Personal AI Agent Setup Guide

OpenCode + Cloudflare Workers AI + AI Gateway + Google Workspace MCP

Free plan • sean.connellan@gmail.com

Cloudflare

Architecture

How It All Connects

YOUR LAPTOP

OpenCode — Terminal AI client

Connects to MCP servers for tools, sends LLM requests through AI Gateway

LOCAL MCP

Google Workspace — Gmail, Calendar, Drive, Docs, Sheets, Slides, Chat

Runs locally, OAuth credentials stay on your machine

CLOUDFLARE

AI Gateway — Caching, rate limiting, logging, DLP

Sits between OpenCode and Workers AI, saves neurons

WORKERS AI

Free LLM Models — Llama, Qwen, Granite, Gemma

10,000 neurons/day on free plan, OpenAI-compatible API

Data Flow

1. You type a prompt in **OpenCode**
2. OpenCode sends the request through **AI Gateway** (cached? logged? rate-limited?)
3. AI Gateway forwards to **Workers AI** for inference
4. If the AI needs a tool, OpenCode calls the **Google Workspace MCP** server
5. MCP server uses **OAuth** to call Google APIs, returns results to the AI

What is OpenCode?

A **terminal-based AI assistant** that orchestrates everything:

- Chat interface for LLM interactions
- Connects to any LLM provider (not locked to one)
- Discovers and calls MCP server tools
- Loads skills for specialised workflows
- Handles file editing, bash, web fetch
- Permission system (allow/ask/deny)

Install

```
# macOS (Homebrew)
brew install anomalyco/tap/opencode

# Or via curl
curl -fsSL https://opencode.ai/install | bash
```

Verify

```
opencode --version
opencode            # launch TUI
```

What is MCP?

Model Context Protocol — a standardised way for AI models to discover and invoke external tools. Think of it as "USB for AI".

LOCAL MCP

stdio transport

Runs on your laptop as a child process. Google Workspace MCP uses this — your OAuth credentials never leave your machine.

REMOTE MCP

Streamable HTTP transport

Runs on Cloudflare Workers. Accessible from anywhere, can be shared, scales to zero. Optional for personal use.

AI Gateway — Your AI Control Plane

Caching
Identical prompts served from cache — zero neurons consumed

Rate Limiting
Prevent runaway usage from burning your free allocation

Analytics
Track every request — tokens, latency, cost, errors

DLP
Redact sensitive data before it reaches the LLM

Logging
100K logs stored free — audit what your AI is doing

Model Routing
Switch between models/providers without changing client config

Cloudflare

Workers AI — Free LLM Models

10,000 neurons/day on the free plan. Serverless GPU inference on Cloudflare's global network.

Model	Size	Neurons/M input	Best for
`llama-3.1-8b-instruct-fp8-fast`	8B	4,095	General purpose (recommended)
`qwen3-30b-a3b-fp8`	30B	4,577	Reasoning tasks
`llama-3.3-70b-instruct-fp8-fast`	70B	26,668	Complex tasks
`granite-4.0-h-micro`	Small	1,542	Quick tasks
`gemma-4-26b-a4b-it`	26B	~4,500	General purpose

8B model: ~2.4M input tokens/day free • 70B model: ~375K input tokens/day free

Cloudflare

Setup Steps

Cloudflare

Step 1 — Cloudflare Account

CREATE ACCOUNT

1. Sign up at `dash.cloudflare.com/sign-up`
2. Use sean.connellan@gmail.com
3. Free plan — no credit card

FIND ACCOUNT ID

Dashboard → Workers & Pages → Account ID in sidebar

CREATE API TOKEN

1. Go to `dash.cloudflare.com/profile/api-tokens`
2. Use "Edit Cloudflare Workers" template
3. Add: Workers Scripts Edit, Workers AI Edit, Account Settings Read
4. Copy token — shown only once

~/.zshrc

```
export CLOUDFLARE_ACCOUNT_ID="your-id"
export CLOUDFLARE_API_KEY="your-token"
```

Cloudflare

Step 2 — AI Gateway

CREATE GATEWAY

1. Dashboard → AI → AI Gateway
2. Click **Create Gateway**
3. Name: `personal-ai-gateway`

Your gateway URL:

```
https://gateway.ai.cloudflare.com/v1/
  {ACCOUNT_ID}/personal-ai-gateway/
  workers-ai/v1
```

