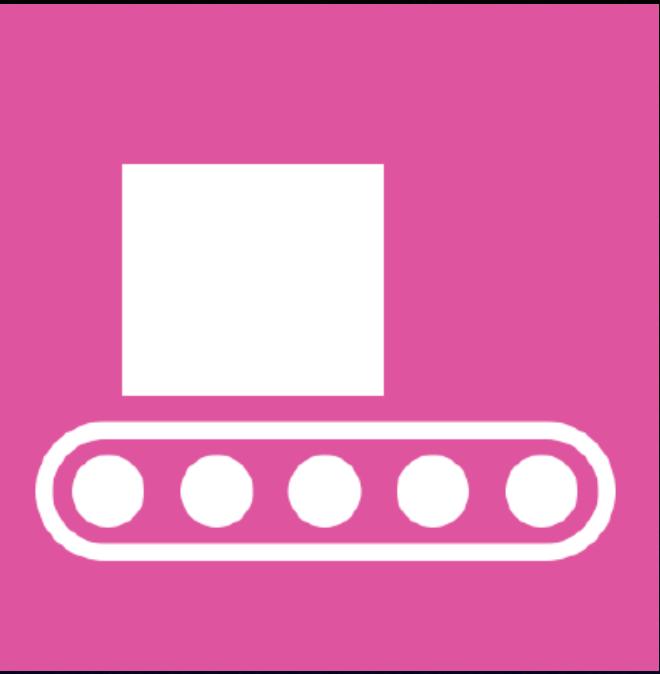


# Resources For This Session:



[github.com/jazzace/mactech-2019-autopkg](https://github.com/jazzace/mactech-2019-autopkg)

Anthony Reimer

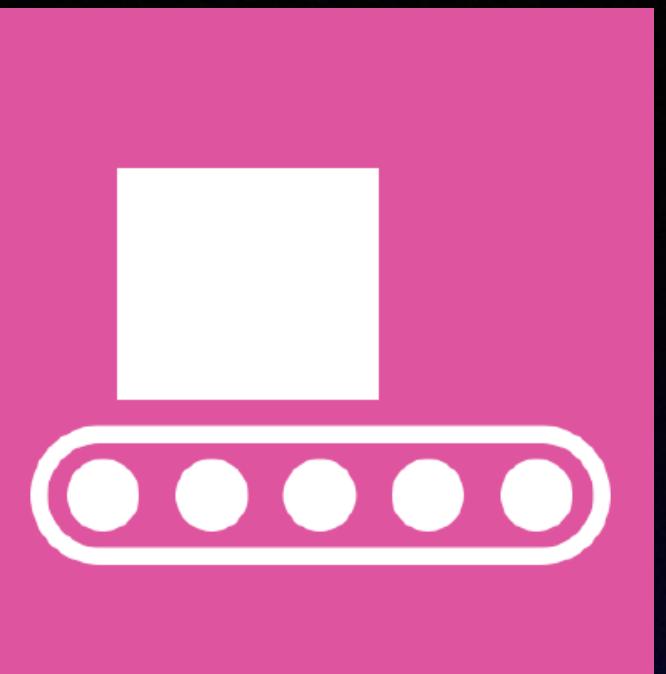
 [@AnthonyReimer](https://twitter.com/AnthonyReimer)

 [jazzace](https://github.com/jazzace)

 [jazzace](https://www.linkedin.com/in/jazzace)



# Writing and Understanding AutoPkg Recipes



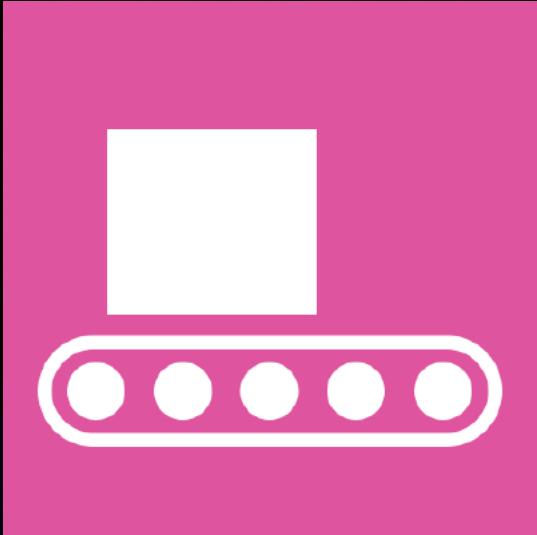
Anthony Reimer

 *AnthonyReimer*

 *jazzace*

 *jazzace*

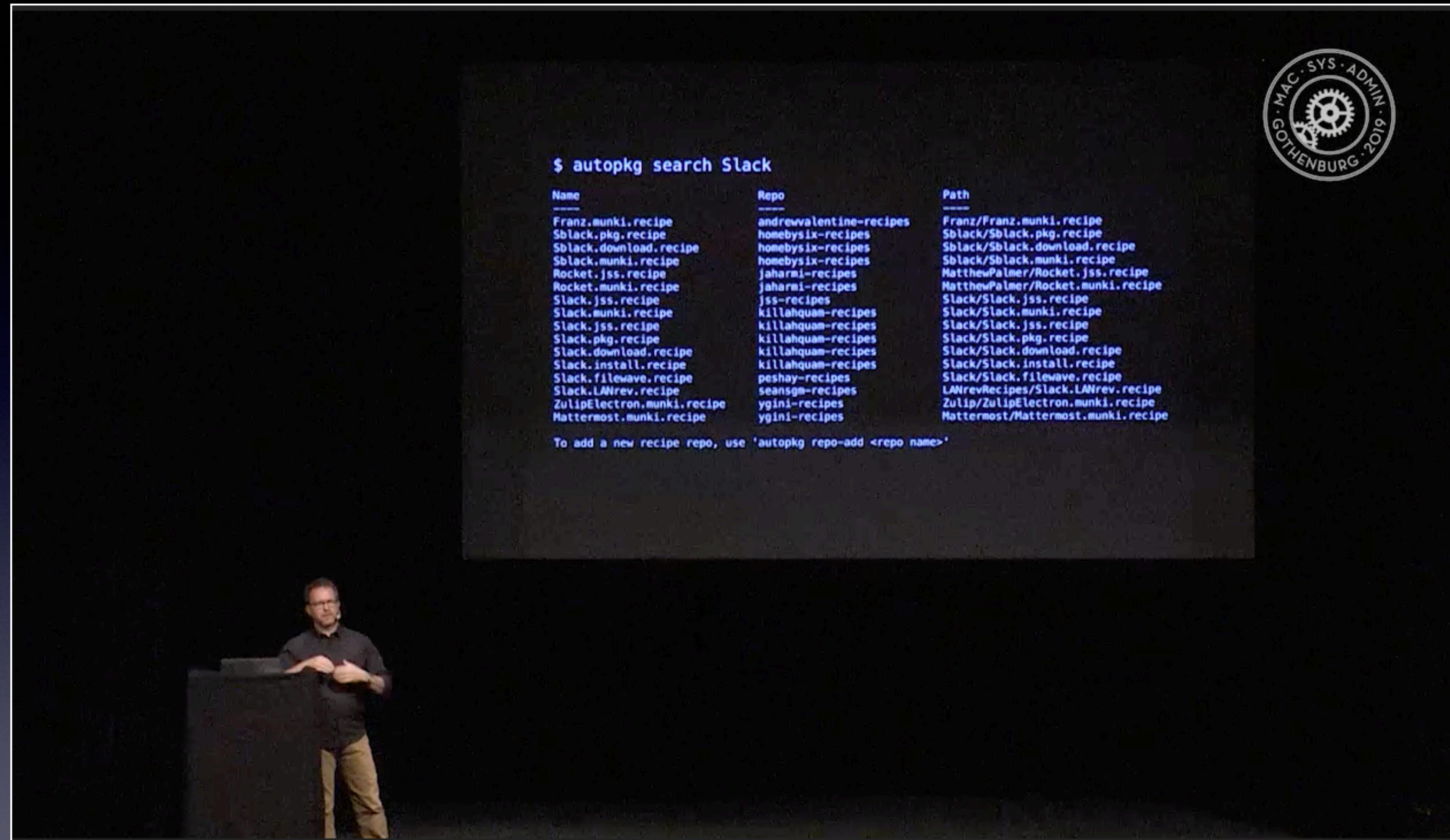




# The Line Up

1. Talking About Recipes
2. Creating Recipes
3. Testing
4. Case Study

# Survey says...



Greg Neagle – AutoPkg 2020  
[docs.macsysadmin.se/2019/](http://docs.macsysadmin.se/2019/)



# Talking About Recipes

## Self Saucing Fudge Dessert

When Joan serves this to company, she puts a scoop of ice cream on the pudding base before topping it with the sauce.

### Pudding Base

1 cup	flour	250 mL
2 tsp.	baking powder	10 mL
1/4 tsp.	salt	1 mL
1/2 cup	sugar	125 mL
3 tbsp.	cocoa	50 mL
1/2 cup	chopped nuts	125 mL
1/2 cup	milk	125 mL
1 tsp.	vanilla	125 mL
2 tbsp.	melted butter or margarine	5 mL
		25 mL

### Sauce

4 tbsp.	cocoa	65 mL
1/2 cup	sugar	125 mL
2 cups	hot water	500 mL

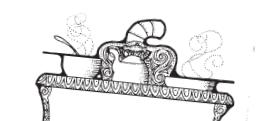
Sift into a two quart (litre) casserole, flour, baking powder, salt, 1/2 cup (125 mL) of sugar and 3 tablespoons (50 mL) of cocoa. Stir in the nuts. Make a well in the centre and add milk, vanilla and melted butter. Stir lightly and smooth evenly over bottom of the casserole.

