

Two milestones. About 7 minutes total. Keep this open next to Claude.

THE FOUR THINGS THAT TRIP ALMOST EVERYONE

1. **Create the API key inside your SUB-ACCOUNT, not the agency view.** An agency-level key looks identical and will not work.
2. **Click "Select all" on the scopes step.** Hand-picking gives you something that half-works, which is harder to debug than broken.
3. **Copy the WHOLE Private Integration key.** It is longer than the visible box: click into the field and select all of it.
4. **Fully quit Claude when told to.** Not just close the window. Cmd-Q on Mac, or exit from the tray on Windows.

MILESTONE 1 OF 2 Get the core tools live

~5 min

Claude can now read and run your GHL account: contacts, pipelines, calendars, invoices, bulk operations. This is the free tier working.

- 1 **Find your license key.** It's in your welcome email. Search your inbox for "GHL Command". Can't find it? Email support and we'll resend within the day. Don't spend twenty minutes hunting.
- 2 **Install in Claude. One command does it.** Open a terminal (Mac: the Terminal app. Windows: press the Windows key, type `cmd`, press Enter), paste this, press Enter:

```
npx -y @elitedcs/ghl-mcp cli install
```

It finds Claude Desktop's settings file, adds GHL Command **alongside any tools you already have** (nothing gets overwritten), and saves a backup first. It says "Added GHL Command to Claude Desktop" when it's done. Using Claude Code in the terminal instead? `claude mcp add --scope user -t stdio ghl -- npx -y @elitedcs/ghl-mcp@latest`

Then **fully quit Claude**. Not just close the window: Cmd-Q on Mac, or right-click the Claude icon by the clock on Windows and choose Quit. Reopen it. Skipping this is the #1 reason people think the install failed.

Manual-edit fallback, only if you cannot run the command: Edit Config opens a folder, not a text box. Open `claude_desktop_config.json` in a text editor; if it already has content, merge, don't replace: add only the "ghl": {...} entry inside "mcpServers". Unsure? `cli install --dry-run` previews, `--print-only` prints the finished file.

- 3 **Create your GHL API key.** In the **sub-account** you want Claude working on: **Settings** → **Integrations** → **Private Integrations** → **Create**. Click **Select all** on scopes. Copy the **whole** key. It starts with `pit-` and it's longer than the box it sits in. (See the box at the top.)
- 4 **Activate, then check it.** Tell Claude: Run `setup_ghl_mcp`. Give it your email, the license key, the `pit-` key, and your Location ID (same Settings area, also in your welcome email). Restart Claude once more when it says setup is complete.

CHECKPOINT Ask Claude: "List my pipelines." Real pipelines back = Milestone 1 done. That's the free read-only tier working.

MILESTONE 2 OF 2 Unlock the Workflow Builder

~2 min, one time

Builds and publishes workflows, funnels and forms from one sentence. Uses GHL's internal API, so it needs one credential capture from your browser. Every member does this. It's the feature you bought.

- 5 **Say: "Unlock the Workflow Builder."** A Chrome window opens on the GHL login page. It will say "**controlled by automated test software**" across the top. **That is expected.** That banner is the tool doing its job.
- 6 **Log into GHL in that window.** That's your whole job. The tool reads the credentials, validates them, and saves them. Restart Claude one last time. Builders are now live: workflows, funnels, forms, pipelines, and the pre-deploy validator.
No window opened, or a locked-down work machine? Ask Claude to Run `auto_capture_firebase_script` instead: the console-paste path. If Chrome blocks the paste, type `allow` pasting in the console and try again.

YOUR FIRST WIN Paste this into Claude, then open GHL → Automation and look at it.

```
Build and publish a workflow called "New Lead Welcome" that sends an SMS  
"Hi {{contact.first_name}}, thanks for reaching out!", waits 1 day, sends a  
follow-up email, then adds the tag "nurtured". Summarize what you built.
```

Prefer to start read-only? Ask Claude "**Audit my workflows**" instead: two minutes, and it finds the dead pipeline and field references GHL never warns you about. Most people find something on the first run.

Stuck anywhere? Reply to any email from us with a **screenshot of where it stopped**. Don't write an essay, just show the screen. You'll get back a short recording for your exact situation, usually the same day. We're across many time zones, so we'd rather fix it while you sleep. Or email support@ghlcommand.com.