jumpcloud™

Works with_
→ **Work**brew

JumpCloud Deployment Guide

Step-by-step instructions for a fast, error-free setup.

jumpcloud™

Workbrew streamlines secure, automated Homebrew package deployment for macOS, integrating seamlessly with JumpCloud to give IT teams centralized device management.

Homebrew is the de-factor package manager on macOS, installed on tens of millions of devices and offering more than 15,000 packages. With zero-touch deployment, policy enforcement, and real-time monitoring, Workbrew lets you leverage the power of Homebrew, whilst ensuring compliance and eliminating security risks.

TABLE OF CONTENTS

By the end of this deployment guide, you will:

Understand the available mechanisms to deploy Workbrew through JumpCloud

Configure JumpCloud to allow Workbrew to manage your fleet's Homebrew installations

Be ready to deploy Workbrew to your devices

Before you begin following this guide, you should:

1. Have access to a JumpCloud instance

- with **API access** to create an API Key. You will need an "Admin with Billing access" to enable API access, if you do not have it already.

2. Have a Workbrew workspace

New to Workbrew? Create a **free account** and follow the **Getting Started** guide.

3. Be aware of the system requirements for Workbrew (and Homebrew):

- Everything **Homebrew requires:**
  - An Apple Silicon CPU or 64-bit Intel CPU.
  - macOS Ventura (13) (or higher) installed on officially supported hardware.
  - The Bourne-again shell for installation (i.e. bash)
  - Don't worry about the Command Line Tools (CLT) for Xcode requirement, Xcode CLT will be installed as part of deployment.
- Device enrolled in JumpCloud.
- User account in the `admin` group or in the `workbrew_users` group

Are you an experienced JumpCloud administrator? These steps will get you up and going quickly. Read on for more detailed explanations.

1. **In JumpCloud,** select **My API Key** or generate a new API Key, if you haven't generated an API key before.

2. **In the Workbrew console**, enter the workspace settings and select JumpCloud as the **MDM Type**. Enter your JumpCloud API token, and then save the Workbrew Workspace API key and installation script.

3. **In JumpCloud,** add the Workbrew Workspace API key and installation script as a **Command After Agent** Install. Set the timeout to be **3600** seconds to allow enough time to install Xcode Command Line Tools.

> **Important:** The installation script must run and complete successfully before the next step. This setup ensures that for any new devices going through enrollment, the install script will run immediately after agent install. Custom apps will not run before the setup process completes.
>
> For already enrolled devices, JumpCloud has no way of enforcing the order in which commands run and apps are installed, we recommend you add the installation script and ensure all devices have successfully run it before deploying the Custom App to those devices.

4. **In JumpCloud,** add the **Workbrew .pkg** with JumpCloud Software Management as a new Custom App using JumpCloud Private Repo.

5. Bind and install the custom app on specific devices or device groups.

6. **In the Workbrew console**, after deployment to a device, check **Devices** to ensure the expected device appears (please be aware that device inventory is updated periodically, not in real time).

7. If needed, check the **Troubleshooting guide and FAQ** or **contact us for support.**

DEPLOYMENT OVERVIEW

Workbrew is installed using a **signed .pkg file**, which installs several components:
- The Workbrew agent
- The Secure Workbrew CLI, a wrapper around the standard Homebrew CLI.

Before installing the Workbrew .pkg on each device, you must run a (bash) script which connects the Workbrew agent to your Workbrew Console. The script also installs Command Line Tools for Xcode if your devices do not already have it. The Workbrew Console connection wizard will guide you through customization to your install script.

You can deploy the installation script as a JumpCloud Command After Agent Install. All new devices going through enrollment will run the install script immediately after agent install. As Custom apps will not run before the setup process completes, this provides sufficient control over execution ordering for newly enrolled devices.

For devices that are already enrolled, JumpCloud currently offers no way of enforcing the order in which commands run and apps are installed. We recommend you add the installation script and ensure all devices have successfully run it before continuing with installing the Workbrew .pkg. You can add the Workbrew .pkg as a **Custom App**, after which you can bind and install the custom app on specific devices or device groups.

In brief, you will perform these steps to ready Workbrew for deployment:
- Get a JumpCloud API key,
- Complete the Workbrew Console connection wizard, adding the API Token in the process,
- Add the setup script to JumpCloud as a Command,
- Add the Workbrew Package to JumpCloud as a Custom App.

CONNECTING WORKBREW TO JUMPCLOUD

1. Creating an API key in JumpCloud

To populate your Workbrew Console with information about your devices and users, Workbrew requires API access to your JumpCloud instance. In this section, you will retrieve or create an **API key** and retain the credentials for input into Workbrew.

To complete this step, follow the instructions in the **JumpCloud Documentation** to **Access Your API Key** until you reach the following numbered step:

4. You will receive an API key, it has the prefix 'jca_' before it to help you search for it easier. Save this for the next step.

2. Add JumpCloud to your Workbrew workspace

The API Token created in the previous step will allow Workbrew to read from your JumpCloud instance using the API.

In this section, you will register JumpCloud as your MDM of choice within Workbrew.

From the Workbrew Console, select Settings. Ensure you are in the Workspace tab.

Under MDM Type, select "JumpCloud".

In the JumpCloud API Token field, enter the JumpCloud API Token generated in the previous section.

Click Update Workspace.

Open Workbrew Workspace API key and installation script, copy the script, and store it for later. This script will run as a Command After Agent Install.

PREPARING THE DEPLOYMENT ARTIFACTS

The Workbrew Workspace API key and installation script saved in the previous step prepares the device for a Workbrew installation, setting environment variables for workspace directories and the Workbrew Workspace API key. It also installs a Homebrew dependency, Command Line Tools for Xcode, using MacOS's `softwareupdate` utility. In this section, you will add the script to Jumpcloud as a Command, executing after Agent Install.

To add the script as a new Command, follow the instructions under **Creating a Command after Agent Install** until you reach the following numbered steps:

5. Select Mac.

6. Paste the Workbrew Workspace API key and installation script.

7. Set the Timeout to be 3600 seconds to allow ample time for Command Line Tools for Xcode to install.

Important: The installation script must run and complete successfully before the next step. SimpleMDM has no builtin way of enforcing the order in which jobs run and apps are installed, we recommend you add the installation script and ensure all devices have successfully run it before continuing setup. Alternatively you can use **Munki** which allows you to run the installation script as a command during pre-installation.

2. Add the package

The Workbrew .pkg installs Workbrew, including the agent, CLI, and Homebrew. In this section, you will add the package to JumpCloud so that it can be distributed to your devices or device groups.

Download the package, and then follow the instructions under **Apple Custom Application** until you reach the following numbered step:

> 8. When this message disappears, you can bind and install the application on specific devices or device groups. See **Binding and Installing Applications on Devices or Device Groups.**

DEPLOYMENT

The Custom App will not be installed until you bind it to devices or device groups. In this step, you will bind and install the Workbrew .pkg.

Follow the steps in **Binding and Installing Applications on Devices or Device Groups** until you reach the following numbered step:

> 7. Click **Save & Install** to proceed with the software installation. A success message indicates whether the installation was successful.

Once Workbrew has been deployed to a device, it will appear on the Workbrew Console.

SUPPORT

Getting Started

Frequently Asked Questions

Troubleshooting

Workbrew Blog

**Deliver the software your team
needs securely and at scale.**

workbrew.com