Mix 1/2 cup (125 mL) of sugar and 4 tablespoons (65 mL) of cocoa. Sprinkle mixture over pudding base. Pour hot water over cocoa sugar mixture. Bake at 350°F (180°C) for 30 minutes. The batter rises and a rich sauce forms under it. Serve hot.

# Name → Description Ingredients Instructions

<b>Self Saucing Fudge Dessert</b>		
<i>When Joan serves this to company, she puts a scoop of ice cream on the pudding base before topping it with the sauce.</i>		
<b>Pudding Base</b>		
1 cup	flour	250 mL
2 tsp.	baking powder	10 mL
¼ tsp.	salt	1 mL
½ cup	sugar	125 mL
3 tbsp.	cocoa	50 mL
½ cup	chopped nuts	125 mL
½ cup	milk	125 mL
1 tsp.	vanilla	5 mL
2 tbsp.	melted butter or margarine	25 mL
<b>Sauce</b>		
4 tbsp.	cocoa	65 mL
½ cup	sugar	125 mL
2 cups	hot water	500 mL
Sift into a two quart (litre) casserole, flour, baking powder, salt, ½ cup (125 mL) of sugar and 3 tablespoons (50 mL) of cocoa. Stir in the nuts. Make a well in the centre and add milk, vanilla and melted butter. Stir lightly and smooth evenly over bottom of the casserole.		
Mix ½ cup (125 mL) of sugar and 4 tablespoons (65 mL) of cocoa. Sprinkle mixture over pudding base. Pour hot water over cocoa sugar mixture. Bake at 350°F (180°C) for 30 minutes. The batter rises and a rich sauce forms under it. Serve hot.		

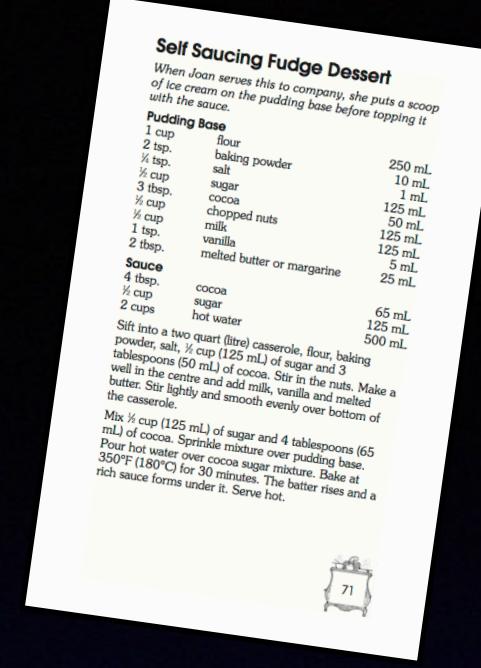
# Identifier Description Input Process



```
<?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>This recipe does nothing.</string>
    <key>Identifier</key>
    <string>com.github.yourgithubusername.download.nothing</string>
    <key>Input</key>
    <dict>
        <key>NAME</key>
        <string>nothing</string>
    </dict>
    <key>MinimumVersion</key>
    <string>1.0</string>
    <key>Process</key>
    <array>
        <dict>
            <key>Arguments</key>
            <dict>
                <key>Argument1</key>
                <string>Value1</string>
                <key>Argument2</key>
                <string>Value2</string>
            </dict>
            <key>Processor</key>
            <string>EndOfCheckPhase</string>
        </dict>
    </array>
</dict>
</plist>
```

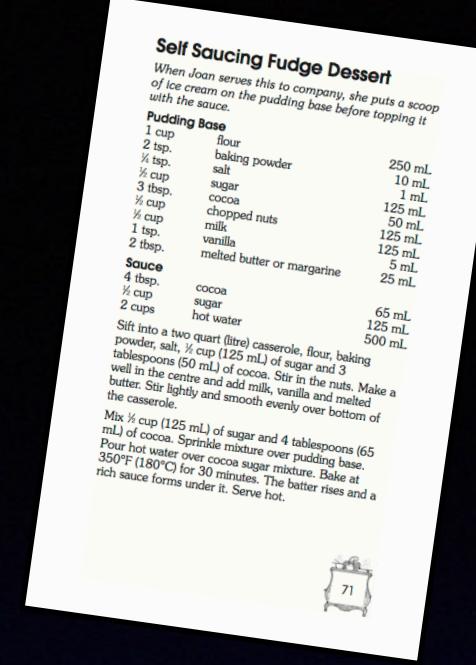
Doc Type Def

Description	<key>Description</key>
Identifier	<string>com.github.yourgithubusername.download.nothing</string>
Input	<key>Input</key>
Minimum Version	<key>MinimumVersion</key>
Processor(s)	<key>Processor</key>



# Categorizing Recipes

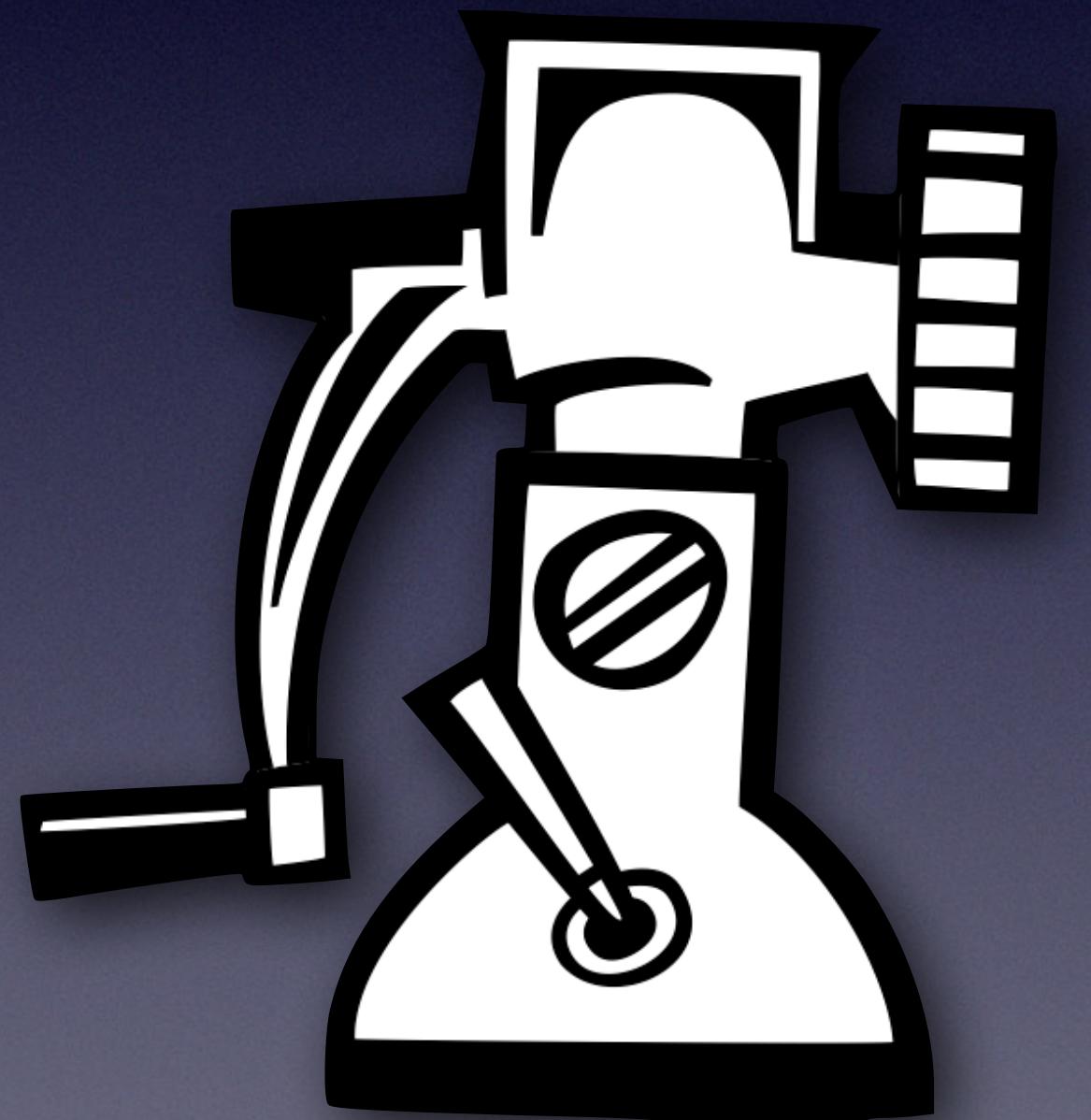
- By Function (End Result)
  - ▶ *download*
  - ▶ *pkg*
  - ▶ *munki, jss, filewave,...*
  
