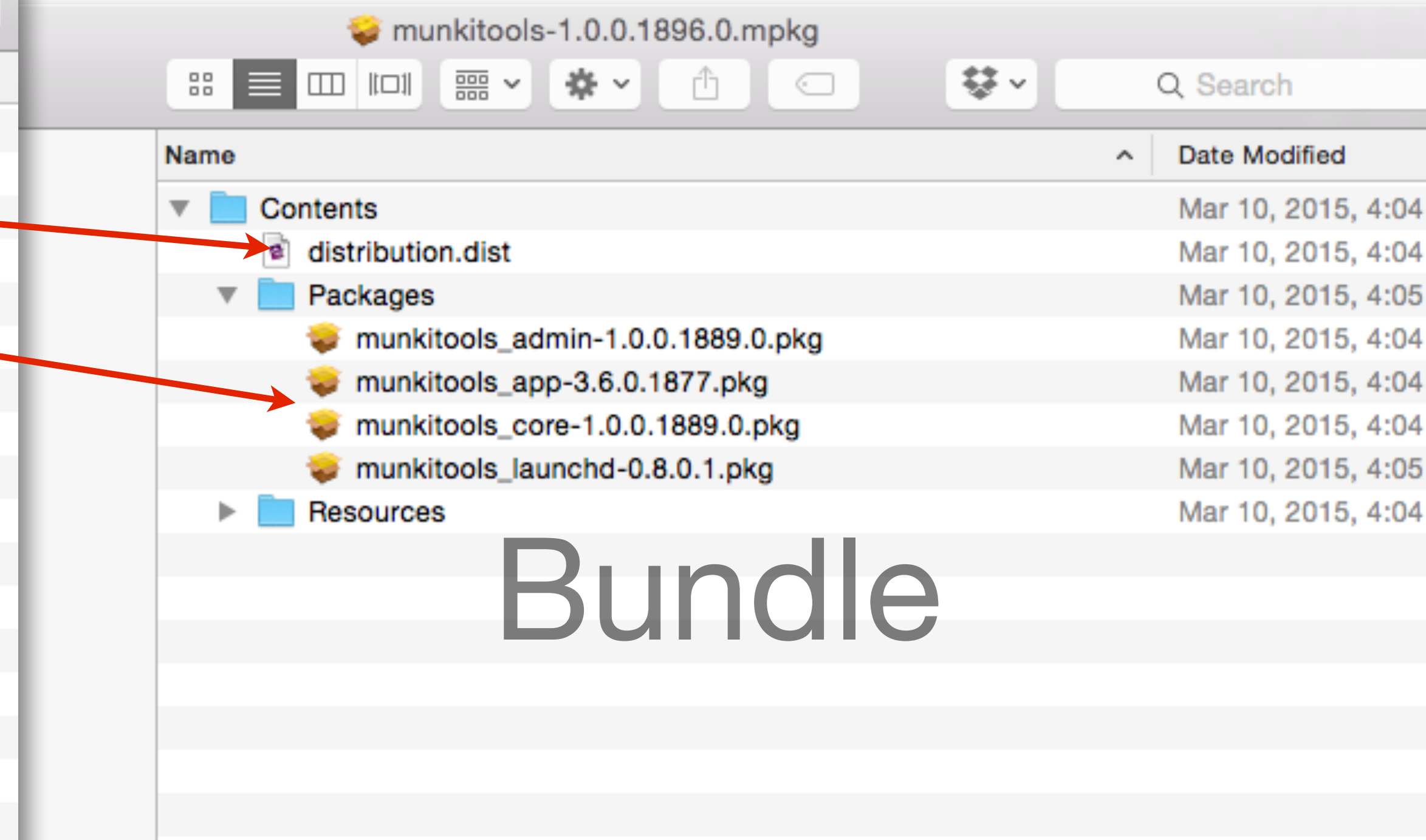
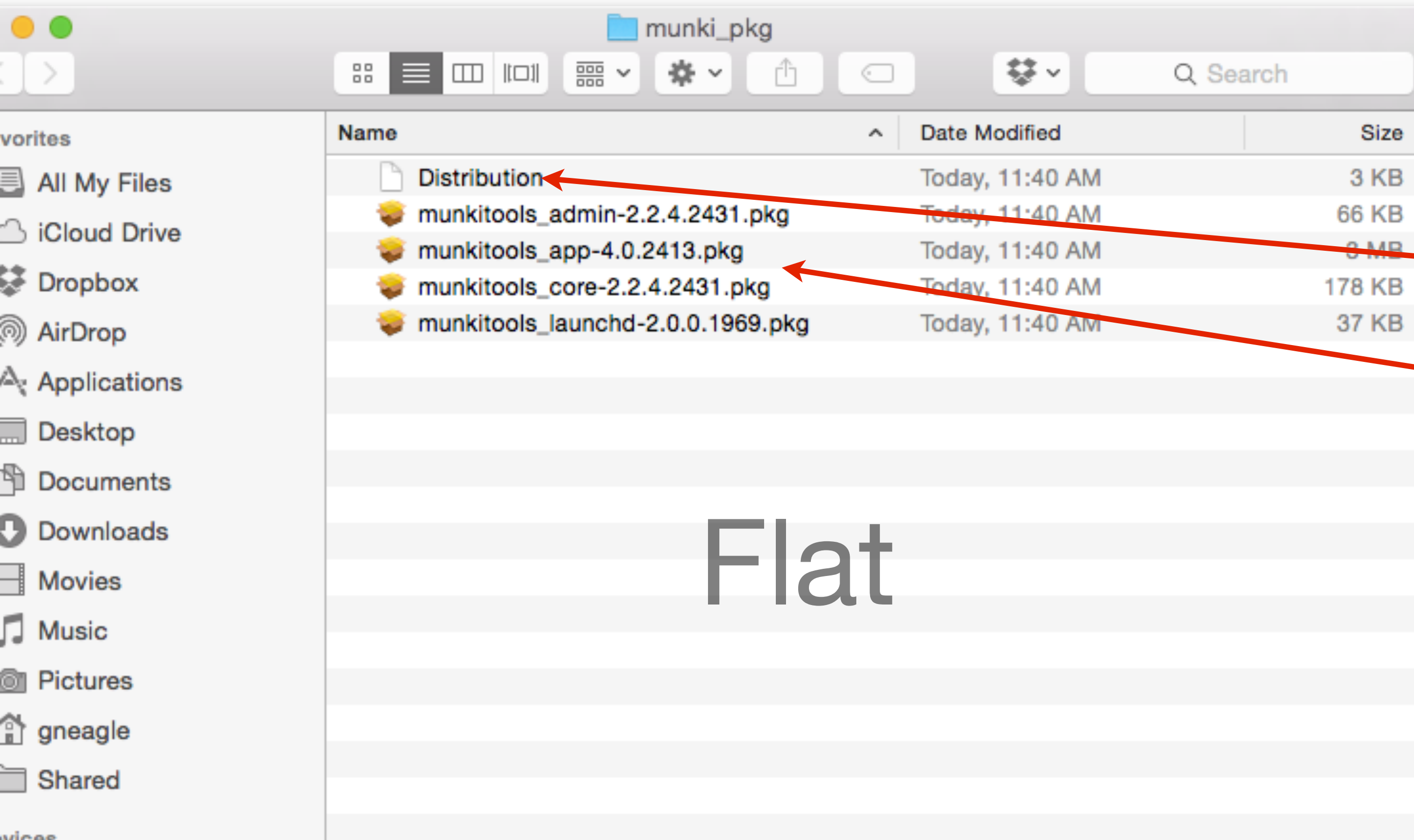
Flat component package



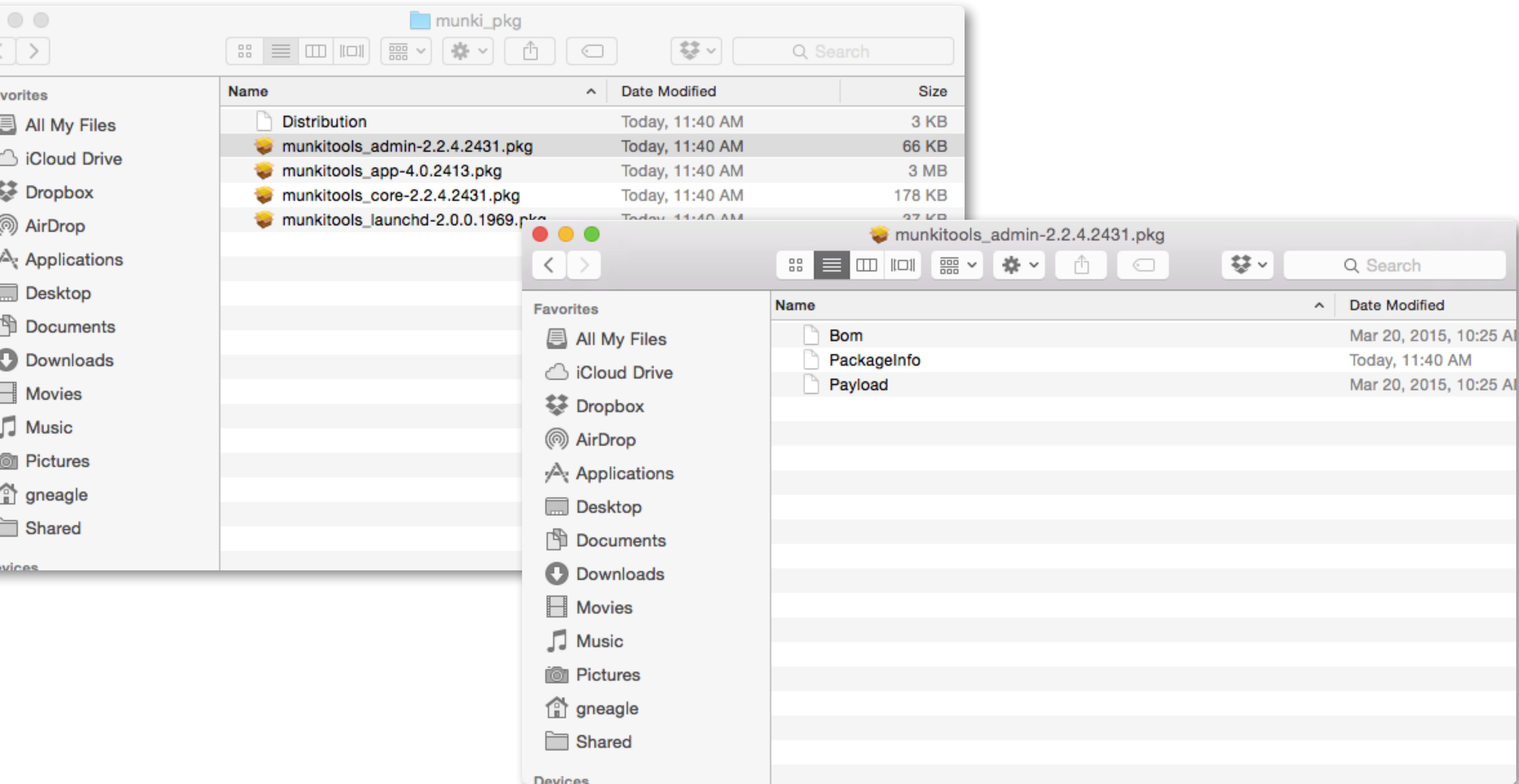
Flat pkg scripts

- ~~• preflight~~
- preinstall
- ~~• preupgrade~~
- ~~• postupgrade~~
- postinstall
- ~~• postflight~~

Distribution packages

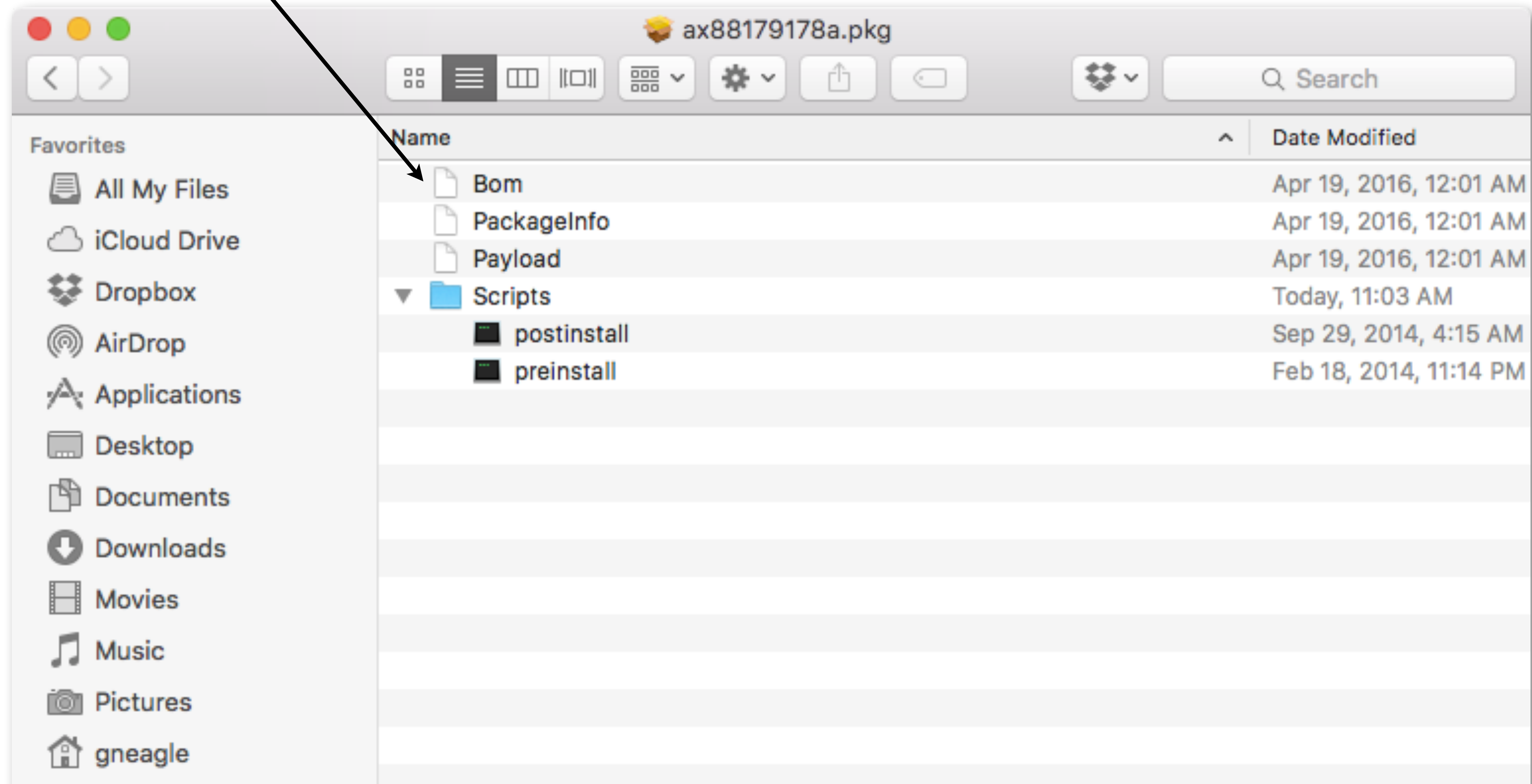


Distribution packages



Examining a pkg archive

Bom/Archive.bom:
list of filesystem items and metadata



Isbom

The archive contains information about **location**, **kind**, **owner**, **group**, and **mode** of each file and directory.

```
% lsbom /tmp/AX88179_178A_v2.8.0_pkg/ax88179178a.pkg/Bom
```

```
. 40700 501/0
```

```
./AX88179_178A.kext 40755 501/20
```

```
./AX88179_178A.kext/Contents 40755 501/20
```

```
./AX88179_178A.kext/Contents/Info.plist 100644 501/20 4882 2338734375
```

```
./AX88179_178A.kext/Contents/MacOS 40755 501/20
```

```
./AX88179_178A.kext/Contents/MacOS/AX88179_178A 100755 501/20 88528 1035438097
```

```
./AX88179_178A.kext/Contents/Resources 40755 501/20
```

```
./AX88179_178A.kext/Contents/Resources/en.lproj 40755 501/20
```

```
./AX88179_178A.kext/Contents/Resources/en.lproj/InfoPlist.strings 100644 501/20 152 221771084
```

```
./AX88179_178A.kext/Contents/_CodeSignature 40755 501/20
```

```
./AX88179_178A.kext/Contents/_CodeSignature/CodeResources 100644 501/20 2370 1632215908
```

```
% lsbom /tmp/AX88179_178A_v2.8.0_pkg/ax88179178a.pkg/Bom
```

```
. 40700 501/0  
./AX88179_178A.kext 40755 501/20  
./AX88179_178A.kext/Contents 40755 501/20  
./AX88179_178A.kext/Contents/Info.plist 100644 501/20 4882 2338734375  
./AX88179_178A.kext/Contents/MacOS 40755 501/20  
./AX88179_178A.kext/Contents/MacOS/AX88179_178A 100755 501/20 88528 1035438097  
./AX88179_178A.kext/Contents/Resources 40755 501/20  
./AX88179_178A.kext/Contents/Resources/en.lproj 40755 501/20  
./AX88179_178A.kext/Contents/Resources/en.lproj/InfoPlist.strings 100644 501/20 152 221771084  
./AX88179_178A.kext/Contents/_CodeSignature 40755 501/20  
./AX88179_178A.kext/Contents/_CodeSignature/CodeResources 100644 501/20 2370 1632215908
```

```
% lsbom /tmp/AX88179_178A_v2.8.0_pkg/ax88179178a.pkg/Bom
```

```
. 40700 501/0  
./AX88179_178A.kext 40755 501/20  
./AX88179_178A.kext/Contents 40755 501/20  
./AX88179_178A.kext/Contents/Info.plist 100644 501/20 4882 2338734375  
./AX88179_178A.kext/Contents/MacOS 40755 501/20  
./AX88179_178A.kext/Contents/MacOS/AX88179_178A 100755 501/20 88528 1035438097  
./AX88179_178A.kext/Contents/Resources 40755 501/20  
./AX88179_178A.kext/Contents/Resources/en.lproj 40755 501/20  
./AX88179_178A.kext/Contents/Resources/en.lproj/InfoPlist.strings 100644 501/20 152 221771084  
./AX88179_178A.kext/Contents/_CodeSignature 40755 501/20  
./AX88179_178A.kext/Contents/_CodeSignature/CodeResources 100644 501/20 2370 1632215908
```

```
% lsbom /tmp/AX88179_178A_v2.8.0_pkg/ax88179178a.pkg/Bom
```

```
. 40700 501/0  
./AX88179_178A.kext 40755 501/20  
./AX88179_178A.kext/Contents 40755 501/20  
./AX88179_178A.kext/Contents/Info.plist 100644 501/20 4882 2338734375  
./AX88179_178A.kext/Contents/MacOS 40755 501/20  
./AX88179_178A.kext/Contents/MacOS/AX88179_178A 100755 501/20 88528 1035438097  
./AX88179_178A.kext/Contents/Resources 40755 501/20  
./AX88179_178A.kext/Contents/Resources/en.lproj 40755 501/20  
./AX88179_178A.kext/Contents/Resources/en.lproj/InfoPlist.strings 100644 501/20 152 221771084  
./AX88179_178A.kext/Contents/_CodeSignature 40755 501/20  
./AX88179_178A.kext/Contents/_CodeSignature/CodeResources 100644 501/20 2370 1632215908
```

```
% lsbom /tmp/AX88179_178A_v2.8.0_pkg/ax88179178a.pkg/Bom
```

```
. 40700 501/0  
./AX88179_178A.kext 40755 501/20  
./AX88179_178A.kext/Contents 40755 501/20  
./AX88179_178A.kext/Contents/Info.plist 100644 501/20 4882 2338734375  
./AX88179_178A.kext/Contents/MacOS 40755 501/20  
./AX88179_178A.kext/Contents/MacOS/AX88179_178A 100755 501/20 88528 1035438097  
./AX88179_178A.kext/Contents/Resources 40755 501/20  
./AX88179_178A.kext/Contents/Resources/en.lproj 40755 501/20  
./AX88179_178A.kext/Contents/Resources/en.lproj/InfoPlist.strings 100644 501/20 152 221771084  
./AX88179_178A.kext/Contents/_CodeSignature 40755 501/20  
./AX88179_178A.kext/Contents/_CodeSignature/CodeResources 100644 501/20 2370 1632215908
```



```
% lsbom /tmp/AX88179_178A_v2.8.0_pkg/ax88179178a.pkg/Bom
```

```
. 40700 501/0
```

```
./AX88179_178A.kext 40755 501/20
```

```
./AX88179_178A.kext/Contents 40755 501/20
```

```
./AX88179_178A.kext/Contents/Info.plist 100644 501/20 4882 2338734375
```

```
./AX88179_178A.kext/Contents/MacOS 40755 501/20
```

```
./AX88179_178A.kext/Contents/MacOS/AX88179_178A 100755 501/20 88528 1035438097
```

```
./AX88179_178A.kext/Contents/Resources 40755 501/20
```

```
./AX88179_178A.kext/Contents/Resources/en.lproj 40755 501/20
```

```
./AX88179_178A.kext/Contents/Resources/en.lproj/InfoPlist.strings 100644 501/20 152 221771084
```

```
./AX88179_178A.kext/Contents/_CodeSignature 40755 501/20
```

```
./AX88179_178A.kext/Contents/_CodeSignature/CodeResources 100644 501/20 2370 1632215908
```

```
% lsbom /tmp/AX88179_178A_v2.8.0_pkg/ax88179178a.pkg/Bom
```

```
. 40700 501/0  
./AX88179_178A.kext 40755 501/20  
./AX88179_178A.kext/Contents 40755 501/20  
./AX88179_178A.kext/Contents/Info.plist 100644 501/20 4882 2338734375  
./AX88179_178A.kext/Contents/MacOS 40755 501/20  
./AX88179_178A.kext/Contents/MacOS/AX88179_178A 100755 501/20 88528 1035438097  
./AX88179_178A.kext/Contents/Resources 40755 501/20  
./AX88179_178A.kext/Contents/Resources/en.lproj 40755 501/20  
./AX88179_178A.kext/Contents/Resources/en.lproj/InfoPlist.strings 100644 501/20 152 221771084  
./AX88179_178A.kext/Contents/_CodeSignature 40755 501/20  
./AX88179_178A.kext/Contents/_CodeSignature/CodeResources 100644 501/20 2370 1632215908
```

```
% cat /tmp/AX88179_178A_v2.8.0_pkg/ax88179178a.pkg/PackageInfo
```

```
<pkg-info format-version="2" identifier="com.asix.pkg.ax88179-178a-10.9"  
version="2.8.0" install-location="/Library/Extensions" auth="root" postinstall-  
action="restart">  
  <payload installKBytes="104" numberOfFiles="11"/>  
  <scripts>  
    <preinstall file="./preinstall"/>  
    <postinstall file="./postinstall"/>  
  </scripts>  
  <bundle id="com.asix.driver.ax88179-178a"  
CFBundleIdentifier="com.asix.driver.ax88179-178a" path="./AX88179_178A.kext"  
CFBundleVersion="1.8.0"/>  
  <bundle-version>  
    <bundle id="com.asix.driver.ax88179-178a"/>  
  </bundle-version>  
</pkg-info>
```

```
% ls -al /Library/Extensions/AX88179_178A.kext
```

```
total 0
```

```
drwxr-xr-x  3 root wheel 102 Dec 25  2015 .  
drwxr-xr-x 16 root wheel 544 Jun  3 14:48 ..  
drwxr-xr-x  6 root wheel 204 Dec 25  2015 Contents
```

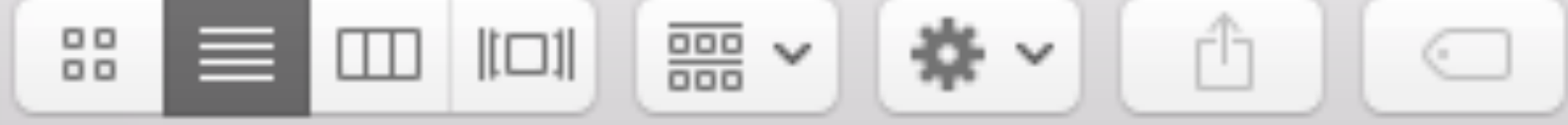
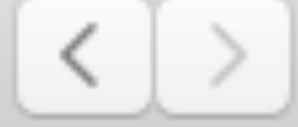
```
% lsbom /tmp/AX88179_178A_v2.8.0_pkg/ax88179178a.pkg/Bom
```

```
. 40700 501/0  
./AX88179_178A.kext 40755 501/20  
./AX88179_178A.kext/Contents 40755 501/20  
./AX88179_178A.kext/Contents/Info.plist 100644 501/20 4882 2338734375  
./AX88179_178A.kext/Contents/MacOS 40755 501/20  
./AX88179_178A.kext/Contents/MacOS/AX88179_178A 100755 501/20 88528 1035438097  
./AX88179_178A.kext/Contents/Resources 40755 501/20  
./AX88179_178A.kext/Contents/Resources/en.lproj 40755 501/20  
./AX88179_178A.kext/Contents/Resources/en.lproj/InfoPlist.strings 100644 501/20 152 221771084  
./AX88179_178A.kext/Contents/_CodeSignature 40755 501/20  
./AX88179_178A.kext/Contents/_CodeSignature/CodeResources 100644 501/20 2370 1632215908
```

A package **may** also
optionally run some scripts.



ax88179178a.pkg



Search

- Favorites
- All My Files
 - iCloud Drive
 - Dropbox
 - AirDrop
 - Applications
 - Desktop
 - Documents
 - Downloads
 - Movies
 - Music
 - Pictures
 - gneagle
 - Shared

Name	Date Modified
Bom	Apr 19, 2016, 12:01 AM
PackageInfo	Apr 19, 2016, 12:01 AM
Payload	Apr 19, 2016, 12:01 AM
Scripts	Today, 11:03 AM
postinstall	Sep 29, 2014, 4:15 AM
preinstall	Feb 18, 2014, 11:14 PM

6 items, 34.79 GB available

```
% cat /tmp/AX88179_178A_v2.8.0_pkg/ax88179178a.pkg/Scripts/postinstall
```

```
#!/bin/sh
```

```
#filename:autopack.sh
```

```
if [ -d /System/Library/Extensions/IONetworkingFamily.kext/Contents/PlugIns/  
AX88179_178A.kext ]; then  
sudo chown -R root:wheel /System/Library/Extensions/IONetworkingFamily.kext/Contents/PlugIns/  
AX88179_178A.kext  
sudo chmod 755 /System/Library/Extensions/IONetworkingFamily.kext/Contents/PlugIns/  
AX88179_178A.kext/Contents/MacOS/AX88179_178A  
fi
```

```
if [ -d /Library/Extensions/AX88179_178A.kext ]; then  
sudo chown -R root:wheel /Library/Extensions/AX88179_178A.kext  
sudo chmod 755 /Library/Extensions/AX88179_178A.kext/Contents/MacOS/AX88179_178A  
fi
```

```
sudo touch /System/Library/Extensions
```



```
% cat /tmp/AX88179_178A_v2.8.0_pkg/ax88179178a.pkg/Scripts/postinstall
```

```
#!/bin/sh
```

```
#filename:autopack.sh
```

```
if [ -d /System/Library/Extensions/IONetworkingFamily.kext/Contents/PlugIns/  
AX88179_178A.kext ]; then  
sudo chown -R root:wheel /System/Library/Extensions/IONetworkingFamily.kext/Contents/PlugIns/  
AX88179_178A.kext  
sudo chmod 755 /System/Library/Extensions/IONetworkingFamily.kext/Contents/PlugIns/  
AX88179_178A.kext/Contents/MacOS/AX88179_178A  
fi
```

```
if [ -d /Library/Extensions/AX88179_178A.kext ]; then  
sudo chown -R root:wheel /Library/Extensions/AX88179_178A.kext  
sudo chmod 755 /Library/Extensions/AX88179_178A.kext/Contents/MacOS/AX88179_178A  
fi
```

```
sudo touch /System/Library/Extensions
```

The archive contains information about location, kind, **owner**, **group**, and **mode** of each file and directory.

Info alternatives

```
% pkgutil --payload-files /Volumes/AX88179/AX88179_178A_v2.8.0.pkg
```

```
.
```

```
./AX88179_178A.kext
```

```
./AX88179_178A.kext/Contents
```

```
./AX88179_178A.kext/Contents/_CodeSignature
```

```
./AX88179_178A.kext/Contents/_CodeSignature/CodeResources
```

```
./AX88179_178A.kext/Contents/Info.plist
```

```
./AX88179_178A.kext/Contents/MacOS
```

```
./AX88179_178A.kext/Contents/MacOS/AX88179_178A
```

```
./AX88179_178A.kext/Contents/Resources
```

```
./AX88179_178A.kext/Contents/Resources/en.lproj
```

```
./AX88179_178A.kext/Contents/Resources/en.lproj/InfoPlist.strings
```

```
.
```

```
./AX88179_178A.kext
```

```
./AX88179_178A.kext/Contents
```

```
./AX88179_178A.kext/Contents/Info.plist
```

```
./AX88179_178A.kext/Contents/MacOS
```

```
./AX88179_178A.kext/Contents/MacOS/AX88179_178A
```

```
./AX88179_178A.kext/Contents/Resources
```

```
./AX88179_178A.kext/Contents/Resources/en.lproj
```

```
./AX88179_178A.kext/Contents/Resources/en.lproj/InfoPlist.strings
```

```
% pkgutil --bom /Volumes/AX88179/AX88179_178A_v2.8.0.pkg
/tmp/AX88179_178A_v2.8.0.pkg.boms.4Cb2Zq/ax88179178a.pkg/Bom
/tmp/AX88179_178A_v2.8.0.pkg.boms.4Cb2Zq/ax88179178a-1.pkg/Bom
% lsbom /tmp/AX88179_178A_v2.8.0.pkg.boms.4Cb2Zq/ax88179178a.pkg/Bom
. 40700 501/0
./AX88179_178A.kext 40755 501/20
./AX88179_178A.kext/Contents 40755 501/20
./AX88179_178A.kext/Contents/Info.plist 100644 501/20 4882 2338734375
./AX88179_178A.kext/Contents/MacOS 40755 501/20
./AX88179_178A.kext/Contents/MacOS/AX88179_178A 100755 501/20 88528 1035438097
./AX88179_178A.kext/Contents/Resources 40755 501/20
./AX88179_178A.kext/Contents/Resources/en.lproj 40755 501/20
./AX88179_178A.kext/Contents/Resources/en.lproj/InfoPlist.strings 100644 501/20 152 221771084
./AX88179_178A.kext/Contents/_CodeSignature 40755 501/20
./AX88179_178A.kext/Contents/_CodeSignature/CodeResources 100644 501/20 2370 1632215908
```

```
% cat /tmp/AX88179_178A_v2.8.0_pkg/Distribution
<?xml version="1.0" encoding="utf-8" standalone="no"?>
<installer-script minSpecVersion="1.000000" authoringTool="com.apple.PackageMaker"
authoringToolVersion="3.0.6" authoringToolBuild="201">
  <title>AX88179/178A USB3.0/2.0 Gigabit Ethernet Adapter Driver</title>
  <options customize="allow" allow-external-scripts="no"/>
  <domains enable_localSystem="true"/>
  <script>
function pm_choice3_selected() {
  result = true;
  result = result &&& (/* >= */
system.compareVersions(system.version.ProductVersion, '10.9.0') >= 0);
  return result;
}

function pm_choice3_enabled() {
  result = true;
  result = result &&& (/* >= */
system.compareVersions(system.version.ProductVersion, '10.9.0') >= 0);
  return result;
}
```

```
}</script>
```

```
<choices-outline>
```

```
<line choice="choice3"/>
```

```
<line choice="choice58"/>
```

```
</choices-outline>
```

```
<choice id="choice3" title="OS X 10.9 to 10.11 driver " description="AX88179/178A  
Macintosh 10.9 to 10.11 Driver v1.8.0" start_enabled="false" selected="pm_choice3_selected()"  
enabled="pm_choice3_enabled(">
```

```
<pkg-ref id="com.asix.pkg.ax88179-178a-10.9"/>
```

```
</choice>
```

```
<choice id="choice58" title="OS X 10.6 to 10.8 driver" description="AX88179/178A Macintosh  
10.6 to 10.8 Driver v1.11.0" start_enabled="false" selected="pm_choice58_selected()"  
enabled="pm_choice58_enabled(">
```

```
<pkg-ref id="com.asix.pkg.ax88179-178a-10.8"/>
```

```
</choice>
```

```
<pkg-ref id="com.asix.pkg.ax88179-178a-10.9" installKBytes="104" version="2.8.0"  
auth="Root" onConclusion="RequireRestart">#ax88179178a.pkg</pkg-ref>
```

```
<pkg-ref id="com.asix.pkg.ax88179-178a-10.8" installKBytes="176" version="2.8.0"  
auth="Root" onConclusion="RequireRestart">#ax88179178a-1.pkg</pkg-ref>
```

```
</installer-script>
```

```
}</script>
  <choices-outline>
    <line choice="choice3"/>
    <line choice="choice58"/>
  </choices-outline>
  <choice id="choice3" title="OS X 10.9 to 10.11 driver " description="AX88179/178A
Macintosh 10.9 to 10.11 Driver v1.8.0" start_enabled="false" selected="pm_choice3_selected()"
enabled="pm_choice3_enabled(">
    <pkg-ref id="com.asix.pkg.ax88179-178a-10.9"/>
  </choice>
  <choice id="choice58" title="OS X 10.6 to 10.8 driver" description="AX88179/178A Macintosh
10.6 to 10.8 Driver v1.11.0" start_enabled="false" selected="pm_choice58_selected()"
enabled="pm_choice58_enabled(">
    <pkg-ref id="com.asix.pkg.ax88179-178a-10.8"/>
  </choice>
  <pkg-ref id="com.asix.pkg.ax88179-178a-10.9" installKBytes="104" version="2.8.0"
auth="Root" onConclusion="RequireRestart">#ax88179178a.pkg</pkg-ref>
  <pkg-ref id="com.asix.pkg.ax88179-178a-10.8" installKBytes="176" version="2.8.0"
auth="Root" onConclusion="RequireRestart">#ax88179178a-1.pkg</pkg-ref>
</installer-script>
```


Flat package review

Examine the contents of the Adobe Reader DC package* and answer these questions:

- What are the component packages inside the distribution package?
- Which component packages include scripts?
- What are the install-locations for the component packages?

*AcroRdrDC_1501620039_MUI.pkg, in the AcroRdrDC_1501620039_MUI.dmg

Examine the contents of the Adobe Reader DC package and answer these questions:

- What are the component packages inside the distribution package?

```
% pkgutil --expand /Volumes/AcroRdrDC_1501620039_MUI/AcroRdrDC_1501620039_MUI.pkg /tmp/AdobeReader_pkg
```

```
% ls /tmp/AdobeReader_pkg
```

```
Distribution          Resources          armagent.pkg          support.pkg
RdrServicesUpdater.pkg application.pkg      browser.pkg
```

```
% ls /tmp/AdobeReader_pkg | grep .pkg
```

```
RdrServicesUpdater.pkg
application.pkg
armagent.pkg
browser.pkg
support.pkg
```

Examine the contents of the Adobe Reader DC package and answer these questions:

- Which component packages include scripts?

```
% ls /tmp/AdobeReader_pkg/RdrServicesUpdater.pkg
Bom          PackageInfo Payload      Scripts
% ls /tmp/AdobeReader_pkg/application.pkg
Bom          PackageInfo Payload      Scripts
% ls /tmp/AdobeReader_pkg/armagent.pkg
Bom          PackageInfo Payload      Scripts
% ls /tmp/AdobeReader_pkg/browser.pkg
Bom          PackageInfo Payload      Scripts
% ls /tmp/AdobeReader_pkg/support.pkg
Bom          PackageInfo Payload
```

Examine the contents of the Adobe Reader DC package and answer these questions:

- What are the install-locations for the component packages?

```
% cat /tmp/AdobeReader_pkg/RdrServicesUpdater.pkg/PackageInfo
<?xml version="1.0" encoding="utf-8" standalone="no"?>
<pkg-info overwrite-permissions="true" relocatable="false"
identifier="com.adobe.RdrServicesUpdater" postinstall-action="none" version="0" format-version="2"
generator-version="InstallCmds-554 (15C50)" install-location="/tmp/webinstaller" auth="root">
  <payload numberOfFiles="10" installKBytes="11601"/>
  <bundle id="com.adobe.AdobeRNAWebInstaller" CFBundleVersion="15.016.20039" path="./
RdrServicesUpdater.app"/>
  <bundle-version/>
  <upgrade-bundle>
    <bundle id="com.adobe.AdobeRNAWebInstaller"/>
  </upgrade-bundle>
  <update-bundle/>
  <atomic-update-bundle/>
  <strict-identifier>
    <bundle id="com.adobe.AdobeRNAWebInstaller"/>
```

Examine the contents of the Adobe Reader package and answer these questions:

- What are the install-locations for the component packages?

