

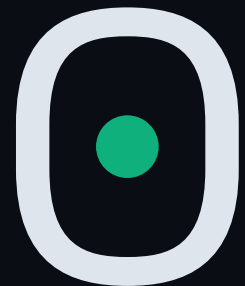
POISON/ZERO

Onboarding

Six steps for admins and developers, from the install key to feedback.

- 01 Prepare the console
- 02 Install
- 03 Set up
- 04 Use
- 05 View telemetry
- 06 Give feedback

For **poison0** **exec0**



1 Prepare the console

PLAN

CONSOLE

Enterprise `https://<tenant>.poison0.com` or your own domain

Solo Dev, Pro `https://console.poisonzero.com`

- 1 Invite more admins: **Settings** → **Access & Security** → **People** → enter the email → **Invite**. The link is valid for 7 days.
- 2 Create an install key: **Devices** → **Roll out many devices with a key** → **Choose variant: Cloud** → **Generate key**. The key is shown only once.
- 3 Pass the key to developers or IT. It is valid for 30 days and for as many devices as your plan allows (Enterprise: unlimited).

2 Install

Option A or B. Both set up `/pz` and the status line in Claude Code.

Check after installing: restart Claude Code. The status line shows  **PoisonZero** .

2A. Developer installs it

MACOS / LINUX · TERMINAL

```
curl -fsSL https://poisonzero.com/install.sh | sudo PZ_ENROLL_CODE=<install-key> PZ_SETUP_AGENTS=claude PZ_SETUP_STATUSLINE=1 sh
```

WINDOWS · POWERSHELL AS ADMINISTRATOR

```
$env:PZ_ENROLL_CODE='<install-key>'; $env:PZ_SETUP_AGENTS='claude'; $env:PZ_SETUP_STATUSLINE='1'; irm https://poisonzero.com/install.ps1 | iex
```

Headless devices (e.g. over SSH): also set `PZ_HEADLESS=strict`. Unclear changes are then rolled back immediately instead of asking in a dialog. Without it, the poison0 agent detects this itself.

2B. IT deploys via Jamf, Intune or Ansible

Step 1: install (as root or SYSTEM)

MACOS / LINUX · ROOT

```
PZ_ENROLL_CODE=<install-key> sh -c "$(curl -fsSL https://poisonzero.com/install.sh)"
```

WINDOWS · SYSTEM

```
$env:PZ_ENROLL_CODE='<install-key>'; irm https://poisonzero.com/install.ps1 | iex
```

Step 2: set up Claude Code (as the signed-in user)

MACOS / LINUX · USER

```
poisonzero setup-claude
poisonzero setup-claude --statusline-only
```

WINDOWS · USER

```
& "$env:ProgramFiles\PoisonZero\poisonzero.exe" setup-claude
& "$env:ProgramFiles\PoisonZero\poisonzero.exe" setup-claude --statusline-only
```

JAMF · ROOT, IN THE SIGNED-IN USER'S ACCOUNT

```
u=$(stat -f%Su /dev/console)
sudo -u "$u" -H env -u SUDO_USER /usr/local/bin/poisonzero setup-claude
sudo -u "$u" -H env -u SUDO_USER /usr/local/bin/poisonzero setup-claude --statusline-only
```

TOOL SETTING

- Intune "Run this script using the logged on credentials: Yes". The full path is needed because `PATH` does not include the program folder.
- Jamf Runs as root; the Jamf tab runs step 2 in the account of the signed-in user.

2C. Device name at install time (optional)

Set one of these variables next to `PZ_ENROLL_CODE` . The first one set, in this order, applies:

VARIABLE	EFFECT
<code>PZ_DEVICE_NAME="Platform / Laptop 07"</code>	The name is taken as is
<code>PZ_DEVICE_NAMES_CSV=<file or https URL></code>	CSV with the header <code>hostname,name</code> . The installer looks up its own hostname and takes the name. Only the name is sent, never the hostname
<code>PZ_DEVICE_NAME_FROM_HOSTNAME=1</code>	The hostname becomes the device name. Not recommended

PZ-NAMES.CSV

```
hostname,name
mbp-0231,Platform / Laptop 07
```

Note: The device name is chosen by the administrator. The name should not contain personal data, for example no person's name.