- By Relationship
  - ▶ *Parent*
  - ▶ *Child*
  - ▶ *Override*

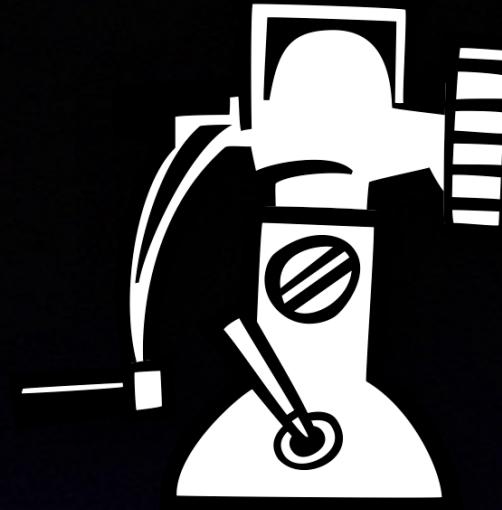


# Root-level Recipe Keys

Key	Parent	Child	Override
Description	✓	✓	
Identifier	✓	✓	✓
Input	optional	optional	optional
Minimum Version	✓	✓	
ParentRecipe	sometimes	✓	✓
Process	✓	✓	
ParentRecipeTrustInfo			✓

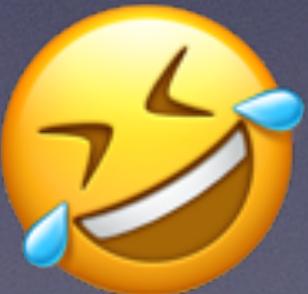
# Creating Recipes

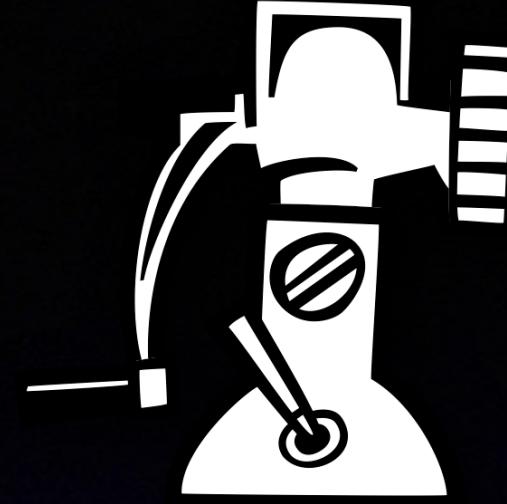




# Creating a New Recipe

- ~~Steal~~ Adapt an existing recipe
- Recipe Robot
  - ▶ *github.com/homebysix/recipe-robot*
- Write it from scratch

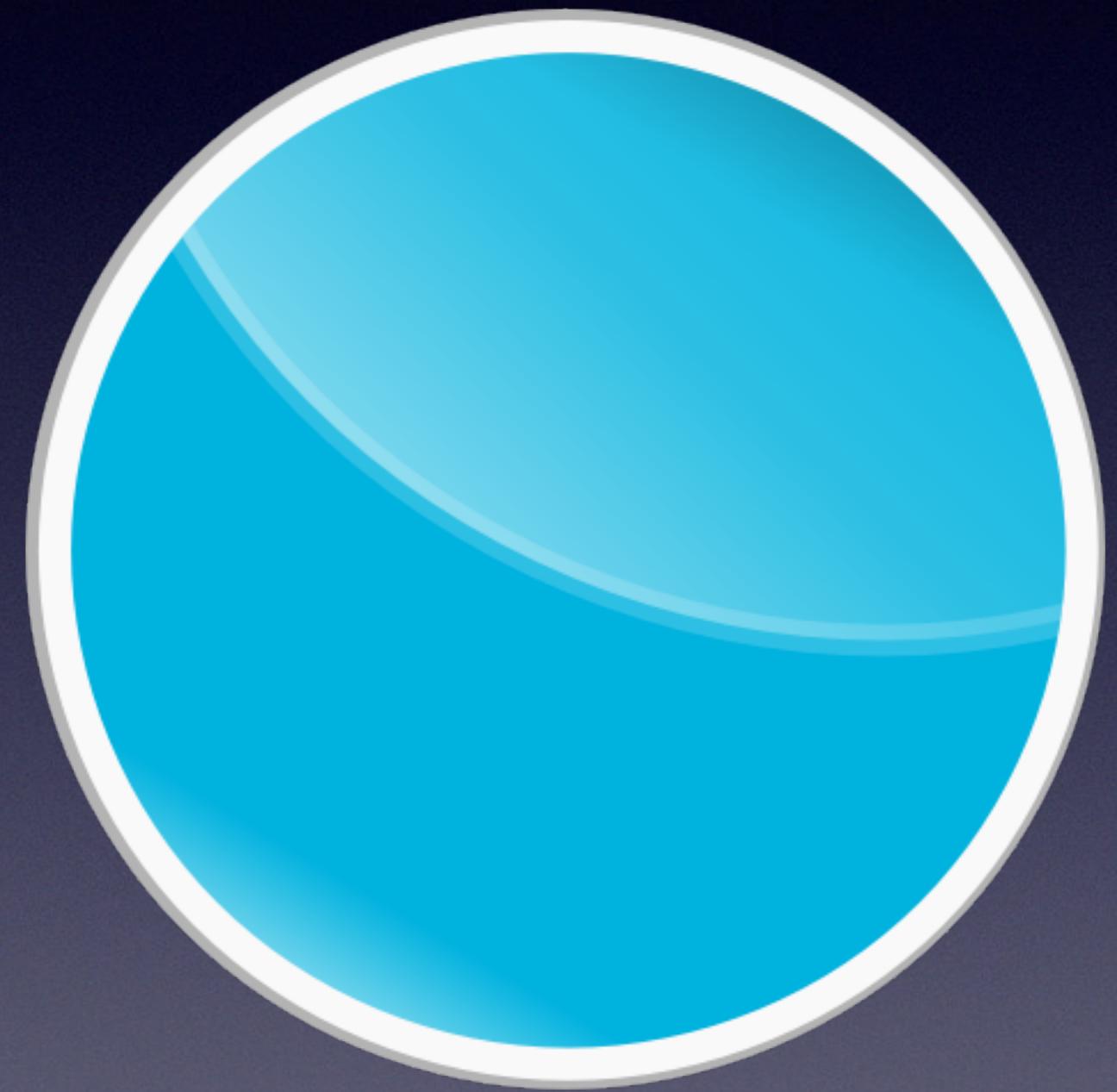




# Creating a New Recipe

- ~~Steal~~ Adapt an existing recipe
- Recipe Robot
  - ▶ *github.com/homebysix/recipe-robot*
- Write it from scratch
  - ✓ `autopkg new-recipe recipename.function.recipe`

# ClipGrab



<https://clipgrab.org>



# Does it already exist?

	July '14	June '16	June '18	Sep '19
Total Recipes	630	4079	5936	<b>7567</b>
.download	258	1156	1667	<b>2109</b>
.munki	267	1037	1537	<b>1944</b>
.pkg	95	794	1205	<b>1599</b>
.install	0	596	723	<b>837</b>
.jss	8	269	430	<b>551</b>
.filewave	0	14	171	<b>255</b>

```
$ autopkg search ClipGrab
```

```
$ autopkg search ClipGrab
```

Name	Repo	Path
ClipGrab.filewave.recipe	peshay-recipes	ClipGrab/ClipGrab.filewave.recipe
ClipGrab.download.recipe	peshay-recipes	ClipGrab/ClipGrab.download.recipe
ClipGrab.munki.recipe	wardsparadox-recipes	ClipGrab/ClipGrab.munki.recipe

To add a new recipe repo, use 'autopkg repo-add <repo name>'

```

<?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 latest version of ClipGrab.</string>                                Description
    <key>Identifier</key>
    <string>com.github.peshay.download.ClipGrab</string>                                Identifier
    <key>Input</key>
    <dict>
        <key>NAME</key>
        <string>ClipGrab</string>
        <key>os</key>

<!--SELECT YOUR OS VERSION AND INPUT BELOW
    "dmg" for MAC OS X
    "exe" for WINDOWS EXE
    "bz2" for LINUX
-->

        <string>dmg</string><!-- <<<<< INPUT HERE <<<< -->
    </dict>
    <key>MinimumVersion</key>
    <string>0.6.1</string>                                                               Minimum Version

```

<https://github.com/autopkg/peshay-recipes/blob/master/ClipGrab/ClipGrab.download.recipe>

```
<key>Process</key>
<array>
<dict>
    <key>Processor</key>
    <string>URLTextSearcher</string>
    <key>Arguments</key>
    <dict>
        <key>url</key>
        <string>https://clipgrab.org</string>
        <key>re_pattern</key>
        <string>https://download.clipgrab.org/(ClipGrab-[^"]*.\%os%)"</string>
        <key>re_flags</key>
        <array>
            <string>IGNORECASE</string>
        </array>
        <key>result_output_var_name</key>
        <string>url</string>
    </dict>
</dict>
<dict>
    <key>Processor</key>
    <string>URLDownloader</string>
    <key>Arguments</key>
    <dict>
        <key>url</key>
        <string>https://download.clipgrab.org/%url%</string>
    </dict>
</dict>
<dict>
    <key>Processor</key>
    <string>EndOfCheckPhase</string>
</dict>
</array>
</dict>
</plist>
```

URLTextSearcher

URLDownloader

EndOfCheckPhase

clipgrab.org

# ClipGrab

Download ClipGrab   FAQs & Support   Donate

A friendly downloader for YouTube and other sites

[Free Download!](#)  
ClipGrab 3.8.5 for Mac OS X

[Donate now!](#)  
4194 people have donated so far.