RdrServicesUpdater.pkg:	/tmp/webinstaller
application.pkg:	/Applications
armagent.pkg:	/tmp/com.adobe.AcrobatRefreshManager
browser.pkg:	/Library/Internet Plug-Ins
support.pkg:	/Library/Application Support

Auditing Packages




Suspicious Package

http://www.mothersruin.com/software/SuspiciousPackage/

The image shows a screenshot of a web browser displaying the website for 'Suspicious Package'. The browser's address bar shows the URL 'www.mothersruin.com/software/SuspiciousPackage/'. The website has a navigation menu with 'PRODUCTS' and 'ABOUT' links, and a logo in the top right corner. On the left side of the website, there is a large blue circular icon containing a white open cardboard box. Below this icon is a vertical sidebar with buttons for 'DOWNLOAD', 'USE', 'FAQ', 'SCRIPTING', and 'HISTORY'. The main content area features the title 'Suspicious Package' and the subtitle 'An Application for Inspecting OS X Installer Packages'. It includes three rhetorical questions: 'Do you know what files that OS X Installer package actually installs?', 'Do you know what scripts it runs during installation, and what they do?', and 'Do you know who the package really came from?'. A paragraph explains that the application can answer these questions and more, allowing users to inspect the contents of an installer package. Below the text, there is a preview of the application's interface. The application window is titled 'JavaForOSX.pkg' and shows a 'Package Info' tab. It displays a yellow and orange cardboard box icon, the name 'JavaForOSX.pkg', and the description 'Apple Installer Package'. It also shows that the package was previously installed on 'Little My' using 'Apple Installer' on 'December 13, 2015'. At the bottom, it indicates that the package 'Installs 1935 items — 111.1 MB on disk'.

PRODUCTS ABOUT



Suspicious Package

An Application for Inspecting OS X Installer Packages

Do you know what files that OS X Installer package actually installs?

Do you know what scripts it runs during installation, and what they do?

Do you know who the package *really* came from?

With Suspicious Package, you can answer these questions and more. Maybe you're quite literally suspicious of a package you've downloaded. Or perhaps you're just curious about what some package does. Or maybe you want to find out after the fact exactly what files a package scattered across your computer. Whatever the reason, Suspicious Package allows you to see inside an installer package. (And it's completely free.)

Suspicious Package is actually both an OS X application ...

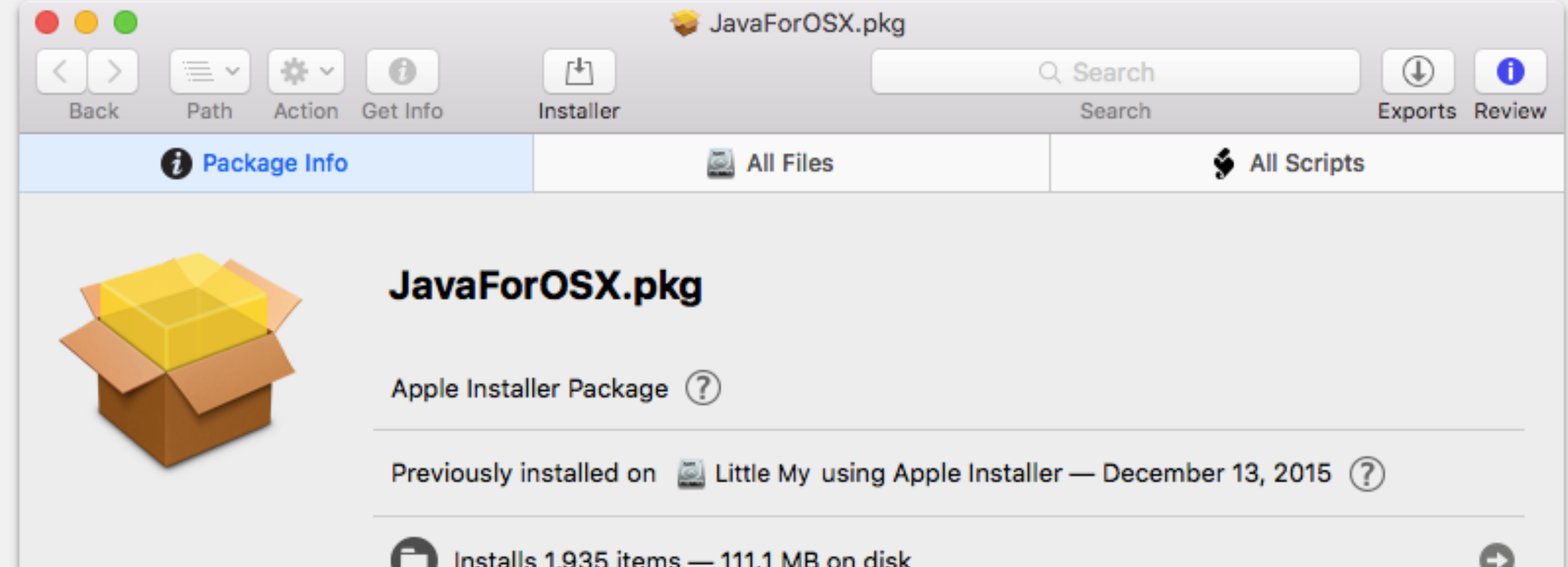
DOWNLOAD

USE

FAQ

SCRIPTING


HISTORY




JavaForOSX.pkg


Back Path Action Get Info Installer Search Search Exports Review

Package Info All Files All Scripts

 **JavaForOSX.pkg**

Apple Installer Package ?

Previously installed on  Little My using Apple Installer — December 13, 2015 ?

 Installs 1935 items — 111.1 MB on disk



Adobe Flash Player.pkg

Open with Installer



Open in Suspicious Package



Adobe Flash Player.pkg

Developer ID Installer Package



DEVELOPER ID

Signed by "Developer ID Installer: Adobe Systems, Inc."



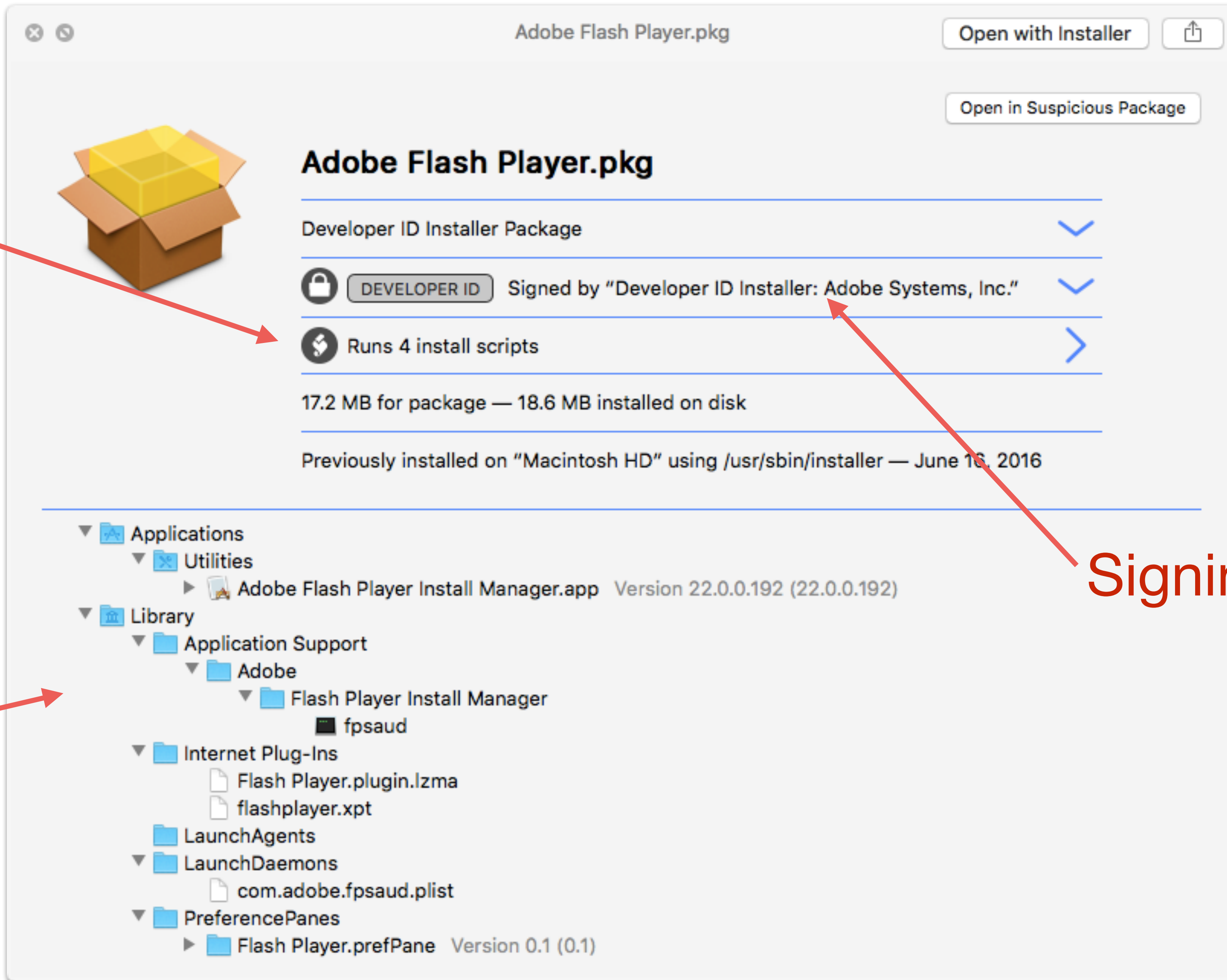
Runs 4 install scripts



17.2 MB for package — 18.6 MB installed on disk

Previously installed on "Macintosh HD" using /usr/sbin/installer — June 16, 2016

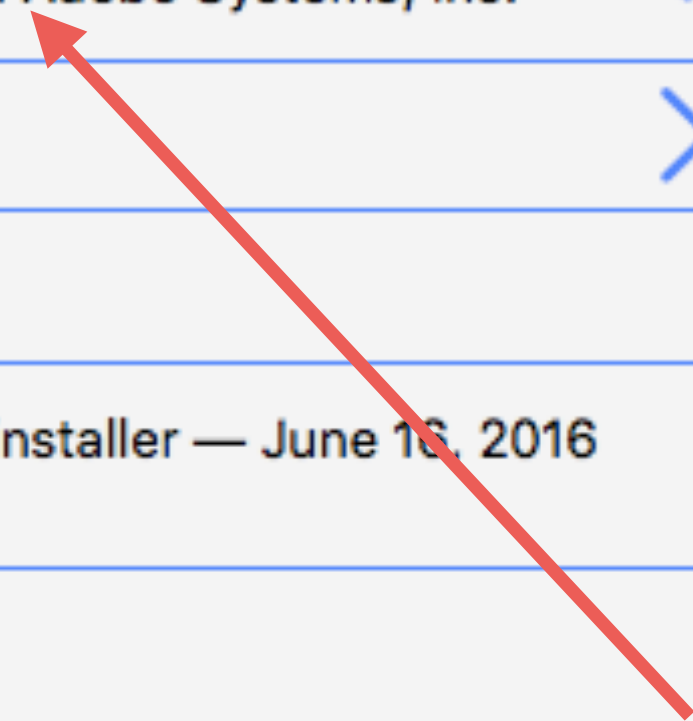
- ▼ Applications
 - ▼ Utilities
 - ▶ Adobe Flash Player Install Manager.app Version 22.0.0.192 (22.0.0.192)
- ▼ Library
 - ▼ Application Support
 - ▼ Adobe
 - ▼ Flash Player Install Manager
 - fpsaud
 - ▼ Internet Plug-Ins
 - Flash Player.plugin.lzma
 - flashplayer.xpt
 - LaunchAgents
 - ▼ LaunchDaemons
 - com.adobe.fpsaud.plist
 - ▼ PreferencePanels
 - ▶ Flash Player.prefPane Version 0.1 (0.1)



Scripts



Signing certificate




Payload



Adobe Flash Player.pkg

Open with Installer

Open in Suspicious Package



Adobe Flash Player.pkg

Developer ID Installer Package

DEVELOPER ID Signed by "Developer ID Installer: Adobe Systems, Inc."

Apple Root CA
Root certificate authority
Expires: Feb 9, 2035, 1:40:36 PM
✔ This certificate is valid

Developer ID Certification Authority
Intermediate certificate authority
Expires: Feb 1, 2027, 2:12:15 PM
✔ This certificate is valid

Developer ID Installer: Adobe Systems, Inc.
Expires: Feb 21, 2017, 11:42:10 AM
✔ This certificate is valid

[Read more](#) about how to interpret package signatures.

Runs 4 install scripts

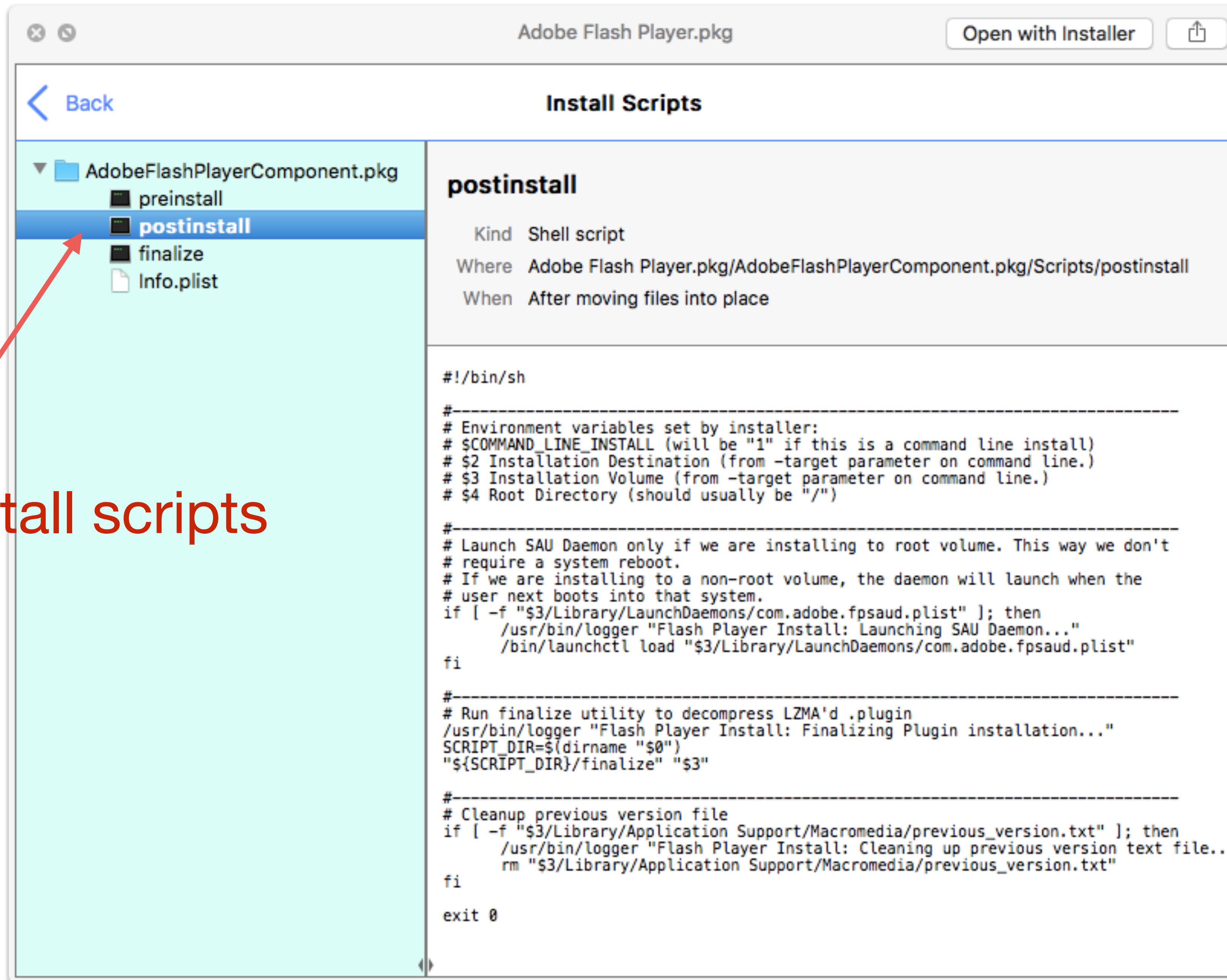
17.2 MB for package — 18.6 MB installed on disk

Previously installed on "Macintosh HD" using /usr/sbin/installer — June 16, 2016

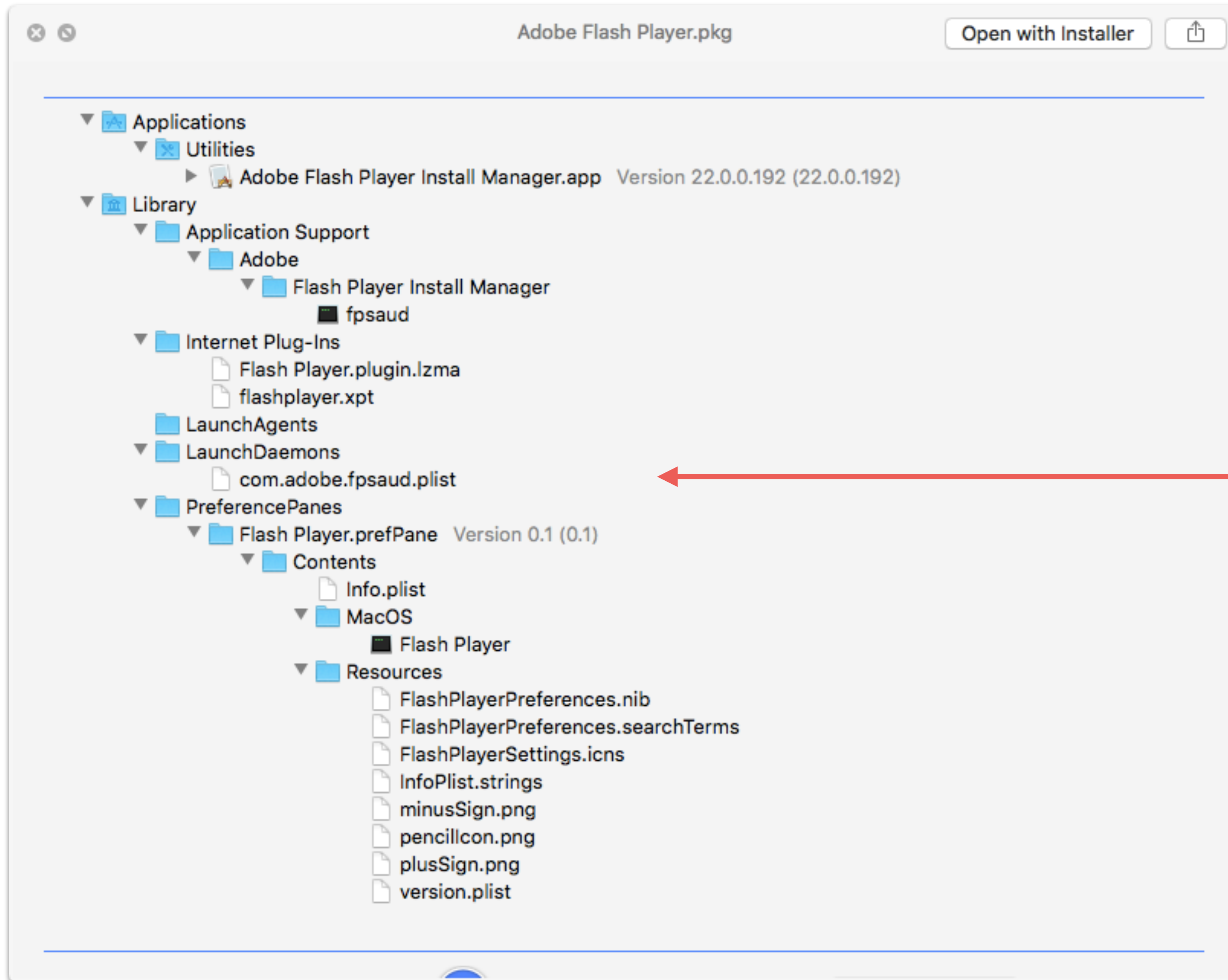
Applications

Signing certificate chain





Package install scripts




Payload

Adobe Flash Player.pkg

Open with Installer

Open in Suspicious Package



Adobe Flash Player.pkg

Developer ID Installer Package

DEVELOPER ID Signed by "Developer ID Installer: Adobe Systems, Inc."

Runs 4 install scripts

17.2 MB for package — 18.6 MB installed on disk


Previously installed on "Macintosh HD" using /usr/sbin/installer — June 16, 2016

- ▼ Applications
 - ▼ Utilities
 - ▶ Adobe Flash Player Install Manager.app Version 22.0.0.192 (22.0.0.192)
- ▼ Library
 - ▼ Application Support
 - ▼ Adobe
 - ▼ Flash Player Install Manager
 - fpsaud
 - ▼ Internet Plug-Ins
 - Flash Player.plugin.lzma
 - flashplayer.xpt
 - LaunchAgents
 - ▼ LaunchDaemons
 - com.adobe.fpsaud.plist
 - ▼ PreferencePanels
 - ▶ Flash Player.prefPane Version 0.1 (0.1)

Adobe Flash Player.pkg

Back Path Action Get Info Installer Search Exports Review

Package Info All Files Info.plist



Adobe Flash Player.pkg

Developer ID Installer Package ?

Previously installed on Macintosh HD using /usr/sbin/installer — June 16, 2016 ?

- Installs 92 items — 18.5 MB on disk →
- DEVELOPER ID** Signed by "Developer ID Installer: Adobe Systems, Inc." →
- Runs 4 install scripts →
- ✓ Didn't find any issues for review

Suspicious Package gives you a lot of convenient information, but not the details that you can get with command line tools.

Demo



Pacifist

https://www.charlessoft.com

CharlesSoft
Software you always wished someone would write

Advertisement
Charter Spectrum More Channels, Internet and Voice

Pacifist 3.5.9

Pacifist is a shareware application that opens Mac OS X .pkg package files, .dmg disk images, and .zip, .tar, .tar.gz, .tar.bz2, and .xar archives and allows you to extract individual files and folders out of them. This is useful, for instance, if an application which is installed by the operating system becomes damaged and needs to be reinstalled without the hassle of reinstalling all of Mac OS X, or if you want to inspect a downloaded package to see what it will install before installing it. Pacifist is also able to verify existing installations and find missing or altered files*, and Pacifist can also examine the kernel extensions installed in your system to let you see what installer installed them, and whether the installer was made by Apple or a third-party.

Pacifist is compatible with Mac OS X 10.8 or higher, including Mac OS X 10.11 "El Capitan". Pacifist requires a 64-bit Intel™ processor.

- Download the latest version of Pacifist in [zip](#) or [disk image](#) format.
- Read the release notes for Pacifist [here](#).
- For users using versions of Mac OS X older than 10.8, download older versions of Pacifist [here](#).
- See some [screenshots](#) of Pacifist in action!
- Pacifist is \$20 shareware - you can [click here to register via Kagi](#).

Pacifist has received **5 mice** from Macworld magazine on two separate occasions:

Advertisement
IBM Cloud SOFTLAY-R
Reliable Dedicated Server
More Power, Faster to Provision and More Customizable Dedicated Servers

SuspiciousPackage.pkg [UNREGISTERED]

Install Extract To... Verify Show Info View Files Search Find

Package is a metapackage
Package requests authentication
Package has a valid signature: Developer ID Installer: Randy Saldinger (936EB786NH) (Developer ID Certification Authority, Apple Root CA)
Size: 230.2 KiB compressed, 609.0 KiB uncompressed
Size of selected files: 0 bytes

Contents Resources

Filename	Size	Owner	Group	Permissions	Modification Date	V
▼ Contents of SuspiciousPackage.pkg	584.8 KiB	root	wheel	drwxr-xr-x		
▶ Contents of SuspiciousPackagePlugin.pkg	584.8 KiB	root	wheel	drwxr-xr-x		

Install Extract To... Verify Show Info View Files

Search Find

Package is a metapackage
Package requests authentication
Package has a valid signature: Developer ID Installer: Adobe Systems, Inc. (Developer ID Certification Authority, Apple Root CA)
Size: 16.4 MiB compressed, 17.7 MiB uncompressed
Size of selected files: 0 bytes

Signing certificate



Contents Resources

Filename	Size	Owner	Group	Permissions	Modification Date
▼ Contents of Adobe Flash Player.pkg	17.7 MiB	root	wheel	drwxr-xr-x	
▶ Contents of AdobeFlashPlayerComponent.pkg	17.7 MiB	root	wheel	drwxr-xr-x	

Metapackage payload



- Install
- Extract To...
- Verify
- Show Info
- View Files

Search Find

Package is a metapackage

Package requests authentication

Package has a valid signature: Developer ID Installer: Adobe Systems, Inc. (Developer ID Certification Authority, Apple Root CA)

Size: 16.4 MiB compressed, 17.7 MiB uncompressed

Size of selected files: 0 bytes

Contents Resources

Filename	Size	Owner	Group	Permissions	Modification Date
▼ Contents of Adobe Flash Player.pkg	17.7 MiB	root	wheel	drwxr-xr-x	
▼ Contents of AdobeFlashPlayerComponent.pkg	17.7 MiB	root	wheel	drwxr-xr-x	
▼ Applications	729.4 KiB	root	wheel	drwxrwxr-x	
▼ Utilities	729.4 KiB	root	wheel	drwxr-xr-x	
▶ Adobe Flash Player Install Manager.app	729.4 KiB	root	wheel	drwxr-xr-x	
▼ Library	17.0 MiB	root	wheel	drwxr-xr-x	
▶ Application Support	79.4 KiB	root	wheel	drwxr-xr-x	
▼ Internet Plug-Ins	15.5 MiB	root	wheel	drwxr-xr-x	
Flash Player.plugin.lzma	15.5 MiB	root	wheel	-rw-rw-r--	6/13/16, 3:07 PM
flashplayer.xpt	856 bytes	root	wheel	-rw-r--r--	6/13/16, 2:49 PM
LaunchAgents	0 bytes	root	wheel	drwxr-xr-x	
▼ LaunchDaemons	462 bytes	root	wheel	drwxr-xr-x	
com.adobe.fpsaud.plist	462 bytes	root	wheel	-rw-r--r--	6/13/16, 2:49 PM
▶ PreferencePanels	1.4 MiB	root	wheel	drwxr-xr-x	

Expanded payload

- Install
- Extract To...
- Verify
- Show Info
- View Files

Search Find

Package is a metapackage

Package requests authentication

Package has a valid signature: Developer ID Installer: Adobe Systems, Inc. (Developer ID Certification Authority, Apple Root CA)

Size: 16.4 MiB compressed, 17.7 MiB uncompressed

Size of selected files: 0 bytes

Contents Resources

Filename	Size	Owner	Group	Permissions	Modification Date
▼ Adobe Flash Player.pkg	16.5 MiB	gnea...	_lpop...	drwxr-xr-x	6/13/16, 3:21 PM
Distribution	4.5 KiB	root	wheel	-rw-r--r--	
▶ Resources	168.8 KiB	root	wheel	drwx-----	12/31/69, 4:00 PM
▶ AdobeFlashPlayerComponent.pkg	16.4 MiB	root	wheel	drwx-----	12/31/69, 4:00 PM

Install
Extract To...
Verify
Show Info
View Files

Find

Package is a metapackage





Package requests authentication

Package has a valid signature: Developer ID Installer: Adobe Systems, Inc. (Developer ID Certification Authority, Apple Root CA)

Size: 16.4 MiB compressed, 17.7 MiB uncompressed

Size of selected files: 0 bytes

Contents
Resources

Filename	Size	Owner	Group	Permissions	Modification Date
<ul style="list-style-type: none"> ▼  Adobe Flash Player.pkg <ul style="list-style-type: none">  Distribution ▶  Resources ▶  AdobeFlashPlayerComponent.pkg 	16.5 MiB	gnea...	_lpop...	drwxr-xr-x	6/13/16, 3:21 PM
	4.5 KiB	root	wheel	-rw-r--r--	
	168.8 KiB	root	wheel	drwx-----	12/31/69, 4:00 PM
	16.4 MiB	root	wheel	drwx-----	12/31/69, 4:00 PM

- Install
- Extract To...
- Verify
- Show Info
- View Files

Search Find

Package is a metapackage

Package requests authentication

Package has a valid signature: Developer ID Installer: Adobe Systems, Inc. (Developer ID Certification Authority, Apple Root CA)

Size: 16.4 MiB compressed, 17.7 MiB uncompressed

Size of selected files: 0 bytes

Contents Resources

Filename	Size	Owner	Group	Permissions	Modification Date
▼ Adobe Flash Player.pkg	16.5 MiB	gnea...	_lpop...	drwxr-xr-x	6/13/16, 3:21 PM
Distribution	4.5 KiB	root	wheel	-rw-r--r--	
▶ Resources	168.8 KiB	root	wheel	drwx-----	12/31/69, 4:00 PM
▶ AdobeFlashPlayerComponent.pkg	16.4 MiB	root	wheel	drwx-----	12/31/69, 4:00 PM

Install
Extract To...
Verify
Show Info
View Files

Find

Package is a metapackage

Package requests authentication

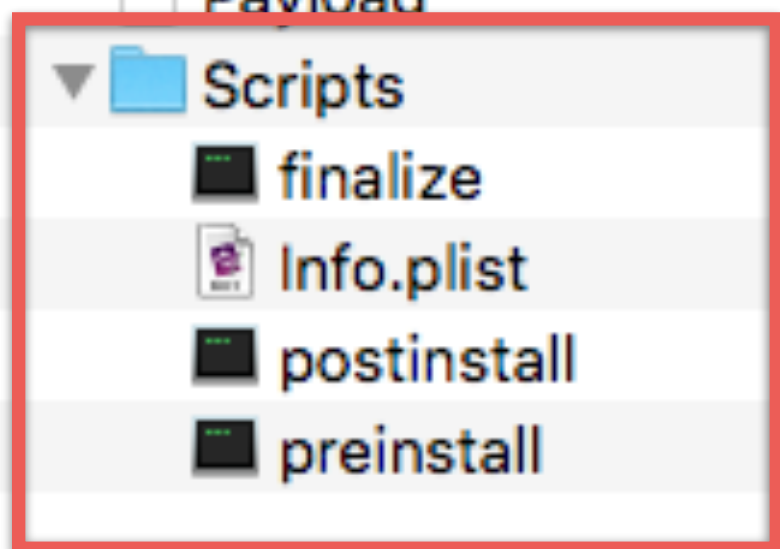
Package has a valid signature: Developer ID Installer: Adobe Systems, Inc. (Developer ID Certification Authority, Apple Root CA)

Size: 16.4 MiB compressed, 17.7 MiB uncompressed

Size of selected files: 0 bytes

Contents
Resources

Filename	Size	Owner	Group	Permissions	Modification Date
▼ Adobe Flash Player.pkg	16.5 MiB	gnea...	_lpop...	drwxr-xr-x	6/13/16, 3:21 PM
Distribution	4.5 KiB	root	wheel	-rw-r--r--	
▶ Resources	168.8 KiB	root	wheel	drwx-----	12/31/69, 4:00 PM
▼ AdobeFlashPlayerComponent.pkg	16.4 MiB	root	wheel	drwx-----	12/31/69, 4:00 PM
Bom	48.3 KiB	204	staff	-rw-r--r--	6/13/16, 3:07 PM
PackageInfo	1.3 KiB	root	wheel	-rw-r--r--	
Payload	16.3 MiB	204	staff	-rw-r--r--	6/13/16, 3:07 PM
▶ Scripts	40.5 KiB	204	staff	-rwxr-xr-x	6/13/16, 3:07 PM
finalize	93.0 KiB	root	staff	-rwxr-xr-x	6/13/16, 2:49 PM
Info.plist	2.0 KiB	root	staff	-rw-r--r--	6/13/16, 2:49 PM
postinstall	1.5 KiB	root	staff	-rwxr-xr-x	6/13/16, 2:49 PM
preinstall	1.8 KiB	root	staff	-rwxr-xr-x	6/13/16, 2:49 PM



You can look inside individual files (like scripts):

```
postinstall
#!/bin/sh

#-----
# Environment variables set by installer:
# $COMMAND_LINE_INSTALL (will be "1" if this is a command line install)
# $2 Installation Destination (from -target parameter on command line.)
# $3 Installation Volume (from -target parameter on command line.)
# $4 Root Directory (should usually be "/")

#-----
# Launch SAU Daemon only if we are installing to root volume. This way we don't
# require a system reboot.
# If we are installing to a non-root volume, the daemon will launch when the
# user next boots into that system.
if [ -f "$3/Library/LaunchDaemons/com.adobe.fpsaud.plist" ]; then
    /usr/bin/logger "Flash Player Install: Launching SAU Daemon..."
    /bin/launchctl load "$3/Library/LaunchDaemons/com.adobe.fpsaud.plist"
fi

#-----
# Run finalize utility to decompress LZMA'd .plugin
/usr/bin/logger "Flash Player Install: Finalizing Plugin installation..."
SCRIPT_DIR=$(dirname "$0")
"${SCRIPT_DIR}/finalize" "$3"

#-----
# Cleanup previous version file
if [ -f "$3/Library/Application Support/Macromedia/previous_version.txt" ]; then
    /usr/bin/logger "Flash Player Install: Cleaning up previous version text file..."
    rm "$3/Library/Application Support/Macromedia/previous_version.txt"
fi

exit 0
```

Take a peek at the Distribution file:

```
Distribution
<?xml version="1.0" encoding="utf-8" standalone="no"?>
<installer-gui-script minSpecVersion="1">
  <title>Adobe Flash Player</title>
  <!-- If we want to customize GUI installer later:
    <welcome file="Welcome.rtf"/>
  <readme file="ReadMe.rtf"/>
  <background file="background.tiff" scaling="proportional" alignment="bottomleft"/>
  -->
  <pkg-ref id="com.adobe.pkg.FlashPlayer">
    <bundle-version>
      <bundle CFBundleShortVersionString="0.1" CFBundleVersion="0.1" SourceVersion="270000" BuildVersion="1"
id="com.adobe.flashplayerpreferences" path="Library/PreferencePanels/Flash Player.prefPane"/>
      <bundle CFBundleShortVersionString="22.0.0.192" CFBundleVersion="22.0.0.192"
id="com.adobe.flashplayer.installmanager" path="Applications/Utilities/Adobe Flash Player Install Manager.app"/>
    </bundle-version>
  </pkg-ref>
  <options customize="never" require-scripts="false"/>
  <choices-outline>
    <line choice="default">
      <line choice="com.adobe.pkg.FlashPlayer"/>
    </line>
  </choices-outline>
  <choice id="default"/>
  <choice id="com.adobe.pkg.FlashPlayer" visible="false">
    <pkg-ref id="com.adobe.pkg.FlashPlayer"/>
  </choice>
  <volume-check script="VolumeCheck()"/>
  <script><![CDATA[
function VolumeCheck()
{
  system.log("Flash Player Install: Performing volume check for " + my.target.mountpoint);

  // Check for an OSX install on Volume
  var osVersionPlistPath = my.target.mountpoint + "/System/Library/CoreServices/SystemVersion.plist";
```

Exercise

- Using Suspicious Package:
 - How many scripts are in the Adobe Reader DC install pkg?
 - What is the version of the PDF reader browser plugin?

