

Canopy RMM Agent “Leaf”

Frequently Asked Questions

What is Leaf?	2
What functionality does Leaf enable on the endpoint?	2
How is Leaf deployed?	2
What are Leaf Plug-Ins?	2
What are out-of-the-box Leaf Plug-Ins?	3
What are advanced or custom Leaf Plug-Ins?	3
What are self-enabled Leaf Plug-Ins?	4
What kind of remote actions on devices does Leaf enable?	4
Does Leaf support advanced device management actions?	4
What kind of Operating Systems does Leaf work with?	5
Which chipset architectures work with Leaf?	6
Does Leaf have device operating requirements?	6

What is Leaf?

Leaf is Canopy's endpoint software Agent that is deployed onto a host controller device in the target environment. Leaf is designed to be installable anywhere, allowing you to install on any Operating System environment out-of-the-box.

Leaf is OS agnostic and has out-of-the-box Linux, Windows, and MacOS integration capabilities. Leaf is designed to use minimal resources while in operation and carries a very minimal footprint, allowing it to be deployed on very small devices.

What functionality does Leaf enable on the endpoint?

Leaf enables a range of functionality including data capture and transmit, command execution, software distribution, and local watchdog. You can then extend the Agent's capabilities through its Plug-In architecture, allowing you to leverage Plug-Ins that interface to unique hardware devices, control systems, and software applications that make-up your endpoint environment.

How is Leaf deployed?

Leaf has very flexible deployment capabilities, allowing your team to onboard devices into Canopy in the best manner for your device type, environment, and remote management operations. Deployment capabilities include:

- Silent Installer - Local

- Over the Air
- SSH Implementation
- SW Imaging Process
- Kubernetes
- Docker

What are Leaf Plug-Ins?

The Leaf agent is a tool that handles the basic transmission of data and actions between devices and Canopy. Leaf Plug-Ins leverage that tool to power your device management needs. Canopy offers three types of Leaf Plug-Ins:

1. **Out-of-the-Box** – Basic data capture and management features that come out-of-the-box with Leaf / Canopy that are widely used to jump start your Remote Monitoring and Management (RMM) journey.
2. **Advanced** – Custom Plug-Ins Canopy provides that are use case or device specific and can be easily implemented.
3. **Self-Enabled** – Building your own Leaf Plug-In utility is easy. It can often be as simple as writing out a json file to directly send data from your internal applications and scripts to Leaf and then into Canopy.



What are out-of-the-box Leaf Plug-Ins?

Built-in Leaf services provide basic data capture and management features that are widely used to jump start your Remote Monitoring and Management (RMM) journey. A few examples include:

PLUG-IN TYPE	DESCRIPTION
System Utilization and Alerting	Monitoring of CPU, Memory, Disk Space
System Information	OS version, network, BIOS, HW info, etc.
Software State Monitor	Alert and watchdog important applications and services
Ping Monitor	Keep an eye on network attached devices
SNMP Data Monitor	Extract information from an SNMP enabled device
File Monitor	Report on file updates to monitor configuration, logs, or file flags
Database Monitor	Extract information directly from a database at the IoT edge

What are advanced or custom Leaf Plug-Ins?

Advanced Plug-Ins are customized Leaf utilities that can be easily implemented. These utilities are often use case or device specific and enable greater visibility and control at the edge. A few common examples of advanced Leaf Plug-Ins are:

PLUG-IN TYPE	DESCRIPTION
Process Sentinel	Keep unwanted applications from starting and alert if they try (great for public facing kiosks)
System Monitor	Deep hardware metrics and analytics leveraging Libre Hardware to monitor CPU cores, temperatures, disks, fans, etc
Windows Event Monitor	Alerts on specific event types that are important to your device
Display Info	Alert on changes in resolution, disconnected cables



What are self-enabled Leaf Plug-Ins?