ClipGrab is a free downloader and converter for YouTube, Vimeo, Facebook and [many other online video sites](#).  
It converts downloaded videos to MPEG4, MP3 or [other formats](#) in just one easy step.  
[Show all download options](#)

FAQs & Support

[Howto: Downloading videos from YouTube?](#)  
[Howto: Downloading videos from Facebook?](#)  
[Howto: Saving YouTube videos on iPad/iPhone](#)  
[Supported video sites](#)  
[Supported output formats](#)

More questions or feedback? Take a look at the [ClipGrab FAQs](#), visit the [forums](#) or drop us an [email](#)!

Follow ClipGrab on Twitter   Legal mentions   vanbittern

<https://clipgrab.org>

Web Inspector — clipgrab.org

Elements Network Debugger Resources Timelines Storage Canvas Audit Console + ⚙

E > E > E section#... > E section.with-sidebar > E article > E section.buttons > E a#download-button.button.download.download-with-icon

```
<!DOCTYPE html>
<html lang="en">
  <head>...
  </head>
  <body>
    <header>...
    <section id="main">
      <section class="with-sidebar">
        <article>
          <h2>A friendly downloader for YouTube and other sites</h2>
          <section class="buttons">
            <a id="download-button" class="button download download-with-icon" href="https://download.clipgrab.org/ClipGrab-3.8.5.dmg" = $0>
              <span class="label">Free Download!</span>
              <span class="details">ClipGrab 3.8.5 for Mac OS X</span>
            </a>
            <a class="button donate" href="/donate">...</a>
          </section>
          <p>...</p>
          <a href="#" id="show-options">Show all download options</a>
        </article>
        <aside>...</aside>
        ::after
      </section>
      <div id="additional-info" style="display: none;">...</div>
      ::after
    </section>
    <footer>...
      <script type="text/javascript" defer async src="https://tracking.vanbittern.com/piwik.js"></script>
      <script src="//ajax.googleapis.com/ajax/libs/jquery/1.9.1/jquery.min.js"></script>
      <script>...
        <!-- Piwik -->
      <script type="text/javascript">...
        <!-- End Piwik Code -->
      </script>
    </footer>
  </body>
```

href="https://download.clipgrab.org/ClipGrab-3.8.5.dmg">

```
<dict>
    <key>Processor</key>
    <string>URLTextSearcher</string>
    <key>Arguments</key>
    <dict>
        <key>url</key>
        <string>https://clipgrab.org</string>
        <key>re_pattern</key>
        <string>https://download.clipgrab.org/(ClipGrab-[^"]*.\%os%)"</string>
        <key>re_flags</key>
        <array>
            <string>IGNORECASE</string>
        </array>
        <key>result_output_var_name</key>
        <string>url</string>
    </dict>
</dict>
```

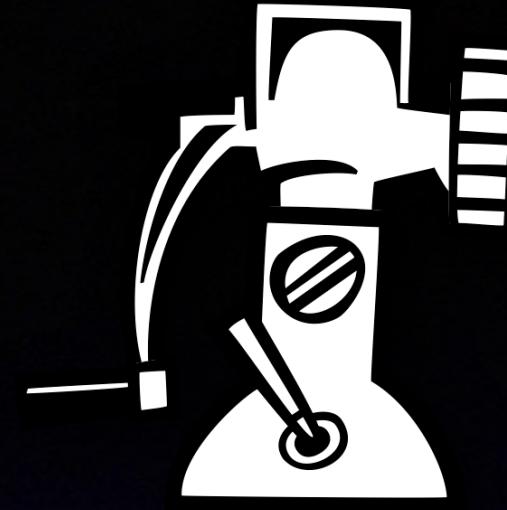
```
<key>Process</key>
<array>
  <dict>
    <key>Processor</key>
    <string>URLTextSearcher</string>
    <key>Arguments</key>
    <dict>
      <key>url</key>
      <string>https://clipgrab.org</string>
      <key>re_pattern</key>
      <string>https://download.clipgrab.org/(ClipGrab-[^"]*.\%os%)"</string>
      <key>re_flags</key>
      <array>
        <string>IGNORECASE</string>
      </array>
      <key>result_output_var_name</key>
      <string>url</string>
    </dict>
  </dict>
  <dict>
    <key>Processor</key>
    <string>URLDownloader</string>
    <key>Arguments</key>
    <dict>
      <key>url</key>
      <string>https://download.clipgrab.org/%url%</string>
    </dict>
  </dict>
  <dict>
    <key>Processor</key>
    <string>EndOfCheckPhase</string>
  </dict>
</array>
</dict>
</plist>
```

**URLDownloader**



# Common Download Recipe Workflow

1. Determine the URI of the item we want to download
2. Download a copy of the item
3. EndOfCheckPhase
4. Check the code signature of the downloaded item



# Choices, Choices, Choices

- Use as is
- Create a Pull Request (PR) with CodeSignatureVerifier added
- Write our own recipe
- Write a child recipe with CodeSignatureVerifier added

```
$ autopkg new-recipe ClipGrabMac.download.recipe  
-p com.github.peshay.download.ClipGrab  
-i com.github.jazzace.download.ClipGrabMac
```

```
<?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>Recipe description</string>
    <key>Identifier</key>
    <string>com.github.jazzace.download.ClipGrabMac</string>
    <key>Input</key>
    <dict>
        <key>NAME</key>
        <string>ClipGrabMac</string>
    </dict>
    <key>MinimumVersion</key>
    <string>1.0</string>
    <key>ParentRecipe</key>
    <string>com.github.peshay.download.ClipGrab</string>
    <key>Process</key>
    <array>
        <dict>
            <key>Arguments</key>
            <dict>
                <key>Argument1</key>
                <string>Value1</string>
                <key>Argument2</key>
                <string>Value2</string>
            </dict>
            <key>Processor</key>
            <string>ProcessorName</string>
        </dict>
    </array>
</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>Description</key>
    <string>Downloads the latest copy of ClipGrab for Mac and verifies the code signature.  
Requires peshay-recipes.</string>
    <key>Identifier</key>
    <string>com.github.jazzace.download.ClipGrabMac</string>
    <key>Input</key>
    <dict>
        <key>NAME</key>
        <string>ClipGrabMac</string>
    </dict>
    <key>MinimumVersion</key>
    <string>1.0</string>
    <key>ParentRecipe</key>
    <string>com.github.peshay.download.ClipGrab</string>
    <key>Process</key>
    <array>
        <dict>
            <key>Arguments</key>
            <dict>
                <key>Argument1</key>
                <string>Value1</string>
                <key>Argument2</key>
                <string>Value2</string>
            </dict>
            <key>Processor</key>
            <string>ProcessorName</string>
        </dict>
    </array>
</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>Description</key>
    <string>Downloads the latest copy of ClipGrab for Mac and verifies the code signature.  
Requires peshay-recipes.</string>
    <key>Identifier</key>
    <string>com.github.jazzace.download.ClipGrabMac</string>
    <key>Input</key>
    <dict>
        <key>NAME</key>
        <string>ClipGrab</string>
        <key>os</key>
        <string>dmg</string>
    </dict>
    <key>MinimumVersion</key>
    <string>1.0</string>
    <key>ParentRecipe</key>
    <string>com.github.peshay.download.ClipGrab</string>
    <key>Process</key>
    <array>
        <dict>
            <key>Arguments</key>
            <dict>
                <key>Argument1</key>
                <string>Value1</string>
                <key>Argument2</key>
                <string>Value2</string>
            </dict>
            <key>Processor</key>
            <string>ProcessorName</string>
        </dict>
    </array>
</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>Description</key>
    <string>Downloads the latest copy of ClipGrab for Mac and verifies the code signature.  
Requires peshay-recipes.</string>
    <key>Identifier</key>
    <string>com.github.jazzace.download.ClipGrabMac</string>
    <key>Input</key>
    <dict>
        <key>NAME</key>
        <string>ClipGrab</string>
        <key>os</key>
        <string>dmg</string>
    </dict>
    <key>MinimumVersion</key>
    <string>1.0</string>
    <key>ParentRecipe</key>
    <string>com.github.peshay.download.ClipGrab</string>
    <key>Process</key>
    <array>
        <dict>
            <key>Arguments</key>
            <dict>
                <key>Argument1</key>
                <string>Value1</string>
                <key>Argument2</key>
                <string>Value2</string>
            </dict>
            <key>Processor</key>
            <string>CodeSignatureVerifier</string>
        </dict>
    </array>
</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>Description</key>
    <string>Downloads the latest copy of ClipGrab for Mac and verifies the code signature.  
Requires peshay-recipes.</string>
    <key>Identifier</key>
    <string>com.github.jazzace.download.ClipGrabMac</string>
    <key>Input</key>
    <dict>
        <key>NAME</key>
        <string>ClipGrab</string>
        <key>os</key>
        <string>dmg</string>
    </dict>
    <key>MinimumVersion</key>
    <string>1.0</string>
    <key>ParentRecipe</key>
    <string>com.github.peshay.download.ClipGrab</string>
    <key>Process</key>
    <array>
        <dict>
            <key>Processor</key>
            <string>CodeSignatureVerifier</string>
            <key>Arguments</key>
            <dict>
                <key>Argument1</key>
                <string>Value1</string>
                <key>Argument2</key>
                <string>Value2</string>
            </dict>
        </dict>
    </array>
</dict>
</plist>
```



# Retrieve Code Signature

- Application Bundle (.app)
  - ▶ `codesign --display -r- --deep /path/to/the.app`
  - ▶ *If .zip, unarchive first (Unarchiver processor in recipe)*
- Package Installer (.pkg)
  - ▶ `pkgutil --check-signature /path/to/the.pkg`

