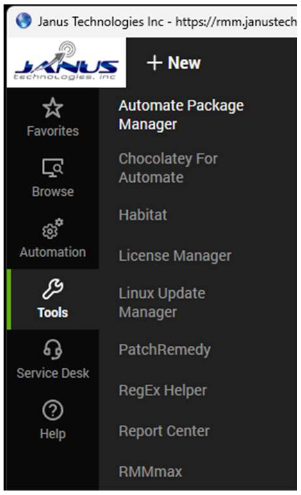
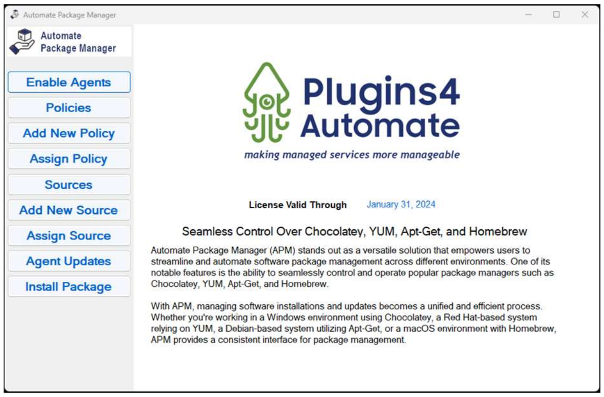
Automate Package Manager – Quick Start

Description: This innovative tool facilitates seamless software management across Windows, Debian Linux, RedHat-style Linux, and Mac OSX operating systems. By integrating popular package managers like Chocolatey, Apt-get, YUM, and Homebrew, Automate Package Manager simplifies the deployment and update processes, enabling MSPs to efficiently control software installations across large-scale computer environments.

One Interface to manage them all!





License Valid Through January 31, 2024

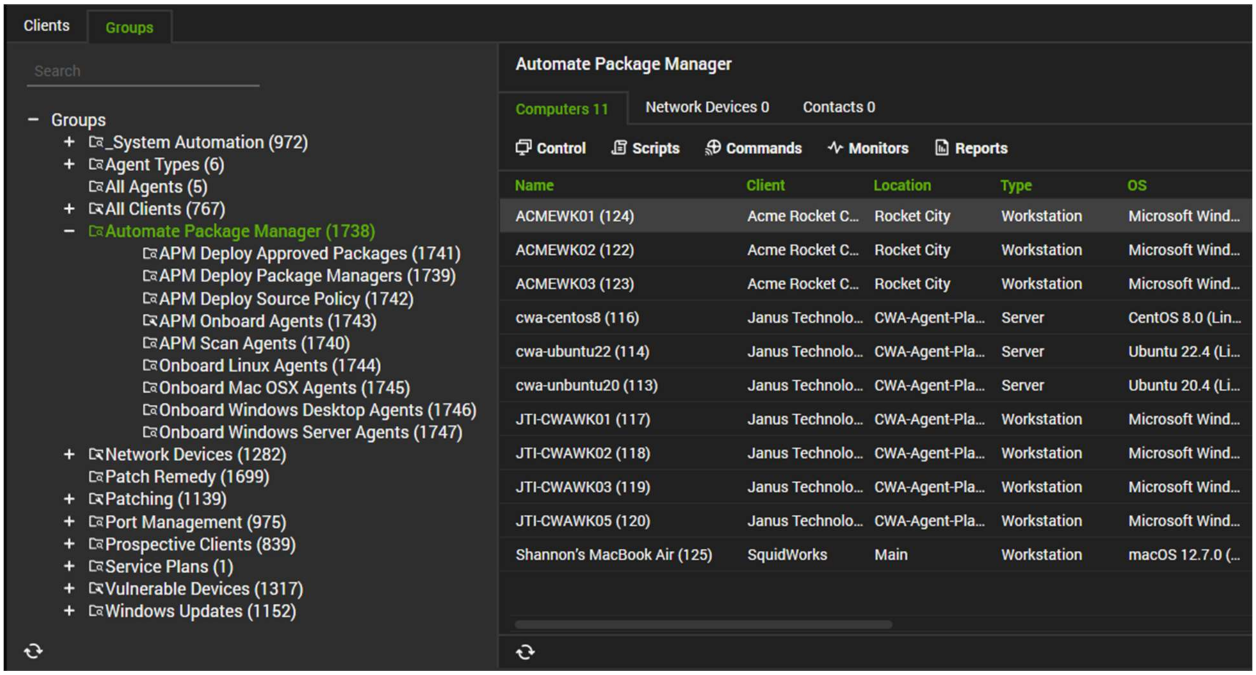
Seamless Control Over Chocolatey, YUM, Apt-Get, and Homebrew

Automate Package Manager (APM) stands out as a versatile solution that empowers users to streamline and automate software package management across different environments. One of its notable features is the ability to seamlessly control and operate popular package managers such as Chocolatey, YUM, Apt-Get, and Homebrew.

With APM, managing software installations and updates becomes a unified and efficient process. Whether you're working in a Windows environment using Chocolatey, a Red Hat-based system relying on YUM, a Debian-based system utilizing Apt-Get, or a macOS environment with Homebrew, APM provides a consistent interface for package management.

Located under the Main Tools Menu is the Automate Package Manager. From this interface you can enable agents, locations, and clients with a single click. Create Update and Install policies that conform to your needs and assign policies to agents, locations, and clients in the same manner as you enable agents. This granularity allows you manage agent installs and updates in ways that makes sense to you.

