

Log Keeper: install, deploy and operate

A technical guide for IT managers to the CodeDelta AI Agent Tracer Log Keeper: why coding-agent logs disappear, what the keeper copies and where, and how to install it on one machine or many.

1. Why it is needed: the logs vanish

AI coding agents (Claude Code, OpenAI Codex, Cursor, GitHub Copilot CLI, Gemini CLI, Google Antigravity) write a session log on the developer's machine. The log records what the agent was asked, every command it ran, the files it touched and the hosts it contacted. It is the only record of what the agent did on that machine.

The agents own those logs and remove them on their own schedule:

- **Claude Code deletes session transcripts after 30 days by default.** The deletion runs as a background sweep after a session starts and shows no message. The setting is `cleanupPeriodDays` (default `30`, minimum `1`); sub-agent transcripts are removed with their parent session. Source: Claude Code documentation, [settings reference](#) and [application data, cleaned up automatically](#). Sessions started or last continued in Claude Desktop or Cowork are kept at any age unless `desktopSessionCleanupPeriodDays` is set.
- The other agents' retention is set by each vendor and can change between versions. The keeper does not depend on it.
- Any user can delete the logs at any time; they are ordinary files in the user's home folder.

A review, an audit or an incident investigation that starts more than a month after the event will usually find the agent's record gone. The Log Keeper copies each machine's agent logs, on a schedule, into a keep folder before the agents remove them. The AI Agent Tracer reads a keep folder exactly as it reads the live logs.

For Claude Code only, retention can also be extended at the source through managed settings:

`"cleanupPeriodDays": 3650` (the value `0` fails validation). This lengthens Claude Code's own retention; it does not protect the logs from deletion by the user or cover the other agents.

2. What the keeper copies

AGENT	SOURCE ON THE MACHINE
Claude Code	<code>~/.claude/projects</code> (session transcripts and sub-agent transcripts)
OpenAI Codex	<code>~/.codex/sessions</code>
Cursor	<code>~/.cursor/projects</code> , plus Cursor's state database <code>state.vscdb</code> (macOS: <code>~/Library/Application Support/Cursor/User/globalStorage/</code> ; Linux: <code>~/.config/Cursor/User/globalStorage/</code> ; Windows: <code>%APPDATA%\Cursor\User\globalStorage\</code>), which holds each session's name and model
GitHub Copilot CLI	<code>~/.copilot/session-state</code>
Gemini CLI	<code>~/.gemini/tmp</code>
Google Antigravity	<code>~/.gemini/antigravity/conversations</code>

Only folders that exist are read. The keeper reads the sources and never modifies or deletes them.

3. How it copies

- **Layout.** Each machine writes to its own subfolder, `<keep folder>/<host>-<user>/` , mirroring the source paths (`.claude/projects/...` , `.codex/sessions/...`). Several machines can share one keep folder.
- **Incremental.** A file is copied when it is new or its size or modification time has changed since the last copy; unchanged files are skipped.
- **Compressed.** Log files are stored xz-compressed as `.jsonl.xz` (the agents' own JSONL format, one record per line). Database files (`state.vscdb` , Antigravity `.db`) are copied unchanged. Compression can be turned off with `--no-compress` .

- **Compaction (optional).** `--compact` replaces every base64 run of 2,000 characters or more (embedded screenshots and context snapshots) with a short marker. The JSON stays valid and every command and message is kept.
- **Never deletes.** Nothing is ever removed from the keep folder. A log the agent later deletes stays kept, and the manifest marks its source as gone.
- **Manifest.** `codedelta-keep-manifest.json` in each machine folder records, per file, the source path, original size and modification time, the SHA-256 of the original, the kept path and size, and the time kept; and, per run, the counts copied and skipped and any refusal.
- **Lock.** One run per machine folder at a time (`.codedelta-keep.lock`); a lock older than 6 hours is treated as left by a crash and taken over.

