

Introduction to Workspace ONE Admin Assistant for macOS

Last Updated Feb 24, 2025 | 3 minute read | [Summarize](#) |

[Omnissa Workspace ONE Admin Assistant](#)

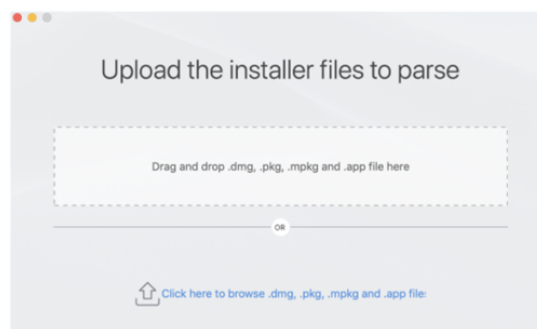
[Product Documentation](#)

The Omnissa Workspace ONE Admin Assistant tool helps you to manage applications through Internal Apps (Full Software Distribution) in Omnissa Workspace ONE UEM. This tool is built using components of Munki, to aid in parsing application installers for the required metadata.

Admin Assistant App

Due to limitations of the Omnissa Workspace ONE UEM server to natively process the macOS software metadata, the Admin Assistant app was created to help administrators generate the required metadata to upload to Workspace ONE UEM. Using the **makepkginfo** tool from Munki, the Admin Assistant app extracts metadata from macOS software installers into a *plist* file, sometimes referred to as a metadata file or *pkginfo* file.

Workspace ONE UEM requires the *pkginfo* metadata file along with the application installer file to manage the deployment in the UEM console. The Admin Assistant also tries to extract an icon image file that can be optionally uploaded into the UEM console. The Admin Assistant app allows you to easily upload the *.dmg*, *.pkg*, *mpkg*, and *.app* files.



Command-Line Interface

In the Admin Assistant 3.0, a command-line interface (CLI) is introduced to work alongside the app interface. Admin Assistant CLI provides a streamlined experience where just one command allows you upload the application and metadata straight to the console. The Admins can now use the GUI to parse packages and use the CLI to upload packages to the console. Future

ON THIS TOPIC

[Admin Assistant App](#)

Command-Line Interface

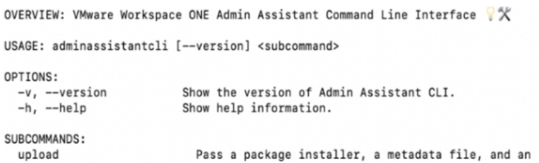
versions of Admin Assistant will add more parity between the app and the CLI functionality.

Admin Assistant CLI Usage Help

Using the CLI, there are several ways to read more about how to use the tool and the built-in subcommands. To access the help information in the CLI, enter the following in the terminal:

- `/usr/local/bin/adminassistantcli`
- `/usr/local/bin/adminassistantcli -h`
- `/usr/local/bin/adminassistantcli -help`
- `/usr/local/bin/adminassistantcli <subcommand> -help`

The following image depicts the Admin Assistant CLI interface



The following tables lists the basic commands that are used to run the Admin Assistant in the CLI mode.

Commands	Description
<code>adminassistantcli config --signin --groupID "groupID1" --hostname "hostname1"</code>	Required. Prompts interactive login terminal. Allows you to perform a successful sign-in.
<code>adminassistantcli config --list</code>	Allows you to list the configurations saved in the keychain.
<code>adminassistantcli upload --installer-path "/Users/example.dmg" --metadata-path "example.plist" --icon-path "icon.png"</code>	Allows you to upload the files and displays the URL to view in console. GroupID can optionally be changed. Uploading an icon is optional.
<code>adminassistantcli --signout</code>	Allows you to signout from the Admin Assistant tool.

Best Practice to Parse Arguments

Admin assistant CLI accepts information using both double quotes and single quotes. However, the best practice is to use single quotes as it accepts all the arguments as a String. Instead of taking the data and the

special characters as commands, it is processed as just the plain text

© 2025 Omnisia, LLC
590 E Middlefield Road,
Mountain View CA 94043
All Rights Reserved.

Offerings

- Omnissa platform
- Platform services
- Products

Resources

- Blog
- Partners
- Security response
- Trust center
- User portal
- Glossary
- Data rights request

Company

- About
- News
- Careers
- Contact us