Groups and Searches

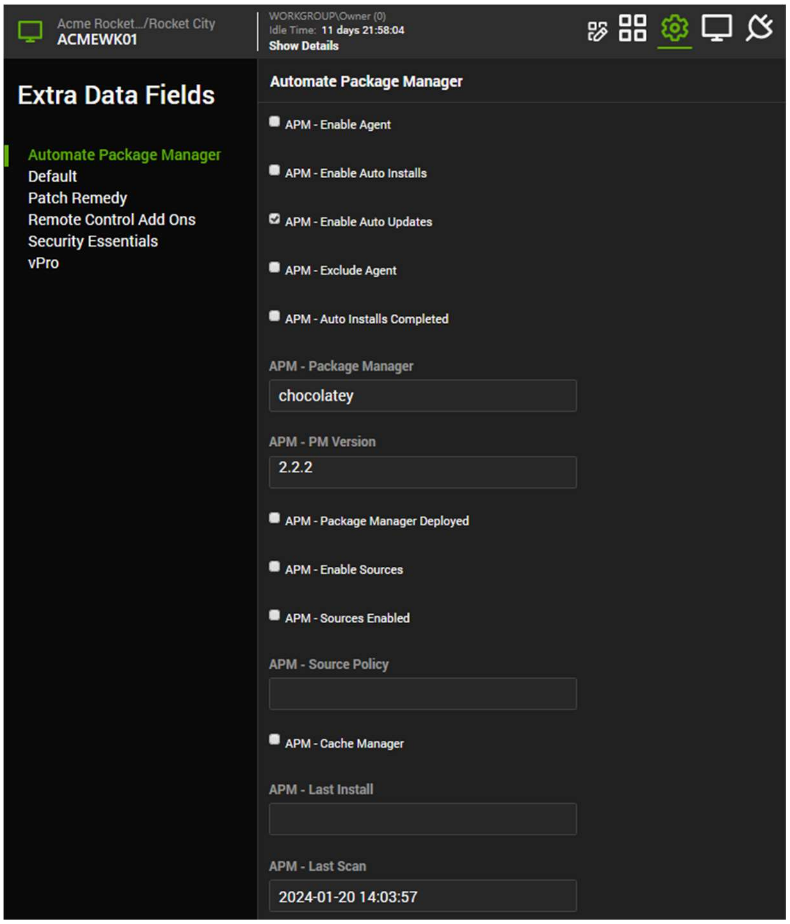


- Groups
  - _System Automation (972)
  - Agent Types (6)
  - All Agents (5)
  - All Clients (767)
  - Automate Package Manager (1738)
    - APM Deploy Approved Packages (1741)
    - APM Deploy Package Managers (1739)
    - APM Deploy Source Policy (1742)
    - APM Onboard Agents (1743)
    - APM Scan Agents (1740)
    - Onboard Linux Agents (1744)
    - Onboard Mac OSX Agents (1745)
    - Onboard Windows Desktop Agents (1746)
    - Onboard Windows Server Agents (1747)
  - Network Devices (1282)
  - Patch Remedy (1699)
  - Patching (1139)
  - Port Management (975)
  - Prospective Clients (839)
  - Service Plans (1)
  - Vulnerable Devices (1317)
  - Windows Updates (1152)

Name	Client	Location	Type	OS
ACMEWK01 (124)	Acme Rocket C...	Rocket City	Workstation	Microsoft Wind...
ACMEWK02 (122)	Acme Rocket C...	Rocket City	Workstation	Microsoft Wind...
ACMEWK03 (123)	Acme Rocket C...	Rocket City	Workstation	Microsoft Wind...
cwa-centos8 (116)	Janus Technolo...	CWA-Agent-Pla...	Server	CentOS 8.0 (Lin...
cwa-ubuntu22 (114)	Janus Technolo...	CWA-Agent-Pla...	Server	Ubuntu 22.4 (Li...
cwa-unbuntu20 (113)	Janus Technolo...	CWA-Agent-Pla...	Server	Ubuntu 20.4 (Li...
JTI-CWAWK01 (117)	Janus Technolo...	CWA-Agent-Pla...	Workstation	Microsoft Wind...
JTI-CWAWK02 (118)	Janus Technolo...	CWA-Agent-Pla...	Workstation	Microsoft Wind...
JTI-CWAWK03 (119)	Janus Technolo...	CWA-Agent-Pla...	Workstation	Microsoft Wind...
JTI-CWAWK05 (120)	Janus Technolo...	CWA-Agent-Pla...	Workstation	Microsoft Wind...
Shannon's MacBook Air (125)	SquidWorks	Main	Workstation	macOS 12.7.0 (...

Automate Package Manager works with ConnectWise Automate's Groups functions to automate the deployment and validation of each agent's package manager. Each automated process will spawn on the agent automatically as the agent progresses through the setup, install, and update processes. You can edit the groups and adjust the Automate script executions to fine tune the process for your MSP or enjoy the preset settings we provide with the plugin.

Location and Agent EDFs



Automate Package Manager

☐ APM - Enable Agent

☐ APM - Enable Auto Installs

☑ APM - Enable Auto Updates

☐ APM - Exclude Agent

☐ APM - Auto Installs Completed

APM - Package Manager: chocolatey

APM - PM Version: 2.2.2

☐ APM - Package Manager Deployed

☐ APM - Enable Sources

☐ APM - Sources Enabled

APM - Source Policy:

☐ APM - Cache Manager

APM - Last Install:

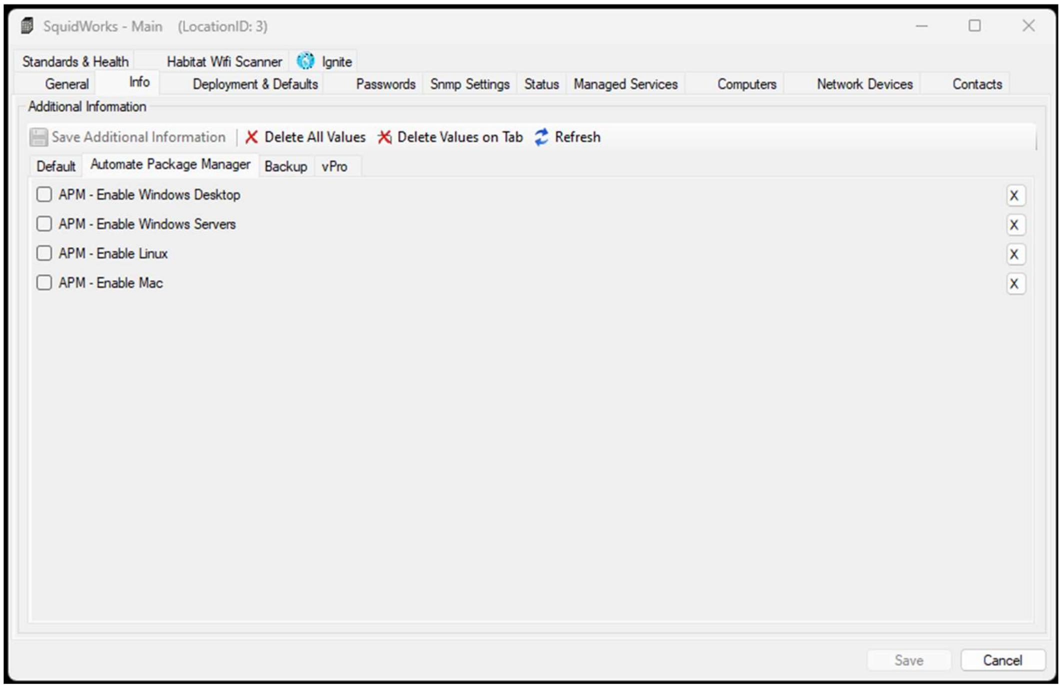
APM - Last Scan: 2024-01-20 14:03:57

Automate Package Manager uses agent and location EDFs to store agent status and to allow individual control over automated package management. By default, all these settings are managed in the main Automate package Manager tool, but you can use the EDFs to validate the status of each agent and in a pinch make quick changes to a single agent quickly.

Example Usage Scenario: You enable a client or a location full of Windows desktops for auto updates, but you are instructed to exclude the CEO of the client in questions PC from management. You would use the main tool to enable the client or location, then opening the agent console for the CEO's computer and selecting to uncheck APM- Enable Agent and check APM – Exclude Agent.

This would disable the agent from any automation and then would exclude agent so that any onboarding you may do for client will not reenable the disabled agent.