Length	1 to 60 characters, no control characters
Invalid name	is ignored, the device is still enrolled
When the name is applied	only when a device is enrolled for the first time. For a device that is already enrolled, change the name in the console: Devices → Rename device .
Available from	version 1.16.45

3 Set up

1 Name devices that have no name: **Devices** → on the device **Rename device** → enter the **Display name of the device**, for example "Platform / Laptop 07".

2 Separate teams (optional): create one install key per team, each with its own **Profile (optional)**. Manage profiles: **Settings** → **Protection & Devices** → **Profiles**.

3

Check tool calls with `exec0` (optional)

`exec0` checks every tool call of your AI agents before it runs. The device needs a GPU suitable for Melira Ultra. The device page shows whether a device can run `exec0`.

To set it up: **Settings** → **Protection & Devices** → **Profiles** → **Edit** on the profile → **`exec0` with Melira AI** → **Mode**

MODE	EFFECT
Off (default)	No checks
Observe	Checks every call. Hard rules still block, dangerous calls are only counted. 14 days by default, after that <code>exec0</code> keeps observing and does not switch to Protection by itself.
Protection	Blocks dangerous calls before they run

Unclear cases: in both modes the person at the device decides.

Check not possible: in Protection the call is blocked (default).

Private devices: set it on the device; on Windows without `sudo` in a terminal run as administrator.

COMMAND	EFFECT
<code>sudo poison0 exec0 observe</code>	Observe
<code>sudo poison0 exec0 enforce</code>	Protection
<code>sudo poison0 exec0 off</code>	Off
<code>poison0 exec0 fail open</code>	Let a call through when <code>exec0</code> cannot check it

More: [How `exec0` checks tool calls](#)

4

Use

WHERE	INPUT	RESULT
Claude Code	<code>/pz</code>	Protection stats for this month
Claude Code	Status line	● protection running; ● service running but reports a problem; ● service not running
Command line	<code>poisonzero stats</code>	Protection stats
Command line	<code>sudo poisonzero status</code>	All decisions, open quarantine
Command line	<code>sudo poisonzero quarantine list</code>	Held-back changes; <code>show</code> , <code>restore</code> , <code>discard <id></code>
Command line	<code>poisonzero doctor</code>	Self-test
Command line	<code>sudo poisonzero egress</code>	What left the machine (Linux: also without <code>sudo</code>)
Command line	<code>poisonzero exec0 status</code>	exec0 mode on this device, hook in Claude Code, last 24 hours

Windows: no `sudo` ; use PowerShell as administrator and the full path `& "$env:ProgramFiles\PoisonZero\poisonzero.exe"` .

UPDATE VIA	<code>/PZ</code> IN CLAUDE CODE
<code>install.sh</code> , <code>install.ps1</code>	Existing <code>/pz</code> files of the installing user are renewed automatically
Homebrew, Scoop, <code>.deb</code> / <code>.rpm</code> , automatic update	Each user runs <code>poisonzero setup-claude</code> once

5

View telemetry

CONSOLE	CONTENT
Devices	State and incidents per device
Analytics	Blocked changes by type and day; filter by device, agent, operating system, 7/30/90 days
Audit	Every incident with its redacted attack content

Never transmitted: full file contents (only redacted changes), files outside the protected paths, process and system information.

6

Give feedback

WHERE	INPUT
Claude Code	<code>/pz feedback <text></code>
Command line	<code>poisonzero feedback "<text>"</code> (no <code>sudo</code>)
ITEM	RULE
Length	At most 2000 characters
Redaction	Locally before sending: secrets, email addresses, IBANs, card numbers, IP addresses, phone numbers, key and token assignments, and the user name in home paths (<code>/Users/<name></code> , <code>/home/<name></code> , <code>C:\Users\<name></code>). The command shows the text after redaction
Sending	With the next check-in. The device sends only the text, version and platform; on arrival PoisonZero stores the device's current name with it
Recipient	Only the PoisonZero team sees the text, not your own admins
Reply	If the team replies, the account owner is told the date and device of the feedback, not its text
Private mode	Blocked; use the contact form on poisonzero.com instead

Read next

Every installer variable and path: [installing the poison0 agent](#). What the console shows per incident: [incidents and audit](#). How updates arrive: [updates and lifecycle](#).