<https://github.com/autopkg/autopkg/wiki/Using-CodeSignatureVerification/>



```
$ codesign --display -r- --deep -v /Volumes/ClipGrab\ 3.8.5/ClipGrab.app
```

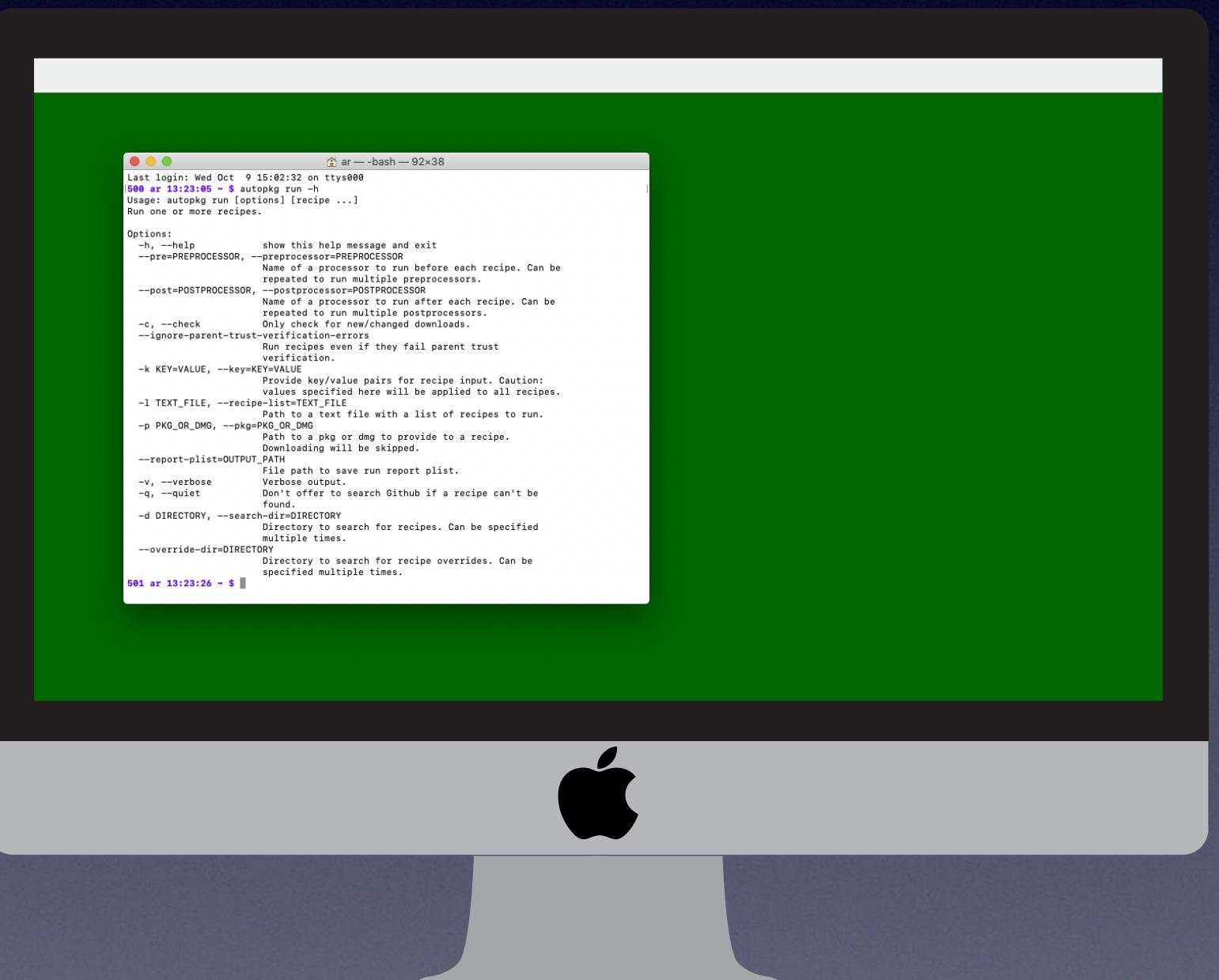
```
$ codesign --display -r- --deep -v /Volumes/ClipGrab\ 3.8.5/ClipGrab.app
Executable=/Volumes/ClipGrab 3.8.5/ClipGrab.app/Contents/MacOS/ClipGrab
Identifier=de.clipgrab.ClipGrab
Format=app bundle with Mach-O thin (x86_64)
CodeDirectory v=20200 size=10036 flags=0x0(none) hashes=308+3 location=embedded
Signature size=9015
Timestamp=Feb 28, 2019 at 12:25:30 AM
Info.plist entries=10
TeamIdentifier=E8BJ3ZV5W8
Sealed Resources version=2 rules=13 files=31
Nested=Frameworks/QtWebEngineCore.framework
Nested=PlugIns/imageformats/libqmacjp2.dylib
Nested=PlugIns/imageformats/libqgif.dylib
Nested=PlugIns/imageformats/libqico.dylib
Nested=Frameworks/QtPositioning.framework
Nested=PlugIns/imageformats/libqtga.dylib
Nested=Frameworks/QtWidgets.framework
Nested=Frameworks/QtSvg.framework
Nested=PlugIns/platforms/libqcocoa.dylib
Nested=Frameworks/QtCore.framework
Nested=Frameworks/QtXml.framework
Nested=Frameworks/QtPrintSupport.framework
Nested=Frameworks/QtWebEngineWidgets.framework
Nested=Frameworks/QtWebChannel.framework
Nested=PlugIns/imageformats/libqwbmp.dylib
Nested=PlugIns/imageformats/libqwepp.dylib
Nested=PlugIns/printsupport/libcocoaprintersupport.dylib
Nested=Frameworks/QtQml.framework
Nested=Frameworks/QtQuick.framework
Nested=Frameworks/QtGui.framework
Nested=MacOS/ffmpeg
Nested=Frameworks/QtNetwork.framework
Nested=PlugIns/imageformats/libqicns.dylib
Nested=PlugIns/bearer/libqgenericbearer.dylib
Nested=PlugIns/imageformats/libqjpeg.dylib
Nested=Frameworks/QtQuickWidgets.framework
Nested=PlugIns/imageformats/libqtiff.dylib
Nested=PlugIns/iconengines/libqsvgicon.dylib
```

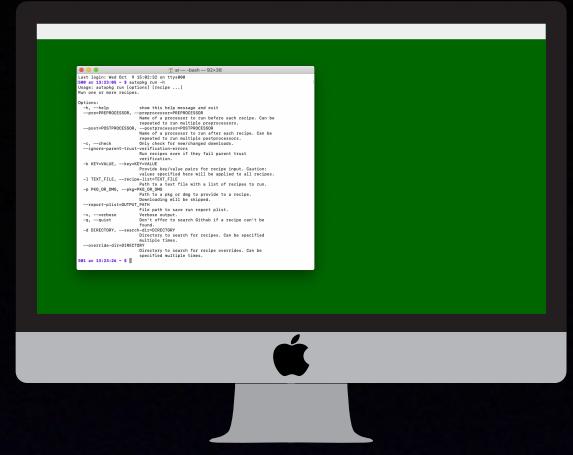
designated => identifier "de.clipgrab.ClipGrab" and anchor apple generic and certificate  
1[field.1.2.840.113635.100.6.2.6] /\* exists \*/ and certificate  
leaf[field.1.2.840.113635.100.6.1.13] /\* exists \*/ and certificate leaf[subject.OU] =  
E8BJ3ZV5W8

```
<dict>
    <key>Description</key>
    <string>Downloads the latest copy of ClipGrab for Mac and verifies the code signature.
Requires peshay-recipes.</string>
    <key>Identifier</key>
    <string>com.github.jazzace.download.ClipGrabMac</string>
    <key>Input</key>
    <dict>
        <key>NAME</key>
        <string>ClipGrab</string>
        <key>os</key>
        <string>dmg</string>
    </dict>
    <key>MinimumVersion</key>
    <string>1.0</string>
    <key>ParentRecipe</key>
    <string>com.github.peshay.download.ClipGrab</string>
    <key>Process</key>
    <array>
        <dict>
            <key>Processor</key>
            <string>CodeSignatureVerifier</string>
            <key>Arguments</key>
            <dict>
                <key>input_path</key>
                <string>%pathname%/*.app</string>
                <key>requirement</key>
                <string>identifier "de.clipgrab.ClipGrab" and anchor apple generic and 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] = E8BJ3ZV5W8</string>
            </dict>
        </dict>
    </array>
</dict>
</plist>
```

```
<dict>
    <key>Description</key>
    <string>Downloads the latest copy of ClipGrab for Mac and verifies the code signature.
Requires peshay-recipes.</string>
    <key>Identifier</key>
    <string>com.github.jazzace.download.ClipGrabMac</string>
    <key>Input</key>
    <dict>
        <key>NAME</key>
        <string>ClipGrab</string>
        <key>os</key>
        <string>dmg</string>
    </dict>
    <key>MinimumVersion</key>
    <string>1.0</string>
    <key>ParentRecipe</key>
    <string>com.github.peshay.download.ClipGrab</string>
    <key>Process</key>
    <array>
        <dict>
            <key>Processor</key>
            <string>CodeSignatureVerifier</string>
            <key>Arguments</key>
            <dict>
                <key>input_path</key>
                <string>%pathname%/*.app</string>
                <key>requirement</key>
                <string>identifier "de.clipgrab.ClipGrab" and anchor apple generic and 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] = E8BJ3ZV5W8</string>
            </dict>
        </dict>
    </array>
</dict>
</plist>
```

# Testing 1, 2, 3





# Testing 1, 2, 3

1. Test the parent
2. Test my new child recipe
3. Test it again

```
$ autopkg run -h
```

# \$ autopkg run -h

Usage: autopkg run [options] [recipe ...]  
Run one or more recipes.