Onboarding EDFs



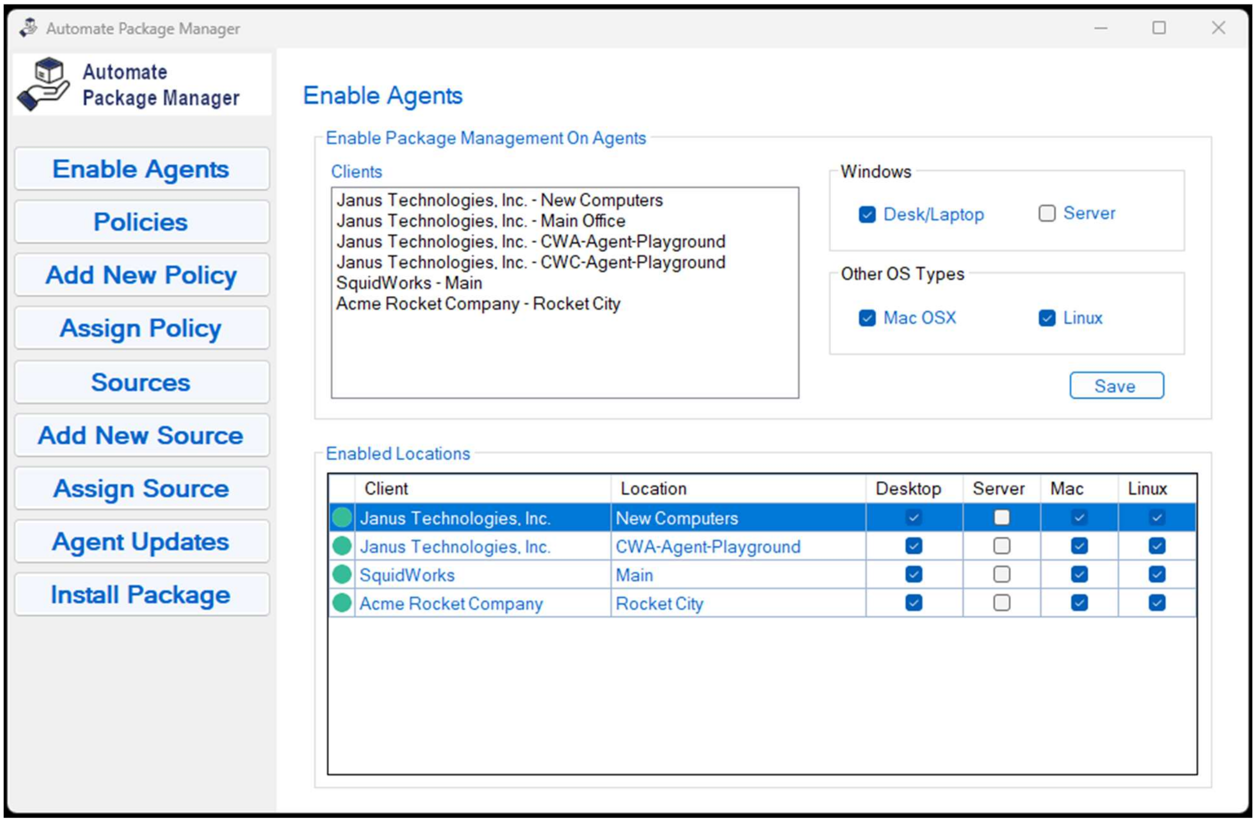
- ☐ APM - Enable Windows Desktop
- ☐ APM - Enable Windows Servers
- ☐ APM - Enable Linux
- ☐ APM - Enable Mac

Under the Location consoles are EDFs placed here by Automate Package Manager to manage onboarding of new agents. These are automatically set when you enable a location in the main tool. You can, however, make temporary adjustments here if needed to stop the onboarding process.

Example Usage Scenario: You have enabled a client's location(s) and onboarding has been running for a while, but you are about to add a new subset of computers that you do not want to onboard. You need to temporarily stop onboarding for Windows Desktops.

By unchecking **APM – Enable Windows Desktop,** you stop the Automate Group automation from looking for new Windows Desktop agents in this location.

Enable Agents



Enable Agents

Enable Package Management On Agents

Clients

Janus Technologies, Inc. - New Computers
Janus Technologies, Inc. - Main Office
Janus Technologies, Inc. - CWA-Agent-Playground
Janus Technologies, Inc. - CWC-Agent-Playground
SquidWorks - Main
Acme Rocket Company - Rocket City

Windows

☑ Desk/Laptop ☐ Server

Other OS Types

☑ Mac OSX ☑ Linux

Enabled Locations

	Client	Location	Desktop	Server	Mac	Linux
●	Janus Technologies, Inc.	New Computers	☑	☐	☑	☑
●	Janus Technologies, Inc.	CWA-Agent-Playground	☑	☐	☑	☑
●	SquidWorks	Main	☑	☐	☑	☑
●	Acme Rocket Company	Rocket City	☑	☐	☑	☑

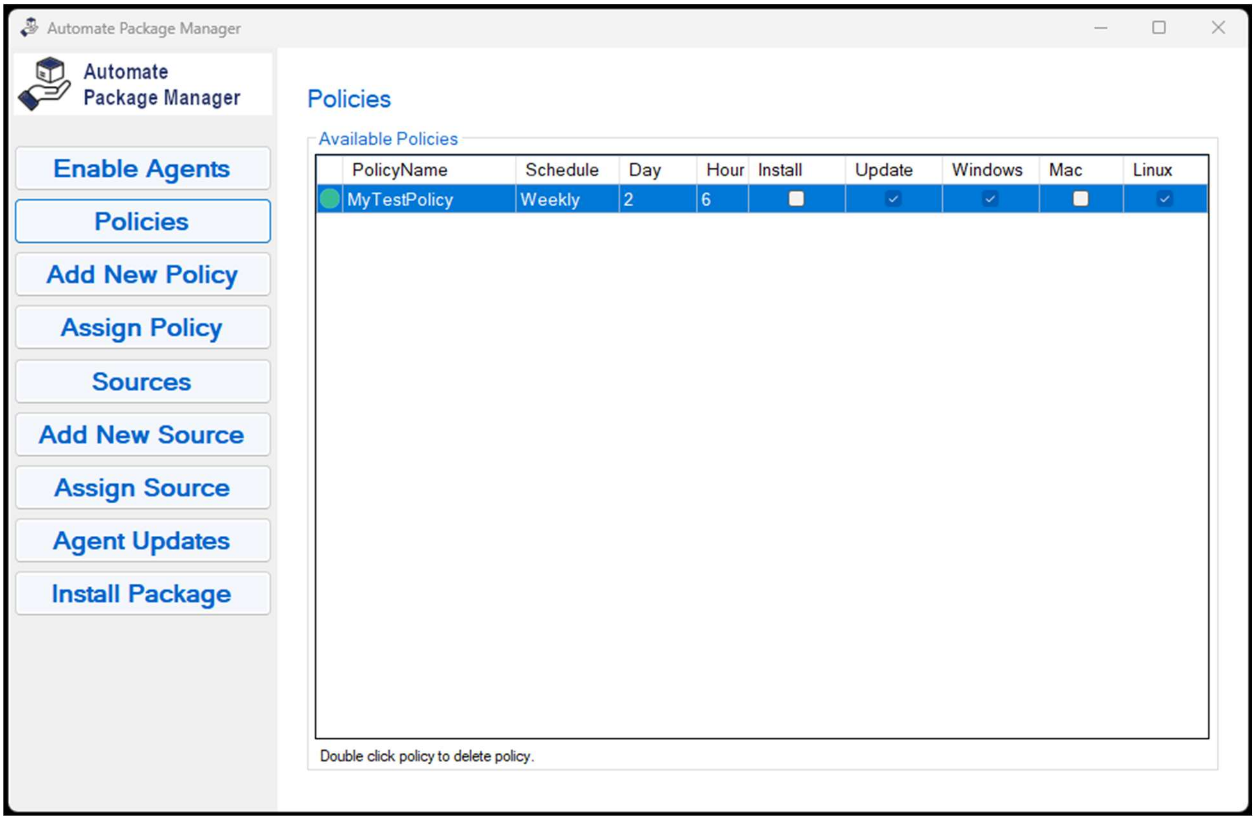
Now that you have a basic understanding of how Automate Package Manager integrates with ConnectWise Automate we can get into how to use this tool.