Apple packaging tools

pkgutil

Query and manipulate OS X
Installer packages and receipts.

pkgutil

Receipt database commands:

--pkgs

--pkg-info

--files

--file-info

--forget

pkgutil --pkgs

```
% pkgutil --pkgs  
adobe.adobeShockwavePlayer.shockwave11.pkg  
adobe.adobeShockwavePlayer.shockwave12.pkg  
com.adobe.pkg.AIR  
com.adobe.pkg.FlashPlayer  
com.alias.pkg.AliasUninstaller1.0  
com.alinofsoftware.alinoftimer  
com.amazon.Kindle  
com.apple.pkg.AdditionalEssentials  
com.apple.pkg.AdditionalSpeechVoices  
com.apple.pkg.AirPortUtility  
com.apple.pkg.Aperture_AppStore  
com.apple.pkg.AppleConfigurator  
com.apple.pkg.AsianLanguagesSupport  
com.apple.pkg.BaseSystemBinaries  
com.apple.pkg.BaseSystemResources
```

pkgutil --pkgs

```
% pkgutil --pkgs | grep -i flash  
com.adobe.pkg.FlashPlayer  
com.googlecode.munki.disableflash10autoupdate
```

pkgutil --pkg-info

```
% pkgutil --pkg-info com.googlecode.munki.disableflash10autoupdate  
package-id: com.googlecode.munki.disableflash10autoupdate  
version: 1.0  
volume: /  
location:  
install-time: 1375297642
```

```
% date -r 1375297642  
Wed Jul 31 12:07:22 PDT 2013
```

pkgutil --files

```
% pkgutil --files com.googlecode.munki.disableflash10autoupdate  
Library  
Library/Application Support  
Library/Application Support/Macromedia  
Library/Application Support/Macromedia/mms.cfg
```

pkgutil --file-info

```
% pkgutil --file-info /Library/Application\ Support/Macromedia/mms.cfg  
volume: /  
path: /Library/Application Support/Macromedia/mms.cfg  
  
pkgid: com.googlecode.munki.disableflash10autoupdate  
pkg-version: 1.0  
install-time: 1375297642  
uid: 0  
gid: 80  
mode: 664
```

pkgutil --forget

```
% pkgutil --forget com.adobe.pkg.FlashPlayer
```

pkgutil

Flat package commands:

--expand

--flatten

--bom

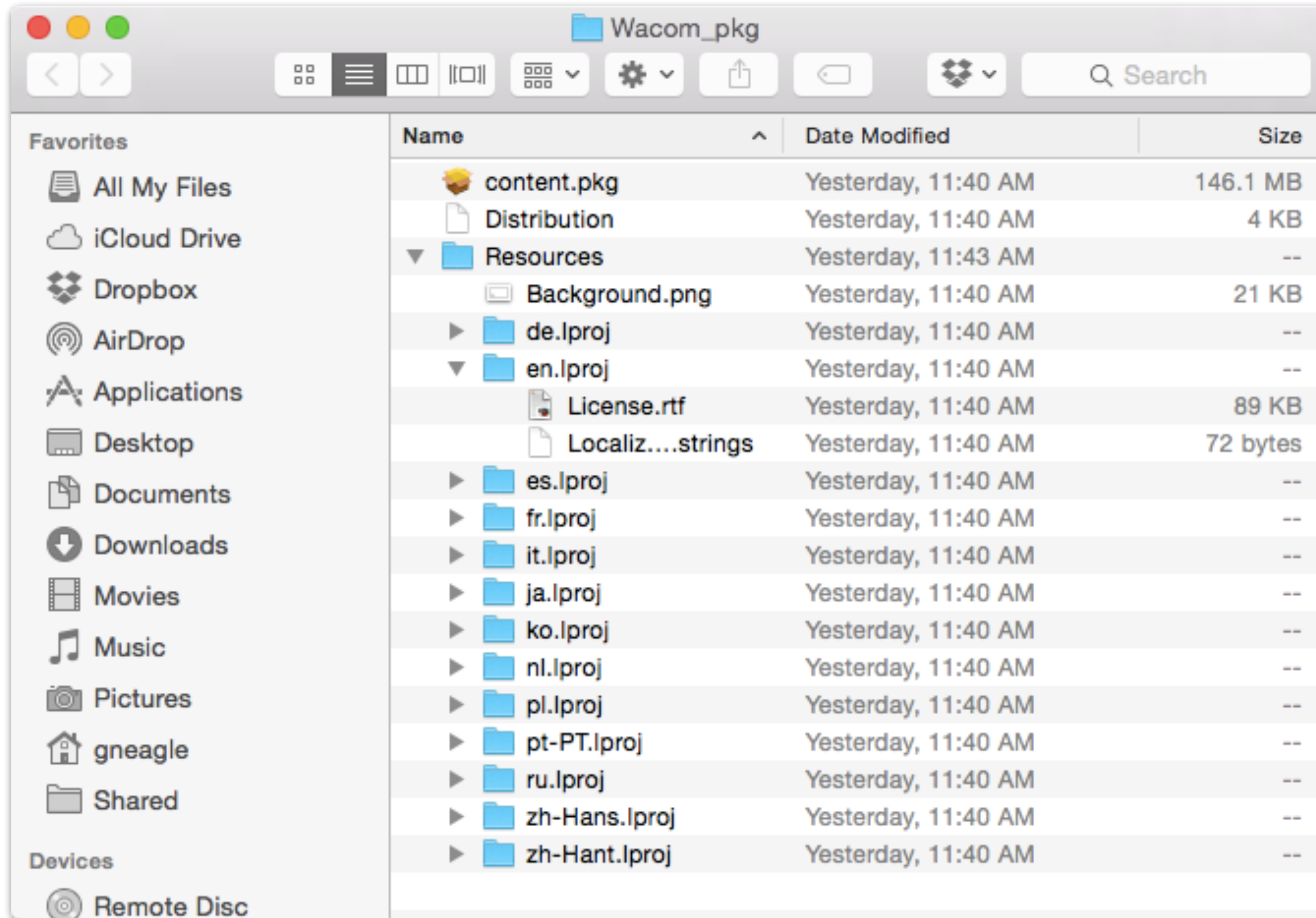
--payload-files

--check-signature

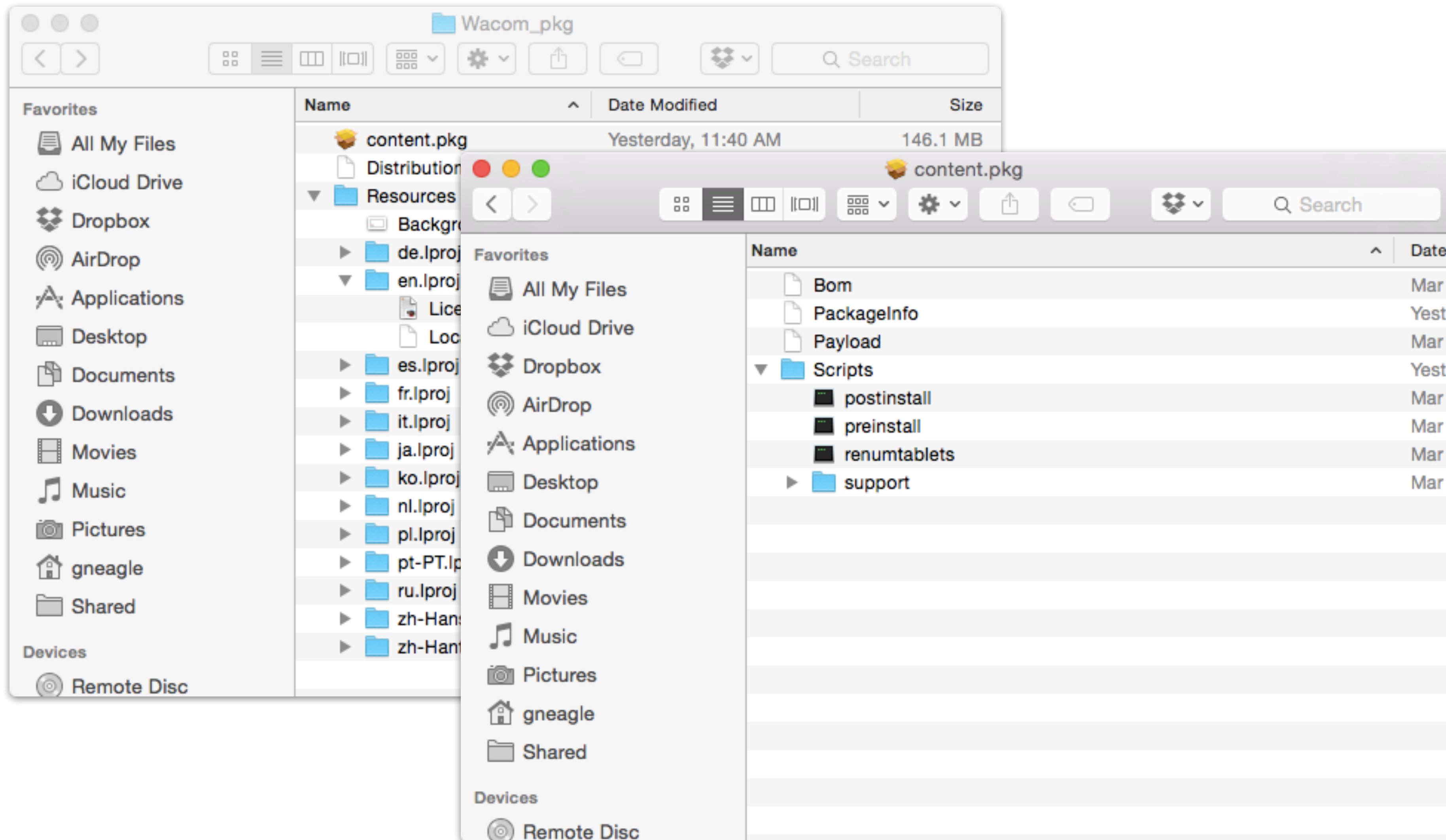
pkgutil --expand

```
% pkgutil --expand Install\ Wacom\ Tablet.pkg /tmp/Wacom_pkg
```

pkgutil --expand



pkgutil --expand



pkgutil --flatten

```
% pkgutil --flatten /tmp/Wacom_pkg ~/Desktop/CustomWacom.pkg
```

pkgutil --bom

```
% pkgutil --bom Install\ Wacom\ Tablet.pkg  
/tmp/Install Wacom Tablet.pkg.boms.ye2sPU/content.pkg/Bom
```

```
% lsbom "/tmp/Install Wacom Tablet.pkg.boms.ye2sPU/content.pkg/Bom"  
. 407550/0  
./Applications 407750/0  
./Applications/Wacom Tablet.localized 40775 0/0  
./Applications/Wacom Tablet.localized/.localized40775 0/0  
./Applications/Wacom Tablet.localized/.localized/de.strings 100664 0/072 2630072004  
./Applications/Wacom Tablet.localized/.localized/en.strings 100664 0/070 2525589374  
./Applications/Wacom Tablet.localized/.localized/es.strings 100664 0/072 3567506170  
./Applications/Wacom Tablet.localized/.localized/fr.strings 100664 0/074 3315303271  
./Applications/Wacom Tablet.localized/.localized/it.strings 100664 0/076 3843434223  
./Applications/Wacom Tablet.localized/.localized/ja.strings 100664 0/064 3782196401  
./Applications/Wacom Tablet.localized/.localized/ko.strings 100664 0/058 981171454  
./Applications/Wacom Tablet.localized/.localized/nl.strings 100664 0/070 3200743802  
./Applications/Wacom Tablet.localized/.localized/pl.strings 100664 0/070 2340425475  
./Applications/Wacom Tablet.localized/.localized/pt.strings 100664 0/082 1558099759  
./Applications/Wacom Tablet.localized/.localized/ru.strings 100664 0/072 3166065194  
./Applications/Wacom Tablet.localized/.localized/zh_CN.strings 100664 0/064 3933362247  
./Applications/Wacom Tablet.localized/.localized/zh_TW.strings 100664 0/064 3696104737  
./Applications/Wacom Tablet.localized/Wacom Desktop Center.app 40775 0/0
```

pkgutil --payload-files

```
% pkgutil --payload-files /Install\ Wacom\ Tablet.pkg
```

```
•  
./Applications  
./Applications/Wacom Tablet.localized  
./Applications/Wacom Tablet.localized/.localized  
./Applications/Wacom Tablet.localized/.localized/de.strings  
./Applications/Wacom Tablet.localized/.localized/en.strings  
./Applications/Wacom Tablet.localized/.localized/es.strings  
./Applications/Wacom Tablet.localized/.localized/fr.strings  
./Applications/Wacom Tablet.localized/.localized/it.strings  
./Applications/Wacom Tablet.localized/.localized/ja.strings  
./Applications/Wacom Tablet.localized/.localized/ko.strings  
./Applications/Wacom Tablet.localized/.localized/nl.strings  
./Applications/Wacom Tablet.localized/.localized/pl.strings  
./Applications/Wacom Tablet.localized/.localized/pt.strings  
./Applications/Wacom Tablet.localized/.localized/ru.strings  
./Applications/Wacom Tablet.localized/.localized/zh_CN.strings  
./Applications/Wacom Tablet.localized/.localized/zh_TW.strings  
./Applications/Wacom Tablet.localized/aft.tar  
./Applications/Wacom Tablet.localized/Wacom Desktop Center.app
```

pkgutil --check-signature

```
% pkgutil --check-signature Install\ Wacom\ Tablet.pkg
```

```
Package "Install Wacom Tablet.pkg":
```

```
Status: signed by a certificate trusted by Mac OS X
```

```
Certificate Chain:
```

```
1. Developer ID Installer: Wacom Technology Corp.
```

```
SHA1 fingerprint: 62 F5 E7 EC B3 21 8F B7 CE 63 70 4B E7 B9 8A 0B 3B 03 0D 8D
```

```
-----
```

```
2. Developer ID Certification Authority
```

```
SHA1 fingerprint: 3B 16 6C 3B 7D C4 B7 51 C9 FE 2A FA B9 13 56 41 E3 88 E1 86
```

```
-----
```

```
3. Apple Root CA
```

```
SHA1 fingerprint: 61 1E 5B 66 2C 59 3A 08 FF 58 D1 4A E2 24 52 D1 98 DF 6C 60
```

pkgutil review

pkgutil review tasks

- How many packages are installed on your machine?
- What pkgs are responsible for the current contents of these files?
 - /Applications/Chess.app/Contents/MacOS/Chess
 - /Applications/Safari.app/Contents/MacOS/Safari
- What files do the XProtectPlistConfigData packages manage?

pkgutil review tasks

- How many packages are installed on your machine?

```
% pkgutil --pkgs | wc -l
```

pkgutil review tasks

- How many packages are installed on your machine?

```
% pkgutil --pkgs | wc -l
```

- Bonus: How many Microsoft packages are installed on your machine?

```
% pkgutil --pkgs | grep -i microsoft | wc -l
```

pkgutil review tasks

- What pkgs are responsible for the current contents of these files?
 - /Applications/Chess.app/Contents/MacOS/Chess
 - /Applications/Safari.app/Contents/MacOS/Safari

```
% pkgutil --file-info /Applications/Chess.app/Contents/MacOS/Chess
```

```
% pkgutil --file-info /Applications/Safari.app/Contents/MacOS/Safari
```

pkgutil review tasks

- What files do the XProtectPlistConfigData packages manage?

```
% pkgutil --pkgs | grep XProtect
com.apple.pkg.XProtectPlistConfigData.14U4036
com.apple.pkg.XProtectPlistConfigData.14U4037
com.apple.pkg.XProtectPlistConfigData.14U4041
com.apple.pkg.XProtectPlistConfigData.14U4042
com.apple.pkg.XProtectPlistConfigData.14U4043
com.apple.pkg.XProtectPlistConfigData.14U4046
% pkgutil --files com.apple.pkg.XProtectPlistConfigData.14U4046
System
System/Library
System/Library/CoreServices
System/Library/CoreServices/XProtect.bundle
System/Library/CoreServices/XProtect.bundle/Contents
System/Library/CoreServices/XProtect.bundle/Contents/Info.plist
System/Library/CoreServices/XProtect.bundle/Contents/Resources
```

pkgutil review tasks

- What files do the XProtectPlistConfigData packages manage?

```
% pkgutil --files com.apple.pkg.XProtectPlistConfigData.14U4046
```

```
System
```

```
System/Library
```

```
System/Library/CoreServices
```

```
System/Library/CoreServices/XProtect.bundle
```

```
System/Library/CoreServices/XProtect.bundle/Contents
```

```
System/Library/CoreServices/XProtect.bundle/Contents/Info.plist
```

```
System/Library/CoreServices/XProtect.bundle/Contents/Resources
```

```
System/Library/CoreServices/XProtect.bundle/Contents/Resources/XProtect.meta.plist
```

```
System/Library/CoreServices/XProtect.bundle/Contents/Resources/XProtect.plist
```

```
System/Library/CoreServices/XProtect.bundle/Contents/_CodeSignature
```

```
System/Library/CoreServices/XProtect.bundle/Contents/_CodeSignature/CodeDirectory
```

```
System/Library/CoreServices/XProtect.bundle/Contents/_CodeSignature/CodeRequirements
```

```
System/Library/CoreServices/XProtect.bundle/Contents/_CodeSignature/CodeResources
```

```
System/Library/CoreServices/XProtect.bundle/Contents/_CodeSignature/CodeSignature
```

```
System/Library/CoreServices/XProtect.bundle/Contents/version.plist
```

pkgbuild

Build an OS X Installer component package from on-disk files

pkgbuild

We're going to build a simple package to bypass the OS X Registration / Setup Assistant that launches on first boot.

Two files need to be placed:

- `/private/var/db/.AppleSetupDone`
- `/Library/Receipts/.SetupRegComplete`

pkgbuild

```
# need some working space  
% cd ~/Desktop
```

pkgbuild

```
# make a root directory for our files  
% mkdir ClearRegistration
```

pkgbuild

```
# create needed subdirectories
```

```
% mkdir -p ClearRegistration/private/var/db
```

```
% mkdir -p ClearRegistration/Library/Receipts
```

pkgbuild

```
# create our two files in the correct subdirectories  
% touch ClearRegistration/private/var/db/.AppleSetupDone  
% touch ClearRegistration/Library/Receipts/.SetupRegComplete
```

pkgbuild

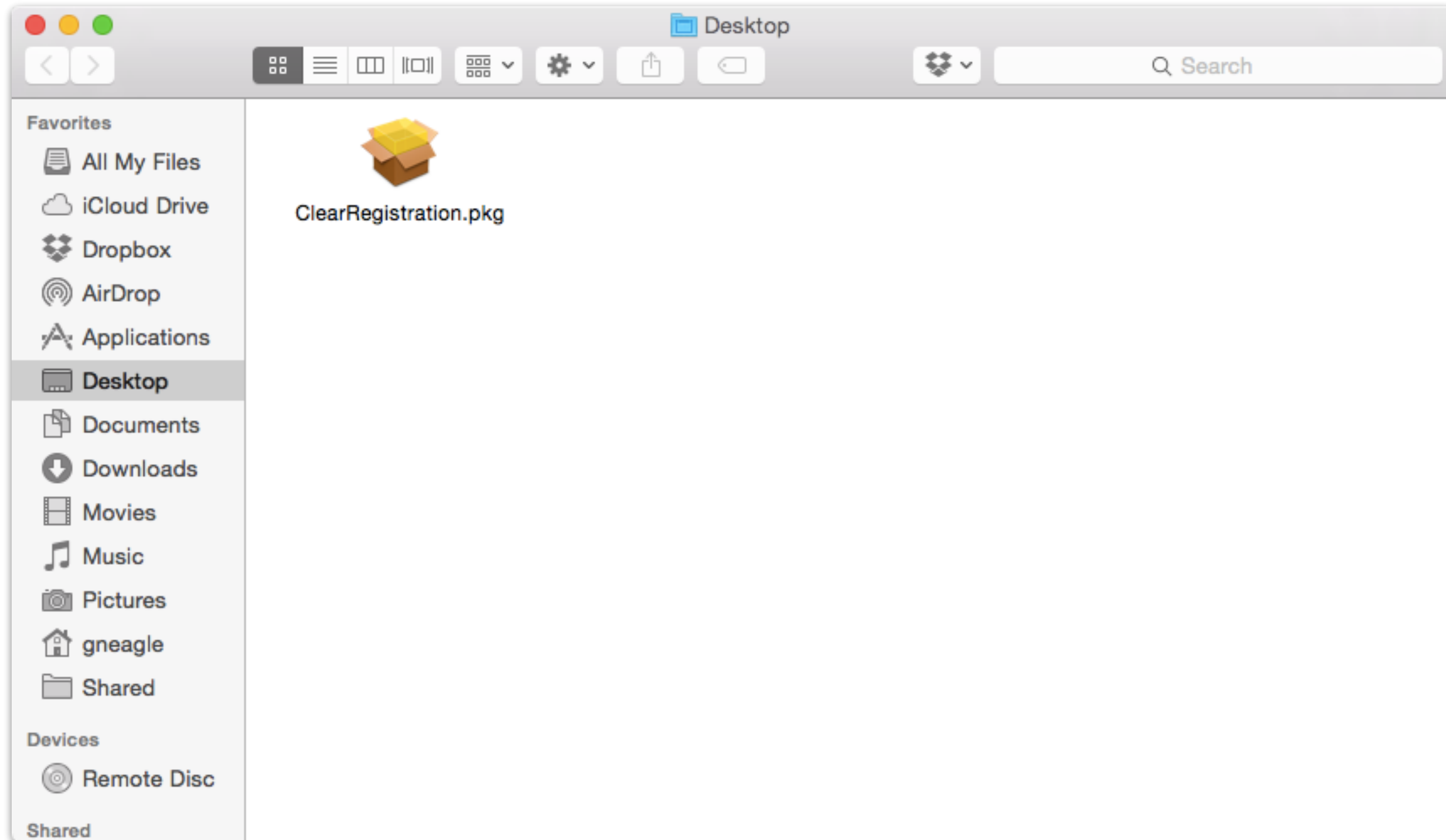
```
# build the pkg
```

```
% pkgbuild --root ClearRegistration \  
            --identifier com.mycompany.pkg.ClearRegistration \  
            --version 1.0 \  
            ClearRegistration.pkg
```

```
pkgbuild: Inferring bundle components from contents of fakeroot
```

```
pkgbuild: Wrote package to ClearRegistration.pkg
```

pkgbuild



Let's check our work...

```
% pkgutil --payload-files ClearRegistration.pkg  
.  
./Library  
./Library/Receipts  
./Library/Receipts/.SetupRegComplete  
./private  
./private/var  
./private/var/db  
./private/var/db/.AppleSetupDone
```

Or fancier...

```
% lsbom `pkgutil --bom ClearRegistration.pkg`  
. 40775 0/0  
./Library 40775 0/0  
./Library/Receipts 40775 0/0  
./Library/Receipts/.SetupRegComplete100664 0/0 0 4294967295  
./private 40775 0/0  
./private/var 40775 0/0  
./private/var/db40775 0/0  
./private/var/db/.AppleSetupDone 100664 0/0 0 4294967295
```


pkgbuild

Package from application



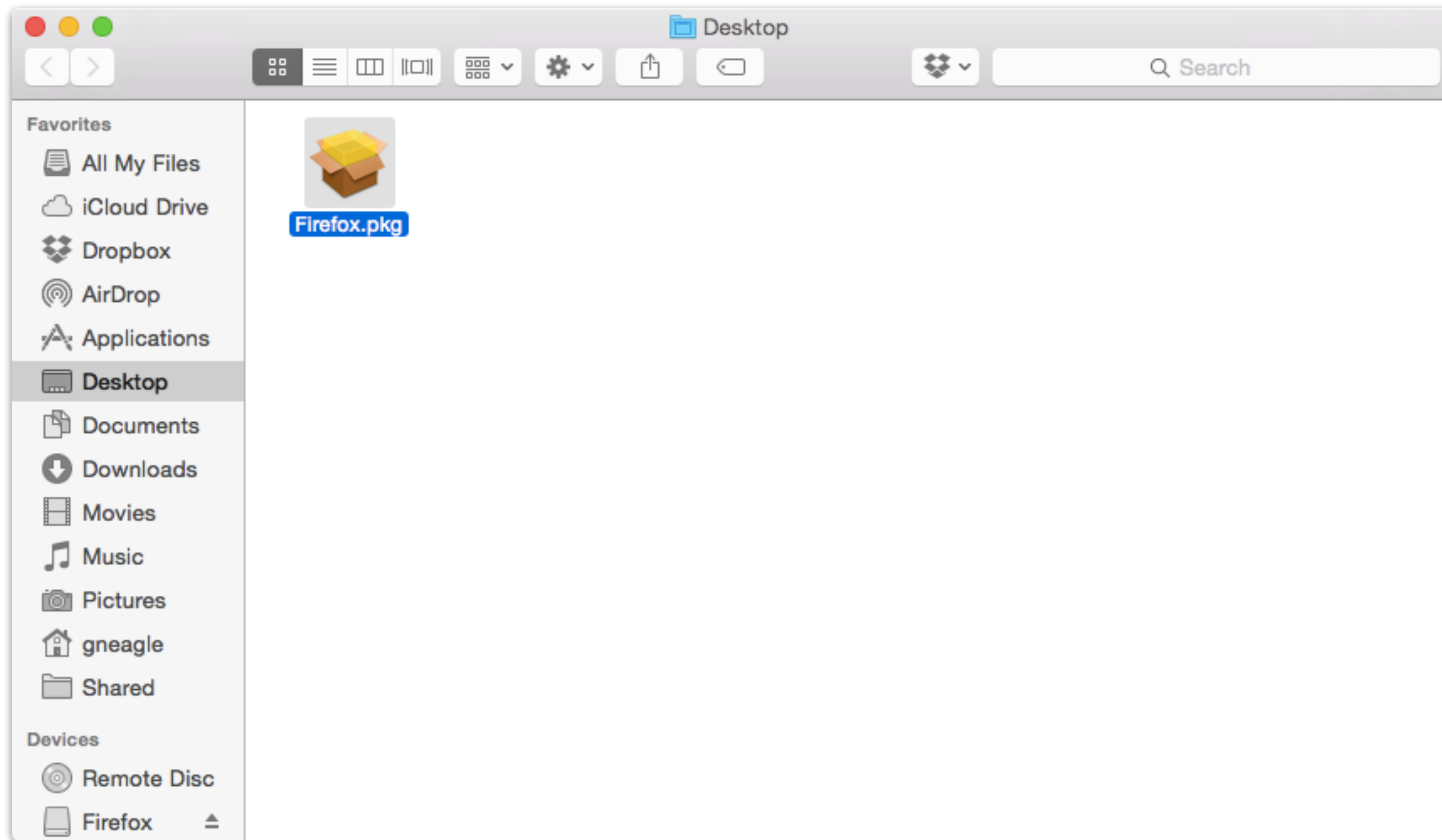
pkgbuild

```
% pkgbuild --component /Volumes/Firefox/Firefox.app \  
            --install-location /Applications \  
            Firefox.pkg
```

```
pkgbuild: Adding component at /Volumes/Firefox/Firefox.app
```

```
pkgbuild: Wrote package to Firefox.pkg
```

pkgbuild



Let's check our work...

```
% pkgutil --payload-files Firefox.pkg
```

```
.  
./Firefox.app  
./Firefox.app/Contents  
./Firefox.app/Contents/_CodeSignature  
./Firefox.app/Contents/_CodeSignature/CodeResources  
./Firefox.app/Contents/_CodeSignature/._CodeResources  
./Firefox.app/Contents/._CodeSignature  
./Firefox.app/Contents/Info.plist  
./Firefox.app/Contents/._Info.plist  
./Firefox.app/Contents/MacOS  
./Firefox.app/Contents/MacOS/crashreporter.app  
./Firefox.app/Contents/MacOS/crashreporter.app/Contents  
./Firefox.app/Contents/MacOS/crashreporter.app/Contents/_CodeSignature  
./Firefox.app/Contents/MacOS/crashreporter.app/Contents/_CodeSignature/CodeResources  
./Firefox.app/Contents/MacOS/crashreporter.app/Contents/_CodeSignature/._CodeResources  
./Firefox.app/Contents/MacOS/crashreporter.app/Contents/._CodeSignature  
./Firefox.app/Contents/MacOS/crashreporter.app/Contents/Info.plist  
./Firefox.app/Contents/MacOS/crashreporter.app/Contents/._Info.plist  
./Firefox.app/Contents/MacOS/crashreporter.app/Contents/MacOS  
./Firefox.app/Contents/MacOS/crashreporter.app/Contents/MacOS/crashreporter
```

Let's check our work...

```
% pkgutil --expand Firefox.pkg Firefox_pkg
% cat Firefox_pkg/PackageInfo
<?xml version="1.0" encoding="utf-8" standalone="no"?>
<pkg-info overwrite-permissions="true" relocatable="false" identifier="org.mozilla.firefox"
postinstall-action="none" version="47.0.0" format-version="2" generator-version="InstallCmds-554
(15F34)" install-location="/Applications" auth="root">
  <payload numberOfFiles="149" installKBytes="201018"/>
  <bundle path="./Firefox.app" id="org.mozilla.firefox" CFBundleShortVersionString="47.0"
CFBundleVersion="4716.6.4"/>
  <bundle-version>
    <bundle id="org.mozilla.firefox"/>
  </bundle-version>
  <upgrade-bundle>
    <bundle id="org.mozilla.firefox"/>
  </upgrade-bundle>
  <update-bundle/>
  <atomic-update-bundle/>
  <strict-identifier>
    <bundle id="org.mozilla.firefox"/>
  </strict-identifier>
  <relocate>
    <bundle id="org.mozilla.firefox"/>
  </relocate>
</pkg-info>
```

Let's check our work...

```
% pkgutil --expand Firefox.pkg Firefox_pkg
% cat Firefox_pkg/PackageInfo
<?xml version="1.0" encoding="utf-8" standalone="no"?>
<pkg-info overwrite-permissions="true" relocatable="false" identifier="org.mozilla.firefox"
postinstall-action="none" version="47.0.0" format-version="2" generator-version="InstallCmds-554
(15F34)" install-location="/Applications" auth="root">
  <payload numberOfFiles="149" installKBytes="201018"/>
  <bundle path="./Firefox.app" id="org.mozilla.firefox" CFBundleShortVersionString="47.0"
CFBundleVersion="4716.6.4"/>
  <bundle-version>
    <bundle id="org.mozilla.firefox"/>
  </bundle-version>
  <upgrade-bundle>
    <bundle id="org.mozilla.firefox"/>
  </upgrade-bundle>
  <update-bundle/>
  <atomic-update-bundle/>
  <strict-identifier>
    <bundle id="org.mozilla.firefox"/>
  </strict-identifier>
  <relocate>
    <bundle id="org.mozilla.firefox"/>
  </relocate>
</pkg-info>
```

Let's check our work...

```
% pkgutil --expand Firefox.pkg Firefox_pkg
% cat Firefox_pkg/PackageInfo
<?xml version="1.0" encoding="utf-8" standalone="no"?>
<pkg-info overwrite-permissions="true" relocatable="false" identifier="org.mozilla.firefox"
postinstall-action="none" version="47.0.0" format-version="2" generator-version="InstallCmds-554
(15F34)" install-location="/Applications" auth="root">
  <payload numberOfFiles="149" installKBytes="201018"/>
  <bundle path="./Firefox.app" id="org.mozilla.firefox" CFBundleShortVersionString="47.0"
CFBundleVersion="4716.6.4"/>
  <bundle-version>
    <bundle id="org.mozilla.firefox"/>
  </bundle-version>
  <upgrade-bundle>
    <bundle id="org.mozilla.firefox"/>
  </upgrade-bundle>
  <update-bundle/>
  <atomic-update-bundle/>
  <strict-identifier>
    <bundle id="org.mozilla.firefox"/>
  </strict-identifier>
  <relocate>
    <bundle id="org.mozilla.firefox"/>
  </relocate>
</pkg-info>
```

A word about owner,
group and mode

The archive contains information about location, kind, **owner, group, and mode** of each file and directory.

--ownership (recommended | preserve | preserve-other)

`--ownership recommended`

- Default
- `pkgbuild` applies Apple's recommended owner and group to files in the payload
- Generally, this is `root:wheel`

--ownership preserve

- `pkgbuild` uses the owner and group from the actual on-disk source files
- It's *your* responsibility to set these before building the package

`--ownership preserve-other`

- `pkgbuild` will apply the recommended UID and GID to those files that are owned by the user running `pkgbuild`, but leave other files unchanged.

But what about mode?