## Options:

- h, --help show this help message and exit
- pre=PREPROCESSOR, --preprocessor=PREPROCESSOR Name of a processor to run before each recipe. Can be repeated to run multiple preprocessors.
- post=POSTPROCESSOR, --postprocessor=POSTPROCESSOR Name of a processor to run after each recipe. Can be repeated to run multiple postprocessors.
- c, --check Only check for new/changed downloads.
- ignore-parent-trust-verification-errors Run recipes even if they fail parent trust verification.
- k KEY=VALUE, --key=KEY=VALUE Provide key/value pairs for recipe input. Caution: values specified here will be applied to all recipes.
- l TEXT\_FILE, --recipe-list=TEXT\_FILE Path to a text file with a list of recipes to run.
- p PKG\_OR\_DMG, --pkg=PKG\_OR\_DMG Path to a pkg or dmg to provide to a recipe. Downloading will be skipped.
- report-plist=OUTPUT\_PATH File path to save run report plist.
- v, --verbose Verbose output.
- q, --quiet Don't offer to search Github if a recipe can't be found.
- d DIRECTORY, --search-dir=DIRECTORY Directory to search for recipes. Can be specified multiple times.
- override-dir=DIRECTORY Directory to search for recipe overrides. Can be specified multiple times.

## Options:

--ignore-parent-trust-verification-errors

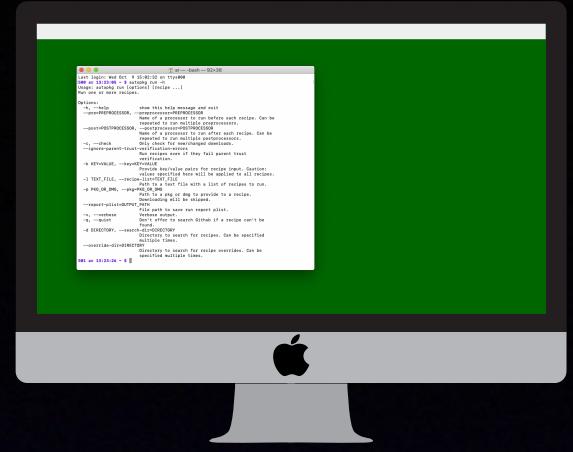
Run recipes even if they fail parent trust verification.

-p PKG\_OR\_DMG, --pkg=PKG\_OR\_DMG

Path to a pkg or dmg to provide to a recipe.  
Downloading will be skipped.

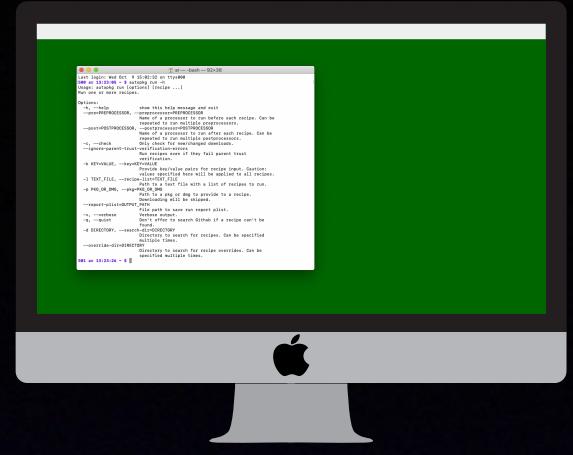
-v, --verbose

Verbose output.



# Strict Trust Verification

- Set Preference
  - ▶ `defaults write com.github.autopkg FAIL_RECIPES_WITHOUT_TRUST_INFO -bool YES`
- Wiki
  - ▶ *<https://github.com/autopkg/autopkg/wiki/AutoPkg-and-recipe-parent-trust-info>*



# Ignore Trust Info

- Persistent
  - ▶ defaults write com.github.autopkg FAIL\_RECIPES\_WITHOUT\_TRUST\_INFO -bool NO
  - ▶ defaults delete com.github.autopkg FAIL\_RECIPES\_WITHOUT\_TRUST\_INFO
- Per Run
  - ▶ autopkg run -vv --ignore-parent-trust-verification-errors *Product.pkg*

```
$ alias aprnt='autopkg run -vv --ignore-parent-trust-  
verification-errors'
```



# autopkg run -p (or --pkg)

- Acts as the result of the recipe download mechanism; downloading skipped
- pkg or dmg

```
$ autopkg run -vv --ignore-parent-trust-verification-errors  
ClipGrab.download.recipe
```

```
$ autopkg run -vv --ignore-parent-trust-verification-errors  
ClipGrab.download.recipe
```

Processing ClipGrab.download.recipe...

WARNING: ClipGrab.download.recipe is missing trust info and FAIL\_RECIPES\_WITHOUT\_TRUST\_INFO is not set. Proceeding...

## URLTextSearcher

```
{'Input': {'re_flags': (  
    IGNORECASE  
)},  
    're_pattern': u'https://download.clipgrab.org/(ClipGrab-[^"]*\\.dmg)"',  
    'result_output_var_name': u'url',  
    'url': u'https://clipgrab.org'}}
```

URLTextSearcher: No value supplied for CURL\_PATH, setting default value of: /usr/bin/curl

URLTextSearcher: Found matching text (url): ClipGrab-3.8.5.dmg

```
{'Output': {u'url': 'ClipGrab-3.8.5.dmg'}}
```

## URLDownloader

```
{'Input': {'CURL_PATH': '/usr/bin/curl',  
    'url': u'https://download.clipgrab.org/ClipGrab-3.8.5.dmg'}}
```

URLDownloader: No value supplied for CHECK\_FILESIZE\_ONLY, setting default value of: False

URLDownloader: Storing new Last-Modified header: Mon, 16 Sep 2019 20:18:27 GMT

URLDownloader: Storing new ETag header: "492c139-592b14e796ac0"

**URLDownloader: Downloaded** /Users/localadmin/Library/AutoPkg/Cache/com.github.peshay.download.ClipGrab/downloads/ClipGrab-3.8.5.dmg

```
{'Output': {'download_changed': True,  
    'etag': '"492c139-592b14e796ac0"',  
    'last_modified': 'Mon, 16 Sep 2019 20:18:27 GMT',  
    'pathname': u'/Users/localadmin/Library/AutoPkg/Cache/com.github.peshay.download.ClipGrab/downloads/ClipGrab-3.8.5.dmg',  
    'url_downloader_summary_result': {'data': {'download_path': u'/Users/localadmin/Library/AutoPkg/Cache/  
com.github.peshay.download.ClipGrab/downloads/ClipGrab-3.8.5.dmg'},  
        'summary_text': 'The following new items were downloaded:'}}}
```

## EndOfCheckPhase

```
{'Input': {}}  
'Output': {}
```

Receipt written to /Users/localadmin/Library/AutoPkg/Cache/com.github.peshay.download.ClipGrab/receipts/ClipGrab.download-  
receipt-20190930-112220.plist

**The following new items were downloaded:**

Download Path

---

/Users/localadmin/Library/AutoPkg/Cache/com.github.peshay.download.ClipGrab/downloads/ClipGrab-3.8.5.dmg

```
$ autopkg run -vv --ignore-parent-trust-verification-errors ClipGrabMac.download.recipe
```

```
$ autopkg run -vv --ignore-parent-trust-verification-errors ClipGrabMac.download.recipe
```

Processing ClipGrabMac.download...

WARNING: ClipGrabMac.download is missing trust info and FAIL\_RECIPES\_WITHOUT\_TRUST\_INFO is not set. Proceeding...

## URLTextSearcher

```
{'Input': {'re_flags': (IGNORECASE),
        're_pattern': u'https://download.clipgrab.org/(ClipGrab-[^"]*\.dmg)"',
        'result_output_var_name': 'url',
        'url': u'https://clipgrab.org'}}}
```

URLTextSearcher: No value supplied for CURL\_PATH, setting default value of: /usr/bin/curl

URLTextSearcher: Found matching text (url): ClipGrab-3.8.5.dmg

```
{'Output': {'url': 'ClipGrab-3.8.5.dmg'}}
```

## URLDownloader

```
{'Input': {'CURL_PATH': '/usr/bin/curl',
        'url': u'https://download.clipgrab.org/ClipGrab-3.8.5.dmg'}}
```

URLDownloader: No value supplied for CHECK\_FILESIZE\_ONLY, setting default value of: False

URLDownloader: Storing new Last-Modified header: Mon, 16 Sep 2019 20:18:27 GMT

URLDownloader: Storing new ETag header: "492c139-592b14e796ac0"

**URLDownloader: Downloaded** /Users/localadmin/Library/AutoPkg/Cache/com.github.jazzace.download.ClipGrabMac/downloads/ClipGrab-3.8.5.dmg

```
{'Output': {'download_changed': True,
        'etag': '"492c139-592b14e796ac0"',
        'last_modified': 'Mon, 16 Sep 2019 20:18:27 GMT',
        'pathname': u'/Users/localadmin/Library/AutoPkg/Cache/com.github.jazzace.download.ClipGrabMac/downloads/ClipGrab-3.8.5.dmg',
        'url_downloader_summary_result': {'data': {'download_path': u'/Users/localadmin/Library/AutoPkg/Cache/com.github.jazzace.download.ClipGrabMac/downloads/ClipGrab-3.8.5.dmg'},
                                         'summary_text': 'The following new items were downloaded:'}}}
```

## EndOfCheckPhase

```
{'Input': {}}
{'Output': {}}
```

## CodeSignatureVerifier

```
{'Input': {'input_path': u'/Users/localadmin/Library/AutoPkg/Cache/com.github.jazzace.download.ClipGrabMac/downloads/ClipGrab-3.8.5.dmg/ClipGrab.app',
        'requirement': u'identifier "de.clipgrab.ClipGrab" and anchor apple generic and 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] = E8BJ3ZV5W8'}}
```

CodeSignatureVerifier: Mounted disk image /Users/localadmin/Library/AutoPkg/Cache/com.github.jazzace.download.ClipGrabMac/downloads/ClipGrab-3.8.5.dmg

CodeSignatureVerifier: Verifying code signature...