We start our configurations with **Enable Agents.** This view allows you to enable agents in bulk (by client location) and for one or more OS types. Keep in mind that you can disable agents that you do not want to manage using the agent EDFs. You can also enable agent in the same manner, if you only want a handful of agents to be enabled, you can enable agent in the same manner.

When you enable a location, it assumes that onboarding new agents is desired. This means you should expect to see the same OS types selected under the location's EDFs. If automated onboarding is not desired, navigate to the location console and modify the EDF's for the Automate Package Manager.

Enabled agents without a policy are passive. This means that you can enable agents without installing or updating them. You may do this to allow an agent to be scanned for currently installed packages and their version numbers without installing any packages or updating installed packages through automation.

Policies



Policies

Available Policies

	PolicyName	Schedule	Day	Hour	Install	Update	Windows	Mac	Linux
●	MyTestPolicy	Weekly	2	6	□	☑	☑	□	☑

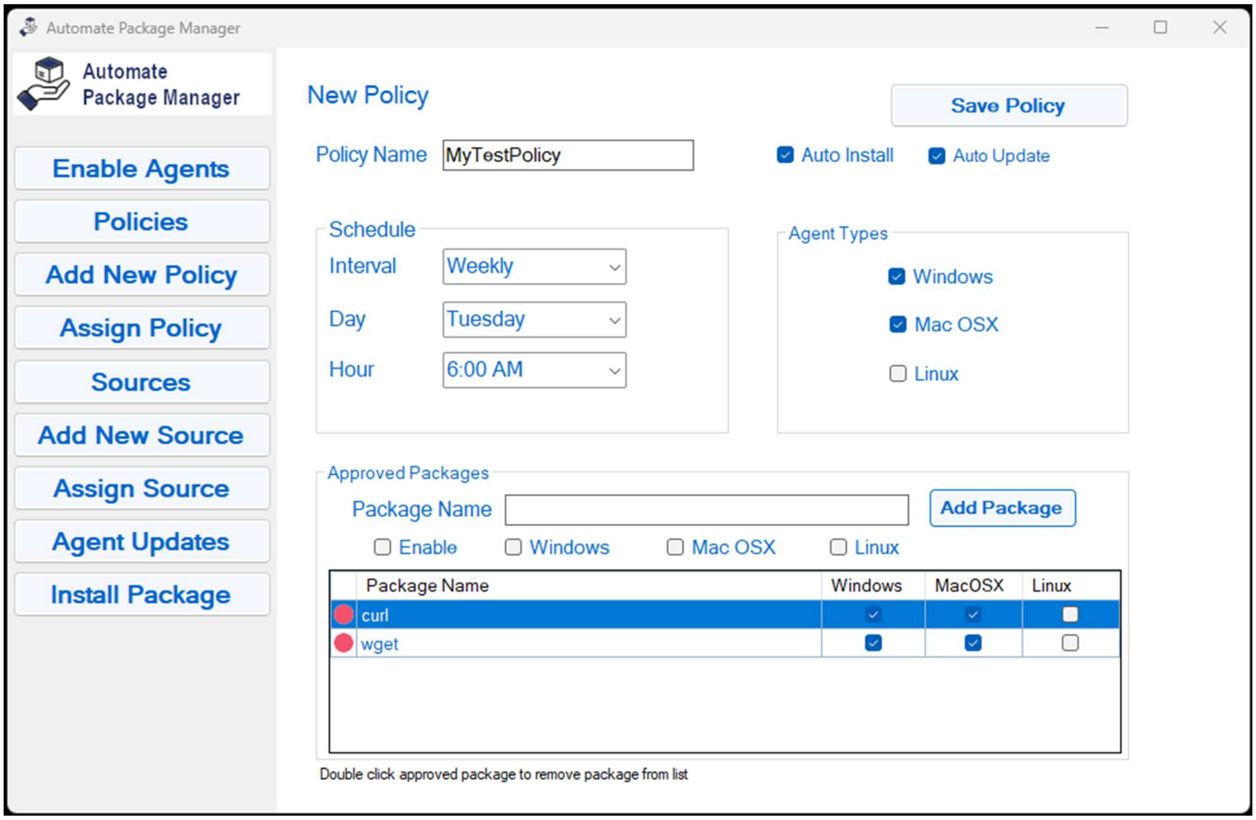
Double click policy to delete policy.

Under this control, you have access to all policies that you have created using the **Add New Policy** control.

A policy defines how Automate Package Manager interacts with the end agents. A policy lets you define what automated processes run, when they run and on what OS types they should run on.

You also can define a Approved Package List with any policy that if the Auto Install function is set, will force the OS type to install packages defines.