4. Safeguards

- **Disk floor.** With less than 10 GB free on the keep volume the keeper copies nothing and reports why. It warns below 20 GB free, and when the current growth rate would fill the volume within 90 days.
- **Unmounted volume.** A keep folder under `/Volumes/<name>` (macOS) whose volume is not mounted, or on a Windows drive letter that does not exist, is refused; nothing is written to the boot disk in its place.
- **Unreadable manifest.** If a machine's manifest exists but cannot be read, the run stops and changes nothing, so the record of earlier copies is never replaced.
- **Protected folders are refused** (next section).

5. Choosing the keep folder: protected folders cannot be used

The scheduled job runs in the background without a window. Operating-system file protection blocks it in certain folders: on macOS a background job was refused files the app itself had written in Desktop and iCloud Drive, with Full Disk Access granted. The keeper therefore refuses these folders in the app (Save, Keep now, Keep automatically), on the command line and in the scheduled job, and nothing is saved, created, copied or scheduled there:

PLATFORM	REFUSED AS KEEP FOLDER (AND ANYTHING INSIDE THEM)
macOS	Desktop, Documents, Downloads, iCloud Drive (<code>~/Library/Mobile Documents</code>), cloud storage (<code>~/Library/CloudStorage</code> — OneDrive, Google Drive, Dropbox, Box; also <code>~/Dropbox</code> , <code>~/Google Drive</code>)
Windows	Desktop, Documents, Downloads, Pictures, Music, Videos, Favorites, OneDrive (including the folders named by <code>%OneDrive%</code> , <code>%OneDriveCommercial%</code> , <code>%OneDriveConsumer%</code>), Dropbox, Google Drive, iCloudDrive. Microsoft Defender's Controlled folder access guards Documents, Pictures, Music, Videos and Favorites, and OneDrive-redirected folders, when it is switched on.
Linux	None refused.

Use a folder directly in the user's home folder, for example `~/tracer logs` . On macOS, a background job wrote to such a folder without any permission grant. Include that folder in the machine's normal backup so a copy also exists off the machine.

6. Requirements

- CodeDelta with the Log Keeper: version 2.2.1 or later.
- No licence is needed for the keeper. A lapsed licence never stops logs being kept.
- The keeper runs as the user whose agent logs it copies. It needs read access to that user's agent folders and write access to the keep folder.
- 10 GB or more free on the keep volume.

7. Install on one machine (the app)

1. Open CodeDelta and choose **AI Agent Tracer** mode. The **Log keeper** line is under the folder field.
2. Enter or select the keep folder (not a protected folder) and choose *Keep images* (full copy, compressed) or *Maximise compaction*. Press **Save**.
3. Press **Keep now** for a first copy. A first copy of several hundred MB of logs takes a minute or two; **Stop** ends it and what is already kept stays.

4. Choose when to keep automatically and tick **Keep automatically**:

- *after log-in, waiting N minutes* (0–240, default 15) — for machines switched off at night;
- *daily at HH:MM* — for machines left on or asleep.

The status line shows the last copy (files copied and unchanged, space used, free space, growth rate), the schedule and the folder the automatic job writes to. Saving a new folder or mode moves the installed job to it. **Remove keeper** removes the automatic job and the setting; kept files are never deleted.

8. Schedules and what each platform installs

PLATFORM	SCHEDULER ENTRY (ONE PER USER)
macOS	launchd user agent <code>~/Library/LaunchAgents/app.codedelta.logkeeper.plist</code> . After log-in: <code>RunAtLoad</code> . Daily: <code>StartCalendarInterval</code> .
Windows	Task Scheduler task <code>CodeDeltaLogKeeper</code> . After log-in: <code>/SC ONLOGON</code> . Daily: <code>/SC DAILY /ST HH:MM</code> .
Linux	A line in the user's crontab tagged <code># app.codedelta.logkeeper</code> . After log-in becomes <code>@reboot</code> (after start-up). Daily: <code>M H * * *</code> .

The wait after log-in is carried in the job's command as `--wait N` . Installing a schedule replaces any earlier keeper job for that user.

Daily runs and power state. launchd starts a job whose time passed while the Mac was asleep when it wakes, but a run whose time passed while the machine was switched off is not made up. For machines that are shut down outside working hours, use the after-log-in schedule.