Building your own Leaf Plug-In utility is easy and flexible, allowing users to get creative. Below are a few common examples of how users leverage Leaf's Plug-In architecture for their own software:

PLUG-IN TYPE	DESCRIPTION
Application Heartbeat	Customer application feeds directly into a health KPI for the device
Connected Peripheral Monitor	Customer app already captured state of various USB and other connected peripherals (connectivity, versions, state etc.), and simply passed the full hardware state to use in Canopy
Configuration Tracker	Directly loading application and device configuration to get a remote view into the current site configuration

What kind of remote actions on devices does Leaf enable?

- Download
- Executor
- Fast Poll
- File Information
- List Windows Updates
- Reboot
- Rename Computer
- Screenshot
- Service Control
- List Directory
- System Updates
- Build Your Own!

Note: Some actions listed may not be available for a certain supported Operating Systems.

Does Leaf support advanced device management actions?

Leaf can be configured to support many kinds of advanced device management actions that can be executed remotely. Example advanced device management capabilities include:

- Software Distribution
- Software and Firmware Management
- Agent Update and Management
- Content Updates
- Offline Execution Logic
- Automation Deployment



What kind of Operating Systems does Leaf work with?

Leaf installs as a native Operating System executable on Linux, Windows, and MacOS devices. Leaf is supported and tested against mainstream supported and long term support (LTS) versions of these Operating Systems. Beyond these native Operating System integrations, the Leaf

architecture is compatible with a much wider range of Operating System versions. Leaf does not require or link to any Operating System libraries, but may require access to certain native functions for the full range of capabilities (for example, functions typically found in `/usr/bin` or `C:\Windows\System32`).

Supported & Compatible OS Versions:

OPERATING SYSTEM	SUPPORTED EXAMPLES	COMPATIBLE EXAMPLES
Linux	<ul style="list-style-type: none"> • Ubuntu 22.04 • Ubuntu 18.04 • CentOS 7-2009 • Raspbian (latest) • RHEL 8 • RHEL 7 • And many more 	<ul style="list-style-type: none"> • Ubuntu 20.10 • CentOS 6.8 • Raspbian (prior versions) • Debian Squeeze 6.0.3 • And many more
Windows	<ul style="list-style-type: none"> • Windows 10 • Windows 10 IoT • Windows 11 • Windows Server 2019 • And many more 	<ul style="list-style-type: none"> • Windows XP • Windows 7 • Windows Embedded Standard 7 • Windows 8.1 • And many more
MacOS	<ul style="list-style-type: none"> • macOS 13 • macOS 12 • macOS 11 	<ul style="list-style-type: none"> • macOS 10.x

For additional questions regarding a specific Operating System version requirement and/or compatibility with Leaf, reach out via our website www.goCanopy.com to get in touch with a Canopy solution architecture who can assist with your evaluation process or email us at info@goCanopy.com.



Which chipset architectures work with Leaf?

Leaf also supports a wide range of chipset architectures across the various Operating Systems:

- amd64 (all 64-bit x86 processors)
- 386 (all 32-bit x86 processors)
- arm (EL and HF)
- arm64 (EL and HF)
- mips

- mips64
- Others upon request

Does Leaf have device operating requirements?

Leaf is designed to operate with a minimal footprint within the target environment. This allows for maximum flexibility and for Leaf to be deployed into very small and/or “basic” devices. The following are broad guidelines for Leaf’s operating requirements:

SYSTEM RESOURCE	LINUX / MAC OS	WINDOWS
CPU (<i>see note</i>)	No minimum requirement	No minimum requirement
Memory	100 MB	256 MB
Storage (<i>see note</i>)	75 MB	75 MB

CPU Note: Any CPU capable of running the host Operating System will be capable of running Leaf. During operation, Leaf maintains a minimal CPU footprint, usually below 1% of available CPU. Because Leaf may be configured to more aggressively monitor and report on telemetry, this percentage could vary per use case.

Storage Note: Leaf’s configuration capabilities allow for variable amounts of storage to be dedicated to queues and logs written to disc. Please work with the Canopy implementation team if you have specific requirements in this area.