Add New Policy



New Policy

Policy Name MyTestPolicy

☑ Auto Install ☑ Auto Update

Schedule

Interval Weekly

Day Tuesday

Hour 6:00 AM

Agent Types

☑ Windows

☑ Mac OSX

☐ Linux

Approved Packages

Package Name

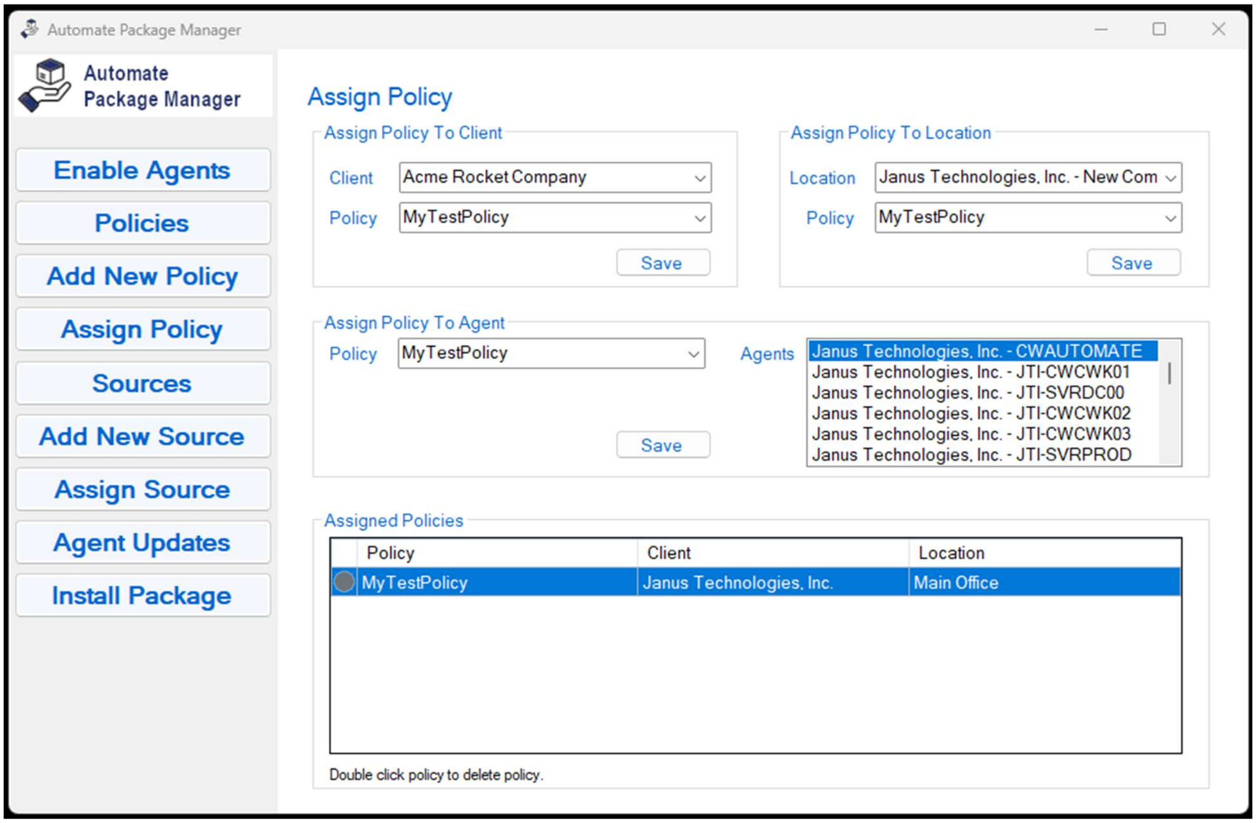
☐ Enable ☐ Windows ☐ Mac OSX ☐ Linux

	Package Name	Windows	MacOSX	Linux
●	curl	☑	☑	☐
●	wget	☑	☑	☐

Double click approved package to remove package from list

When adding a new policy, your policy will need a name, a schedule, and one or more OS types. You can create policies that only update or policies that also include a list of packages to auto install. Once a policy is saved, it will be seen in the Policy list.

Assign Policy



Assign Policy

Assign Policy To Client

Client: Acme Rocket Company

Policy: MyTestPolicy

Assign Policy To Location

Location: Janus Technologies, Inc. - New Com

Policy: MyTestPolicy

Assign Policy To Agent

Policy: MyTestPolicy

Agents:
Janus Technologies, Inc. - CWAUTOMATE
Janus Technologies, Inc. - JTI-CWCWK01
Janus Technologies, Inc. - JTI-SVRDC00
Janus Technologies, Inc. - JTI-CWCWK02
Janus Technologies, Inc. - JTI-CWCWK03
Janus Technologies, Inc. - JTI-SVRPROD

Assigned Policies

Policy	Client	Location
MyTestPolicy	Janus Technologies, Inc.	Main Office

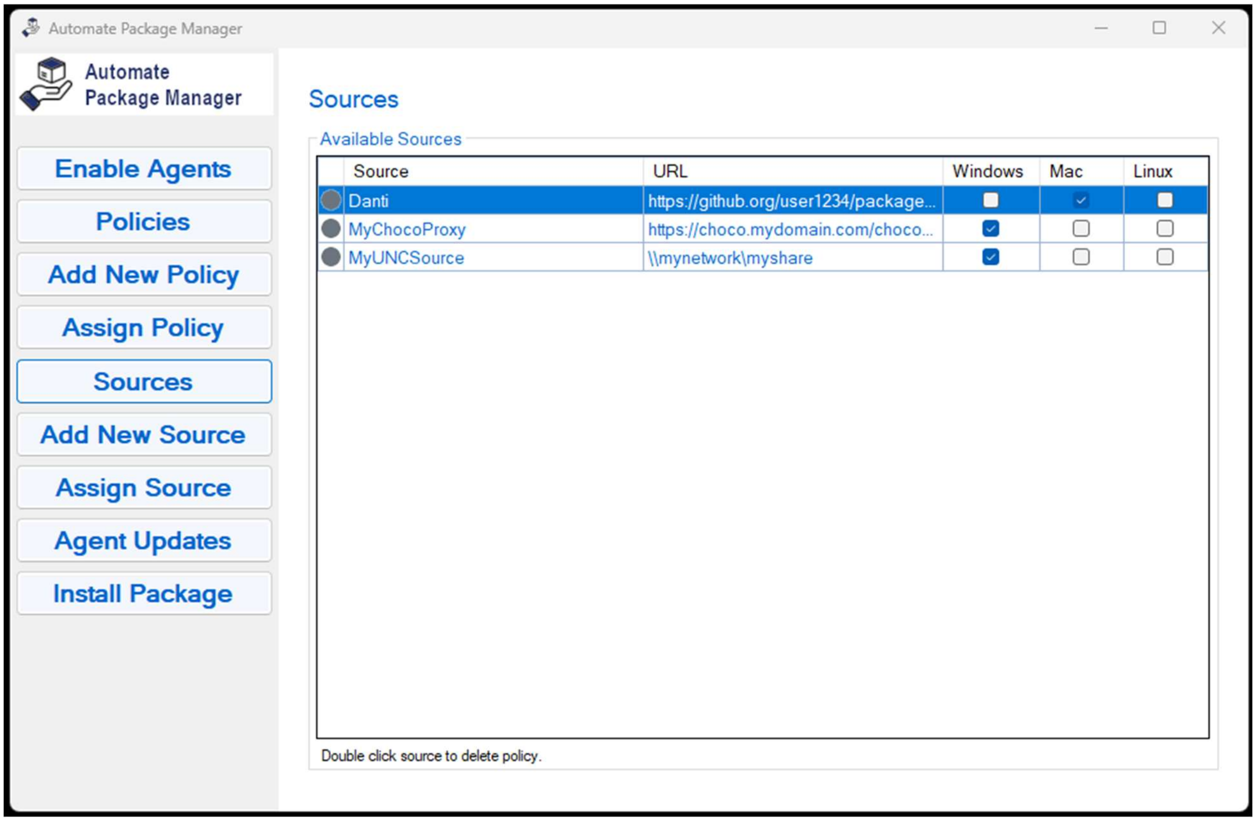
Double click policy to delete policy.

Once you have one or more policies created, you can assign them to Clients, Locations, and Agents. You can assign a policy more than once to multiple clients, locations, and agents. You can modify individual agent's policies by removing or changing the policy name under the agents EDFs for Automate Package Manager.

Example Usage Scenario: You want to be able to update existing packages but not install any package through automation.

Create a policy with a name of "Update Only", select the Auto Update check box, set an update schedule, and select the agent types the policy should apply. If this is a single OS type policy, then adding the OS type to name (Update Only -Windows Server) may be helpful.