- Mode (read/write/execute) is always copied from on-disk files.
- It's *your* responsibility to set these to the right values *before* building the package
- If a package fixes user, group or mode in a postinstall script, it is doing it wrong.

pkgbuild review

pkgbuild review task

- Build a package that installs some application that is normally distributed via “drag-n-drop” disk image.

pkgbuild review task

- Build a package that installs some application that is normally distributed via “drag-n-drop” disk image.

```
% pkgbuild --component /Volumes/Google\ Chrome/Google\ Chrome.app --install-location /Applications ~/Desktop/GoogleChrome.pkg
```

```
% pkgbuild --component /Applications/Skype.app ~/Desktop/Skype.pkg
```

productbuild

Build a product archive for the OS X Installer or the Mac App Store

(Builds “distribution”-style packages)

productbuild

Convert a component pkg to a distribution pkg:

```
% cd ~/Desktop  
% mkdir ComponentPkgs  
% productbuild --package Firefox.pkg ComponentPkgs/Firefox.pkg  
productbuild: Wrote product to ComponentPkgs/Firefox.pkg
```

productbuild

Check our work:

```
% pkgutil --expand ComponentPkgs/Firefox.pkg ComponentPkgs/Firefox_pkg
% ls -1 ComponentPkgs/Firefox_pkg
Distribution
Firefox.pkg
```

productbuild

Create a distribution pkg from multiple component pkgs

```
productbuild --package Firefox.pkg \  
--package ClearRegistration.pkg \  
ComponentPkgs/Combo.pkg
```

productbuild

Check our work!

```
% pkgutil --expand ComponentPkgs/Combo.pkg ComponentPkgs/Combo_pkg  
% ls -1 ComponentPkgs/Combo_pkg  
ClearRegistration.pkg  
Distribution  
Firefox.pkg
```

productbuild review

- Using `pkgbuild` and `productbuild`, build a distribution package that installs both Firefox and Google Chrome (or any two apps of your choice)

- Using `pkgbuild` and `productbuild`, build a distribution package that installs both Firefox and Google Chrome (or any two apps of your choice)
- A solution:

```
% pkgbuild --component /Volumes/Firefox/Firefox.app --install-location /Applications Firefox.pkg  
pkgbuild: Adding component at /Volumes/Firefox/Firefox.app  
pkgbuild: Wrote package to Firefox.pkg
```

```
% pkgbuild --component /Volumes/Google\ Chrome/Google\ Chrome.app \  
--install-location /Applications GoogleChrome.pkg  
pkgbuild: Adding component at /Volumes/Google Chrome/Google Chrome.app  
pkgbuild: Wrote package to GoogleChrome.pkg
```

```
% productbuild --package Firefox.pkg --package GoogleChrome.pkg FreeBrowsers.pkg  
productbuild: Wrote product to FreeBrowsers.pkg
```

GUI Packaging



Apple
PackageMaker



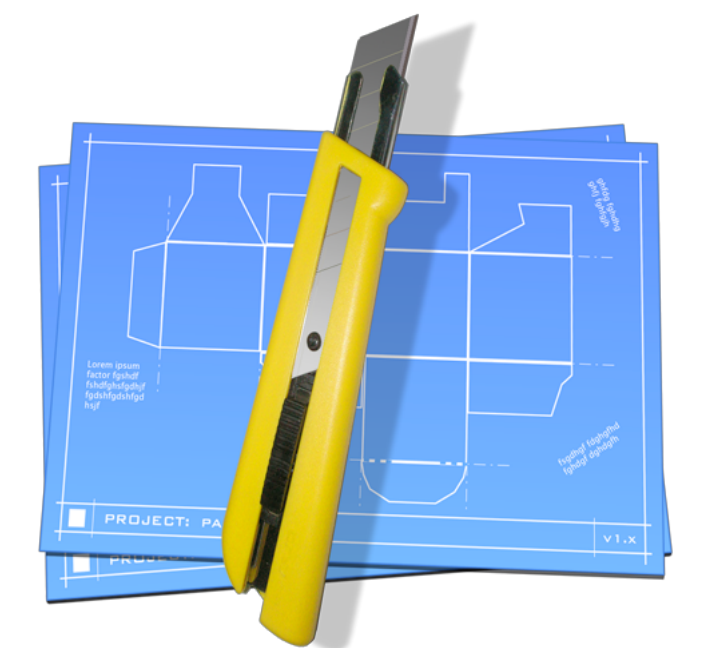
WhiteBox
Iceberg



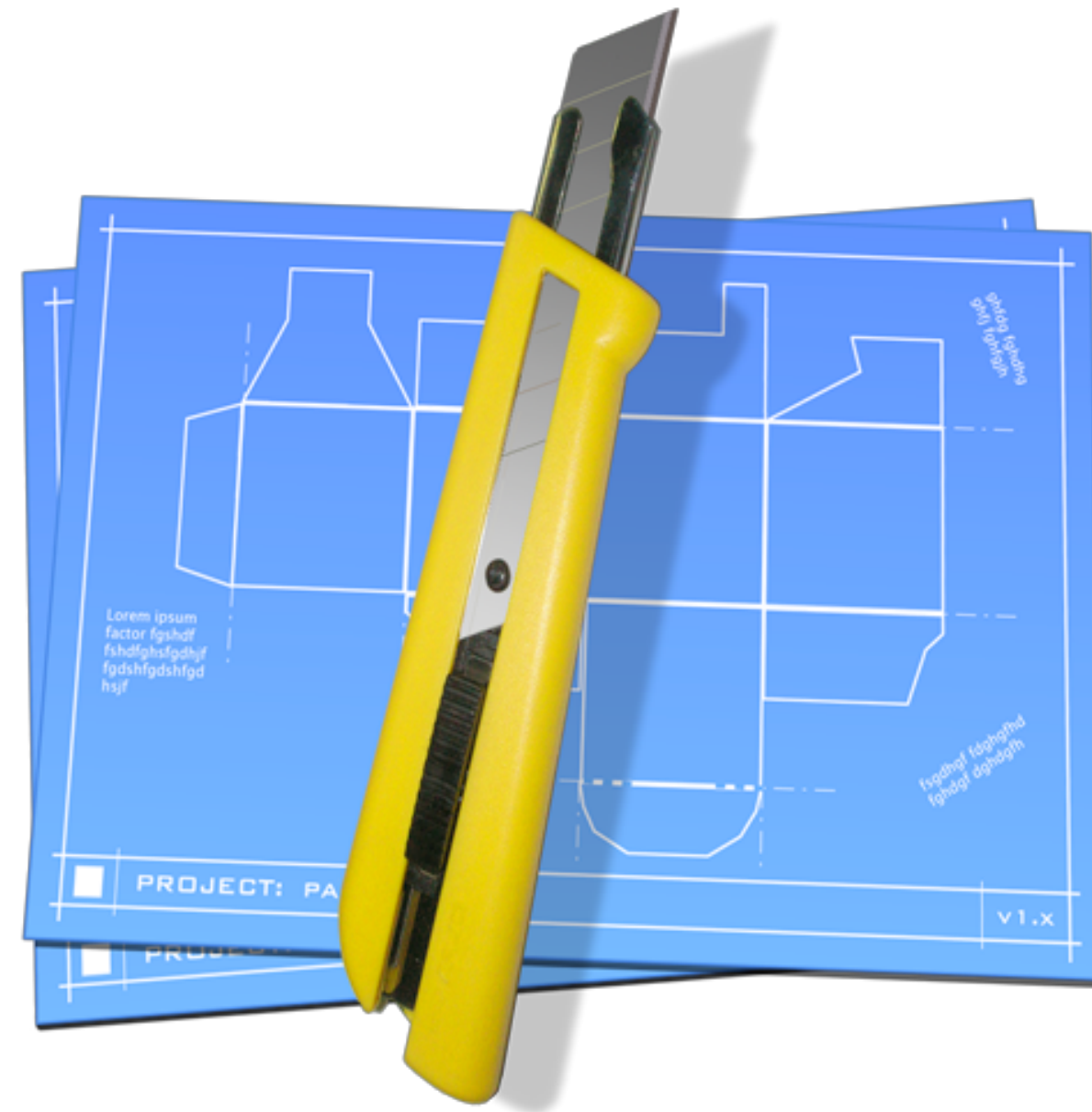
LANrev
InstallEase



JAMF
Composer



WhiteBox
Packages



WhiteBox Packages

<http://s.sudre.free.fr/Software/Packages/about.html>

We're going to build a simple package to bypass the OS X Registration / Setup Assistant that launches on first boot.

Two files need to be placed:

- `/private/var/db/.AppleSetupDone`
- `/Library/Receipts/.SetupRegComplete`

This is going to be slightly tricky because the two files in the payload are **invisible** in the Finder.

We need to create the payload folder hierarchy first, and then add in the **invisible** files.

New Project

Choose a template for your project:



Distribution



Raw Package



Raw Package

A Raw Package project lets you install files at specific locations.

Show on launch

Cancel

Previous

Next

New Project

Choose the name and location for your project:

Project Name:

Project Directory:

The project directory ~/BypassSetupAssistant/ will be created if necessary, and the project file BypassSetupAssistant.pkgproj will be created therein.

Show on launch

Build

Name:

Package file name

Path:

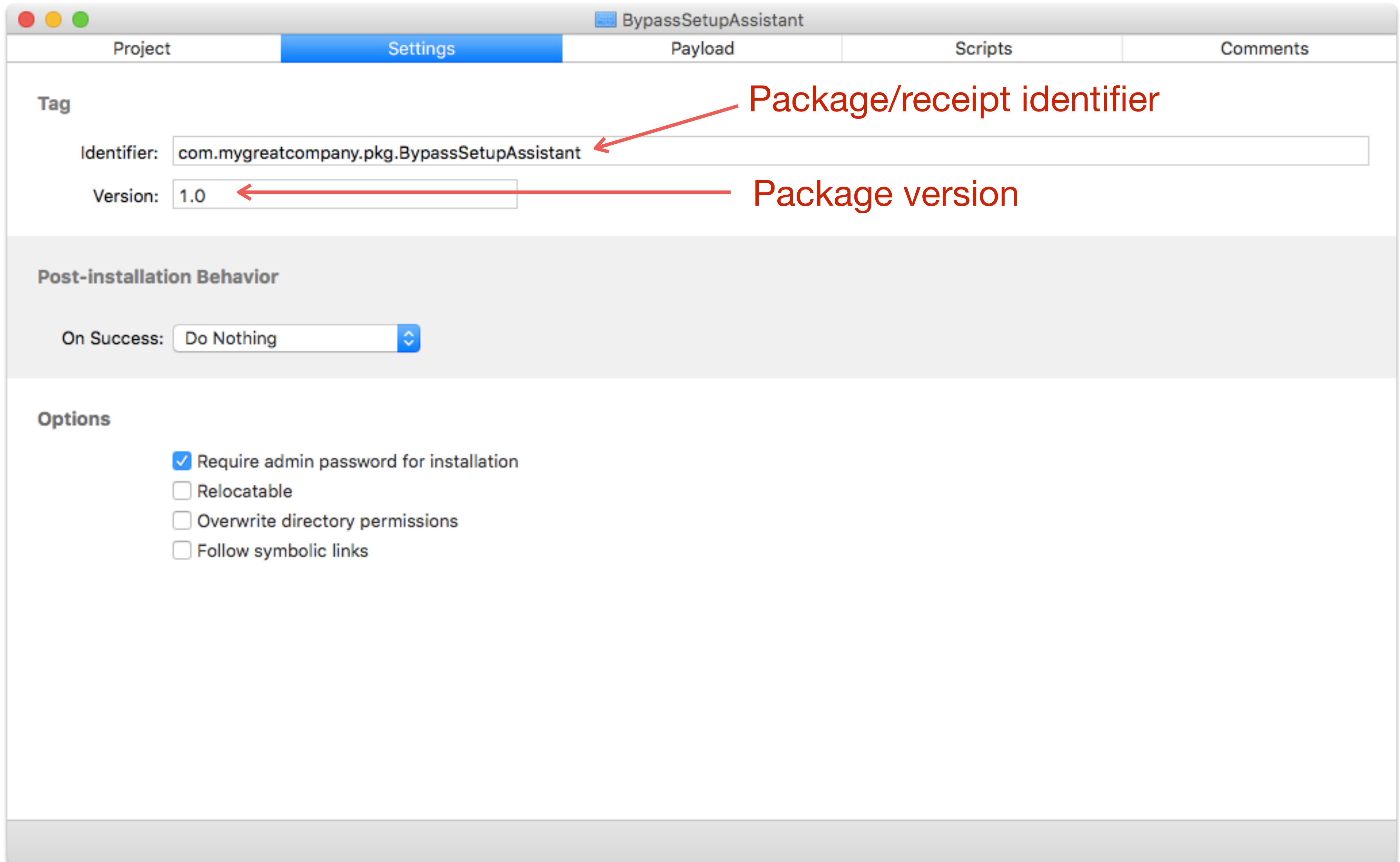
Reference Folder:

Exclusions:

Pattern	Kind	Regex
<input checked="" type="checkbox"/> Remove .DS_Store files		
<input checked="" type="checkbox"/> Remove .pbdevelopment files		
<input checked="" type="checkbox"/> Remove SCM metadata		
<input checked="" type="checkbox"/> Optimize nib files		
<input checked="" type="checkbox"/> Remove Resources Disabled folders		

+ -

Exclude items only from payloads



Settings

Type:

Default Destination: /

Contents

Filename	Owner	Group	Permissions
▼ /	root	wheel	drwxr-xr-x
▼ Applications	root	admin	drwxrwxr-x
▶ Library	root	wheel	drwxr-xr-x
▶ Users	root	admin	drwxr-xr-x

Empty Selection

Settings

Type: Internal

Default Destination: /

Set

Contents

Filename	Owner	Group	Permissions
/	root	wheel	drwxr-xr-x
App	root	admin	drwxrwxr-x
Lib	root	wheel	drwxr-xr-x
User	root	admin	drwxr-xr-x

Add Files...

New Folder

Delete...

Reveal in Finder

Expand

Expand All

Set as Default Location



Modified: -

Architectures: -

Kind: Standard folder

Reference: -

Source: -

Destination: /

Attributes

Name: /

Owner: root

Group: wheel

Access: drwxr-xr-x

	Read	Write	Exec	Bit	
Owner	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	SetUID
Group	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	SetGID
Others	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Sticky

+ -

Settings

Type: Internal

Default Destination: /

Set

Contents

Filename	Owner	Group	Permissions
▼ /	root	wheel	drwxr-xr-x
▼ Applications	root	admin	drwxrwxr-x
▶ Library	root	wheel	drwxr-xr-x
▶ private	root	wheel	drwxr-xr-x
▶ Users	root	admin	drwxr-xr-x

+ -

 **untitled folder**
Modified: -

Architectures: -

Kind: Custom folder

Reference: -

Source: -

Destination: /untitled folder

Attributes

Name: untitled folder

Owner: root

Group: wheel

Access: drwxr-xr-x

	Read	Write	Exec	Bit	
Owner	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	SetUID
Group	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	SetGID
Others	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Sticky

Settings

Type: Internal

Default Destination: /

Set

Contents

Filename	Owner	Group	Permissions
▼ /	root	wheel	drwxr-xr-x
▼ Applications	root	admin	drwxrwxr-x
▶ Library	root	wheel	drwxr-xr-x
▼ private	root	wheel	drwxr-xr-x
▼ var	root	wheel	drwxr-xr-x
▼ db	root	wheel	drwxr-xr-x
▶ Users	root	admin	drwxr-xr-x

+ -



db

Modified: -

Architectures: -

Kind: Custom folder

Reference: -

Source: -

Destination: /private/var/db

Attributes

Name: db

Owner: root

Group: wheel

Access: drwxr-xr-x

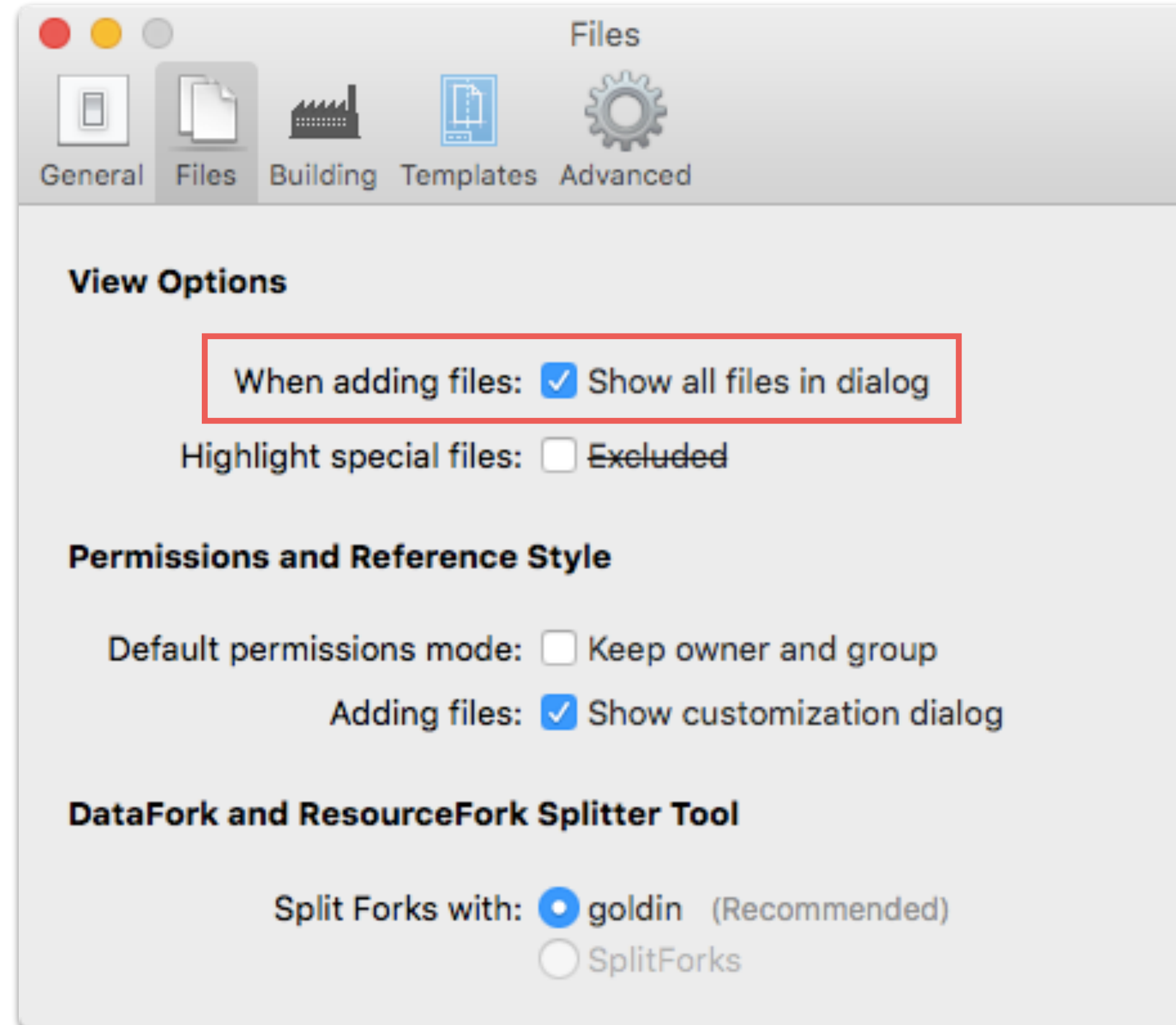
	Read	Write	Exec	Bit	
Owner	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	SetUID
Group	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	SetGID
Others	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Sticky

Create the two invisible files using the Terminal:

```
$ cd ~/BypassSetupAssistant/
```

```
$ touch .AppleSetupDone
```

```
$ touch .SetupRegComplete
```

Settings

Type: Internal

Default Destination: /

Set

Contents

Filename	Owner	Group	Permissions
/	root	wheel	drwxr-xr-x
Applications	root	admin	drwxrwxr-x
Library	root	wheel	drwxr-xr-x
private	root	wheel	drwxr-xr-x
var	root	wheel	drwxr-xr-x
db	root	wheel	drwxr-xr-x
Users	root	admin	drwxr-xr-x

Add Files...

New Folder

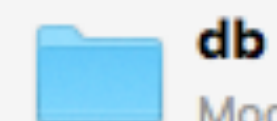
Delete...

Reveal in Finder

Expand

Expand All

Set as Default Location



db

Modified: -

Architectures: -

Kind: Custom folder

Reference: -

Source: -

Destination: /private/var/db

Attributes

Name: db

Owner: root

Group: wheel

Access: drwxr-xr-x

	Read	Write	Exec	Bit	
Owner	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	SetUID
Group	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	SetGID
Others	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Sticky

+ -

Project

Navigation icons: Home, Back, Forward, Grid, List, Columns, Search, Share, Back, Search

Comments

Settings

Type

Default Destination

Contents

Filename

- /
- Applications
- Library
- private
 - var
 - db
- Users

Favorites

Recents

Devices

Remote...

Shared

All...

Media

Music

Photos

Movies

Name

Date Modified

Size

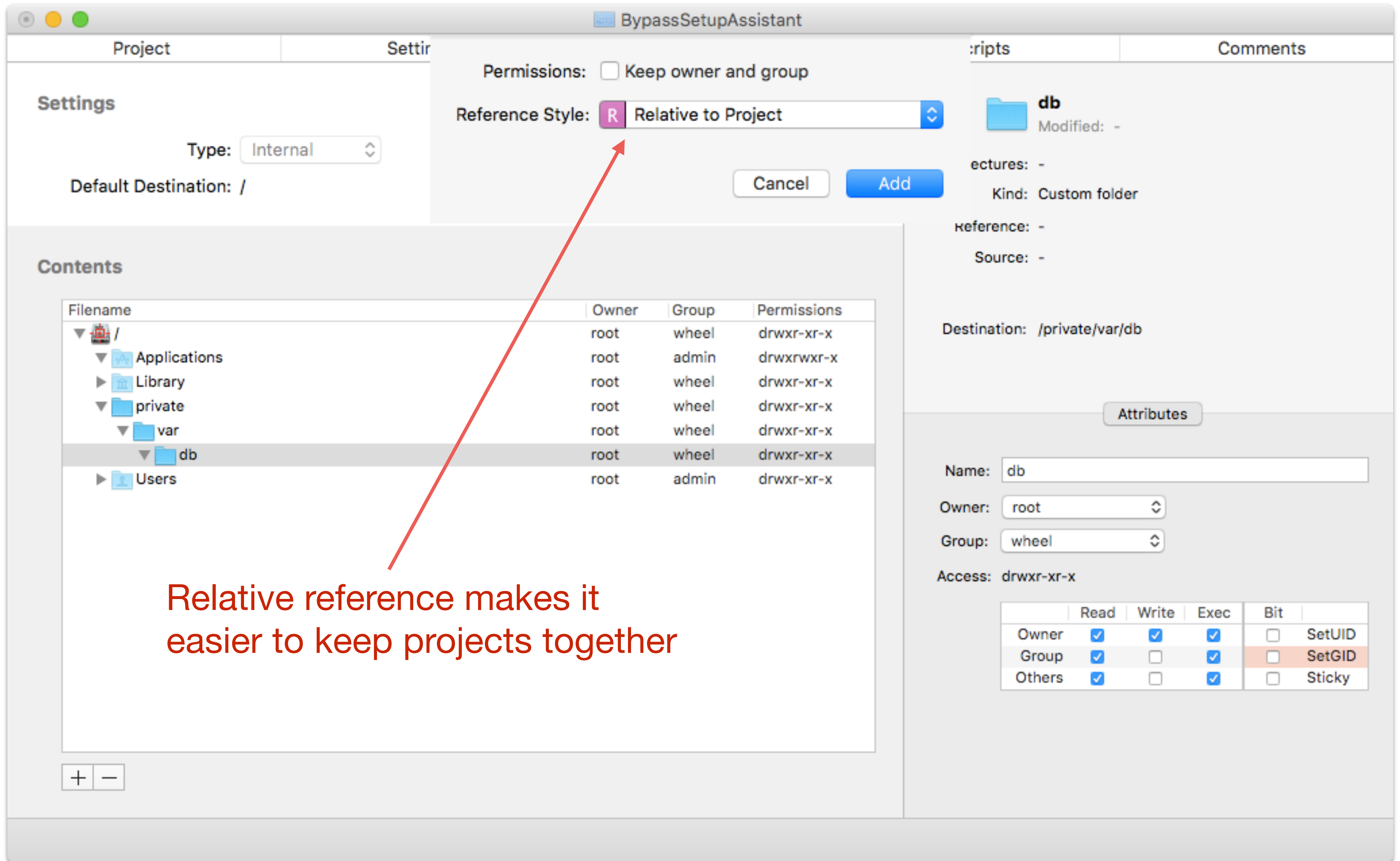
Name	Date Modified	Size
.AppleSetupDone	Today, 2:17 PM	Zero bytes
.SetupRegComplete	Today, 2:18 PM	Zero bytes
BypassSetupAssistant	Today, 1:45 PM	5 KB

Cancel

Add...

	read	write	Exec	Bit	
Owner	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	SetUID
Group	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	SetGID
Others	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Sticky

+ -



Relative reference makes it easier to keep projects together

Settings

Type: Internal

Default Destination: /

Set

Contents

Filename	Owner	Group	Permissions
▼ /	root	wheel	drwxr-xr-x
▼ Applications	root	admin	drwxrwxr-x
▶ Library	root	wheel	drwxr-xr-x
▼ private	root	wheel	drwxr-xr-x
▼ var	root	wheel	drwxr-xr-x
▼ db	root	wheel	drwxr-xr-x
▶ .AppleSetupDone	root	wheel	-rw-rw-r--
▶ Users	root	admin	drwxr-xr-x

+ -

**.AppleSetupDone**

Modified: Jun 21, 2016, 2:17 PM

Architectures: -

Kind: File

Reference: R Relative to Project

Source: .AppleSetupDone

Destination: /private/var/db/.AppleSetupDone

Attributes

Name: .AppleSetupDone

Owner: root

Group: wheel

Access: -rw-rw-r--

	Read	Write	Exec	Bit	
Owner	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	SetUID
Group	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	SetGID
Others	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Sticky

Settings

Type: Internal

Default Destination: /

Set

Contents

Filename	Owner	Group	Permissions
Library	root	wheel	drwxr-xr-x
Application Support	root	admin	drwxr-xr-x
Automator	root	wheel	drwxr-xr-x
Documentation	root	wheel	drwxr-xr-x
Extensions	root	wheel	drwxr-xr-x
Filesystems	root	wheel	drwxr-xr-x
Frameworks	root	wheel	drwxr-xr-x
Input Methods	root	wheel	drwxr-xr-x
Internet Plug-Ins	root	wheel	drwxr-xr-x
LaunchAgents	root	wheel	drwxr-xr-x
LaunchDaemons	root	wheel	drwxr-xr-x
PreferencePanels	root	wheel	drwxr-xr-x
Preferences	root	wheel	drwxr-xr-x
Printers	root	admin	drwxr-xr-x
PrivilegedHelperTools	root	wheel	drwxr-xr-x
QuickLook	root	wheel	drwxr-xr-x
QuickTime	root	wheel	drwxr-xr-x
Receipts	root	wheel	drwxr-xr-x
.SetupRegComplete	root	wheel	-rw-rw-r--

+ -

.SetupRegComplete

Modified: Jun 21, 2016, 2:18 PM

Architectures: -

Kind: File

Reference: R Relative to Project

Source: .SetupRegComplete

Destination: /Library/Receipts/.SetupRegComplete

Attributes

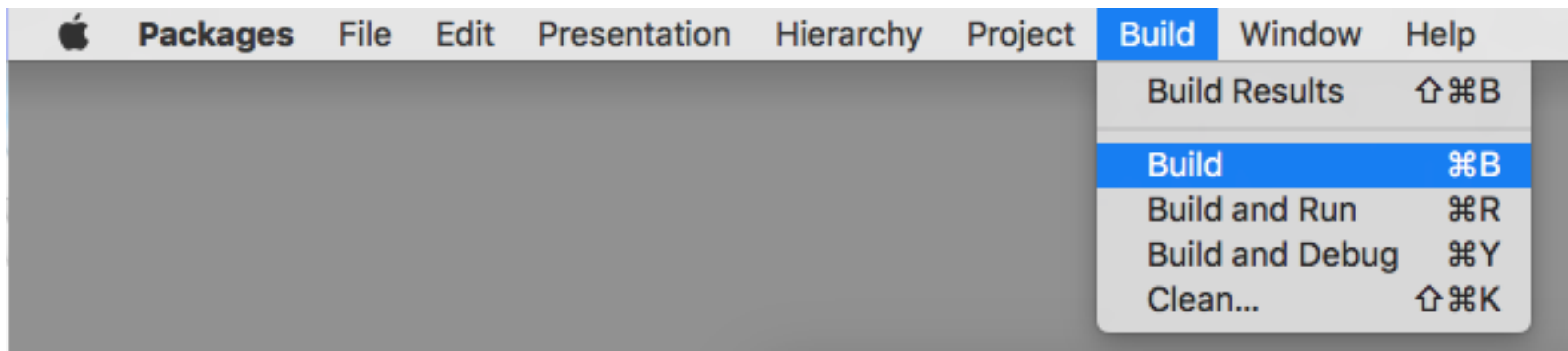
Name: .SetupRegComplete

Owner: root


Group: wheel

Access: -rw-rw-r--


	Read	Write	Exec	Bit	
Owner	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	SetUID
Group	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	SetGID
Others	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Sticky



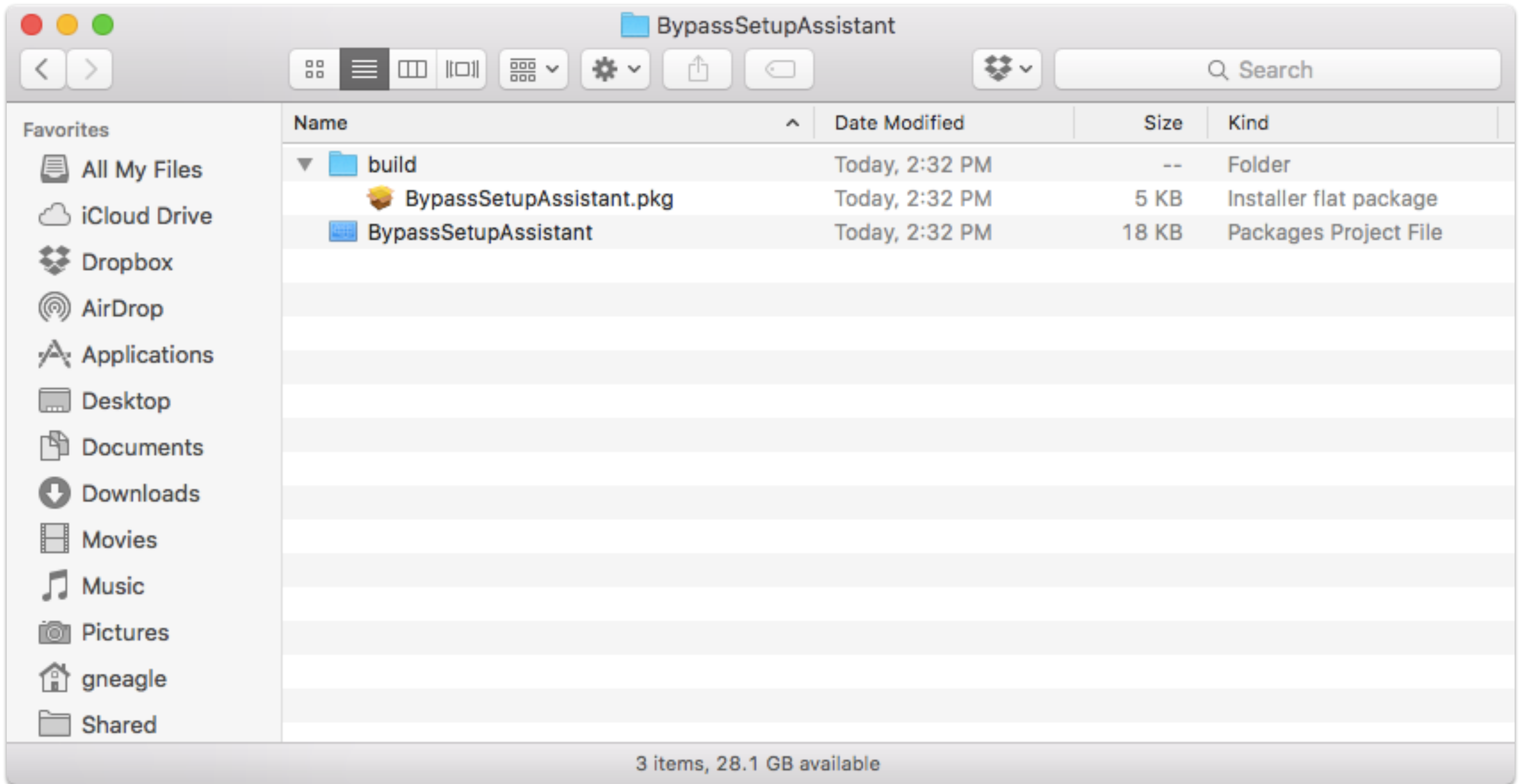
BypassSetupAssistant - Build Log

 **Build BypassSetupAssistant**
Project BypassSetupAssistant

- ✓ Create build folder
- ▼ ✓ Create payload
 - ✓ Assemble payload files
 - ✓ Split forks
 - ✓ Create bill of materials
 - ✓ Create pax archive
- ▼ ✓ Scripts payload
 - ✓ Assemble payload files
 - ✓ Split forks
 - ✓ Create pax archive
- ✓ Create PackageInfo document

 **Build Succeeded** 6/21/16, 2:32 PM
No issues

Build succeeded



BypassSetupAssistant

Navigation and utility icons including back, forward, view modes, settings, share, and search.

Favorites

- All My Files
- iCloud Drive
- Dropbox
- AirDrop
- Applications
- Desktop
- Documents
- Downloads
- Movies
- Music
- Pictures
- gneagle
- Shared

Name	Date Modified	Size	Kind
build	Today, 2:32 PM	--	Folder
BypassSetupAssistant.pkg	Today, 2:32 PM	5 KB	Installer flat package
BypassSetupAssistant	Today, 2:32 PM	18 KB	Packages Project File

3 items, 28.1 GB available

```
% pkgutil --payload-files ~/BypassSetupAssistant/build/BypassSetupAssistant.pkg
```

```
.  
./Library  
./Library/Receipts  
./Library/Receipts/.SetupRegComplete  
./private  
./private/var  
./private/var/db  
./private/var/db/.AppleSetupDone
```

BypassSetupAssistant

Search

Name	Date Modified	Size	Kind
build	Today, 2:32 PM	--	Folder
BypassSetupAssistant			


Favorites

- All My Files
- iCloud Drive
- Dropbox
- AirDrop
- Applications
- Desktop
- Documents
- Downloads
- Movies
- Music
- Pictures
- gneagle
- Shared

BypassSetupAssistant.pkg

Open with Installer

Open in Suspicious Package




BypassSetupAssistant.pkg

Installer Package

5 KB for package — Zero KB installed on disk

- Library
 - Receipts
 - .SetupRegComplete
- private
 - var
 - db
 - .AppleSetupDone

Quick Look generated by  Suspicious Package version 3.1 (217)

Demo

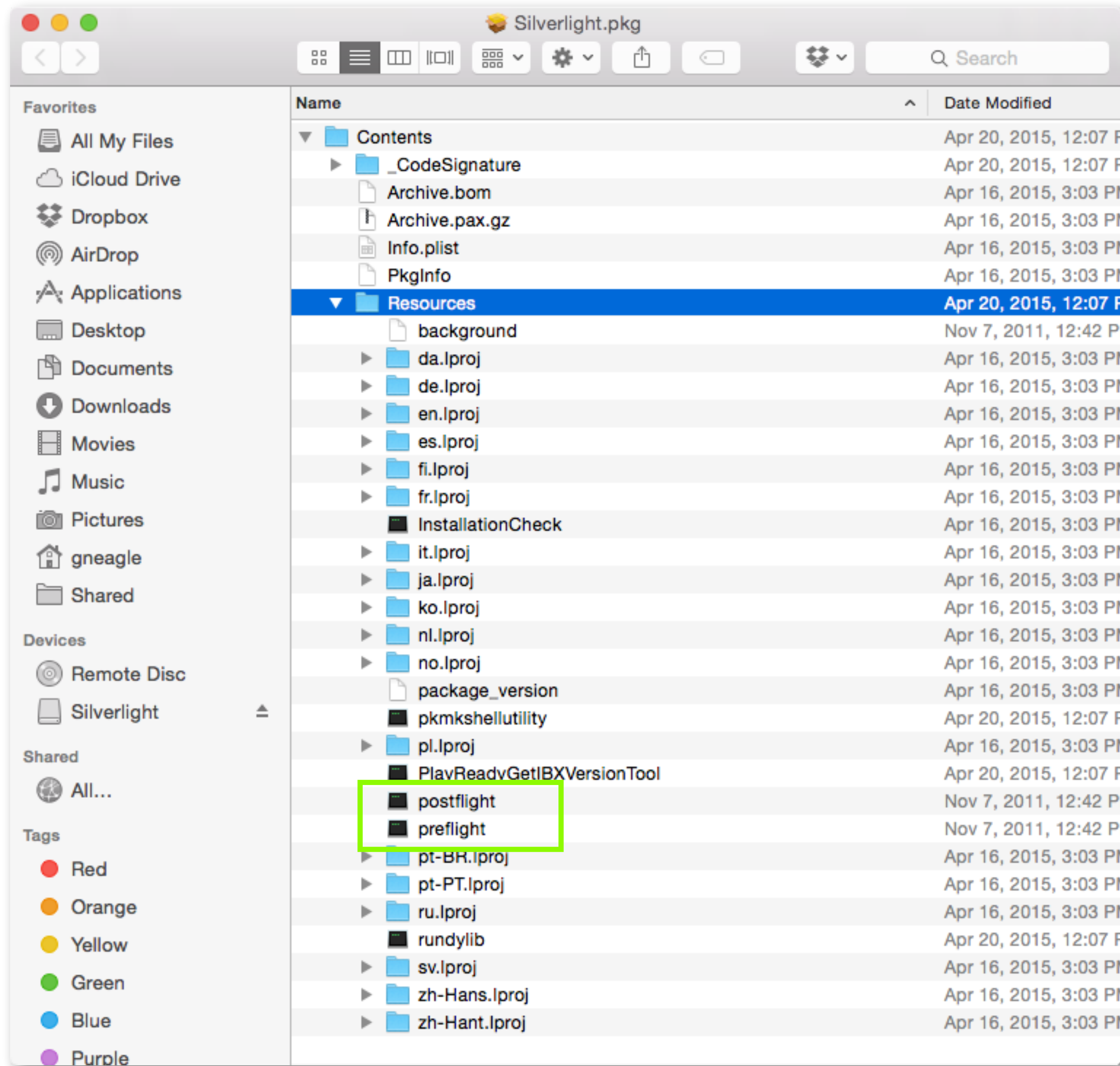
Exercise

Use Packages.app to create a installer package for Google Chrome.

Package Scripting

A package is an archive of files and directories.

A package may also
optionally run some scripts.