RECOMMENDED SETTINGS

Setting	Value
Caching	Enabled
Rate Limiting	50 req/min
Logging	Enabled
DLP	Optional

Free: 10 gateways, 100K logs, core features included

Step 3 — Google Cloud OAuth

GOOGLE CLOUD CONSOLE

1. Create project at `console.cloud.google.com`
2. Enable APIs: Gmail, Calendar, Drive, Docs, Sheets, Slides, Chat, People, Tasks
3. OAuth consent screen → **External** (required for @gmail.com)
4. Add sean.connellan@gmail.com as test user

CREATE CREDENTIALS

1. APIs & Services → Credentials → OAuth client ID
2. Type: **Desktop Application**
3. Download JSON → save to `~/.google-mcp/credentials.json`

> "This app isn't verified" warning is **expected** in Testing mode. Click Advanced → Go to app.

Step 4 — Google Workspace MCP Server

Install

```
git clone https://github.com/gemini-cli-extensions/workspace.git \
  ~/projects/google-workspace-mcp
cd ~/projects/google-workspace-mcp
npm install && npm run build
```

FIRST AUTH

Run `node workspace-server/dist/index.js --debug` — browser OAuth opens automatically

AUTH MANAGEMENT

`node scripts/auth-utils.js status/clear/expire`

Provides 50+ tools: Gmail, Calendar, Drive, Docs, Sheets, Slides, Chat — all via local stdio transport

Cloudflare

OpenCode Config

OpenCode — LLM Provider Config

~/.config/opencode/opencode.jsonc

```
{
  "model": "cf-gateway/@cf/meta/llama-3.3-70b-instruct-fp8-fast",
  "small_model": "cf-gateway/@cf/meta/llama-3.1-8b-instruct-fp8-fast",
  "provider": {
    "cf-gateway": {
      "npm": "@ai-sdk/openai-compatible",
      "name": "Cloudflare AI Gateway",
      "options": {
        "baseURL": "https://gateway.ai.cloudflare.com/v1/{ACCOUNT_ID}/{GATEWAY_ID}/workers-ai/v1",
        "apiKey": "{env:CLOUDFLARE_API_KEY}"
      },
      "models": {
        "@cf/meta/llama-3.3-70b-instruct-fp8-fast": { "name": "Llama 3.3 70B" },
        "@cf/meta/llama-3.1-8b-instruct-fp8-fast": { "name": "Llama 3.1 8B" }
      }
    }
  }
}
```

OpenCode — MCP Servers & Permissions

~/.config/opencode/opencode.jsonc (continued)

```
"mcp": {
  "google-workspace-local": {
    "type": "local",
    "command": ["sh", "-c",
      "cd /Users/sean/projects/google-workspace-mcp && node scripts/start.js"]
  },
  "sean-mcp-server": {
    "type": "remote",
    "url": "https://sean-mcp-server.<subdomain>.workers.dev/mcp"
  }
},
"permission": {
  "edit": "allow",
  "webfetch": "allow",
  "bash": {
    "*": "allow",
    "rm *": "ask",
    "rm -rf ~*": "deny",
    "sudo *": "ask"
```

Cloudflare

Skills — Teach the AI Specialist Workflows

SKILL.md files in
`~/.config/opencode/skills/<name>/`

Loaded dynamically when a task matches their description. Turn a general-purpose AI into a specialist.

EXAMPLE SKILLS

- **email-summariser** — Prioritised email digest
- **daily-briefing** — Calendar + email + tasks
- **call-notes-summary** — Structured call notes

email-summariser/SKILL.md

```
---
name: email-summariser
description: Summarises unread emails
---

## Workflow
1. gmail_search: is:unread newer_than:1d
2. gmail_get: fetch each email
3. Categorise: URGENT / ACTION / FYI
4. Present structured summary
```

Cloudflare

10,000 Neurons/Day Is Plenty

~2.4M input tokens with Llama 3.1 8B • ~375K with Llama 3.3 70B

Hit the limit? Workers Paid is $5/month for unlimited neurons at $0.011/1K

Full guide: `github.com/gemini-cli-extensions/workspace` • `opencode.ai` • `developers.cloudflare.com/agents`