Sources



Sources

Available Sources

	Source	URL	Windows	Mac	Linux
●	Danti	https://github.org/user1234/package...	□	☑	□
●	MyChocoProxy	https://choco.mydomain.com/choco...	☑	□	□
●	MyUNCSource	\\mynetwork\myshare	☑	□	□

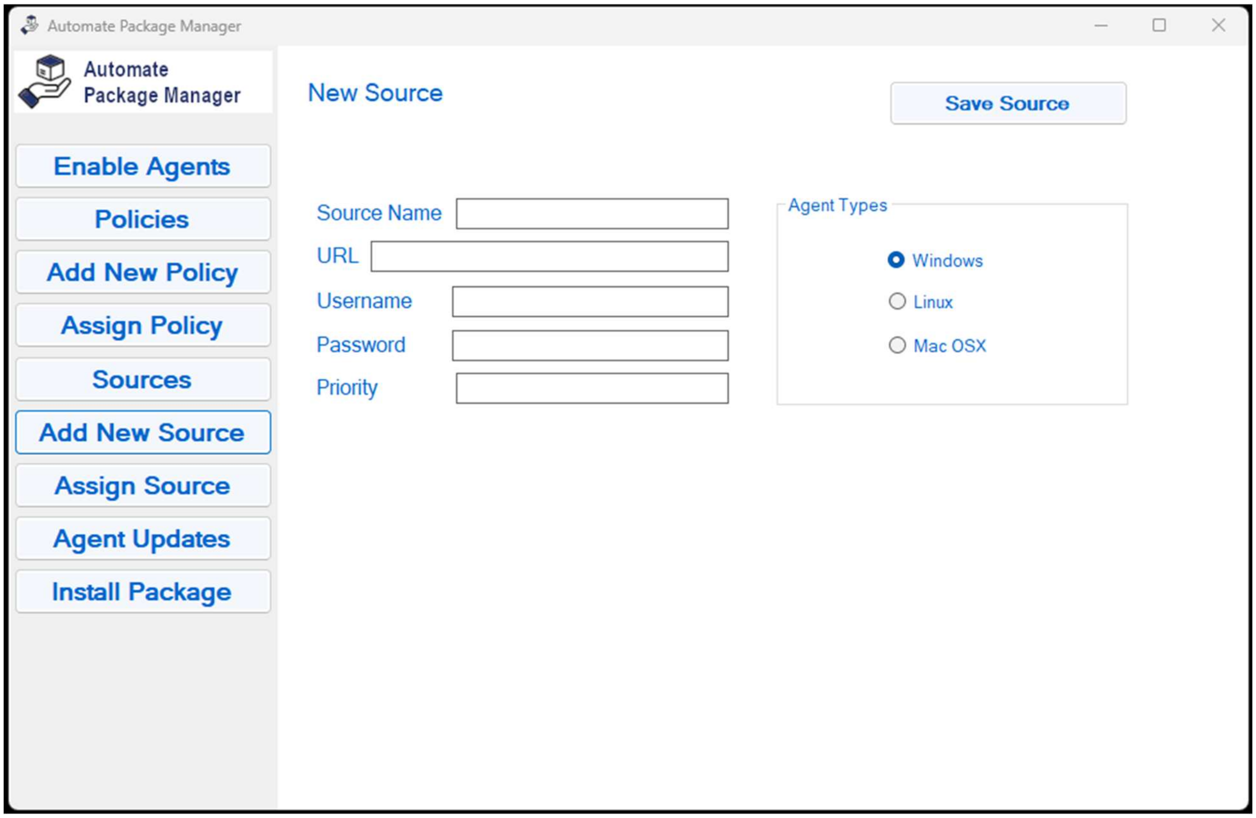
Double click source to delete policy.

Sources is the way you define where to look for desired packages. In some cases, this may be an alternate source or proxy for any of the package managers. The Source view displays the sources you have defined and for what OS types these sources are intended.

Example Usage Scenario: You want to avoid the dreaded 429 error you get with using the Chocolatey Public Repositories.

You have looked over our “Run a $7 a month Chocolatey Proxy” and have setup and configured your own proxy. Adding that proxy to your source list will allow you to assign a source to a client, a location, or to the agent(s). Once assigned, automation will push the source configurations down to agents effected.

Add New Source



New Source

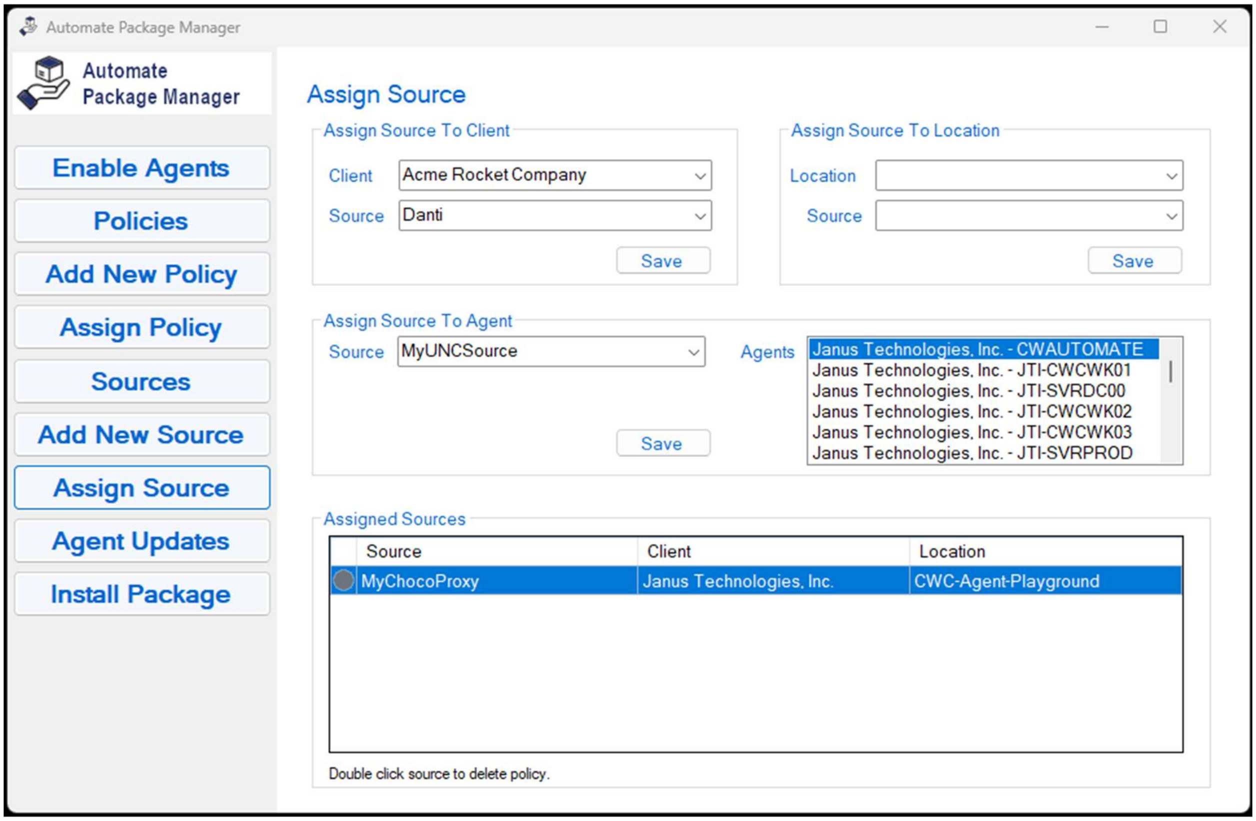
Source Name
URL
Username
Password
Priority

Agent Types

● Windows
○ Linux
○ Mac OSX

To create a new source for package retravel, you will need to provide a friendly name for your source, the URL/UNC where the repository resides, and any authentications needed to access repository. Where applicable you can also set the priority of the source over pre-existing sources configure on the agent(s).