CodeSignatureVerifier: Deep verification enabled...

CodeSignatureVerifier: Strict verification not defined. Using codesign defaults...

CodeSignatureVerifier: /private/tmp/dmg.qL60K9/ClipGrab.app: valid on disk

CodeSignatureVerifier: /private/tmp/dmg.qL60K9/ClipGrab.app: satisfies its Designated Requirement

CodeSignatureVerifier: /private/tmp/dmg.qL60K9/ClipGrab.app: explicit requirement satisfied

CodeSignatureVerifier: Signature is valid

```
{'Output': {}}
```

Receipt written to /Users/localadmin/Library/AutoPkg/Cache/com.github.jazzace.download.ClipGrabMac/receipts/ClipGrabMac-receipt-20190930-114431.plist

## The following new items were downloaded:

Download Path

-----

/Users/localadmin/Library/AutoPkg/Cache/com.github.jazzace.download.ClipGrabMac/downloads/ClipGrab-3.8.5.dmg



```
$ !!
autopkg run -vv --ignore-parent-trust-verification-errors ClipGrabMac.download.recipe
Processing ClipGrabMac.download...
WARNING: ClipGrabMac.download is missing trust info and FAIL_RECIPES_WITHOUT_TRUST_INFO is not set. Proceeding...
URLTextSearcher
{'Input': {'re_flags': (
    IGNORECASE
),
    're_pattern': u'https://download.clipgrab.org/ClipGrab-[^"]*.dmg"',
    'result_output_var_name': u'url',
    'url': u'https://clipgrab.org'}}}
URLTextSearcher: No value supplied for CURL_PATH, setting default value of: /usr/bin/curl
URLTextSearcher: Found matching text (url): ClipGrab-3.8.5.dmg
{'Output': {u'url': 'ClipGrab-3.8.5.dmg'}}

URLDownloader
{'Input': {'CURL_PATH': '/usr/bin/curl',
    'url': u'https://download.clipgrab.org/ClipGrab-3.8.5.dmg'}}
URLDownloader: No value supplied for CHECK_FILESIZE_ONLY, setting default value of: False
URLDownloader: Item at URL is unchanged.
URLDownloader: Using existing /Users/localadmin/Library/AutoPkg/Cache/com.github.jazzace.download.ClipGrabMac/downloads/ClipGrab-3.8.5.dmg
{'Output': {'pathname': u'/Users/localadmin/Library/AutoPkg/Cache/com.github.jazzace.download.ClipGrabMac/downloads/ClipGrab-3.8.5.dmg'}}

EndOfCheckPhase
{'Input': {}}
{'Output': {}}

CodeSignatureVerifier
{'Input': {'input_path': u'/Users/localadmin/Library/AutoPkg/Cache/com.github.jazzace.download.ClipGrabMac/downloads/ClipGrab-3.8.5.dmg/ClipGrab.app',
    'requirement': u'identifier "de.clipgrab.ClipGrab" and anchor apple generic and 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] = E8BJ3ZV5W8'}}
CodeSignatureVerifier: Mounted disk image /Users/localadmin/Library/AutoPkg/Cache/com.github.jazzace.download.ClipGrabMac/downloads/ClipGrab-3.8.5.dmg
CodeSignatureVerifier: Verifying code signature...
CodeSignatureVerifier: Deep verification enabled...
CodeSignatureVerifier: Strict verification not defined. Using codesign defaults...
CodeSignatureVerifier: /private/tmp/dmg.pH407P/ClipGrab.app: valid on disk
CodeSignatureVerifier: /private/tmp/dmg.pH407P/ClipGrab.app: satisfies its Designated Requirement
CodeSignatureVerifier: /private/tmp/dmg.pH407P/ClipGrab.app: explicit requirement satisfied
CodeSignatureVerifier: Signature is valid
{'Output': {}}
Receipt written to /Users/localadmin/Library/AutoPkg/Cache/com.github.jazzace.download.ClipGrabMac/receipts/ClipGrabMac-receipt-20190930-114501.plist
```

**Nothing downloaded, packaged or imported.**

```
<?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 copy of ClipGrab for Mac and makes a pkg installer
from the app. Requires peshay-recipes.</string>
  <key>Identifier</key>
  <string>com.github.jazzace.pkg.ClipGrabMac</string>
  <key>Input</key>
  <dict/>
  <key>MinimumVersion</key>
  <string>1.0</string>
  <key>ParentRecipe</key>
  <string>com.github.jazzace.download.ClipGrabMac</string>
  <key>Process</key>
  <array>
    <dict>
      <key>Processor</key>
      <string>AppPkgCreator</string>
    </dict>
  </array>
</dict>
</plist>
```

AppPkgCreator

# Custom Processors





# Processors

- Python code
- Core Processors
  - ▶ `/Library/AutoPkg/autopkglib`
  - ▶ `autopkg list-processors`
- Custom Processors
  - ▶ *Located in the same directory as a recipe*
  - ▶ *Can be shared ([github.com/autopkg/autopkg/wiki/Processor-Locations#shared-recipe-processors](https://github.com/autopkg/autopkg/wiki/Processor-Locations#shared-recipe-processors))*



# Custom Advantages

- Can deal with complex/unique download and/or versioning situations
- Customized for that product
- Can be coded to use human-friendly Input values
- Can be more efficient
- Allows addition of features not currently covered by core processors



# Custom Disadvantages

- Requires knowledge of Python to write a custom processor
- Requires knowledge of Python or a good testing scheme to audit a custom processor for trust
- If you can't write Python and the custom processor requires an update, you have to wait for someone else to do it
- More code to maintain

# Microsoft Office





# Office Recipe Sets

- Core
  - ▶ <https://github.com/autopkg/recipes>
- Rich Trouton (rtrouton)
  - ▶ <https://github.com/autopkg/rtrouton-recipes>
- Allister Banks et al (arubdesu)
  - ▶ <https://github.com/arubdesu/office-recipes>



# Common Download Recipe Workflow

1. Determine the URI of the item we want to download
2. Download a copy of the item
3. EndOfCheckPhase
4. Check the code signature of the downloaded item



# Common Download Recipe Workflow

1. Determine the URI of the item we want to download
  - ▶ *GitHubReleasesInfoProvider*
  - ▶ *SparkleUpdateInfoProvider*
  - ▶ *URLTextSearcher*
2. Download a copy of the item
3. EndOfCheckPhase
4. Check the code signature of the downloaded item

[Code](#)[Issues 24](#)[Pull requests 2](#)[Projects 0](#)[Wiki](#)[Security](#)[Insights](#)

Branch: master ▾

[recipes / MSOfficeUpdates / MSOfficeMacURLandUpdateInfoProvider.py](#)[Find file](#) [Copy path](#)

homebysix Use Exception instead of general BaseException

1ada4b5 on Sep 3

3 contributors

Executable File | 339 lines (313 sloc) | 14.8 KB

[Raw](#)[Blame](#)[History](#)

```
1 #!/usr/bin/env python
2 #
3 # Copyright 2015 Allister Banks and Tim Sutton,
4 # based on MSOffice2011UpdateInfoProvider by Greg Neagle
5 # much Office 2019 update work done by Carl Ashley
6 #
7 # Licensed under the Apache License, Version 2.0 (the "License");
8 # you may not use this file except in compliance with the License.
9 # You may obtain a copy of the License at
10 #
11 #     http://www.apache.org/licenses/LICENSE-2.0
12 #
13 # Unless required by applicable law or agreed to in writing, software
14 # distributed under the License is distributed on an "AS IS" BASIS,
15 # WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
16 # See the License for the specific language governing permissions and
17 # limitations under the License.
```

*<https://github.com/autopkg/recipes/blob/master/MSOfficeUpdates/MSOfficeMacURLandUpdateInfoProvider.py>*

docs.microsoft.com

## Most current packages for Office for Mac

The following table lists the most current packages for the Office suite and for the individual applications. The Office suite includes all the individual applications, such as Word, PowerPoint, and Excel. All packages are 64-bit only. The build date is listed in parentheses, in a YYMMDD format, after the version number. The install package is used if you don't have the application already installed, while the update package is used to update an existing installation.

To view release notes, see [Release notes](#).