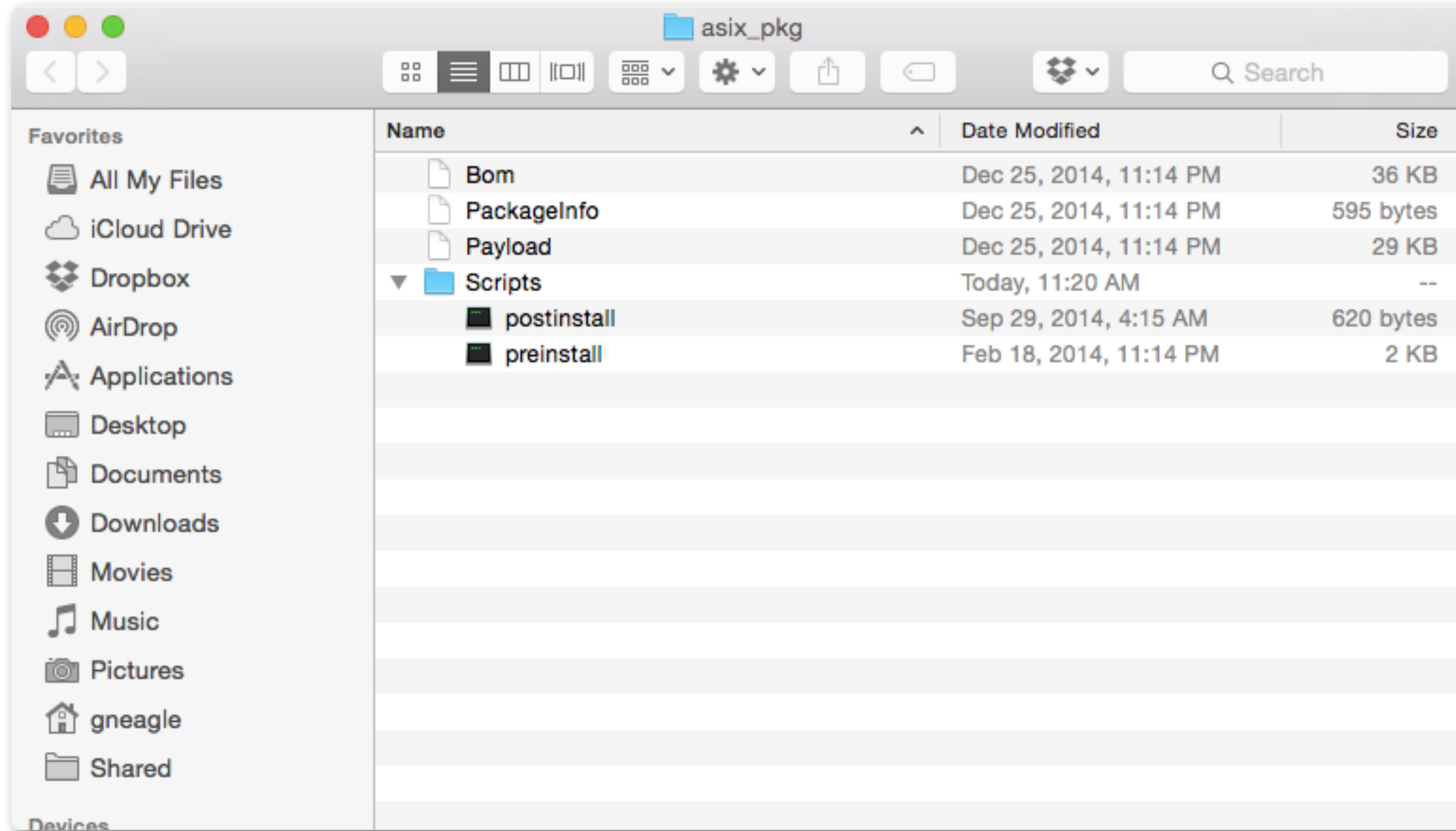
Bundle pkg scripts

- preflight
- preinstall
- preupgrade
- postupgrade
- postinstall
- postflight

Bundle pkg scripts

preflight	Executed before the installation process.
preinstall	Executed before installing a single package.
preupgrade	Executed before upgrading a single package.
postinstall	Executed after installing a single package.
postupgrade	Executed after upgrading a single package.
postflight	Executed after the installation process.

Flat component package



Flat pkg scripts

- ~~• preflight~~
- preinstall
- ~~• preupgrade~~
- ~~• postupgrade~~
- postinstall
- ~~• postflight~~

Script arguments

\$1	The full path to the installation package.
\$2	The full path to the installation destination.
\$3	The mountpoint of the destination volume.
\$4	The root directory for the current System folder

Environment vars

<code>\$PACKAGE_PATH</code>	Same as the \$1 argument.
<code>\$SCRIPT_NAME</code>	The name of the file being executed.
<code>\$COMMAND_LINE_INSTALL</code>	Set when installing via /usr/sbin/installer

Test script

```
#!/bin/sh
```

```
/usr/bin/logger "Positional argument #1: $1"
```

```
/usr/bin/logger "Positional argument #2: $2"
```

```
/usr/bin/logger "Positional argument #3: $3"
```

```
/usr/bin/logger "Positional argument #4: $4"
```

```
/usr/bin/logger ""
```

```
/usr/bin/logger "Env var PACKAGE_PATH          : $PACKAGE_PATH"
```

```
/usr/bin/logger "Env var SCRIPT_NAME           : $SCRIPT_NAME"
```

```
/usr/bin/logger "Env var COMMAND_LINE_INSTALL: $COMMAND_LINE_INSTALL"
```



```
% sudo tail /var/log/system.log
```

```
Jun 26 14:49:45 redacted gneagle[11744]: Positional argument #1: /Users/gneagle/Desktop/ScriptTest.pkg  
Jun 26 14:49:45 redacted gneagle[11745]: Positional argument #2: /  
Jun 26 14:49:45 redacted gneagle[11746]: Positional argument #3: /  
Jun 26 14:49:45 redacted gneagle[11747]: Positional argument #4: /  
Jun 26 14:49:45 redacted gneagle[11749]: Env var PACKAGE_PATH : /Users/gneagle/Desktop/ScriptTest.pkg  
Jun 26 14:49:45 redacted gneagle[11751]: Env var SCRIPT_NAME : postinstall  
Jun 26 14:49:45 redacted gneagle[11752]: Env var COMMAND_LINE_INSTALL: 1
```

postflight script

```
#!/bin/sh
```

```
touch /private/var/db/.AppleSetupDone
```

```
touch /Library/Receipts/.SetupRegComplete
```

postflight script

```
#!/bin/sh
```

```
touch $3/private/var/db/.AppleSetupDone
```

```
touch $3/Library/Receipts/.SetupRegComplete
```

postflight script

```
#!/bin/sh
```

```
touch "$3/private/var/db/.AppleSetupDone"
```

```
touch "$3/Library/Receipts/.SetupRegComplete"
```

postflight script

```
#!/bin/sh
```

```
touch "$3/private/var/db/.AppleSetupDone"
```

```
touch "$3/Library/Receipts/.SetupRegComplete"
```

postflight script

```
#!/bin/sh
```

```
/usr/bin/touch "$3/private/var/db/.AppleSetupDone"
```

```
/usr/bin/touch "$3/Library/Receipts/.SetupRegComplete"
```

A package is an archive of files and directories.

A package may also optionally run some scripts.

Pro Tip:

Don't use a script when a
payload will work.


```
% pkgutil --payload-files SuppressSetupAssistant.pkg  
.  
./Library  
./Library/Receipts  
./Library/Receipts/.SetupRegComplete  
./private  
./private/var  
./private/var/db  
./private/var/db/.AppleSetupDone
```

Common scripting errors

#1: Forgetting to refer
to the target volume

```
#!/bin/bash
sudo rm -rf /Library/Application\ Support/Adobe/Shockwave\ 11
```

```
#!/bin/sh

# Script to delete old versions
# Copyright (c) 2006 Microsoft Corporation

#####
# Remove WPFfe
rm -rf /Library/Internet\ Plug-Ins/WPFfe.plugin/
rm -rf /Library/Internet\ Plug-Ins/Silverlight.plugin/

rm -rf /Library/Receipts/Silverlight.pkg/
rm -rf /Library/Receipts/Silverlight_W2.pkg/
rm -rf /Library/Receipts/Silverlight_W2_MIX.pkg/
```

#2: Interacting with the Finder or Dock

```
osascript -e "tell application \"Finder\" to make new alias at \"$aliaspath\"  
to file \"$filepath\""
```

#3: Assuming there is a GUI user

```
#!/bin/bash
#get the user and group
thisuser="$SUDO_USER"
if [ -z "$thisuser" ]; then
    thisuser="$USER"
    group=$(id -gn "$thisuser")
else
    sudo_user_name="$SUDO_USER"
    group=$(id -gn "$sudo_user_name")
fi

#set proper permissions on the prefpane
sudo chmod -R 555 "/Library/PreferencePanes/utcore-prefpane.prefPane"
#set ownership of the prefpane
sudo chown -R "$thisuser":"$group" "/Library/PreferencePanes/utcore-prefpane.prefPane"

#adding the UTCore login item
sudo su "$thisuser" -c "defaults write loginwindow AutoLaunchedApplicationDictionary -
array-add '<dict><key>Hide</key><true/><key>Path</key><string>/Library/PreferencePanes/
utcore-prefpane.prefPane/Contents/Resources/UTCoreHelper.app</string></dict>'"
```


#4: Other assumptions about the
environment

```
#!/bin/bash

function MoveReaderToTrash {

    newReaderApp=$readerApp
    while [ 1 ]
    do
        if [ -e ~/.Trash/"$newReaderApp" ]
        then
            echo "$newReaderApp exists in Trash. Retrying with a new name."
            sleep 1
            time_to_be_attached=`date "+%k-%M-%S"`
            newReaderApp="Adobe Reader"$time_to_be_attached".app"
        else
            echo "Moving existing $readerApp to Trash as $newReaderApp"
            mv "$installation_path/$readerApp" ~/.Trash/"$newReaderApp"
            if [ $? -ne 0 ]
            then
                echo "ERROR: Could not move existing $readerApp to Trash."
                exit -1
            else
                exit 0
            fi
        fi
    done
}
```

Scripting recommendations

#1: Avoid scripts.

#2: Don't use a script to fix
payload mistakes

#3: Use \$3 to target files
on the target volume

#4: Don't forget that target volume name might have spaces

#5: Test via CLI

#6: Test installing to a volume
other than the current startup
volume

#7: Test installing at the
loginwindow

Stupid Package Tricks

Payload-free packages
(bundle-style)

A package is an archive of files and directories.

A package may also
optionally run some scripts.



PSU MacAdmins.pkg

Name	Date Modified	Size	Kind
▼ Contents	Today, 2:02 PM	--	Folder
Archive.bom	Jun 12, 2007, 5:48 PM	35 KB	Document
Archive.pax.gz	Jun 12, 2007, 5:48 PM	88 bytes	GZip archive
Info.plist	Today, 2:03 PM	1 KB	property list
PkgInfo	Jun 12, 2007, 5:48 PM	9 bytes	TextEd...ument
▼ Resources	Today, 2:04 PM	--	Folder
▶ en.lproj	Jun 10, 2009, 5:06 PM	--	Folder
package_version	Jun 12, 2007, 5:48 PM	17 bytes	TextEd...ument
postflight	Today, 2:04 PM	17 bytes	Unix E...le File
preflight	Today, 2:04 PM	17 bytes	Unix E...le File

```
% lsbom PSU\ MacAdmins.pkg/Contents/Archive.bom  
. 40755501/20
```




PSU MacAdmins.pkg

Name	Date Modified	Size	Kind
▼ Contents	Today, 2:02 PM	--	Folder
Archive.bom	Jun 12, 2007, 5:48 PM	35 KB	Document
Archive.pax.gz	Jun 12, 2007, 5:48 PM	88 bytes	GZip archive
Info.plist	Today, 2:03 PM	1 KB	property list
PkgInfo	Jun 12, 2007, 5:48 PM	9 bytes	TextEd...ument
▼ Resources	Today, 2:04 PM	--	Folder
▶ en.lproj	Jun 10, 2009, 5:06 PM	--	Folder
package_version	Jun 12, 2007, 5:48 PM	17 bytes	TextEd...ument
postflight	Today, 2:04 PM	17 bytes	Unix E...le File
preflight	Today, 2:04 PM	17 bytes	Unix E...le File

managingosx.wordpress.com/2010/02/18/payload-free

My Sites Reader Follow Like Reblog

MANAGING OS X

<https://managingosx.wordpress.com/2010/02/18/payload-free-package-template/>

PAYLOAD-FREE PACKAGE TEMPLATE

February 18, 2010
GregN
[Edit](#)

Payload-free packages are packages that exist only to run scripts. These are typically scripts you need to run a single time to perform a configuration task. They don't install any files, thus the name "payload-free". They are useful in workflows where you are installing packages like InstaDMG or any of the software distribution mechanisms that use packages. You can wrap your script into a payload-free package, and use the same mechanism you use to install software to deliver and run your script.

[Here](#) is a template payload-free package you can use to create custom payload-free packages for your own use.

To use it, duplicate it, rename it, and edit the following files:

`payload-free.pkg/Contents/Info.plist` – edit the `CFBundleIdentifier`, `CFBundleShortVersion`, and possibly the `IFPkgFlagRestartAction`.

`payload-free.pkg/Contents/Resources/en.lproj/Description.plist` – edit the `IFPkgDescriptionDescription` and `IFPkgDescriptionTitle`. Repeat for other languages if desired.

And, why, exactly?

Run a script as root, to...

Make a configuration change

```
#!/bin/sh
```

```
# setup $TARGETVOL so defaults is not unhappy
```

```
if [ "$3" == "/" ]; then
```

```
    TARGETVOL=""
```

```
else
```

```
    TARGETVOL="$3"
```

```
fi
```

```
# turn off Bluetooth
```

```
BLUETOOTHPREFS="$TARGETVOL/Library/Preferences/com.apple.Bluetooth"
```

```
/usr/bin/defaults write "$BLUETOOTHPREFS" ControllerPowerState -int 0
```

```
if [ "$3" == "/" ]; then
```

```
    # we're installing on the startup disk, so
```

```
    # restart bluetooth daemon to pick up our changes
```

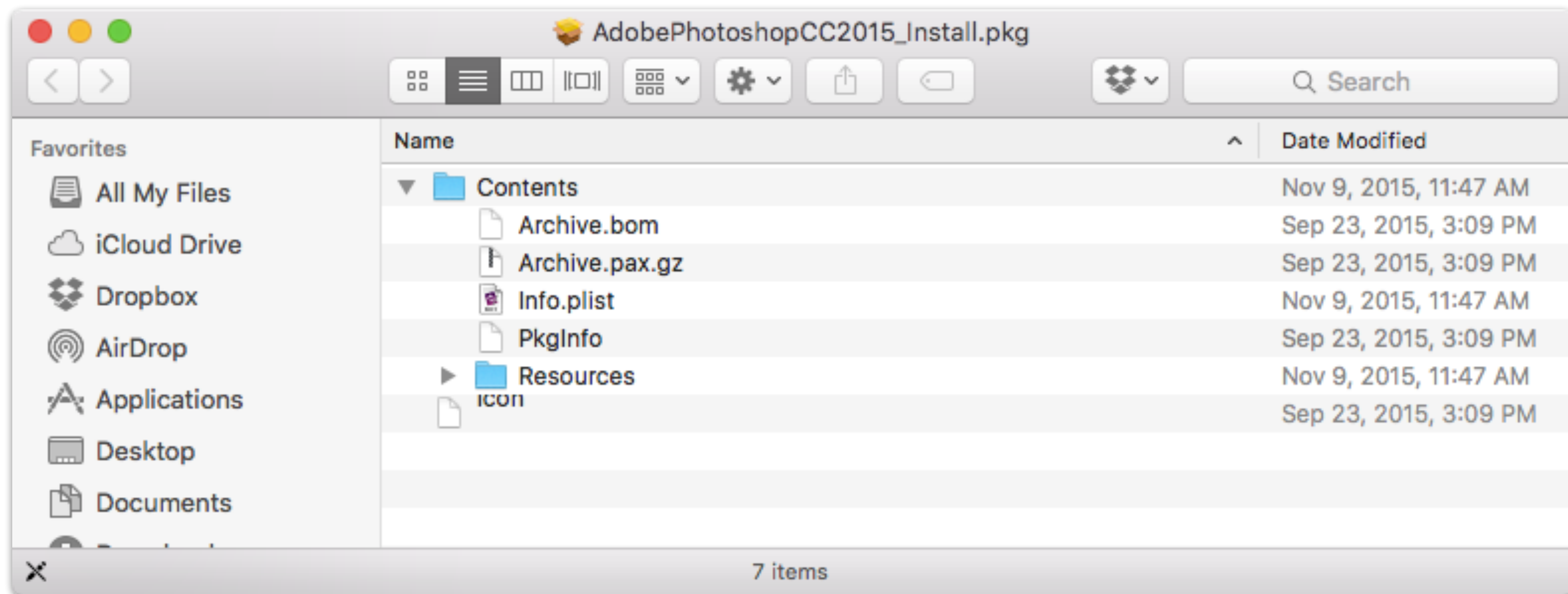
```
    /usr/bin/killall -HUP blued
```

```
fi
```

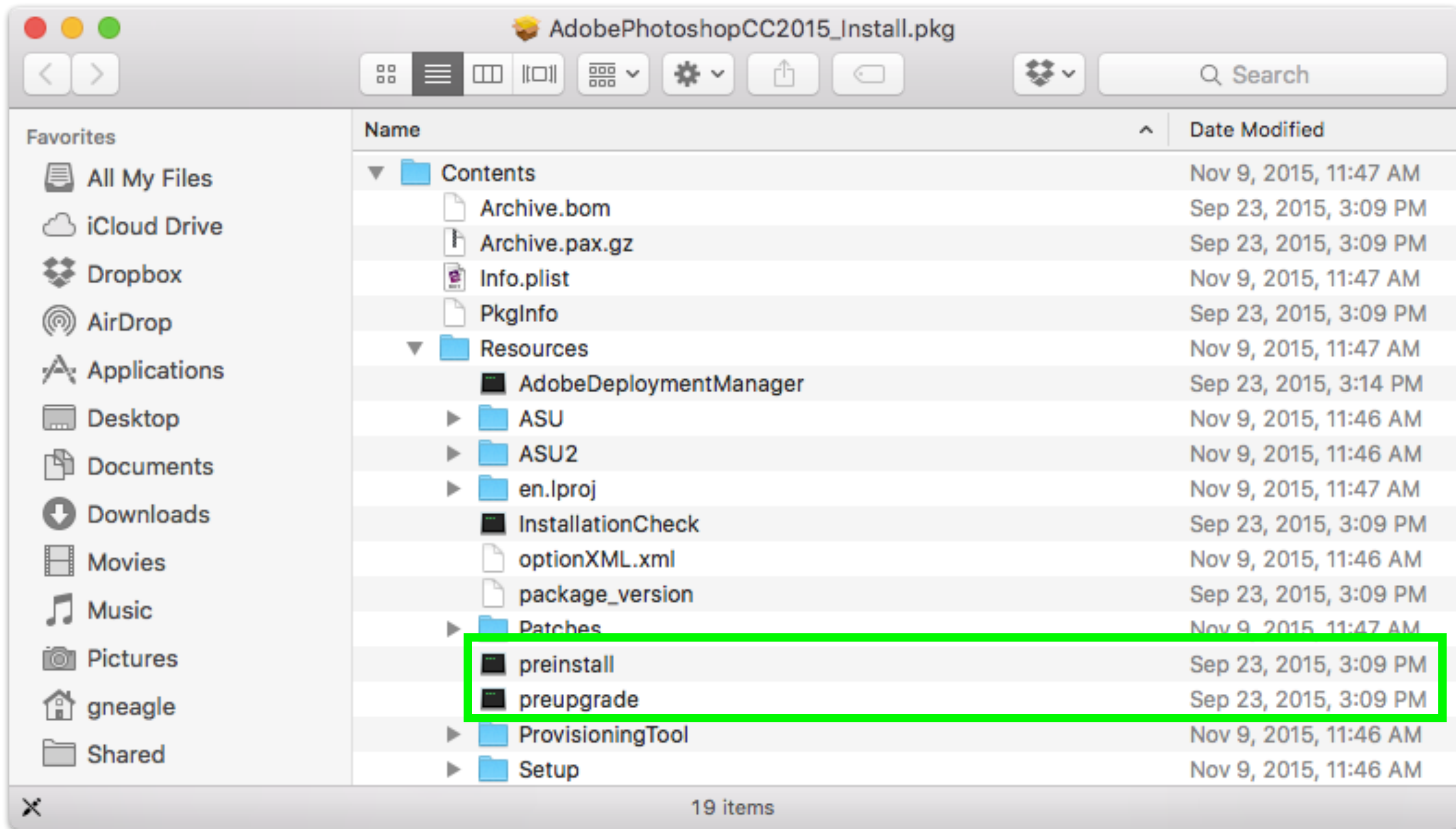
Install a printer

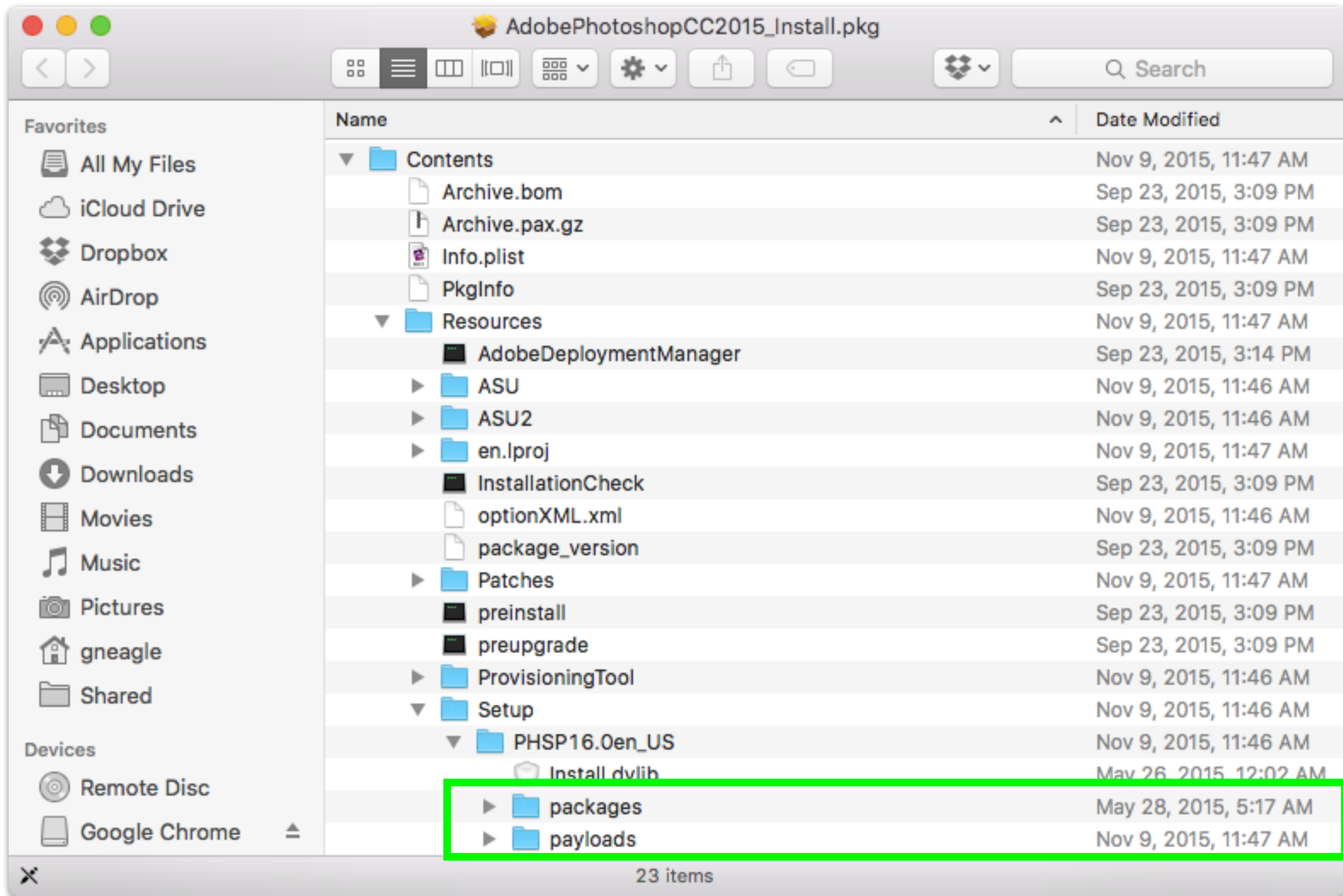
Uninstall software

Install software :- (



```
% lsbom AdobePhotoshopCC2015_Install.pkg/Contents/Archive.bom  
. 40755 501/20
```





Flat payload-free packages

```
#!/bin/sh
```

```
# setup $TARGETVOL so defaults is not unhappy
```

```
if [ "$3" == "/" ]; then
```

```
    TARGETVOL=""
```

```
else
```

```
    TARGETVOL="$3"
```

```
fi
```

```
# turn off Bluetooth
```

```
BLUETOOTHPREFS="$TARGETVOL/Library/Preferences/com.apple.Bluetooth"
```

```
/usr/bin/defaults write "$BLUETOOTHPREFS" ControllerPowerState -int 0
```

```
if [ "$3" == "/" ]; then
```

```
    # we're installing on the startup disk, so
```

```
    # restart bluetooth daemon to pick up our changes
```

```
    /usr/bin/killall -HUP blued
```

```
fi
```

```
untitled 11025
1  #!/bin/sh
2  -
3  # setup $TARGETVOL so defaults is not unhappy
4  if [ "$3" == "/" ]; then
5      TARGETVOL=""
6  else
7      TARGETVOL="$3"
8  fi
9  -
10 # turn off Bluetooth
11 BLUETOOTHPREFS="$TARGETVOL/Library/Preferences/com.apple.Bluetooth"
12 /usr/bin/defaults write "$BLUETOOTHPREFS" ControllerPowerState -int 0
13 -
14 if [ "$3" == "/" ]; then
15     # we're installing on the startup disk, so
16     # restart bluetooth daemon to pick up our changes
17     /usr/bin/killall -HUP blue
18 fi
```

Line: 18 Column: 3 Plain Text Soft Tabs: 4

Save As:



Tags:

Navigation bar with back/forward arrows, view icons (grid, list, compare), a folder icon labeled "Scripts", a search field with a magnifying glass icon and the text "Search", and a dropdown arrow.

- Favorites
- Shared
 - All...
- Devices
 - Remote...

Name	Date Modified

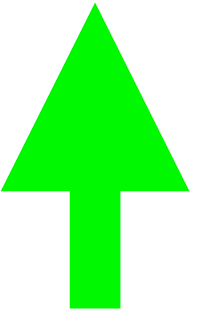
File Encoding:

Line Endings:

New Folder Cancel **Save**


```
% chmod a+x Scripts/postinstall
```

```
% pkgbuild --nopayload \  
--scripts Scripts \  
--identifier "org.psumacadmins.bluetoothconfig" \  
--version 1.0 TurnBluetoothOff.pkg  
pkgbuild: Wrote package to TurnBluetoothOff.pkg
```



```
% sudo installer -pkg TurnBluetoothOff.pkg -target /  
installer: Package name is TurnBluetoothOff  
installer: Installing at base path /  
installer: The install was successful.
```

```
% pkgbuild --nopayload \  
--scripts Scripts \  
--identifier "org.psumacadmins.bluetoothconfig" \  
--version 1.0 TurnBluetoothOff.pkg
```

```
% pkgbuild --nopayload \  
--scripts Scripts \  
--identifier "org.psumacadmins.bluetoothconfig" \  
--version 1.0 TurnBluetoothOff.pkg
```

```
% pkgutil --info org.psumacadmins.bluetoothconfig  
No receipt for 'org.psumacadmins.bluetoothconfig' found at '/'.
```

Flat

pseudo-payload-free packages

```
% mkdir Empty
```

```
% pkgbuild --root Empty \  
            --scripts Scripts \  
            --identifier "org.psumacadmins.bluetoothconfig" \  
            --version 1.0 TurnBluetoothOff.pkg
```

```
pkgbuild: Inferring bundle components from contents of Empty
```

```
pkgbuild: Adding top-level postinstall script
```

```
pkgbuild: Wrote package to TurnBluetoothOff.pkg
```

```
% sudo installer -pkg TurnBluetoothOff.pkg -target /  
installer: Package name is TurnBluetoothOff  
installer: Installing at base path /  
installer: The install was successful.
```



```
% pkgutil --info org.psumacadmins.bluetoothconfig
package-id: org.psumacadmins.bluetoothconfig
version: 1.0
volume: /
location: /
install-time: 1435880253
```

Installer behaviors

Installing a package

- pre-xxx scripts run if present
- Files are copied from the (compressed) archive to the correct location on the target volume
- Owner, group, and mode are set to values contained within the package
- post-xxx scripts run if present

Upgrading a package

- pre-xxx scripts run if present
- Files are copied from the (compressed) archive to the correct location on the target volume
- Owner, group, and mode are set to values contained within the package
- Any files that were in the previous version of the package but are not in this version are removed
- post-xxx scripts run if present

What makes Installer decide
to do “upgrade” behavior?

Matching package identifiers.

```
% sudo pkgutil --forget org.psumacadmins.bluetoothconfig  
Forgot package 'org.psumacadmins.bluetoothconfig' on '/'.
```

Package Signatures

Package Signing

Packages can be **signed** to provide proof against **potential tampering**.

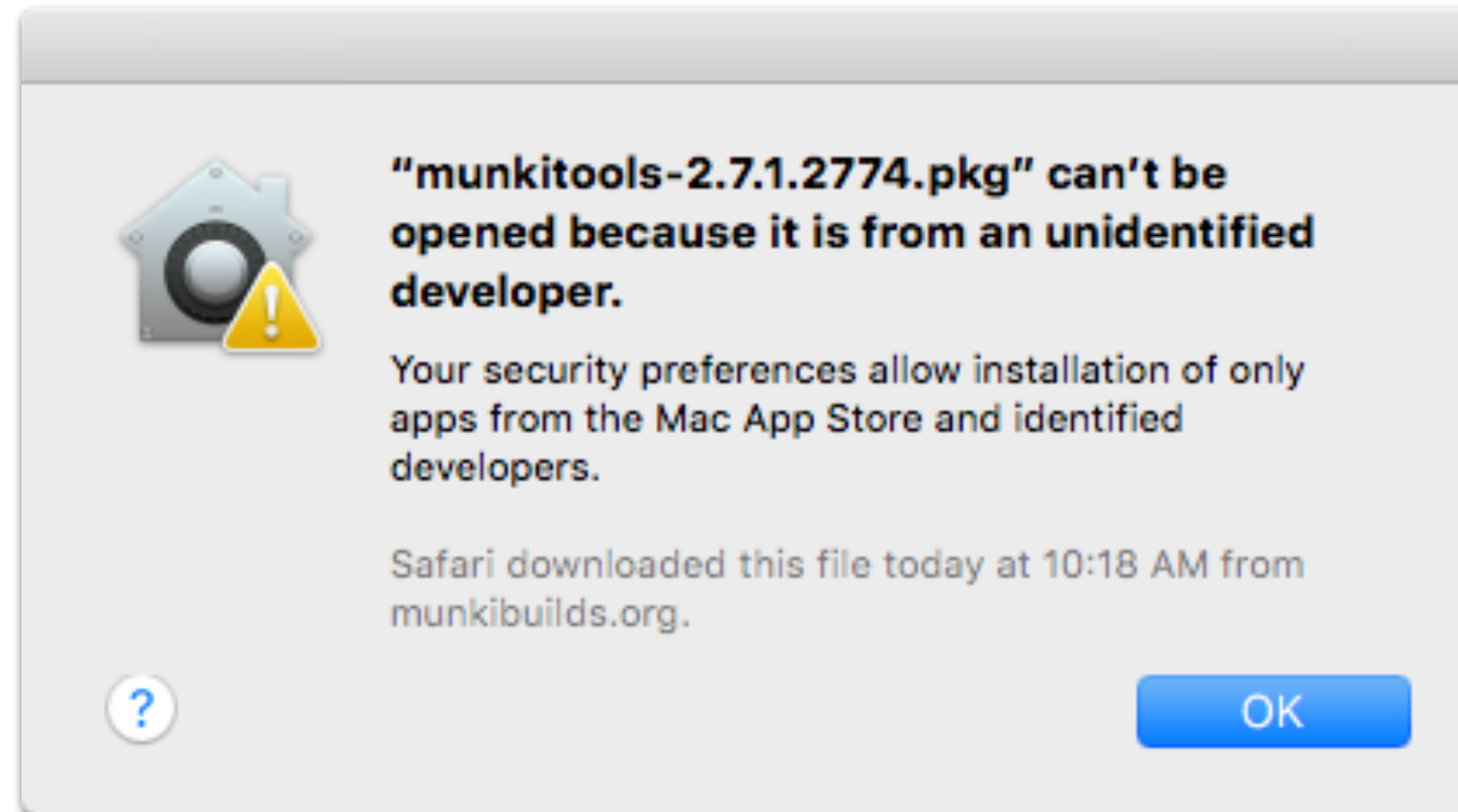
Signing a package with a valid **Developer ID certificate** will avoid the **Gatekeeper** prompt.

Gatekeeper

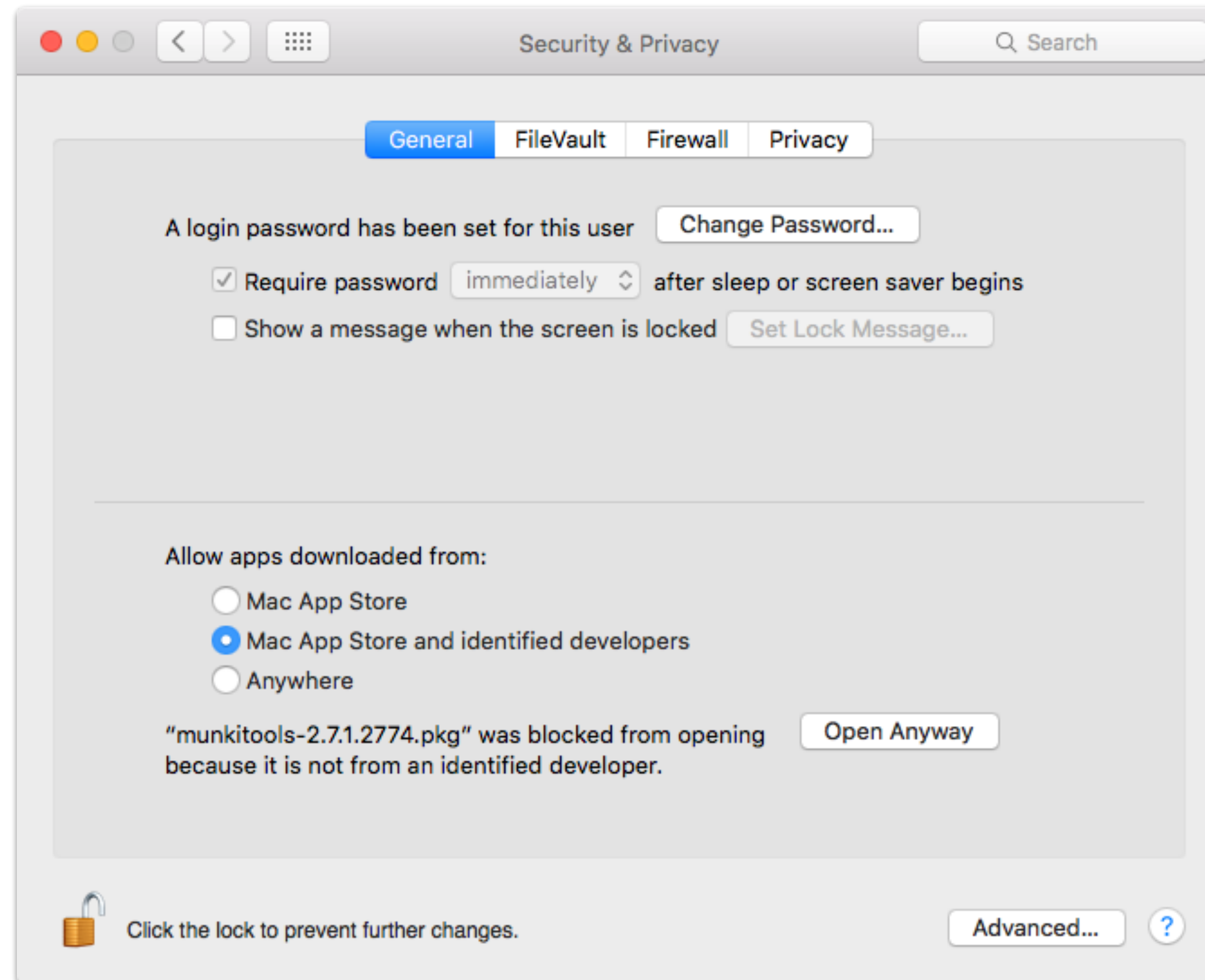
Gatekeeper will by default prevent users from installing unsigned packages **from Installer.app**.