Assign Source



Assign Source

Assign Source To Client

Client: Acme Rocket Company

Source: Danti

Assign Source To Location

Location:

Source:

Assign Source To Agent

Source: MyUNCSource

Agents:

Janus Technologies, Inc. - CWAUTOMATE
Janus Technologies, Inc. - JTI-CWCWK01
Janus Technologies, Inc. - JTI-SVRDC00
Janus Technologies, Inc. - JTI-CWCWK02
Janus Technologies, Inc. - JTI-CWCWK03
Janus Technologies, Inc. - JTI-SVRPROD

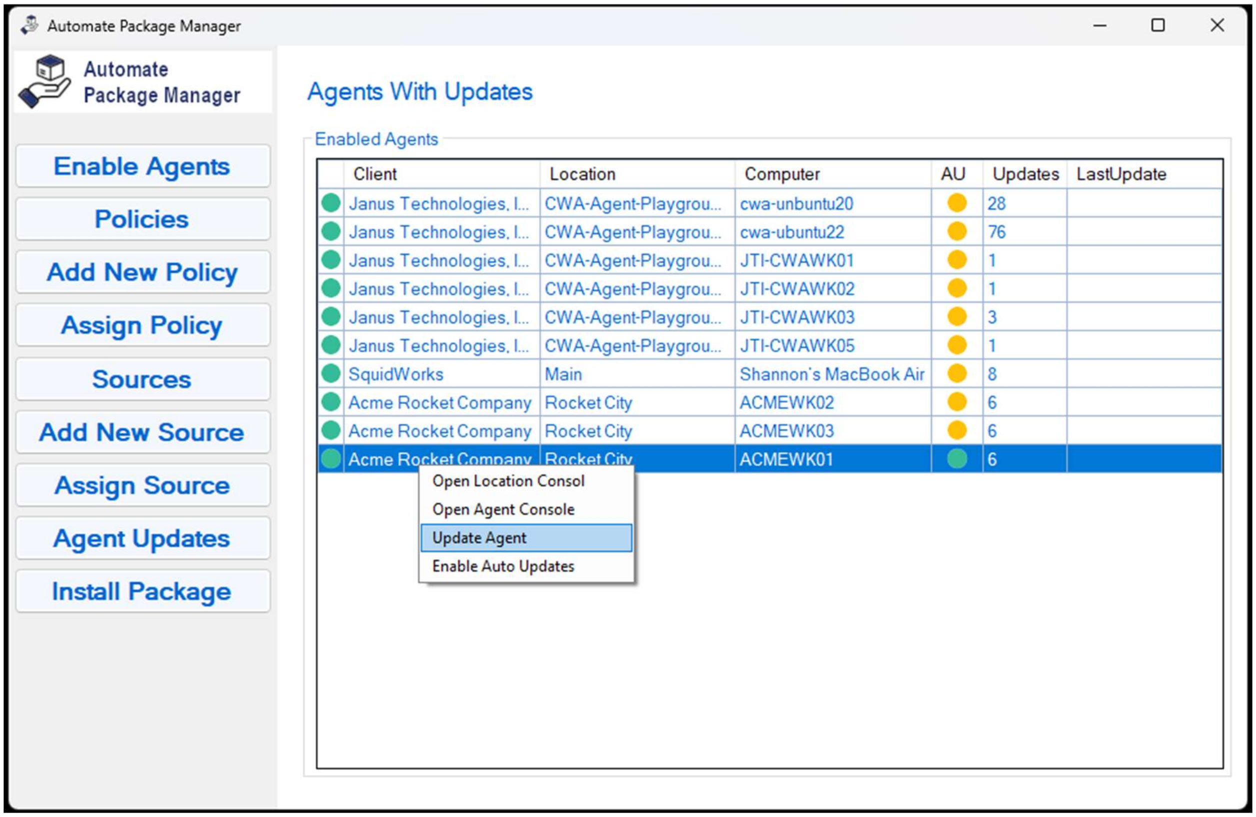
Assigned Sources

	Source	Client	Location
●	MyChocoProxy	Janus Technologies, Inc.	CWC-Agent-Playground

Double click source to delete policy.

Just like policies, you can assign a source to a client, a location, or agent(s). Once assigned, you can see the assignment listed and its status in the EDFs of the agent. Groups are responsible for the automation of the function. Once the agent joins the **APM Deploy Source Policy** group, it will deploy the source to agent and then exit this group once deployed.

Agent Updates



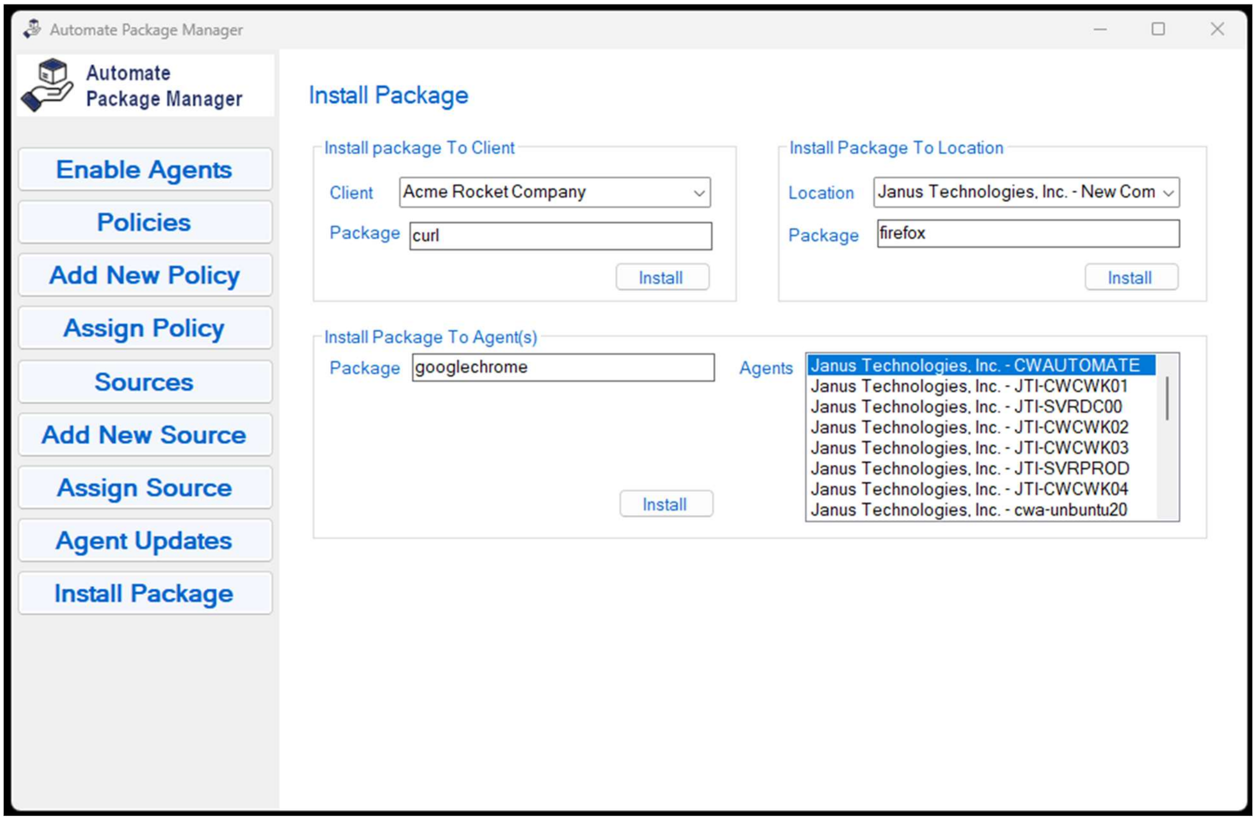
	Client	Location	Computer	AU	Updates	LastUpdate
🟢	Janus Technologies, I...	CWA-Agent-Playgrou...	cwa-unbuntu20	🟡	28	
🟢	Janus Technologies, I...	CWA-Agent-Playgrou...	cwa-ubuntu22	🟡	76	
🟢	Janus Technologies, I...	CWA-Agent-Playgrou...	JTI-CWAWK01	🟡	1	
🟢	Janus Technologies, I...	CWA-Agent-Playgrou...	JTI-CWAWK02	🟡	1	
🟢	Janus Technologies, I...	CWA-Agent-Playgrou...	JTI-CWAWK03	🟡	3	
🟢	Janus Technologies, I...	CWA-Agent-Playgrou...	JTI-CWAWK05	🟡	1	
🟢	SquidWorks	Main	Shannon's MacBook Air	🟡	8	
🟢	Acme Rocket Company	Rocket City	ACMEWK02	🟡	6	
🟢	Acme Rocket Company	Rocket City	ACMEWK03	🟡	6	
🟢	Acme Rocket Company	Rocket City	ACMEWK01	🟢	6	

