

# Homebrew Deployment (macOS Script)

## Description

Homebrew, or also known as simply Brew is the missing package manager for macOS (or Linux), Homebrew is able to install many tools that do not have another installer. Quite a lot of Linux based tools have a Homebrew install. This article will guide you through installing it using FileWave.

Actually, not just Brew, but we will also install:

- Apple Command Line Tools
- Brew
- Cask

## Ingredients

- FileWave Central
- Attached Fileset



## Directions

1. Unzip the download and drag the Fileset into the FileWave Central > Filesets view, into a Fileset group of choice.
2. Associate with a test device and Update Model
3. After all is well with the test, consider expanding this to a larger quantity of devices

## Activation Script

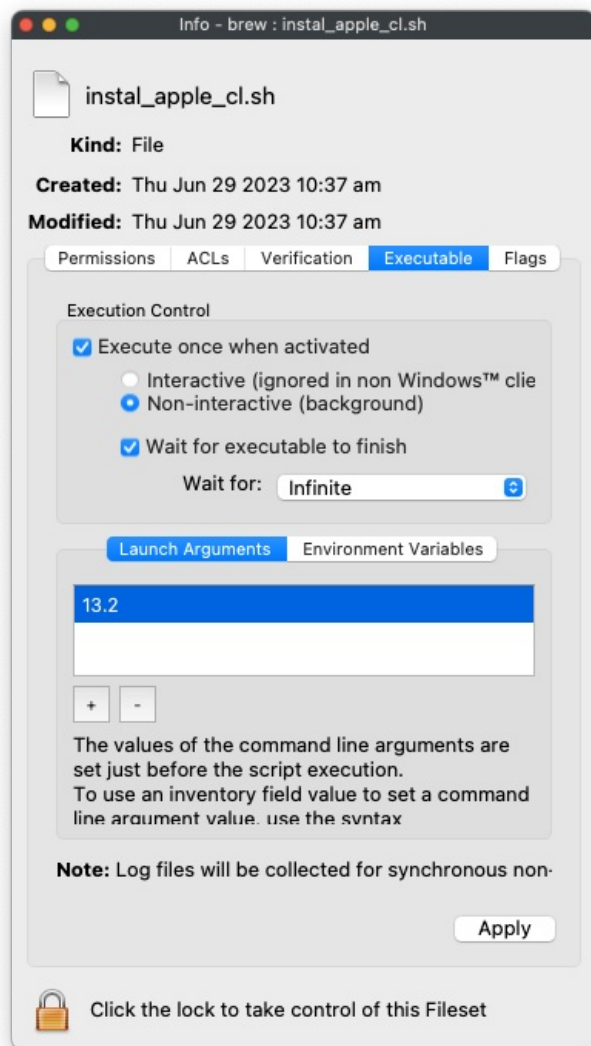
The Activation Script will choose a version of the possible Apple Command Line tools and proceed to download and instal those tools.

 Apple Command Line tools are a requirement and will be installed first.

 The Activation Script will use the newest version of the Command Line tools found, when ran. It is possible to overrule this behaviour by supplying a desired version as a Launch Argument.

### Example:

To instal version 13.2 Command Line tools, select the Activation Script: 'instal\_apple\_cl.sh' from within the Fileset and from the 'Get Info' view, edit the Executable Launch Arguments, as below:



Software Updates may be large in size. This can slow down the installation as well as use high bandwidth. A 'Wait for' as 'Infinite' could easily be required.

## Post Activation Script

The Fileset will attempt to instal Brew with the currently logged in user. If this current user does not have an ID greater than 500, the script will attempt to instal with an Admin user on the device. The script log should report the user that was configured for Brew. Consider altering the script if a different behaviour is desired.

This user shall also be added to the macOS developer group.

Below are the sections of script that defines the user and also adds that user to the development group:

```
current_user=$(stat -f%Su /dev/console)


# ....

# If user is below uid 500 should we abort or pick a user. Picking an admin user for now
if [ $(id -u $current_user) -lt 501 ]
then
    echo "No 500 user. Picking a local admin user"
    current_user=$( /usr/bin/dscl . -read /Groups/admin GroupMembership | awk '{print $NF}')
fi

echo "Configuring user ${current_user}..."

# Add them as a developer
is_developer=$(groups $current_user | awk '/ _developer / ')
if [ ! $is_developer ]
then
```

```
echo "Adding user $current_user to _developer group"
/usr/sbin/dseditgroup -o edit -a "$current_user" -t user _developer
fi
```

 Brew/Cask installations may also be large in size. The same considerations as the Activation Script would therefore also be likely. Additionally, the Fileset will double check there are no further updates and if so, those shall also be installed.

## Uninstall

If you want to remove Brew, the best way is likely these scripted methods because you are probably just fixing a single machine when you remove Brew, but you could also use curl to fetch the `uninstall.sh` and put it in the Fileset if you want to uninstall when you remove the Association to the Fileset.

```
/bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/uninstall.sh)"
```

If you want to run the Homebrew uninstaller non-interactively, you can use:

```
NONINTERACTIVE=1 /bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/uninstall.sh)"
```

Download the uninstall script and run `/bin/bash uninstall.sh --help` to view more uninstall options.

---

🔄Revision #7

★Created 29 June 2023 15:14:48 by Sean Holden

✎Updated 12 September 2023 13:36:38 by Josh Levitsky