Gatekeeper does **not** affect `/usr/bin/installer`.

Gatekeeper



Gatekeeper



Gatekeeper check

```
$ spctl --assess --type install ClearRegistrationSignedInvalid.pkg  
ClearRegistrationSignedInvalid.pkg: rejected
```

```
$ echo $?
```

```
3
```

Fail!

Gatekeeper check

```
$ spctl --assess --type install ClearRegistrationSignedInvalid.pkg
```

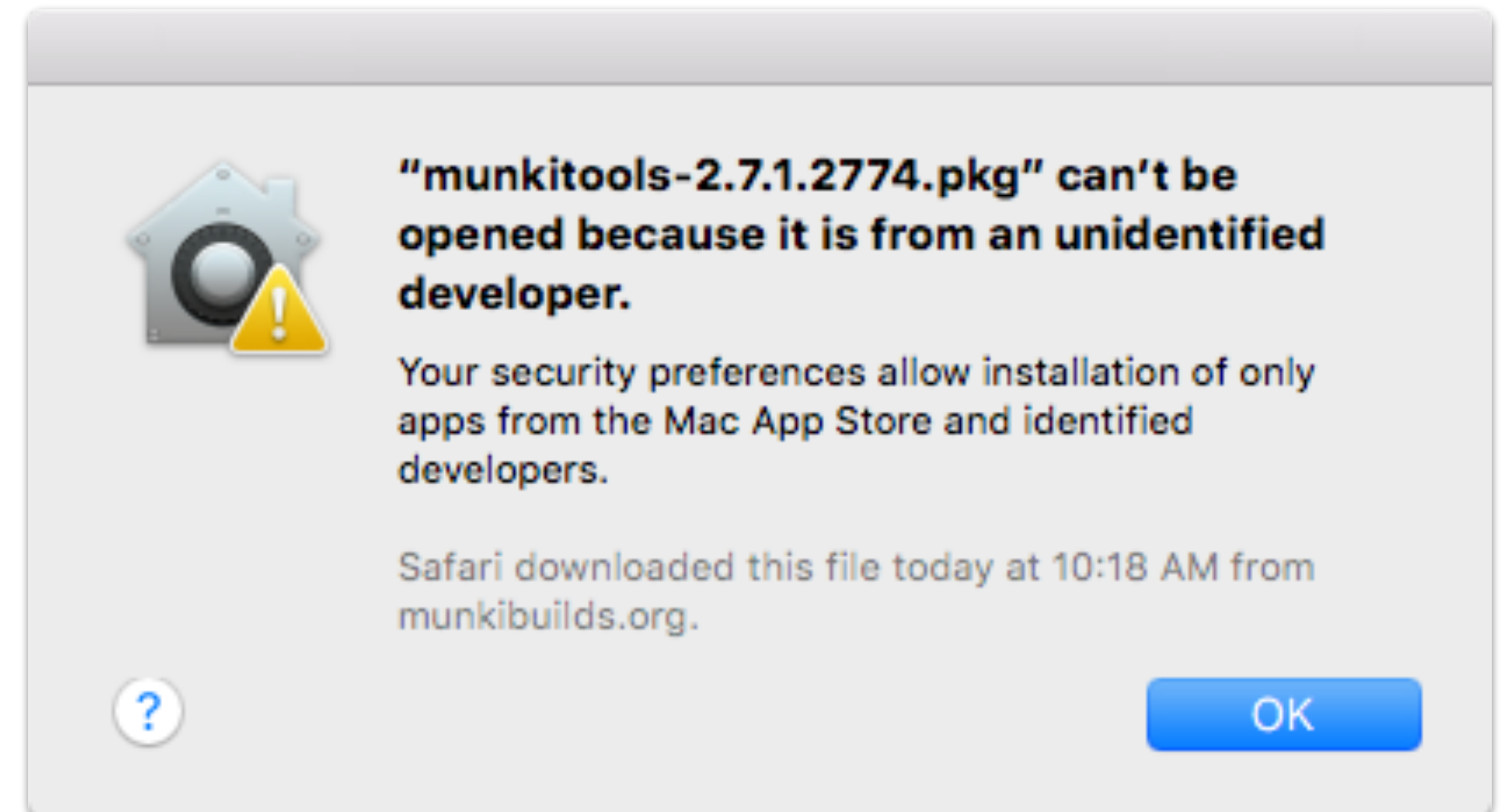
```
$ echo $?
```

```
0
```

Pass!

Gatekeeper penalties

There is a penalty for unsigned packages downloaded from a web browser or email if Gatekeeper is active.



Gatekeeper penalties

There is **no penalty** for **signed*** packages downloaded from a web browser or email if Gatekeeper is active.

*with Apple-issued installer certificates

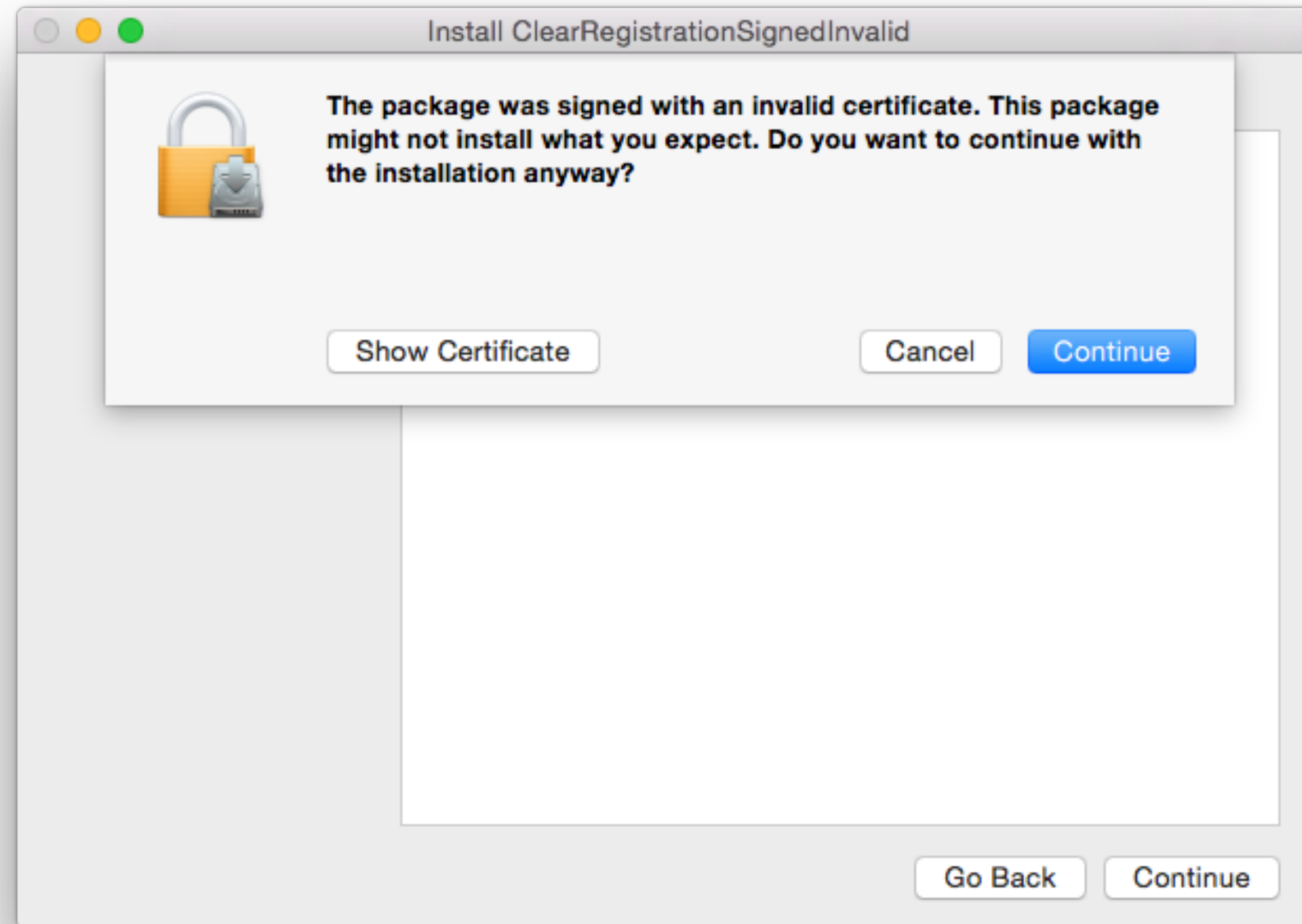
Gatekeeper penalties

There is **no penalty for unsigned packages copied from other sources** if Gatekeeper is active.

Gatekeeper penalties

There is **no penalty for signed packages copied from other sources** if Gatekeeper is active.

Untrusted certs will cause Installer.app to complain



Installer check

```
$ pkgutil --check-signature ClearRegistrationSignedInvalid.pkg
```

```
Package "ClearRegistrationSignedInvalid.pkg":
```

```
Status: signed by untrusted certificate
```

```
Certificate Chain:
```

```
1. My Test Cert
```

```
SHA1 fingerprint: FC F5 3D 0B EF 4F 46 53 99 C6 A7 1F  
8B 95 CD B2 4D C7 FA 70
```

Fail!

Installer check

```
% pkgutil --check-signature Adobe\ Flash\ Player.pkg
```

```
Package "Adobe Flash Player.pkg":
```

```
Status: signed by a certificate trusted by Mac OS X
```

```
Certificate Chain:
```

```
1. Developer ID Installer: Adobe Systems, Inc.
```

```
SHA1 fingerprint: 9D 75 C9 20 01 4A 65 04 94 A7 63 95 E3 91 93 47 04 E8 57 DF
```

```
-----
```

```
2. Developer ID Certification Authority
```

```
SHA1 fingerprint: 3B 16 6C 3B 7D C4 B7 51 C9 FE 2A FA B9 13 56 41 E3 88 E1 86
```

```
-----
```

```
3. Apple Root CA
```

```
SHA1 fingerprint: 61 1E 5B 66 2C 59 3A 08 FF 58 D1 4A E2 24 52 D1 98 DF 6C 60
```

```
Pass!
```

Untrusted certs will make /usr/sbin/installer complain

```
$ sudo /usr/sbin/installer -pkg ClearRegistrationSignedInvalid.pkg -target /  
installer: Package name is ClearRegistrationSignedInvalid  
installer: Certificate used to sign package is not trusted. Use -  
allowUntrusted to override.
```

Installer penalties

There is **no penalty** from Installer.app or /usr/sbin/installer for **unsigned** packages.

Installer penalties

There is **no penalty** from Installer.app or /usr/sbin/installer for **properly signed** packages.

Installer penalties

There is a **penalty** for signed packages with **invalid or untrusted** certificates.

You can sign packages as you build them:

```
$ pkgbuild --sign "Developer ID Installer" ...
```

You can sign distribution packages with productbuild too:

```
$ productbuild --sign "Developer ID Installer" ...
```

You can use productbuild to sign existing packages:

```
$ productsign --sign "Developer ID Installer"  
input.pkg output.pkg
```

Package Signing

Developer ID and Gatekeeper:

<https://developer.apple.com/developer-id/>

Code Signing:

<https://developer.apple.com/support/code-signing/>

OS X Code Signing In Depth

<https://developer.apple.com/library/mac/technotes/tn2206/index.html>

Package Signing

a HOW-TO guide

You can sign a package with a **valid identity**.

A valid identity must include a trusted certificate and the private key.

You can create **your own self-signed certificate** to sign with, but it **won't** be trusted by others.

The best option to sign packages is with a **Developer ID Installer certificate** from the Apple developer site.

Apple WWDC 16

watch()

Everyone Can Code
We're giving everyone the power to learn, write, and teach code.



Apple Pay
An easier way to pay within apps and websites.



Download New Betas
Get the next release of Xcode, macOS, iOS, watchOS, and tvOS.

Apple Developer Program
One membership. All Platforms.



Apple Inc. developer.apple.com/account/#/overview/52ZFZKM6BK

Developer Account Greg Neagle

Greg Neagle
Greg Neagle

Program Resources


- Overview
- Membership
- Certificates, IDs & Profiles
- iTunes Connect
- CloudKit Dashboard
- Code-Level Support

Additional Resources

- Documentation
- Forums
- Bug Reporter
- News & Updates


Greg Neagle

Apple Developer Program



Certificates, Identifiers & Profiles

Manage the certificates, identifiers, profiles, and devices you need to develop and distribute apps.



iTunes Connect

Publish and manage your apps on the App Store with iTunes Connect.


Apple Inc. developer.apple.com/account/ios/certificate/?teamId=52ZF

Developer Discover Design Develop Distribute Support Account

Certificates, Identifiers & Profiles Greg Neagle

iOS, tvOS, watchOS

iOS Certificates +



Getting Started with iOS Certificates

You will need to set up digital certificates to develop and distribute iOS apps. To install your app on a development device or submit it to the App Store, it must first be signed with an Apple-issued certificate. Certificates allow the system to identify who signed the app.

Request Certificates with Xcode
Xcode is the easiest way to request certificates. Connect your device to your Mac and click Use for Development in Xcode's Organizer window. Sign in with the Apple ID associated with your iOS Developer Program membership and Xcode will automatically generate your certificates.

Request Certificates Manually
To request certificates, upload a [Certificate Signing Request](#) from your Mac.

Learn More
[App Distribution Guide](#)

- Certificates
 - All
 - Pending
 - Development
 - Production
- Identifiers
 - App IDs
 - Pass Type IDs
 - Website Push IDs
 - iCloud Containers
 - App Groups
 - Merchant IDs
- Devices
 - All
 - Apple TV
 - Apple Watch
 - iPad
 - iPhone
 - iPod Touch
- Provisioning Profiles
 - All
 - Development
 - Distribution

Apple Inc. developer.apple.com/account/ios/certificate/?teamId=52ZF

Developer Discover Design Develop Distribute Support Account

Certificates, Identifiers & Profiles

Greg Neagle

iOS, tvOS, watchOS

- iOS, tvOS, watchOS
- OS X**
- Safari Extensions

- Development
- Production

Identifiers

- App IDs
- Pass Type IDs
- Website Push IDs
- iCloud Containers
- App Groups
- Merchant IDs


Devices

- All
- Apple TV
- Apple Watch
- iPad
- iPhone
- iPod Touch

Provisioning Profiles

- All
- Development

iOS Certificates



Getting Started with iOS Certificates

You will need to set up digital certificates to develop and distribute iOS apps. To install your app on a development device or submit it to the App Store, it must first be signed with an Apple-issued certificate. Certificates allow the system to identify who signed the app.

Request Certificates with Xcode
Xcode is the easiest way to request certificates. Connect your device to your Mac and click Use for Development in Xcode's Organizer window. Sign in with the Apple ID associated with your iOS Developer Program membership and Xcode will automatically generate your certificates.

Request Certificates Manually
To request certificates, upload a [Certificate Signing Request](#) from your Mac.

Learn More
[App Distribution Guide](#)

Open "https://developer.apple.com/account/mac/" in a new tab

Apple Inc. developer.apple.com/account/mac/certificate/

Developer Discover Design Develop Distribute Support Account

Certificates, Identifiers & Profiles Greg Neagle

OS X

Certificates

- All
- Pending
- Development
- Production

Identifiers

- App IDs
- Website Push IDs
- iCloud Containers


Devices

- All

Provisioning Profiles

- All
- Development
- Distribution

Mac Certificates +



Getting Started with Mac Certificates

You will need to set up certificates to develop and distribute Mac apps. To submit your app to the Mac App Store or distribute it outside of the Mac App Store, it must first be signed with an Apple-issued certificate. Certificates allow the system to identify who signed the app.

Request Certificates with Xcode
Xcode is the easiest way to request certificates. Use the Organizer window in Xcode to begin. Sign in with the Apple ID associated with your Mac Developer Program membership and Xcode will automatically generate your Mac certificates.

Request Certificates Manually
To request certificates, upload a [Certificate Signing Request](#) from your Mac.

Learn More
[App Distribution Guide](#)

Apple Inc. developer.apple.com/account/mac/certificate/c

Developer Discover Design Develop Distribute Support Account


Certificates, Identifiers & Profiles Greg Neagle ▾

OS X ▾

- Certificates**
 - All
 - Pending
 - Development
 - Production
- Identifiers**
 - App IDs
 - Website Push IDs
 - iCloud Containers
- Devices**
 - All
- Provisioning Profiles**
 - All
 - Development
 - Distribution

Add Mac "Certificate"

Select Type Request Generate Download

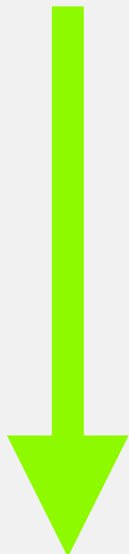
 **What type of certificate do you need?**

Development

- Mac Development**
Sign development versions of your Mac app.
- Apple Push Notification service SSL (Sandbox)**
Establish connectivity between your notification server and the Apple Push Notification service sandbox environment. A separate certificate is required for each app you develop.

Production

- Mac App Store**
Sign your Mac app and Mac Installer Package for submission to the Mac App Store.
- Apple Push Notification service SSL (Production)**
Establish connectivity between your notification server and the Apple Push Notification service production environment. A separate certificate is required for each app you distribute.
- Website Push ID Certificate**
Sign and send updates for Websites.



Apple Inc. developer.apple.com/account/mac/certificate/c

Devices

- All

Provisioning Profiles

- All
- Development
- Distribution

Apple Push Notification service SSL (Sandbox)
Establish connectivity between your notification server and the Apple Push Notification service sandbox environment. A separate certificate is required for each app you develop.

Production



Mac App Store
Sign your Mac app and Mac Installer Package for submission to the Mac App Store.

Apple Push Notification service SSL (Production)
Establish connectivity between your notification server and the Apple Push Notification service production environment. A separate certificate is required for each app you distribute.

Website Push ID Certificate
Sign and send updates for Websites.

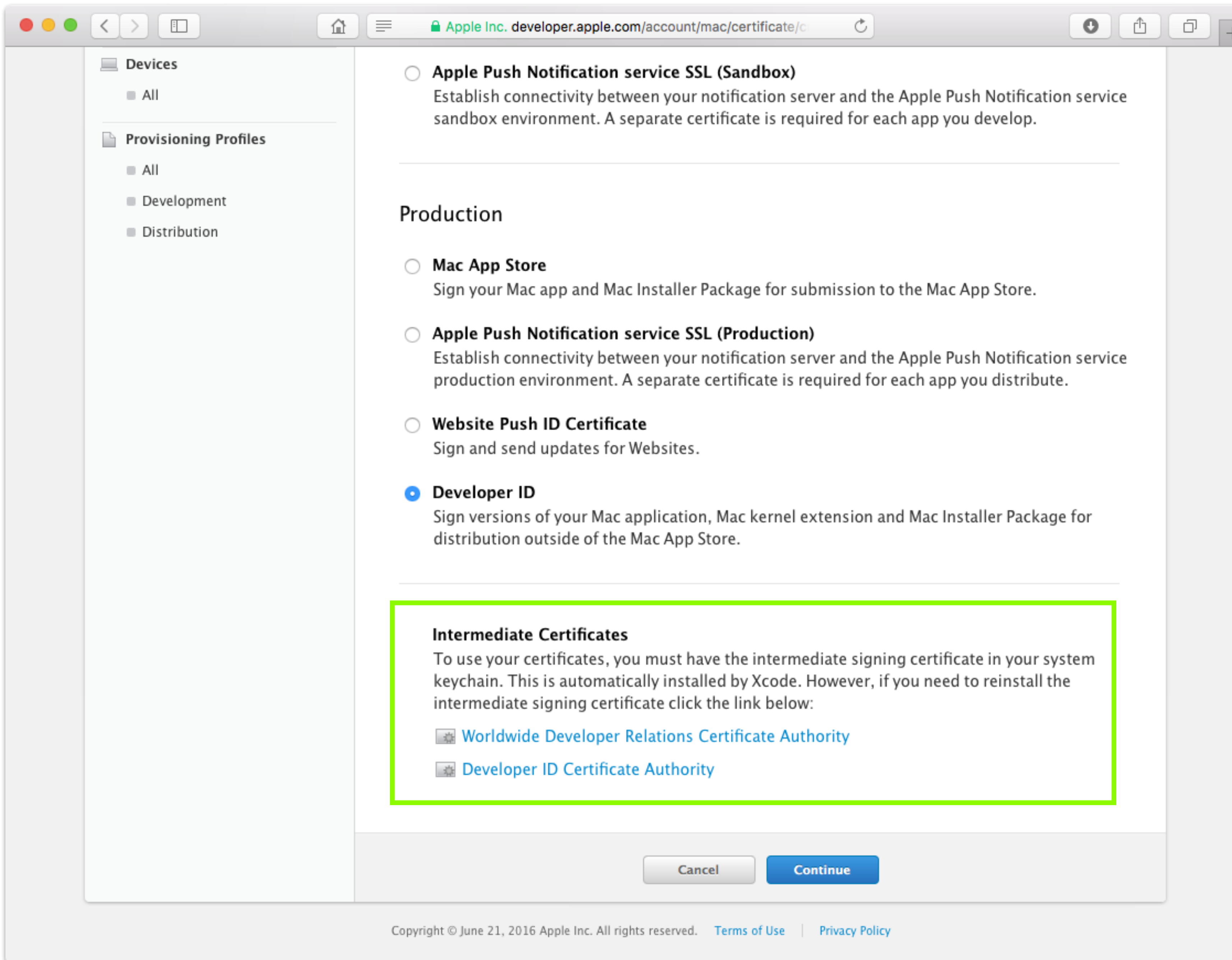
Developer ID
Sign versions of your Mac application, Mac kernel extension and Mac Installer Package for distribution outside of the Mac App Store.

Intermediate Certificates
To use your certificates, you must have the intermediate signing certificate in your system keychain. This is automatically installed by Xcode. However, if you need to reinstall the intermediate signing certificate click the link below:

-  [Worldwide Developer Relations Certificate Authority](#)
-  [Developer ID Certificate Authority](#)

Cancel Continue

Copyright © June 21, 2016 Apple Inc. All rights reserved. [Terms of Use](#) | [Privacy Policy](#)





Apple Push Notification service SSL (Sandbox)
Establish connectivity between your notification server and the Apple Push Notification service sandbox environment. A separate certificate is required for each app you develop.

Production

- Mac App Store**
Sign your Mac app and Mac Installer Package for submission to the Mac App Store.
- Apple Push Notification service SSL (Production)**
Establish connectivity between your notification server and the Apple Push Notification service production environment. A separate certificate is required for each app you distribute.
- Website Push ID Certificate**
Sign and send updates for Websites.
- Developer ID**
Sign versions of your Mac application, Mac kernel extension and Mac Installer Package for distribution outside of the Mac App Store.

Intermediate Certificates
To use your certificates, you must have the intermediate signing certificate in your system keychain. This is automatically installed by Xcode. However, if you need to reinstall the intermediate signing certificate click the link below:

-  [Worldwide Developer Relations Certificate Authority](#)
-  [Developer ID Certificate Authority](#)

Apple Inc. developer.apple.com/account/mac/certificate/c

Developer Discover Design Develop Distribute Support Account


Certificates, Identifiers & Profiles Greg Neagle ▾


OS X ▾

- ✓ Certificates
 - All
 - Pending
 - Development
 - Production
- ID Identifiers
 - App IDs
 - Website Push IDs
 - iCloud Containers
- Devices
 - All
- Provisioning Profiles
 - All
 - Development
 - Distribution

Add Mac "Certificate"

Select Type Request Generate Download

 **Select the certificates you want to generate.**

 **Important:** Xcode is the preferred method for requesting and automatically installing these certificates on a Mac. [Learn how](#)

- Developer ID Application**
This certificate is used to code sign your app for distribution outside of the Mac App Store.
- Developer ID Installer**
This certificate is used to sign your app's Installer Package for distribution outside of the Mac App Store.

Cancel Back Continue

Copyright © June 21, 2016 Apple Inc. All rights reserved. [Terms of Use](#) | [Privacy Policy](#)

Apple Inc. developer.apple.com/account/mac/certificate/c

Developer Discover Design Develop Distribute Support Account

Certificates, Identifiers & Profiles Greg Neagle ▾

OS X

Certificates

- All
- Pending
- Development
- Production

Identifiers

- App IDs
- Website Push IDs
- iCloud Containers

Devices


- All

Provisioning Profiles

- All
- Development
- Distribution

Add Mac "Certificate"

Select Type Request Generate Download



About Creating a Certificate Signing Request (CSR)

To manually generate a Certificate, you need a Certificate Signing Request (CSR) file from your Mac. To create a CSR file, follow the instructions below to create one using Keychain Access.

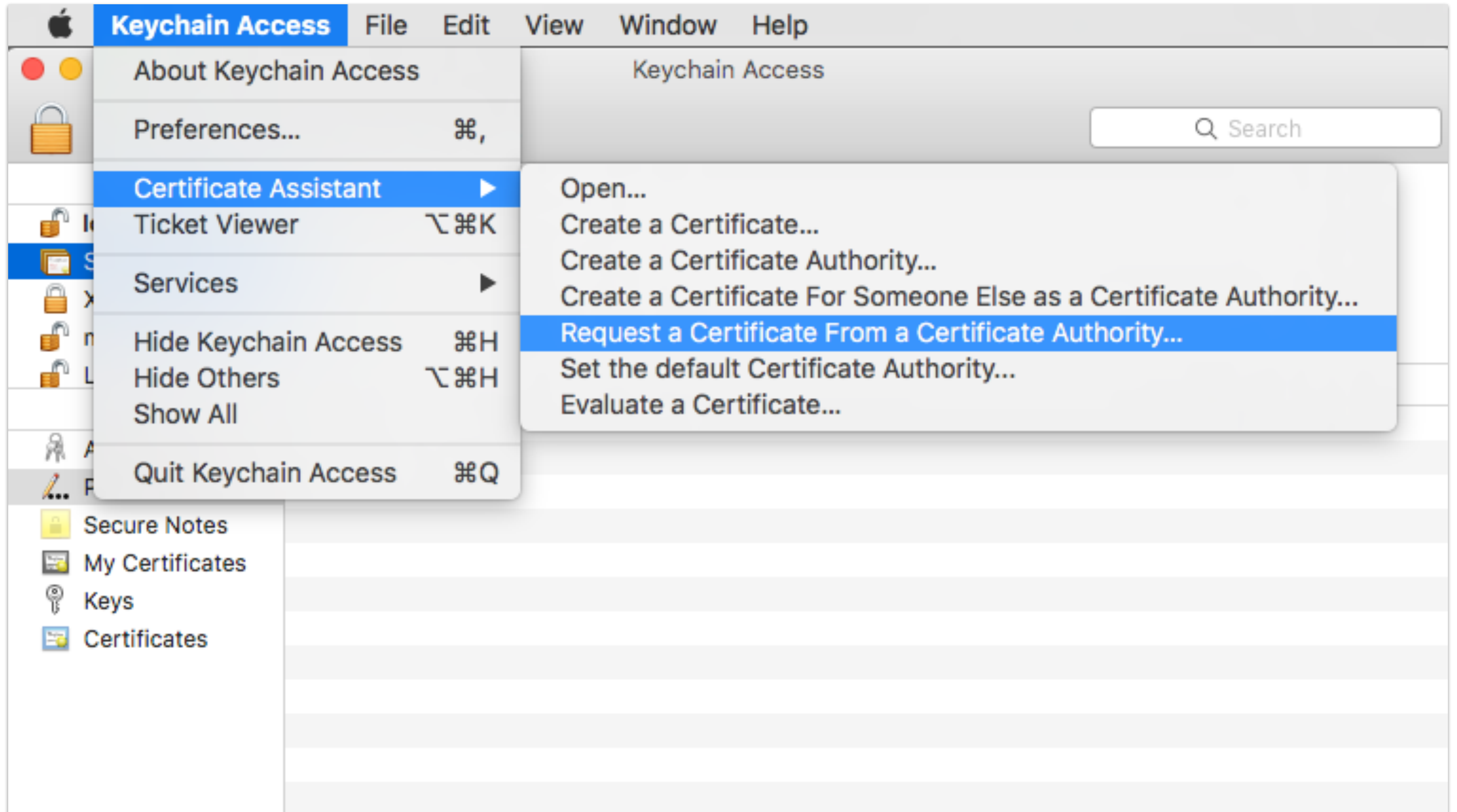
Create a CSR file.
In the Applications folder on your Mac, open the Utilities folder and launch Keychain Access.

Within the Keychain Access drop down menu, select Keychain Access > Certificate Assistant > Request a Certificate from a Certificate Authority.

- In the Certificate Information window, enter the following information:
 - In the User Email Address field, enter your email address.
 - In the Common Name field, create a name for your private key (e.g., John Doe Dev Key).
 - The CA Email Address field should be left empty.
 - In the "Request is" group, select the "Saved to disk" option.
- Click Continue within Keychain Access to complete the CSR generating process.

Cancel Back Continue

Copyright © June 21, 2016 Apple Inc. All rights reserved. [Terms of Use](#) | [Privacy Policy](#)



Certificate Assistant

Certificate Information

Enter information for the certificate you are requesting. Click Continue to request a certificate from the CA.

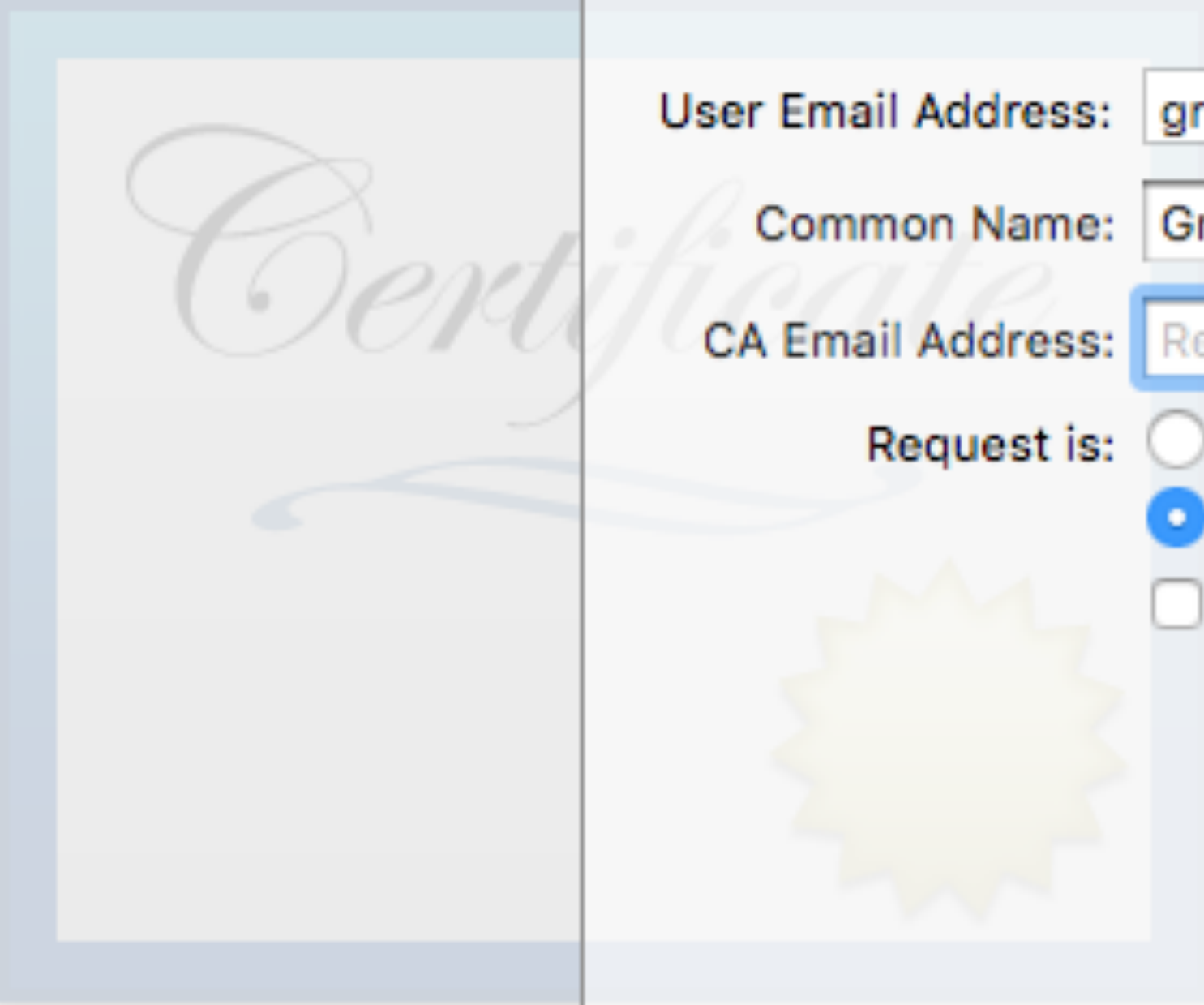
User Email Address:

Common Name:

CA Email Address:

Request is: Emailed to the CA
 Saved to disk
 Let me specify key pair information





Continue




Certificate Assistant

Save As: CertificateSigningRequest.certSigningRe ^

Tags:

< >     Desktop ^ v Search

Devices

 Remote...

Shared

 All...