**September 18, 2019**

Version 16.29.1 (19091700)

Application	Download links
Word	<a href="#">Update package</a>
Excel	<a href="#">Update package</a>
PowerPoint	<a href="#">Update package</a>

**September 10, 2019**

Version 16.29 (19090802)

Application	Download links	SHA-256 hash for install package
Office suite (with Teams)	<a href="#">Install package</a>	0ABECAC4FA523DDC12B9360C0F57F8C19473E1F07D0FA2E5E94FA9F4601935D4
Office suite (without Teams)	<a href="#">Install package</a>	613B9F0DADAFE6D65ACA12C86208848D865230F73C840A0627987F3E10722906
Word	<a href="#">Install package</a> <a href="#">Update package</a>	E29949E29D62783F61425459046F0A815B4C870D624C58F56C4E84FD8DD40F9D
Excel	<a href="#">Install package</a> <a href="#">Update package</a>	AB899FEB07F6BF5B3F72A151715402ACD2BBF0678419EFD244A97E4116EBB72F
PowerPoint	<a href="#">Install package</a> <a href="#">Update package</a>	93A0590ABBA4CFB2B4D9A011BE65053539DC3B569A27AA77F7E1F1C9D5EC24D9
Outlook	<a href="#">Install package</a> <a href="#">Update package</a>	863ED59600A7FE4F4F686C98462850F6CC214001859A2BA4A3E0DFA0F6567FBC
OneNote	<a href="#">Update package</a>	Not applicable

## Most current packages for other Mac programs from Microsoft

The following table provides information about the most current packages, including a download link, for other Mac programs from Microsoft that may be of interest to IT professionals. The build date is listed in parentheses, in a YYMMDD format, after the version number.

**Note:** Microsoft AutoUpdate (MAU) release information has moved to the [MAU Release History](#) page.

↓ Download PDF  
Open "<https://go.microsoft.com/fwlink/?linkid=2009112>" in a new tab

Is this page helpful?  
 Yes  No

In this article

- [Most current packages for Office for Mac](#)
- [Most current packages for other Mac programs from Microsoft](#)
- [Release history for Office for Mac](#)

<https://docs.microsoft.com/en-us/officeupdates/update-history-office-for-mac>



# rtrouton & arubdesu

## Download Recipes

1. Get FWLink as an Input Variable
2. Download a copy of the item using known URL and Input Variable value
3. EndOfCheckPhase
4. Check the code signature of the downloaded item



# Common Download Recipe Workflow

1. Determine the URI of the item we want to download
2. Download a copy of the item
3. EndOfCheckPhase
4. Check the code signature of the downloaded item
5. Collect version information



# Versioning

- Processors that collect version info
  - ▶ *Versioner*
  - ▶ *AppDmgVersioner*
  - ▶ *AppPkgCreator*
  - ▶ *custom processors*



# Office Custom Processor

- Core recipes
  - ▶ *MSOfficeMacURLAndUpdateInfoProvider*  
(*URL provider + versioner*)
- arubdesu recipes
  - ▶ *OfficeSuiteSKULessVersionProvider*  
(executed before *URLDownloader*)

```

<?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 Microsoft's Excel for Office 365 installer package and renames the package.</string>
    <key>Identifier</key>
    <string>com.github.rtrouton.pkg.microsoftexcel365</string>
    <key>Input</key>
    <dict>
        <key>NAME</key>
        <string>Microsoft Excel</string>
        <key>VENDOR</key>
        <string>Microsoft</string>
        <key>SOFTWARETITLE</key>
        <string>Excel</string>
    </dict>
    <key>MinimumVersion</key>
    <string>1.0.0</string>
    <key>ParentRecipe</key>
    <string>com.github.rtrouton.download.microsoftexcel365</string>
    <key>Process</key>
    <array>
        <dict>
            <key>Arguments</key>
            <dict>
                <key>destination_path</key>
                <string>%RECIPE_CACHE_DIR%/downloads/unpack</string>
                <key>flat_pkg_path</key>
                <string>%pathname%</string>
            </dict>
            <key>Processor</key>
            <string>FlatPkgUnpacker</string>
        </dict>
        <dict>
            <key>Arguments</key>
            <dict>
                <key>pattern</key>
                <string>%RECIPE_CACHE_DIR%/downloads/unpack/*Excel*.pkg</string>
            </dict>
            <key>Processor</key>
            <string>FileFinder</string>
        </dict>
        <dict>
            <key>Arguments</key>
            <dict>
                <key>destination_path</key>
                <string>%RECIPE_CACHE_DIR%/downloads/payload</string>
                <key>pkg_payload_path</key>
                <string>%found_filename%/Payload</string>
            </dict>
            <key>Processor</key>
            <string>PkgPayloadUnpacker</string>
        </dict>
        <dict>
            <key>Arguments</key>
            <dict>
                <key>input_plist_path</key>
                <string>%RECIPE_CACHE_DIR%/downloads/payload/Microsoft Excel.app/Contents/Info.plist</string>
                <key>plist_version_key</key>
                <string>CFBundleVersion</string>
            </dict>
            <key>Processor</key>
            <string>Versioner</string>
        </dict>
        <dict>
            <key>Processor</key>
            <string>PkgCopier</string>
            <key>Arguments</key>
            <dict>
                <key>source_pkg</key>
                <string>%pathname%</string>
                <key>pkg_path</key>
                <string>%RECIPE_CACHE_DIR%/%VENDOR%_%SOFTWARETITLE%_%version%.pkg</string>
            </dict>
        </dict>
        <dict>
            <key>Processor</key>
            <string>PathDeleter</string>
            <key>Arguments</key>
            <dict>
                <key>path_list</key>
                <array>
                    <string>%RECIPE_CACHE_DIR%/downloads/unpack</string>
                    <string>%RECIPE_CACHE_DIR%/downloads/payload</string>
                </array>
            </dict>
        </dict>
    </array>
</dict>
</plist>

```

# 1. Run Parent download recipe

► *MicrosoftExcel365.download*

# 2. FlatPkgUnpacker

# 3. FileFinder

► *look for \*Excel\*.pkg inside unpacked installer*

# 4. PkgPayloadUnpacker

# 5. Versioner

# 6. PkgCopier

► *Renames to Microsoft\_Excel\_version*

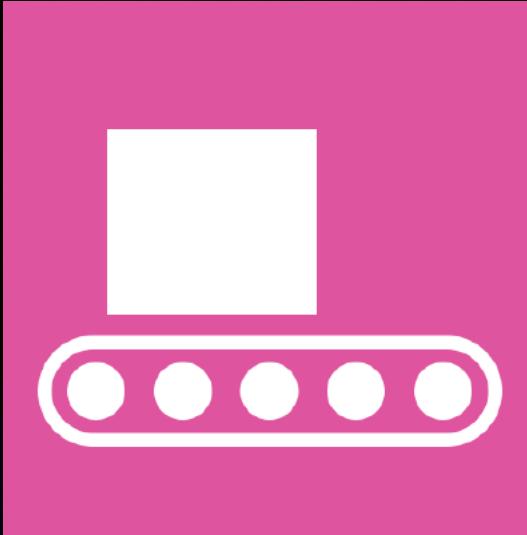
# 7. PathDeleter

	<b>core</b> (autopkg/recipes)	<b>rtrouton</b> (autopkg/rtrouton-recipes)	<b>arubdesu</b> (arubdesu/office-recipes)
<b>Value for URLDownloader</b>		Input variables	Input variables
<b>Versioning</b>	Custom Processor	6 Core Processors	Custom Processor



# Further Reading

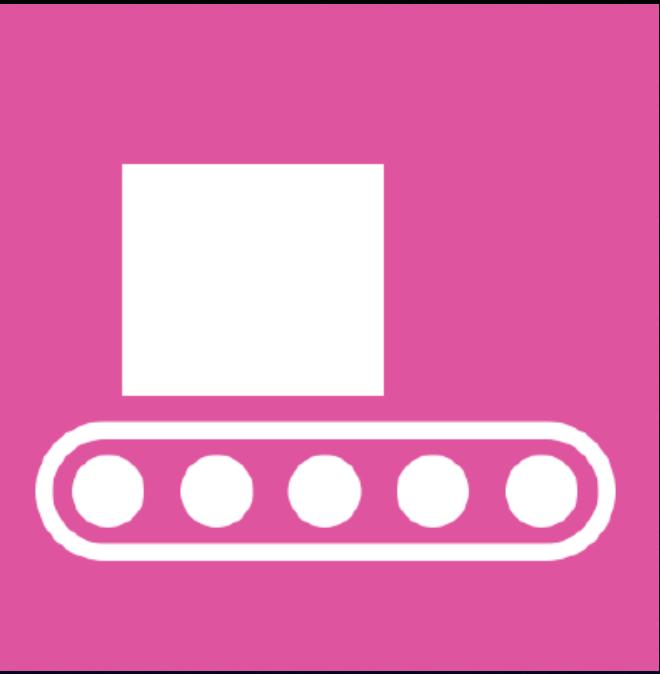
- Anthony's Mac Labs Blog
  - ▶ *maclabs.jazzace.ca* (look for )



# Resources

- AutoPkg Wiki
  - ▶ *github.com/autopkg/autopkg/wiki*
- #autopkg channel in MacAdmins Slack
  - ▶ *MacAdmins.org* to join
- AutoPkg Discussion List
  - ▶ *groups.google.com/group/autopkg-discuss*

# Resources For This Session:



[github.com/jazzace/mactech-2019-autopkg](https://github.com/jazzace/mactech-2019-autopkg)

Anthony Reimer

 [@AnthonyReimer](https://twitter.com/AnthonyReimer)

 [jazzace](https://github.com/jazzace)

 [jazzace](https://www.linkedin.com/in/jazzace)