9. Command line

The program: macOS `/Applications/CodeDelta.app/Contents/Resources/codedelta-gui/codedelta-gui` ; Windows `C:\Program Files\CodeDelta\codedelta-gui.exe` (default install folder); Linux `codedelta-gui` in the extracted bundle. Below, `codedelta-gui` stands for that path.

```
# copy now
codedelta-gui scan "<keep folder>" --mode tracer-keep --compact

# install: N minutes after log-in (default 15; 0-240)
codedelta-gui scan "<keep folder>" --mode tracer-keep --compact --install-at-login
--delay 15

# or install: daily at a set time (default 03:00; --hour H also accepted)
codedelta-gui scan "<keep folder>" --mode tracer-keep --compact --install-nightly
--at 03:00

# remove the scheduled job (kept files are not touched)
codedelta-gui scan "<keep folder>" --mode tracer-keep --remove-nightly
```

OPTION	EFFECT
<code>--compact</code>	Strip base64 runs of 2,000+ characters (images, snapshots).
<code>--no-compress</code>	Store copies uncompressed.
<code>--keep-from DIR</code>	Copy that folder instead of this machine's agent folders.
<code>--wait N</code>	Wait N minutes before copying (used by the after-log-in job).

Exit code `0`: the run completed. Exit code `2`: refused or failed (protected folder, disk floor, unmounted volume, unreadable manifest, files not kept, invalid schedule), with the reason on standard error.

10. Deploying to many machines

The scheduled job is per user: it is installed in the user's own scheduler and copies that user's logs. Deploy the application as usual, then run the install command **in the context of each user** (not as root or SYSTEM), for example from your device-management tool's user-context script or a log-in script:

```
# macOS, run as the logged-in user
"/Applications/CodeDelta.app/Contents/Resources/codedelta-gui/codedelta-gui" \
  scan "$HOME/tracer logs" --mode tracer-keep --compact --install-at-login --delay
15

# Windows, run in the user's context
"C:\Program Files\CodeDelta\codedelta-gui.exe" scan "%USERPROFILE%\tracer logs" --
mode tracer-keep --compact --install-at-login --delay 15

# Linux, run as the user
./codedelta-gui scan "$HOME/tracer logs" --mode tracer-keep --compact --install-
at-login --delay 15
```

The same command with `--remove-nightly` removes the job. The keeper writes to a folder and has no network code; to collect copies centrally, point the keep folder at a location your existing backup or file-collection tooling already gathers.

11. Checking that it runs

WHAT	WHERE
Output of each scheduled run	<code>logkeeper.log</code> in the CodeDelta settings folder: macOS <code>~/Library/Application Support/CodeDelta/</code> ; Windows <code>%APPDATA%\CodeDelta\</code> ; Linux <code>~/.config/codedelta/</code>
Outcome of the last command-line or scheduled run	<code>logkeeper-last.json</code> in the same folder: time, <code>ok</code> , the problem if any, files copied, unchanged and not kept
History of every run	<code>runs</code> in <code>codedelta-keep-manifest.json</code> in the machine folder
At a glance	The Log keeper status line in the app; a failed automatic run is shown first, with its reason

12. Reading the kept logs

A keep folder is a normal AI Agent Tracer input. Select it as the agent-logs folder in the app, or run:

```
codedelta-gui scan "<keep folder>" --mode tracer --out-dir <report folder>
```

The report reads the compressed files directly and states in its header that it was produced from a keep folder and the date the copies run to. A keep folder holding several machines' subfolders produces one report across all of them.

13. Limits of this version

- The job and the settings belong to the user. A user can change the keep folder, remove the job, or edit or delete kept files and the manifest. There is no administrator lock in this version.
- The SHA-256 values in the manifest show whether a kept file still matches the original; they do not prevent changes, because the manifest is in the same folder.
- No direct upload to cloud storage (S3, Azure Blob, Google Cloud Storage) in this version.
- Only the six agents in section 2 are covered.