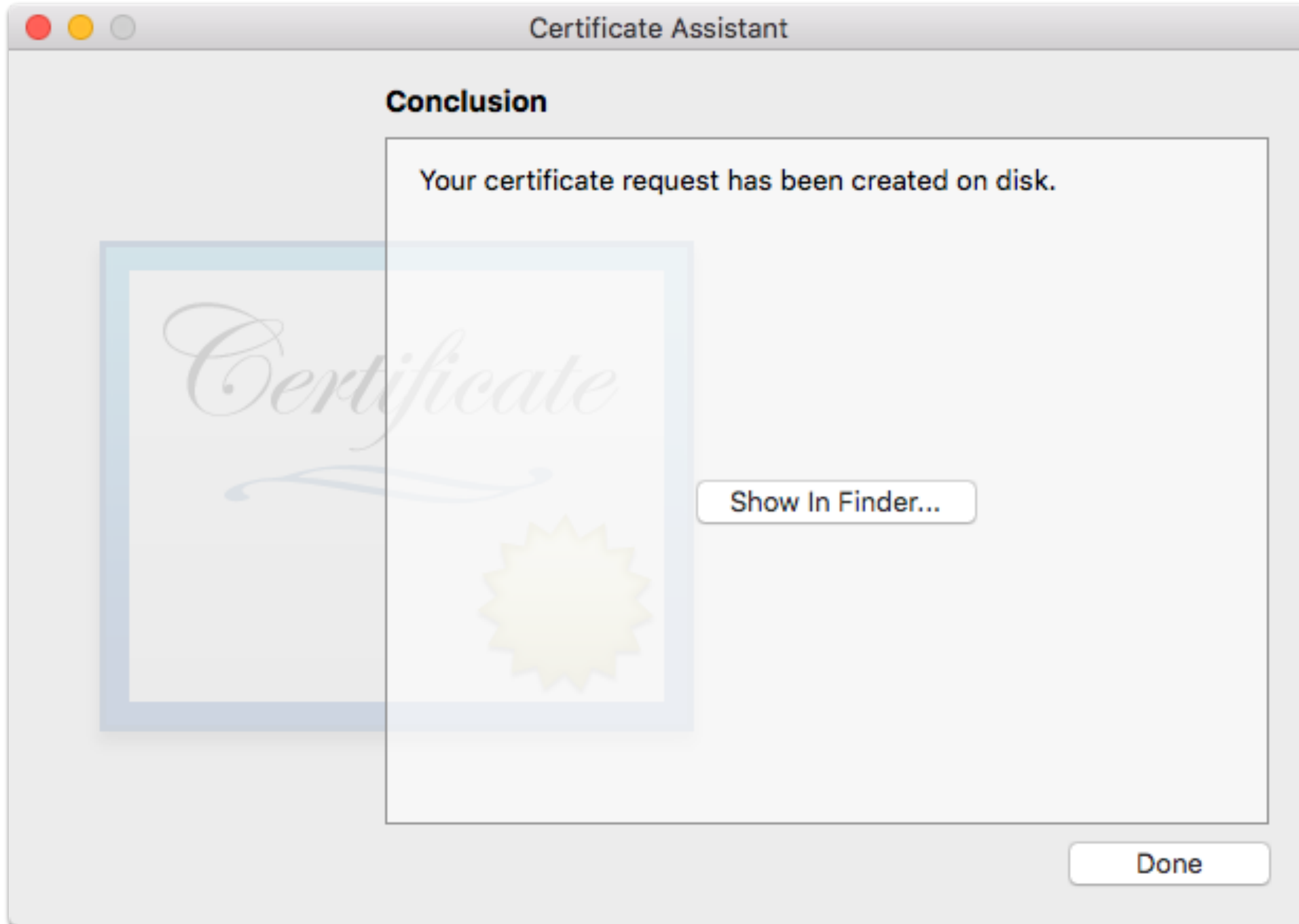
Name ^ Date Modified Size

▶ cleanup Today, 3:22 PM --

New Folder

Cancel

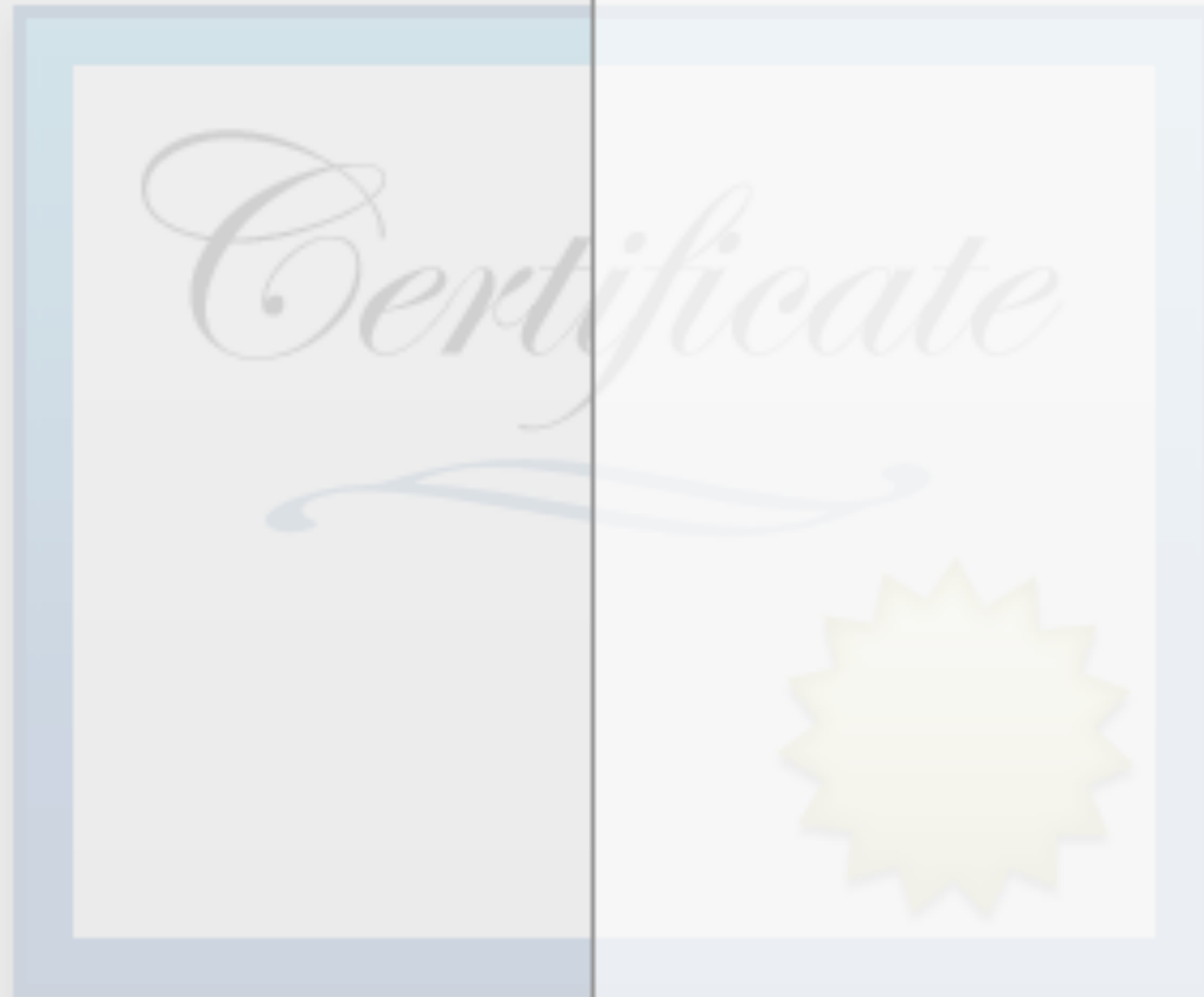
Save



Certificate Assistant

Conclusion

Your certificate request has been created on disk.



Show In Finder...

Done



Desktop



Search

- Favorites
- All My Files
- iCloud Drive
- Dropbox
- AirDrop
- Applications
- Desktop**
- Documents
- Downloads
- Movies
- Music
- Pictures
- gneagle
- Shared

Name	Date Modified	Size	Kind
CertificateSigningRequest.certSigningRequest	Today, 3:23 PM	968 bytes	certSig...equest
cleanup	Today, 3:22 PM	--	Folder

Apple Inc. developer.apple.com/account/mac/certificate/c

Developer Discover Design Develop Distribute Support Account

Certificates, Identifiers & Profiles Greg Neagle ▾

OS X

Certificates

- All
- Pending
- Development
- Production

Identifiers

- App IDs
- Website Push IDs
- iCloud Containers

Devices


- All

Provisioning Profiles

- All
- Development
- Distribution

Add Mac "Certificate"

Select Type Request Generate Download



About Creating a Certificate Signing Request (CSR)

To manually generate a Certificate, you need a Certificate Signing Request (CSR) file from your Mac. To create a CSR file, follow the instructions below to create one using Keychain Access.

Create a CSR file.
In the Applications folder on your Mac, open the Utilities folder and launch Keychain Access.

Within the Keychain Access drop down menu, select Keychain Access > Certificate Assistant > Request a Certificate from a Certificate Authority.

- In the Certificate Information window, enter the following information:
 - In the User Email Address field, enter your email address.
 - In the Common Name field, create a name for your private key (e.g., John Doe Dev Key).
 - The CA Email Address field should be left empty.
 - In the "Request is" group, select the "Saved to disk" option.
- Click Continue within Keychain Access to complete the CSR generating process.

Cancel Back Continue

Copyright © June 21, 2016 Apple Inc. All rights reserved. [Terms of Use](#) | [Privacy Policy](#)

Apple Inc. developer.apple.com/account/mac/certificate/c

Developer Discover Design Develop Distribute Support Account


Certificates, Identifiers & Profiles Greg Neagle ▾

OS X ▾

- ✓ Certificates
 - All
 - Pending
 - Development
 - Production
- ID Identifiers
 - App IDs
 - Website Push IDs
 - iCloud Containers
- Devices
 - All
- Provisioning Profiles
 - All
 - Development
 - Distribution

Add Mac "Certificate"

Select Type Request **Generate** Download



Generate your certificate.

When your CSR file is created, a public and private key pair is automatically generated. Your private key is stored on your computer. On a Mac, it is stored in the login Keychain by default and can be viewed in the Keychain Access app under the "Keys" category. Your requested certificate is the public half of your key pair.

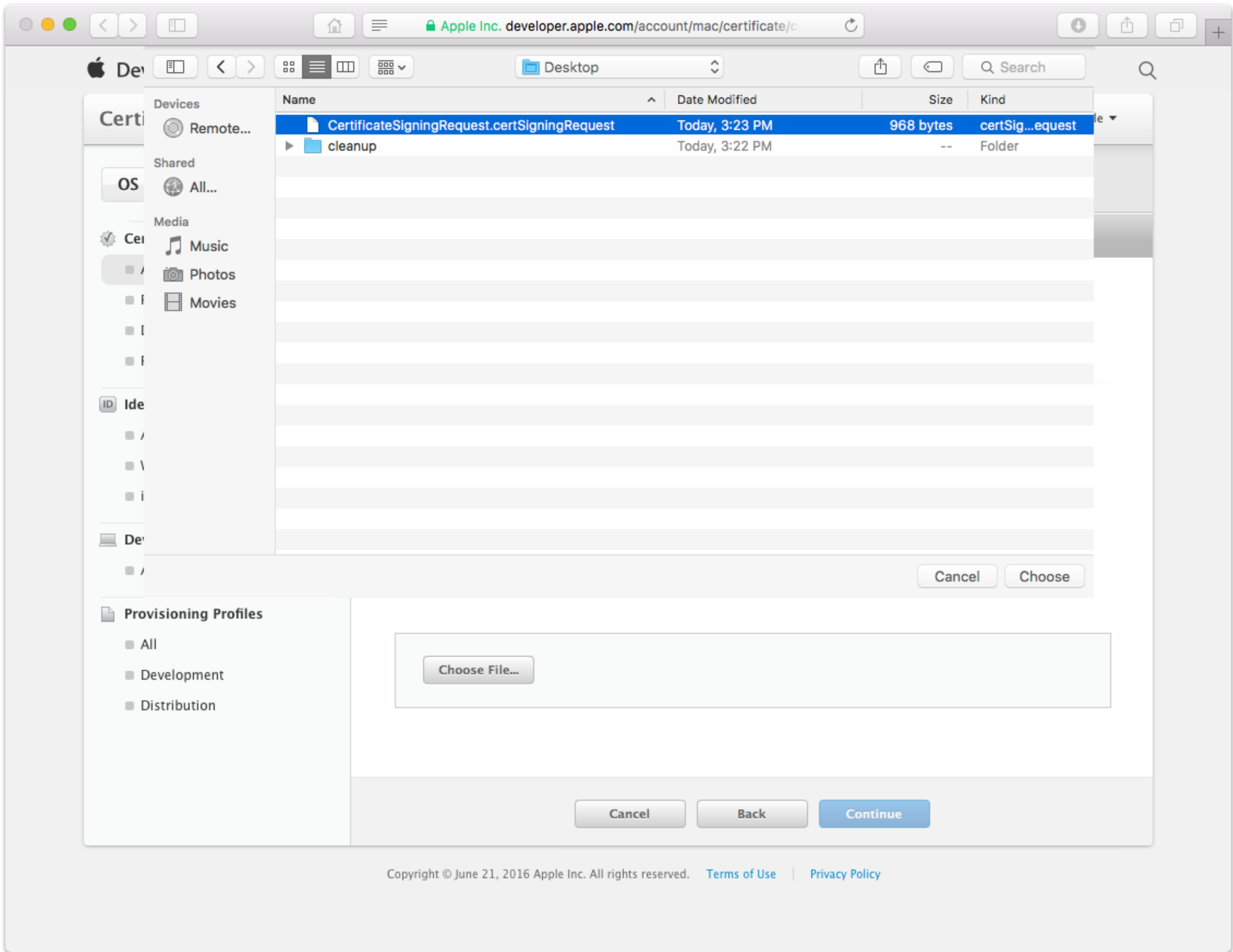
Upload CSR file.

Select .certSigningRequest file saved on your Mac.

Choose File...

Cancel Back **Continue**

Copyright © June 21, 2016 Apple Inc. All rights reserved. [Terms of Use](#) | [Privacy Policy](#)



Apple Inc. developer.apple.com/account/mac/certificate/c

Developer Discover Design Develop Distribute Support Account


Certificates, Identifiers & Profiles Greg Neagle ▾

OS X ▾

- ✓ Certificates
 - All
 - Pending
 - Development
 - Production
- ID Identifiers
 - App IDs
 - Website Push IDs
 - iCloud Containers
- Devices
 - All
- Provisioning Profiles
 - All
 - Development
 - Distribution

Add Mac "Certificate"

Select Type Request **Generate** Download



Generate your certificate.

When your CSR file is created, a public and private key pair is automatically generated. Your private key is stored on your computer. On a Mac, it is stored in the login Keychain by default and can be viewed in the Keychain Access app under the "Keys" category. Your requested certificate is the public half of your key pair.

Upload CSR file.

Select .certSigningRequest file saved on your Mac.

Choose File... CertificateSigningRequest.certSigningRequest

Cancel Back **Continue**

Copyright © June 21, 2016 Apple Inc. All rights reserved. [Terms of Use](#) | [Privacy Policy](#)

Apple Inc. developer.apple.com/account/mac/certificate/c

Developer Discover Design Develop Distribute Support Account


Certificates, Identifiers & Profiles Greg Neagle ▾

OS X ▾


- ✓ Certificates
 - All
 - Pending
 - Development
 - Production
- ID Identifiers
 - App IDs
 - Website Push IDs
 - iCloud Containers
- Devices
 - All
- Provisioning Profiles
 - All
 - Development
 - Distribution

Add Mac "Certificate"

Select Type Request Generate Download

 **Your certificate is ready.**

Download, Install and Backup
Download your certificate to your Mac, then double click the .cer file to install in Keychain Access. Make sure to save a backup copy of your private and public keys somewhere secure.

	Name:	Developer ID Installer: Greg Neagle
	Type:	Developer ID Installer
	Expires:	Jun 22, 2021

[Download](#)

Documentation
For more information on using and managing your certificates read:
[App Distribution Guide](#)

[Add Another](#) [Done](#)

Downloads

Search

Favorites

- All My Files
- iCloud Drive
- Dropbox
- AirDrop
- Applications
- Desktop
- Documents
- Downloads**
- Movies
- Music
- Pictures
- gneagle
- Shared

Devices

- Remote Disc

Shared

- All...

Tags

Today	Date Modified	
developerID_installer.cer	Today, 3:28 PM	1
munkitools-2.7.1.2774.pkg	Jun 16, 2016, 1:28 PM	3.5
Previous 30 Days		
micromdm-darwin-amd64	May 18, 2016, 6:13 AM	14.2
MunkiAdmin-1.4.2.dmg	Apr 26, 2016, 11:06 AM	7.8
MBAM-Mac-1.1.3.72.dmg	Oct 2, 2015, 3:03 PM	3.9
filevaultsetup-1.0.3.dmg	Jan 7, 2014, 12:07 PM	505
May		
13116157_86966402...9739176687498_o.jpg	May 1, 2016, 8:57 PM	72
13131290_86966402...5828358175978_o.jpg	May 1, 2016, 8:57 PM	70
Hopper Disassembler v3	Apr 27, 2016, 3:33 PM	92.7
April		
fonts-master	Apr 29, 2016, 10:12 AM	
IMG_3247.PNG	Apr 21, 2016, 2:17 PM	117
ArrayControllerSearch	Apr 18, 2016, 4:11 PM	
MSCerrorLog.txt	Apr 12, 2016, 10:43 AM	72
ScreenSaverLock.mobileconfig	Apr 7, 2016, 10:25 PM	6
ScreenSaverIdleTime.mobileconfig	Apr 7, 2016, 10:14 PM	5
ConferenceRoomMacE...r-1.0.mobileconfig.html	Apr 7, 2016, 2:26 PM	3
AutoDMG-1.5.5.dmg	Jan 22, 2016, 3:38 AM	1
March		
Metadata	Mar 4, 2016, 7:03 AM	287 by
SuspiciousPackage.xip	Mar 4, 2016, 7:01 AM	2.3
Content	Feb 29, 2016, 9:46 AM	4.4
mas	Feb 21, 2016, 10:20 PM	4.4

1 of 71 selected, 26.9 GB available

Keychain Access

Click to lock the login keychain.

Keychains

- login
- System Roots
- X509Anchors
- munki
- Local Items

Category

- All Items
- Passwords
- Secure Notes
- My Certificates
- Keys
- Certificates

Developer ID Installer: Greg Neagle (52ZFZKM6BK)
Issued by: Developer ID Certification Authority
Expires: Tuesday, June 22, 2021 at 3:16:57 PM Pacific Daylight Time
✔ This certificate is valid

Name	Kind
▶ MDM Vendor: Walt Disney Animation Studios	certificate
▼ Developer ID Installer: Greg Neagle (52ZFZKM6BK)	certificate
🔑 Greg Neagle Dev Key	private key
▶ com.apple.idms.appleid.pr...c306d6977554f3943596c55494b59413d3d	certificate
▶ com.apple.idms.appleid.pr...c306d6977554f3943596c55494b59413d3d	certificate
▶ APSP:f242751a-ac01-4a4b-9247-a33e191c5aff	certificate

+ ⓘ Copy 5 items

```
% security find-identity -v -p basic
```

- 1) FD4E1FC44C5C9CAA401BD0B8A0A83514C7E9FF23 "com.apple.idms.appleid.prd.6f4864726b706c306d6977554f"
- 2) 05A5F8F2143B6B572641EF65B708F1A1F9B7EB8C "com.apple.idms.appleid.prd.6f4864726b706c306d6977554f"
- 3) 4CCD4349B18E11528D18556B6EF195EE025DB623 "MDM Vendor: Walt Disney Animation Studios"
- 4) 6E7EA91A11A21F80C09EA18910C71E56C8E034D3 "APSP:f242751a-ac01-4a4b-9247-a33e191c5aff"
- 5) 31D7D9701EBC6DD71BEF920DC9E81B2D82FE62A8 "Developer ID Installer: Greg Neagle (52ZFZKM6BK)"

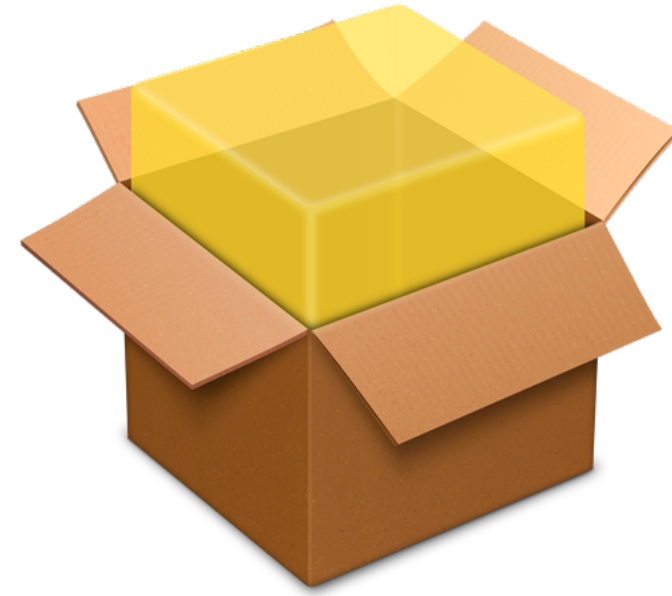
```
5 valid identities found
```



```
% security find-identity -v -p basic
```

- 1) FD4E1FC44C5C9CAA401BD0B8A0A83514C7E9FF23 "com.apple.idms.appleid.prd.6f4864726b706c306d6977554f"
 - 2) 05A5F8F2143B6B572641EF65B708F1A1F9B7EB8C "com.apple.idms.appleid.prd.6f4864726b706c306d6977554f"
 - 3) 4CCD4349B18E11528D18556B6EF195EE025DB623 "MDM Vendor: Walt Disney Animation Studios"
 - 4) 6E7EA91A11A21F80C09EA18910C71E56C8E034D3 "APSP:f242751a-ac01-4a4b-9247-a33e191c5aff"
 - 5) 31D7D9701EBC6DD71BEF920DC9E81B2D82FE62A8 "Developer ID Installer: Greg Neagle (52ZFZKM6BK)"
- 5 valid identities found

Alright, let's sign something!



munkitools-2.7.1.2774.pkg

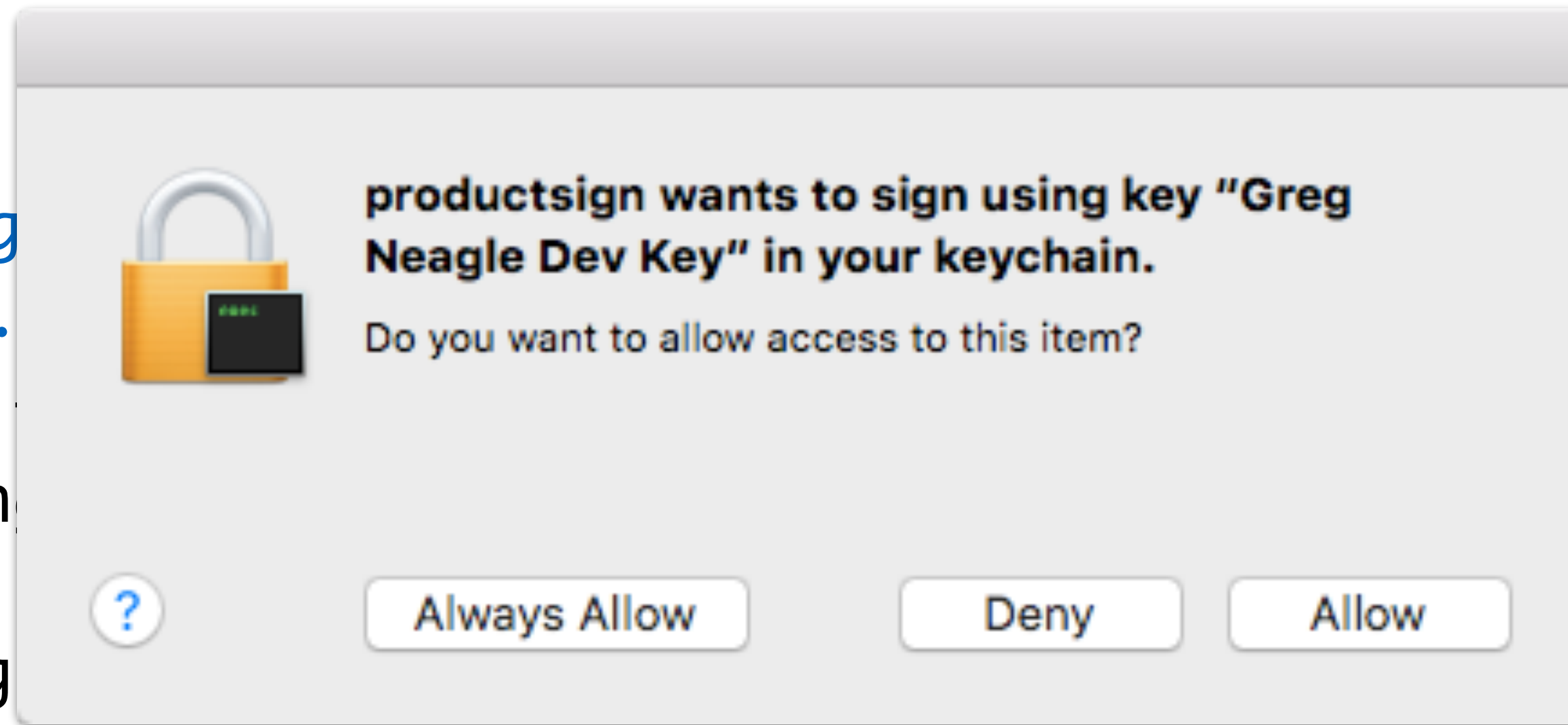
```
% pkgutil --check-signature munkitools-2.7.1.2774.pkg
```

```
Package "munkitools-2.7.1.2774.pkg":
```

```
Status: no signature
```

```
% productsign --sig  
munkitools-2.7.1.
```

```
productsign: using  
productsign: signing  
(52ZFZKM6BK)" from  
productsign: adding  
productsign: adding  
productsign: Wrote signed product archive to munkitools_signed-2.7.1.2774.pkg
```



```
52ZFZKM6BK)" \  
g
```

```
staller: Greg Neagle  
s/login.keychain  
Authority"
```

```
% pkgutil --check-signature munkitools_signed-2.7.1.2774.pkg
```

```
Package "munkitools_signed-2.7.1.2774.pkg":
```

```
Status: signed by a certificate trusted by Mac OS X
```

```
Certificate Chain:
```

```
1. Developer ID Installer: Greg Neagle (52ZFZKM6BK)
```

```
SHA1 fingerprint: 31 D7 D9 70 1E BC 6D D7 1B EF 92 0D C9 E8 1B 2D 82 FE 62 A8
```

```
-----
```

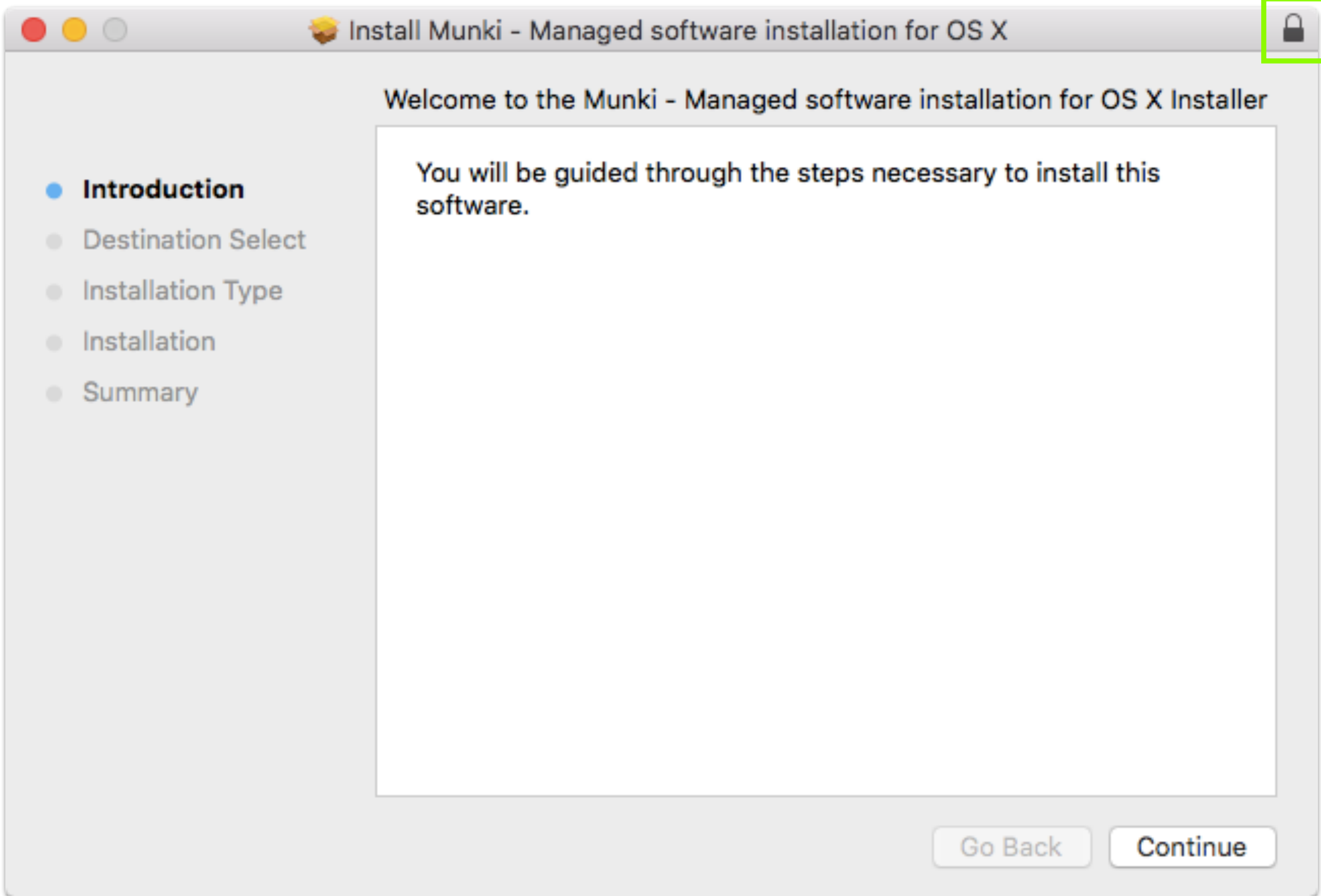
```
2. Developer ID Certification Authority
```

```
SHA1 fingerprint: 3B 16 6C 3B 7D C4 B7 51 C9 FE 2A FA B9 13 56 41 E3 88 E1 86
```

```
-----
```

```
3. Apple Root CA
```


```
SHA1 fingerprint: 61 1E 5B 66 2C 59 3A 08 FF 58 D1 4A E2 24 52 D1 98 DF 6C 60
```



Install Munki - Managed software installation for OS X

- **Introduct**
- Destina
- Installa
- Installa
- Summa

Apple Root CA
↳ Developer ID Certification Authority
↳ **Developer ID Installer: Greg Neagle (52ZFZKM6BK)**

 **Developer ID Installer: Greg Neagle (52ZFZKM6BK)**
Issued by: Developer ID Certification Authority
Expires: Tuesday, June 22, 2021 at 3:16:57 PM Pacific Daylight Time
✔ This certificate is valid

▶ **Details**


?

OK

Go Back Continue

munkitools_signed-2.7.1.2774.pkg Open with Installer Share


Open in Suspicious Package





munkitools_signed-2.7.1.2774.pkg

Developer ID Installer Package ▼

DEVELOPER ID Signed by "Developer ID Installer: Greg Neagle (52ZFZK..." ▲

 **Apple Root CA**
Root certificate authority
Expires: Feb 9, 2035, 1:40:36 PM
✔ This certificate is valid

 **Developer ID Certification Authority**
Intermediate certificate authority
Expires: Feb 1, 2027, 2:12:15 PM
✔ This certificate is valid

 **Developer ID Installer: Greg Neagle (52ZFZKM6BK)**
Expires: Jun 22, 2021, 3:16:57 PM
✔ This certificate is valid

[Read more](#) about how to interpret package signatures.

⏪ Restart required after installation

🔧 Runs 2 install scripts ▶

3.5 MB for package — 5.4 MB installed on disk

Previously installed on "Macintosh HD" using Apple Installer — June 21, 2016

Signing an existing package:

```
productsign --sign "Developer ID Installer: Foo" unsigned.pkg signed.pkg
```

Signing as you build a package:

```
pkgbuild --sign "Developer ID Installer: Foo" --root payload_folder ...
```

```
productbuild --sign "Developer ID Installer: Foo" --package Bar.pkg ...
```

Developer ID and Gatekeeper:

<https://developer.apple.com/developer-id/>

Code Signing:

<https://developer.apple.com/support/code-signing/>

OS X Code Signing In Depth

<https://developer.apple.com/library/mac/technotes/tn2206/index.html>

munki-pkg

munki-pkg

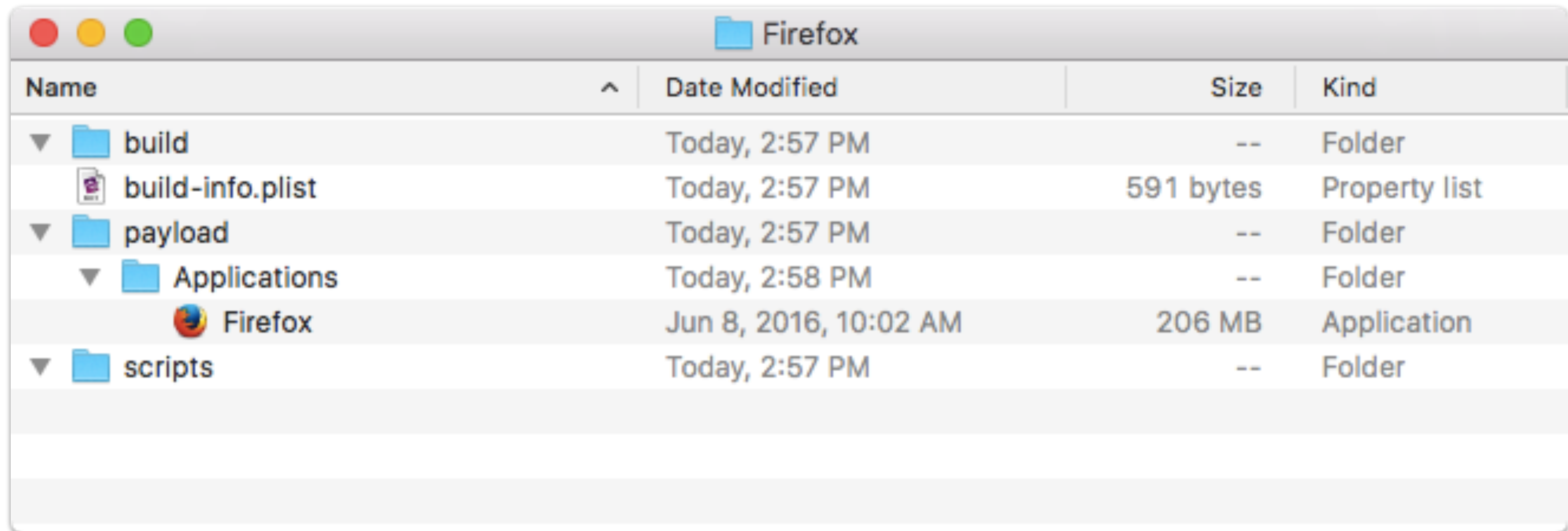
- a simple tool for building packages in a consistent, repeatable manner from source files and scripts in a project directory

munki-pkg

- does not require Munki
- can be run completely standalone

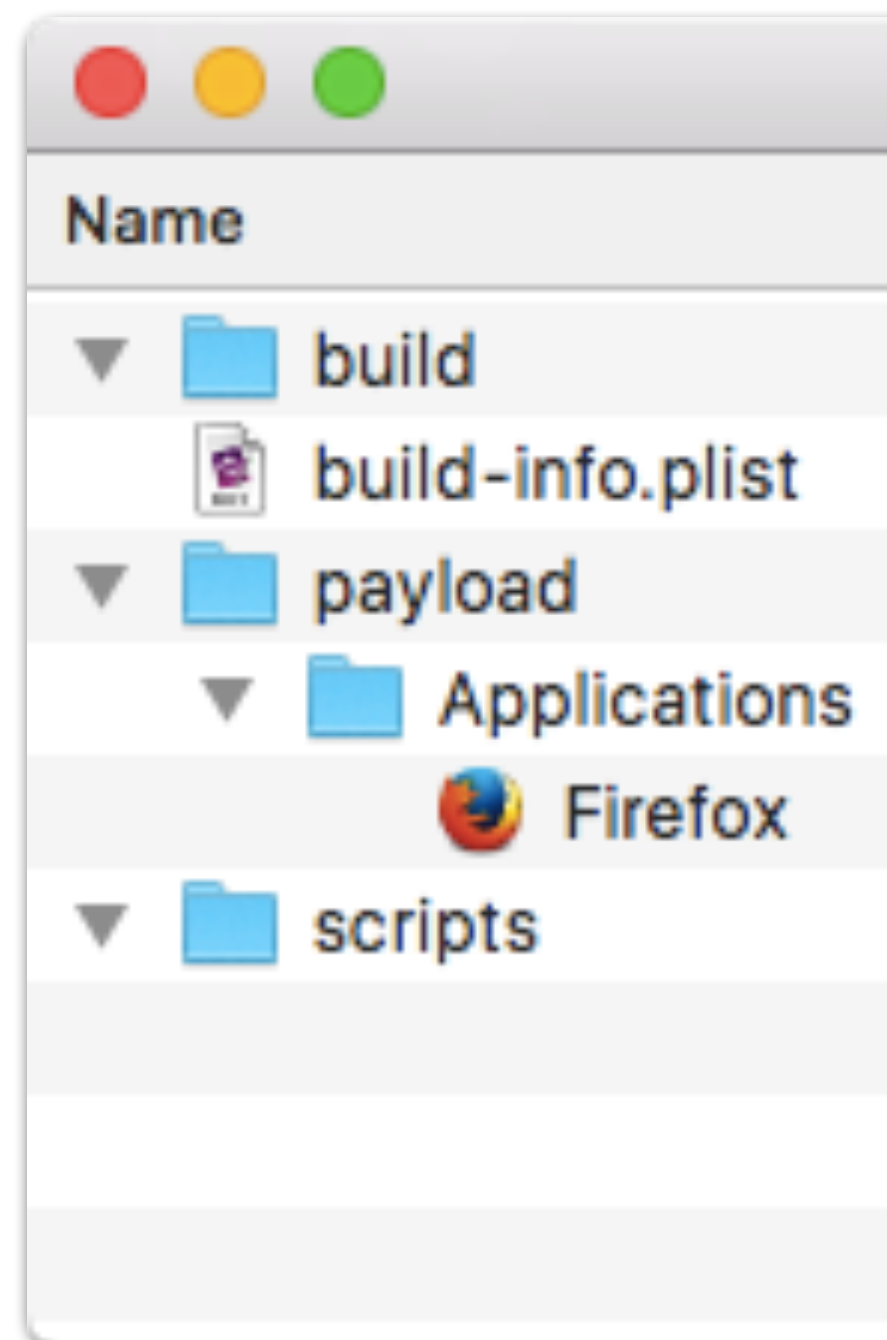
munki-pkg

- <https://github.com/munki/munki-pkg>



The image shows a Mac OS Finder window titled "Firefox". The window displays a list of files and folders in a table format. The columns are "Name", "Date Modified", "Size", and "Kind". The items listed are:

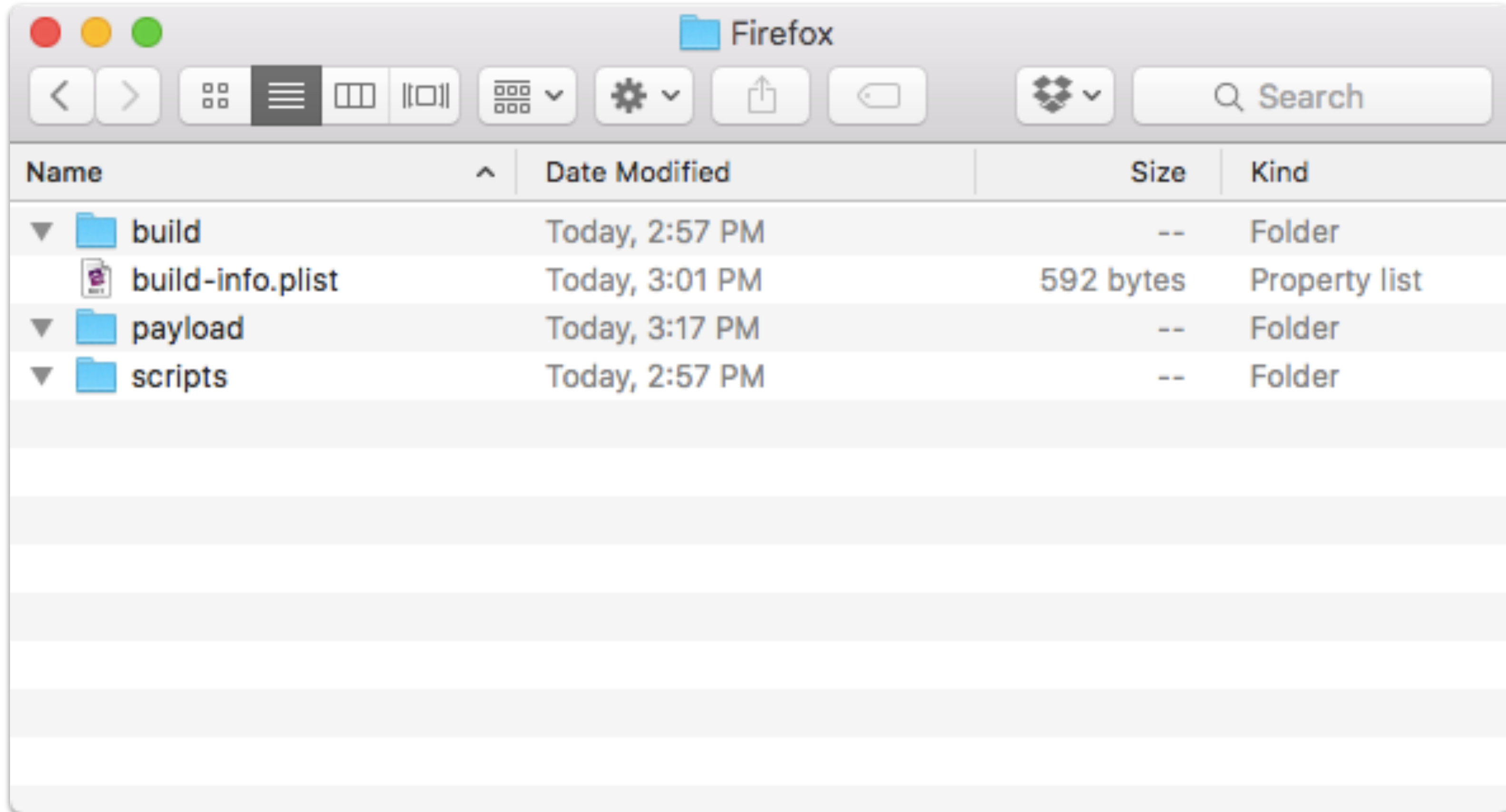
Name	Date Modified	Size	Kind
▼ build	Today, 2:57 PM	--	Folder
build-info.plist	Today, 2:57 PM	591 bytes	Property list
▼ payload	Today, 2:57 PM	--	Folder
▼ Applications	Today, 2:58 PM	--	Folder
Firefox	Jun 8, 2016, 10:02 AM	206 MB	Application
▼ scripts	Today, 2:57 PM	--	Folder



```
build-info.plist
1 <?xml version="1.0" encoding="UTF-8"?>
2 <!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
3 <plist version="1.0">
4 <dict>
5
6 build-info.json
7 {
8   "postinstall_action": "none",
9   "suppress_bundle_relocation": true,
10  "name": "Firefox-${version}.pkg",
11  "distribution_style": false,
12  "install_location": "/",
13  "version": "47.0",
14  "ownership": "recommended",
15  "identifier": "com.github.munki.pkg.Firefox"
16 }
17
18
19
20 Line: 10 Column: 2 Plain Text Soft Tabs: 4
21 </dict>
22 </plist>
23
24 Line: 12 Column: 31 Property List Soft Tabs: 4 name
```

Creating munki-pkg
projects

```
% ./munkipkg --create Firefox  
munkipkg: Created new package project at Firefox
```



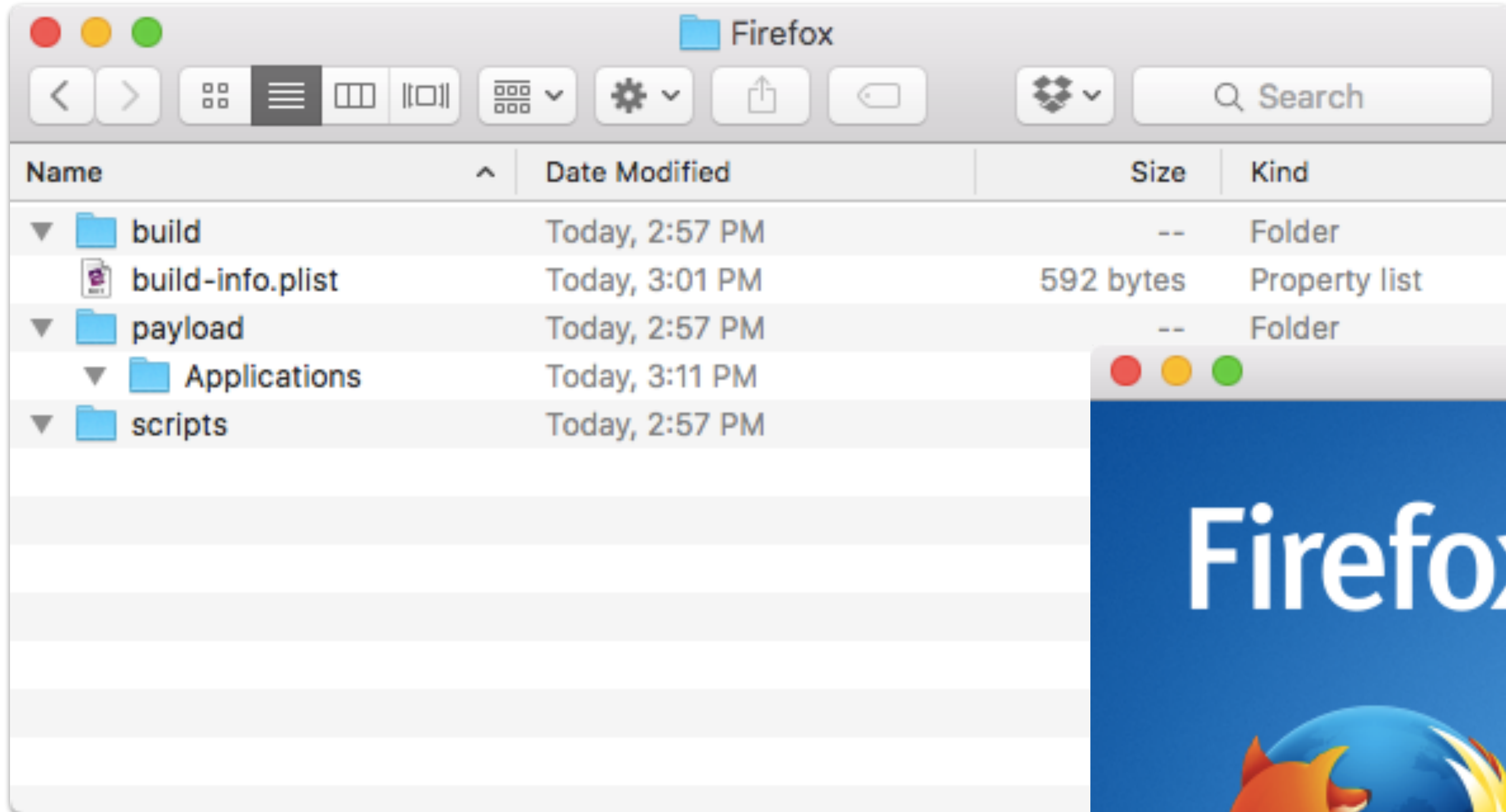
Finder sidebar showing a folder hierarchy:

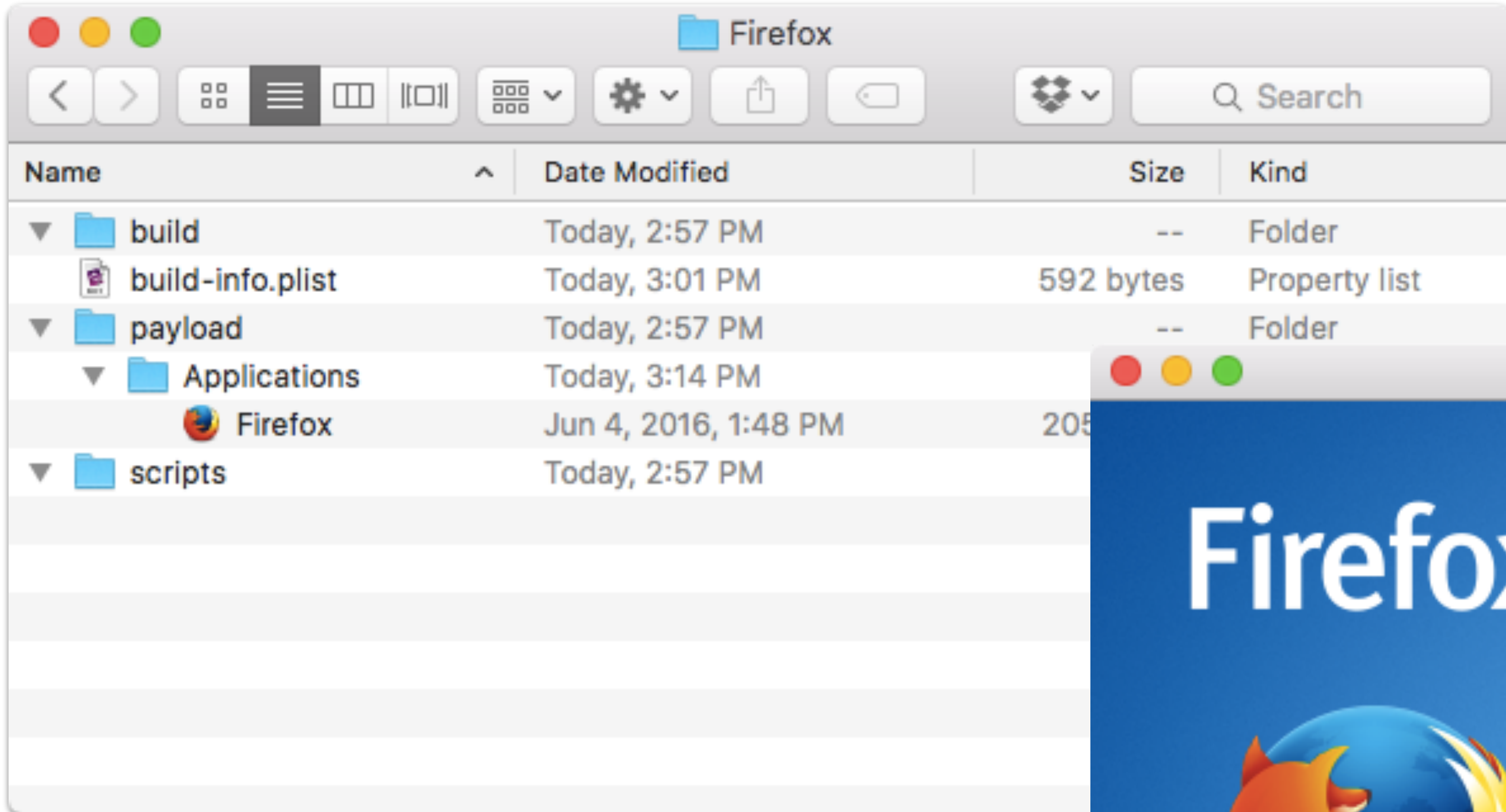
- build
- build-in
- payload** (selected)
- scripts

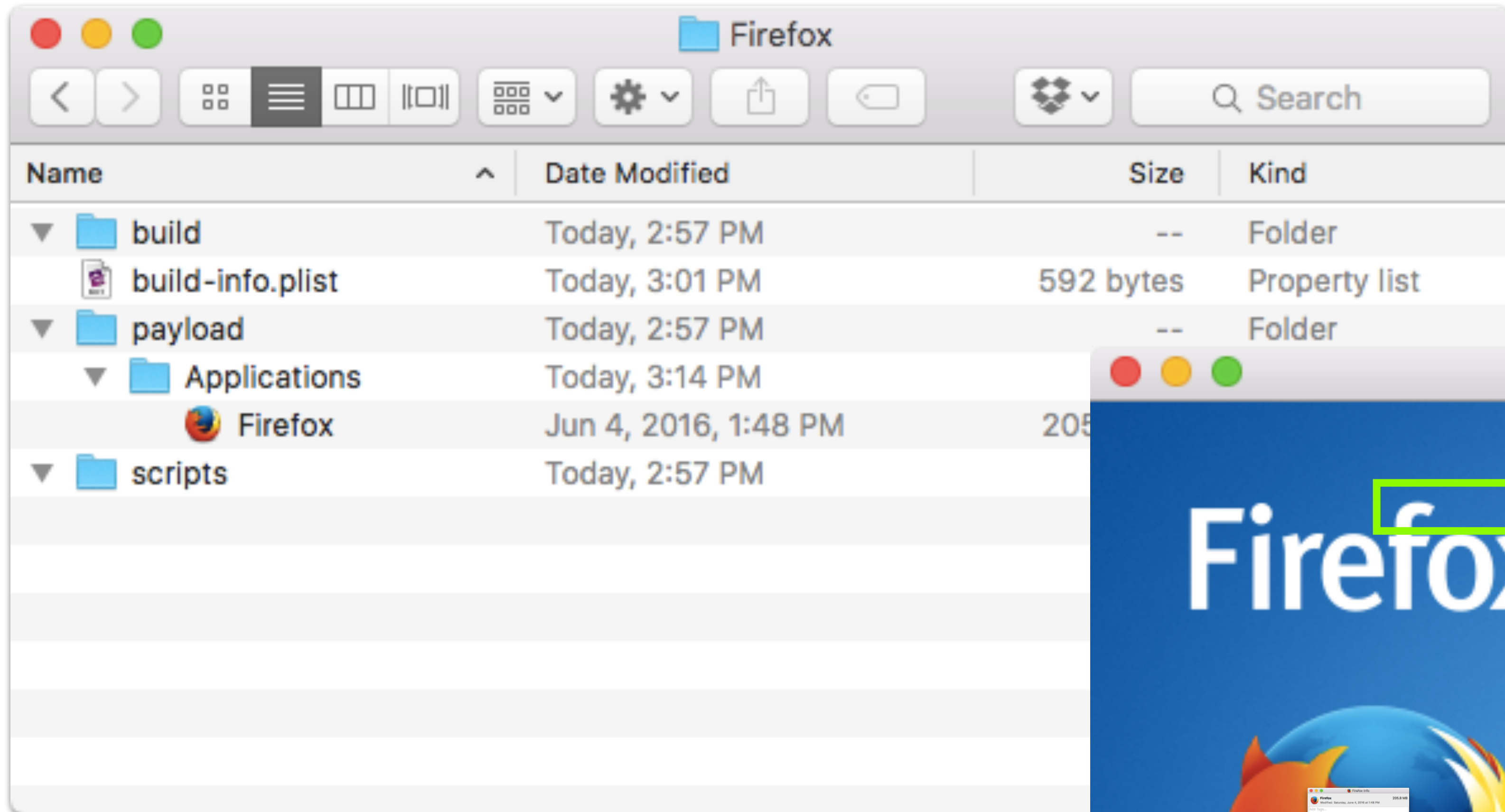
- New Finder Window ⌘N
- New Folder** ⇧⌘N
- New Folder with "payload" ^⌘N
- New Smart Folder ⌘⇧N
- New Tab ⌘T
- Open ⌘O
- Open With ▶
- Print ⌘P
- Close Window ⌘W
- Get Info ⌘I
- Rename
- Compress "payload"
- Duplicate ⌘D
- Make Alias ⌘L
- Quick Look "payload" ⌘Y
- Show Original
- Add to Dock ⇧⇧⌘T
- Move to Trash ⌘⇧⌘
- Eject ⌘E
- Find ⌘F
- Tags...
 - ● ● ● ● ● ●

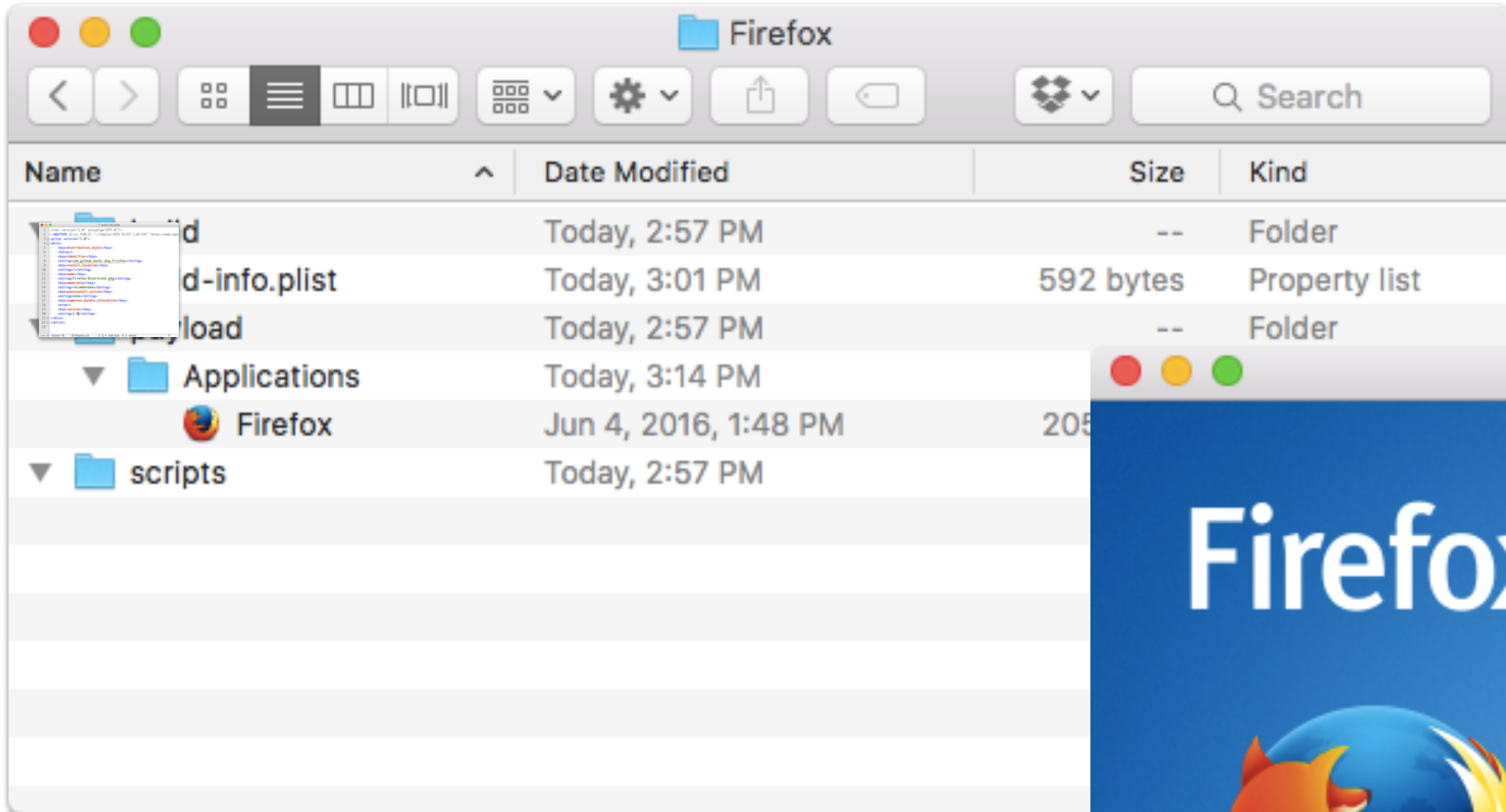
Finder context menu for a folder:

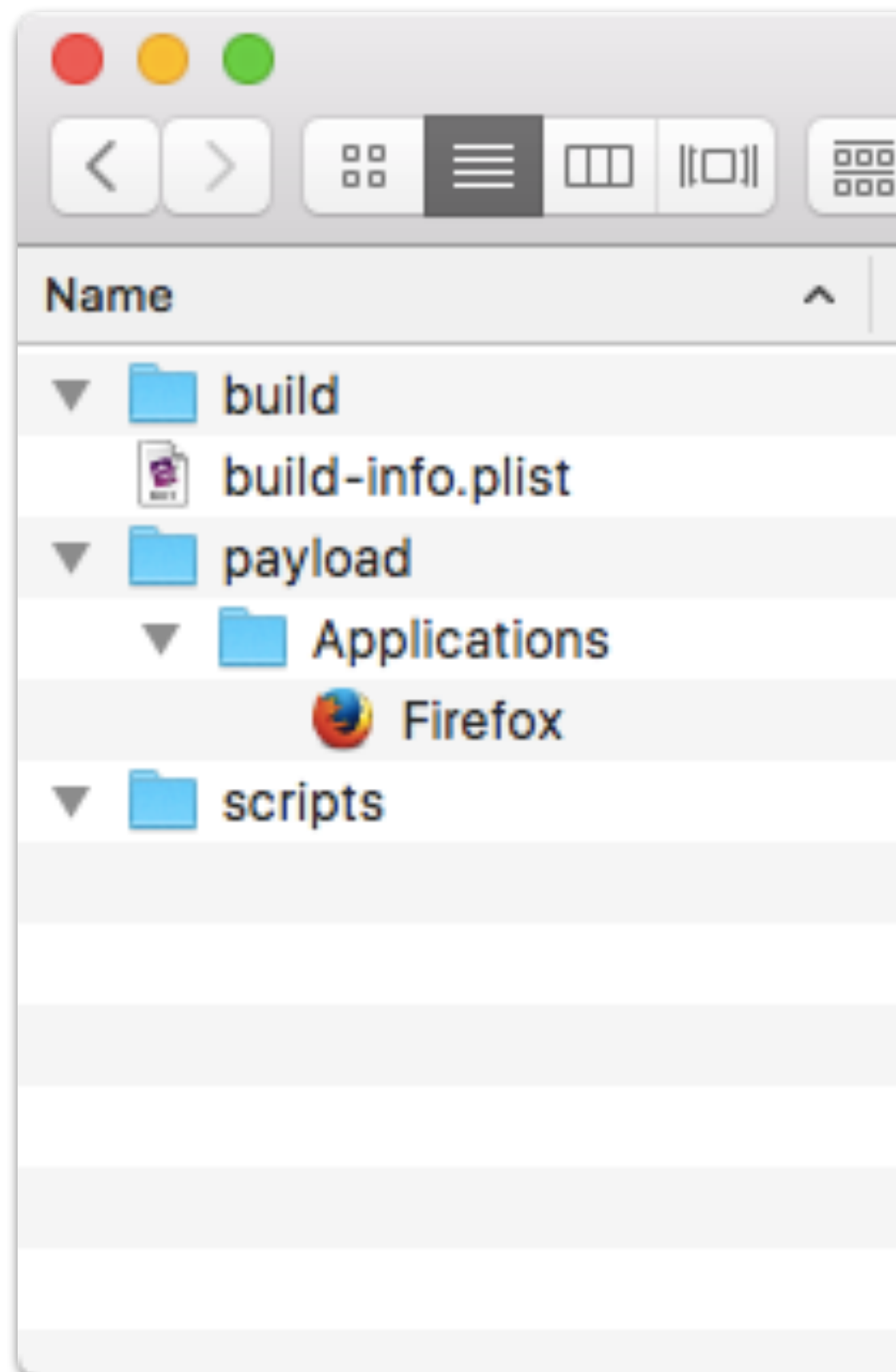
Size	Kind
--	Folder
592 bytes	Property list
--	Folder
--	Folder











```
build-info.plist
1 <?xml version="1.0" encoding="UTF-8"?>
2 <!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
3 <plist version="1.0">
4 <dict>
5   <key>distribution_style</key>
6   <false/>
7   <key>identifier</key>
8   <string>com.github.munki.pkg.Firefox</string>
9   <key>install_location</key>
10  <string>/</string>
11  <key>name</key>
12  <string>Firefox-${version}.pkg</string>
13  <key>ownership</key>
14  <string>recommended</string>
15  <key>postinstall_action</key>
16  <string>none</string>
17  <key>suppress_bundle_relocation</key>
18  <true/>
19  <key>version</key>
20  <string>47.0</string>
21 </dict>
22 </plist>
23
```

Line: 20 Column: 17 Property List Soft Tabs: 4 version



Building a package

Name	Date Modified	Size	Kind
build	Today, 4:28 PM	--	Folder
Firefox-47.0.pkg	Today, 3:49 PM	5 KB	Install...ackage
build-info.plist	Today, 3:01 PM	592 bytes	Property list
payload	Today, 3:57 PM	--	Folder
Applications	Today, 3:57 PM	--	Folder
Firefox	Jun 4, 2016, 1:48 PM	205.8 MB	Application
scripts	Today, 2:57 PM	--	Folder

```
gneagle@:/Users/Shared/munki-git/munki-pkg — -tcsh — 59x12
gneagle@munki-pkg % ./munkipkg Firefox
pkgbuild: Inferring bundle components from contents of Firefox/payload
pkgbuild: Writing new component property list to /var/folders/tc/sd4_mt...
pkgbuild: Reading components from /var/folders/tc/sd4_mt...
pkgbuild: Wrote package to Firefox/build/Firefox-47.0.pkg
gneagle@munki-pkg %
```

build-info

build-info.plist

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
  <key>distribution_style</key>
  <false/>
  <key>identifier</key>
  <string>com.github.munki.pkg.Foo</string>
  <key>install_location</key>
  <string>/</string>
  <key>name</key>
  <string>Foo-${version}.pkg</string>
  <key>ownership</key>
  <string>recommended</string>
  <key>postinstall_action</key>
  <string>none</string>
  <key>suppress_bundle_relocation</key>
  <true/>
  <key>version</key>
  <string>1.0</string>
</dict>
</plist>
```

build-info.json

```
{  
  "postinstall_action": "none",  
  "suppress_bundle_relocation": true,  
  "name": "Foo-${version}.pkg",  
  "distribution_style": false,  
  "install_location": "/",  
  "version": "1.0",  
  "ownership": "recommended",  
  "identifier": "com.github.munki.pkg.Foo"  
}
```

build-info.yaml

```
distribution_style: false
identifier: com.github.munki.pkg.Foo
install_location: /
name: Foo-${version}.pkg
ownership: recommended
postinstall_action: none
suppress_bundle_relocation: true
version: '1.0'
```


build-info keys

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
  <key>distribution_style</key>
  <false/>
  <key>identifier</key>
  <string>com.github.munki.pkg.Foo</string>
  <key>install_location</key>
  <string>/</string>
  <key>name</key>
  <string>Foo-${version}.pkg</string>
  <key>ownership</key>
  <string>recommended</string>
  <key>postinstall_action</key>
  <string>none</string>
  <key>suppress_bundle_relocation</key>
  <true/>
  <key>version</key>
  <string>1.0</string>
</dict>
</plist>
```

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
  <key>distribution_style</key>
  <false/>
  <key>identifier</key>
  <string>com.github.munki.pkg.Foo</string>
  <key>...</key>
  <string>...</string>
  <key>distribution_style</key>
  <string>...</string>
  <key>...</key>
  <string>recommended</string>
  <key>postinstall_action</key>
  <string>none</string>
  <key>suppress_bundle_relocation</key>
  <true/>
  <key>version</key>
  <string>1.0</string>
</dict>
</plist>
```

distribution_style:

If true, build as a “distribution-style” package using `productbuild`.

Otherwise, build as a component package using `pkgbuild`.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
  <key>distribution_style</key>
  <false/>
  <key>identifier</key>
  <string>com.github.munki.pkg.Foo</string>
  <key>install_location</key>
  <string>/</string>
  <key>
    identifier:
    String containing the package identifier.
    If this is missing, one is constructed using the name of the package
    project directory.
  </key>
  <key>suppress_bundle_relocation</key>
  <true/>
  <key>version</key>
  <string>1.0</string>
</dict>
</plist>
```

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
  <key>distribution_style</key>
  <false/>
  <key>identifier</key>
  <string>com.github.munki.pkg.Foo</string>
  <key>install_location</key>
  <string>/</string>
  <key>name</key>
  <string>Foo-${version}.pkg</string>
  <key>ownership</key>
  <string>install_location:
  <key> String. Path to the intended install location of the payload on the target
  <string> disk. Defaults to "/".
  <key>
  <true/>
  <key>version</key>
  <string>1.0</string>
</dict>
</plist>
```

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
  <key>distribution_style</key>
  <false/>
  <key>identifier</key>
  <string>com.github.munki.pkg.Foo</string>
  <key>install_location</key>
  <string>/</string>
  <key>name</key>
  <string>Foo- $\{version\}$ .pkg</string>
  <key>ownership</key>
  <string>recommended</string>
  <key>recommended_install_action</key>
  <string>name:
String containing the package name. If this is missing, one is
constructed using the name of the package project directory.
  <key>
  <true/>
  <key>
  <string>1.0</string>
</dict>
</plist>
```

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
  <key>ownership:
  <false>String. One of "recommended", "preserve", or "preserve-other".
  <key> Defaults to "recommended". See man pkgbuild for more info.
  <string>...
  <key>install_location</key>
  <string>/</string>
  <key>name</key>
  <string>Foo- $\{version\}$ .pkg</string>
  <key>ownership</key>
  <string>recommended</string>
  <key>postinstall_action</key>
  <string>none</string>
  <key>suppress_bundle_relocation</key>
  <true/>
  <key>version</key>
  <string>1.0</string>
</dict>
</plist>
```

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
  <key>postinstall_action:
  <false/>
  <key>String. One of "none", "logout", or "restart". Defaults to "none".
  <string>com.github.munki.pkg.Foo</string>
  <key>install_location</key>
  <string>/</string>
  <key>name</key>
  <string>Foo- $\{version\}$ .pkg</string>
  <key>ownership</key>
  <string>recommended</string>
  <key>postinstall_action</key>
  <string>none</string>
  <key>suppress_bundle_relocation</key>
  <true/>
  <key>version</key>
  <string>1.0</string>
</dict>
</plist>
```



```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
  <key>suppress_bundle_relocation:
  <false/>
  <key> Boolean: true or false. Defaults to true. If present and false, bundle
  <string> relocation will be allowed, which causes the Installer to update
  <key> bundles found in locations other than their default location.
  <string>
  <key>name</key>
  <string>Foo-{$version}.pkg</string>
  <key>ownership</key>
  <string>recommended</string>
  <key>postinstall_action</key>
  <string>none</string>
  <key>suppress_bundle_relocation</key>
  <true/>
  <key>version</key>
  <string>1.0</string>
</dict>
</plist>
```

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
  <key>version:
  <false/>
  <key>A string representation of the version number. Defaults to "1.0".
  <string>com.github.munki.pkg.Foo</string>
  <key>install_location</key>
  <string>/</string>
  <key>name</key>
  <string>Foo- $\{version\}$ .pkg</string>
  <key>ownership</key>
  <string>recommended</string>
  <key>postinstall_action</key>
  <string>none</string>
  <key>suppress_bundle_relocation</key>
  <true/>
  <key>version</key>
  <string>1.0</string>
</dict>
</plist>
```

Package signing

```
<key>signing_info</key>
<dict>
  <key>identity</key>
  <string>Signing Identity Common Name</string>
  <key>keychain</key>
  <string>/path/to/SpecialKeychain</string>
  <key>additional_cert_names</key>
  <array>
    <string>Intermediate CA Common Name 1</string>
    <string>Intermediate CA Common Name 2</string>
  </array>
  <key>timestamp</key>
  <true/>
</dict>
```

signing_info:

The only required key/value in the signing_info dictionary is 'identity'.

See the SIGNED PACKAGES section of the man page for **pkgbuild** or the SIGNED PRODUCT ARCHIVES section of the man page for **productbuild** for more information on the signing options.

```
<key>signing_info</key>
<dict>
  <key>identity</key>
  <string>Developer ID Installer: Greg Neagle (52ZFZKM6BK)</string>
</dict>
```

signing_info:

The only required key/value in the signing_info dictionary is 'identity'.

Importing existing
packages

```
munkipkg --import /path/to/foo.pkg Foo
```

- Creates a new package project directory named "Foo"
- Payload, scripts and build-info are extracted from foo.pkg.
- Complex or non-standard packages may not be extracted with 100% fidelity, and not all package formats are supported.

AutoPkg for Packages

AutoPkg

- Runs recipes to retrieve the latest versions of software
- Software can be packaged, imported to management tools, or simply output.
- Recipes can be shared with others via git

AutoPkg

- Install AutoPkg
- Search for `Firefox`
- Add `recipes` repository
- Run `Firefox.pkg` recipe

Demo

AutoPkg

<https://github.com/autopkg/autopkg>

<http://macadmins.psu.edu/conference/resources/>

Philosophy

Repackaging considered harmful

Why repackage?

- Vendor doesn't supply a deployable package
- Want to customize or configure what is installed

Why not repackage?

- Vendor may not support your installs
- It's tedious and difficult to do right
- It's easy to do wrong
- It's hard to test and replicate

Repackaging alternatives

- Install additional pkgs
- Edit vendor packages
- Post-install scripting
- AutoPkg

Common repackaging pitfalls

- “Snapshotting”:
 - Can miss important changes
 - Can include irrelevant changes
 - Hard to ensure actions of package scripts are replicated

Common repackaging pitfalls

- Can break updates unless you also repackage those

Why repackage?

- Vendor doesn't supply a deployable package

- ~~Want to customize or configure what is installed~~

Why repackage?

- Vendor doesn't supply a deployable package:
 - Not a supported pkg format
 - InstallEase, InstallAnywhere, InstallerVISE, etc
 - Installation application
 - Won't install "silently" (requires human interaction)
 - Won't install when there is no GUI user

Why package?

- Internal files, tools, configurations, apps that need to be deployed

If you must (re)package?

- Repeatable
- Testable
- Documented

- **munki-pkg**
<https://github.com/munki/munki-pkg>
- **The Luggage**
<https://github.com/unixorn/luggage/>
- **AutoPkg**
<http://autopkg.github.io/autopkg/>

- Scripts that use pkgbuild/productbuild
- Put these in Git (or other version control)

<http://managingosx.wordpress.com>