Microsoft Intune

Works with_
→ **Work**brew

Intune Deployment Guide

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

Microsoft Intune

Workbrew streamlines secure, automated Homebrew package deployment for macOS, integrating seamlessly with Microsoft Intune 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 Microsoft Intune

Configure Intune 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 Microsoft Intune tenant**

with user roles to:
- Register an application in Microsoft Entra,
- Manage and create applications in Microsoft Intune,
- Manage device assignments in Microsoft Intune.

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 Microsoft Intune
- User account in the `admin` group or in the `workbrew_users` group

QUICKSTART

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

1. **In Microsoft Entra,** register an application with Application permission to "Read Microsoft Intune devices" (`DeviceManagementManagedDevices.Read.All`), "Read all groups" (`Group.Read.All`) and "Read all group memberships" (`GroupMember.Read.All`).

2. **In the Workbrew console**, enter the workspace settings and select Intune as the **MDM Type**. Enter your Microsoft Intune Tenant ID, Microsoft Graph API Client ID, and Microsoft Graph Client Secret. Save the Workbrew Workspace API key and installation script.

3. **In Microsoft Intune,** add the Workbrew .pkg as a new unmanaged MacOS PKG app. Add the Workbrew Workspace API key and installation script as a **pre-installation script.**

4. **In Microsoft Intune,** assign your newly created app to the desired devices.

5. **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).

6. 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.

In addition to 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 Workbrew .pkg as an unmanaged MacOS PKG app with a pre-installation script for the setup script. The app can then be assigned to enrolled devices for deployment, or made available for other groups to install at their leisure.

Please note that you cannot remotely uninstall unmanaged MacOS PKG apps with Microsoft Intune, nor will the app be automatically removed when a device is unenrolled.

In brief, you will perform these steps to ready Workbrew for deployment:
- Create a Microsoft Graph API application and save the Client ID and Client Secret,
- Complete the Workbrew Console connection wizard, adding the Microsoft Graph API Client ID and Client Secret in the process,
- Add the Workbrew Package and setup script as an unmanaged MacOS PKG app to Microsoft Intune,
- Assign the new app to target devices,
- Optionally, make the app available for self service.

1. Creating an app in Microsoft Graph API

To populate your Workbrew Console with information about your devices and users, Workbrew requires Read-Only API access to your Microsoft Intune tenant. In this section, you will create an Microsoft Graph API application with the "**Read Microsoft Intune devices**" permission, and retain the credentials for input into Workbrew.

To complete this step, follow the instructions in the **Microsoft Graph API documentation** to **Register apps to use the Microsoft Graph API** until you reach the following numbered steps:

4. Enter "Workbrew Console API" as the application's name. You do not need to enter a redirect URI.

5. After clicking Register, save the Application (client) ID from the Overview pane for later.

7.
- Select Add a permission, and then on the Microsoft APIs tab, select the Microsoft Graph tile,
- Select Application permissions,
- Enter `DeviceManagementManagedDevices.Read.All` into the search bar, and check the box on the result,
- Click the Add permissions button.
- Repeat to add `Group.Read.All` and `GroupMember.Read.All`.

After step 7, Create a Client Secret:
- Select Certificates and secrets,
- Select the Client secrets tab,
- Click + New client secret,
- Enter Workbrew Console API as the description, and set the expiry as per your company policies,
- Save the Value of the Client secret for later.

2. Add Microsoft Intune to your Workbrew workspace

The API Token created in the previous step will allow Workbrew to read from your Microsoft Intune tenant using the API. In this section, you will register Intune as your MDM of choice within Workbrew.

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

Under MDM Type, select "Intune".

Under Microsoft Intune Tenant ID, enter the URL for your Intune tenant.

In the Microsoft Graph API Client ID field, enter the Application (client) ID.

In the Microsoft Graph API Client Secret enter the Client secret Value.

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 pre-installation script in the Custom App.

1. Add the package

The Workbrew .pkg installs Workbrew, including the agent, CLI, and Homebrew. In this section, you will add the package as an unmanaged MacOS PKG app application to Microsoft Intune.

Download the package, and then follow the instructions in **Add an unmanaged macOS PKG app to Microsoft Intune** until you reach the following numbered steps:

Step 1 – App information

2. Select the **Workbrew .pkg.**

Set App information

- Name: Set the app name as "Workbrew installer and setup script".
- Publisher: Set the publisher name to "Workbrew".

Step 2 – Program

- Add a pre-install script, and insert the Workbrew Workspace API key and installation script that you saved earlier.

Step 3 – Requirements

- Minimum Operating System: select "MacOS Ventura 13.0".

Step 4 – Detection rules

- Set **Ignore app version** to "Yes".
- Remove any auto-detected App bundle ID (CFBundleIdentifier) and App version (CFBundleShortVersionString) entries. Enter com.workbrew.Workbrew for the App bundle ID and 1.6.1 for the App

Step 6 – Assignments

- Assign the app to the desired device groups. You can also return to this later.

Step 7 – Review + create

- Once finished, click Create. You will be immediately taken to an overview page of your new app, however, the app will take some time to prepare before it can be edited.

DEPLOYMENT

Workbrew will be distributed to the groups assigned to the app in Step 6 above. If you skipped that step or need to add assignments in the future, access the app through the Microsoft Intune admin center, select the Properties pane, and then click Edit next to Assignments towards the bottom of the page.

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