The **Agent Update** view displays all agents that are enabled and have at least 1 update available. Each agent listed that has a policy assigned may be between schedules and may eventually resolve themselves.

You have the option in this view to manually push out the update commands or enable / disable Auto Updates with or without an assigned policy that has not defined auto updates prior. If auto updates have been executed on agent at least once, you will see the date that last update ran on agent.

You can open agent and location consoles directly from this view to quickly navigate the EDFs set for the agent.

Install Package



Install Package

Install package To Client

Client: Acme Rocket Company

Package: curl

Install Package To Location

Location: Janus Technologies, Inc. - New Com

Package: firefox

Install Package To Agent(s)

Package: googlechrome

Agents:
Janus Technologies, Inc. - CWAUTOMATE
Janus Technologies, Inc. - JTI-CWCWK01
Janus Technologies, Inc. - JTI-SVRDC00
Janus Technologies, Inc. - JTI-CWCWK02
Janus Technologies, Inc. - JTI-CWCWK03
Janus Technologies, Inc. - JTI-SVRPROD
Janus Technologies, Inc. - JTI-CWCWK04
Janus Technologies, Inc. - cwa-unbuntu20

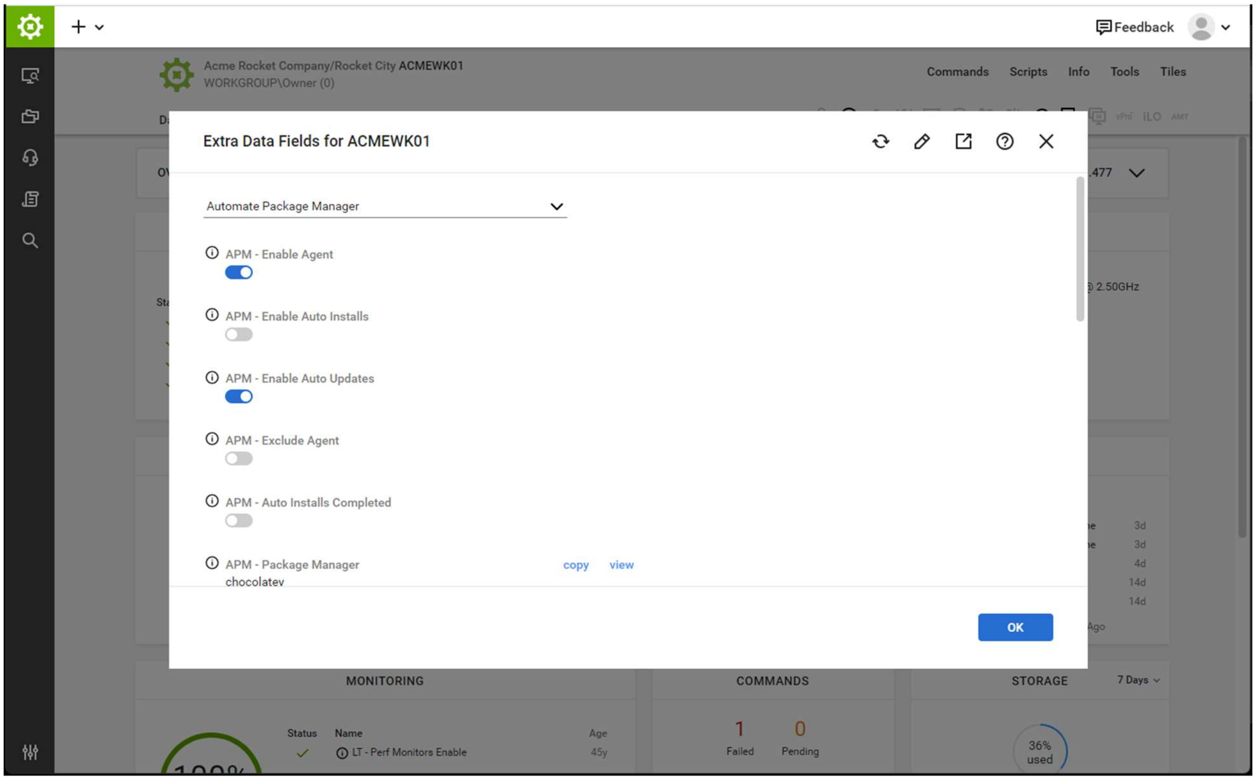
The Install Package view allows you to manually install any package available to the agent or agents. Quickly deploy software to a client, a location, or agent(s) with a single click. Just select the targets and provide a package name. If the package resides on the repositories for the agent, then the package will be installed on agent(s).

Many packages are common across all OS types. Packages like CURL, WGET are common command line packages that can be deployed using any package manager (Chocolatey, Apt-get, YUM, Homebrew). Some packages may require a desktop enabled operating system like Windows, Ubuntu or Mac OSX. Google Chrome and Firefox are examples of available packages for all OS types but require a desktop enabled version of the OS to work correctly. If you install Firefox to a base CentOS system, without a desktop enabled Firefox is unusable.

Example Usage Scenario: You have onboarded a new client, you have several scripts that you run that use cURL to retrieve files from the Internet and want to pre deploy the cURL package to all agents under client, so these scripts work when assigned to agents.

Using **Install Package**, deploy cURL across the entire client and all OS types with just one click.

Manage Agents Via Web Interface



Extra Data Fields for ACMEWK01

Automate Package Manager

- APM - Enable Agent: on
- APM - Enable Auto Installs: off
- APM - Enable Auto Updates: on
- APM - Exclude Agent: off
- APM - Auto Installs Completed: off
- APM - Package Manager: chocolatey

Once the plugin is installed and base configurations set, you will be able to access the contents of the agent’s status and effect the agent settings through the technical side of the ConnectWise Automate Web Portal.