A LANDRY INDUSTRIES PUBLICATION

The Viblogger System

How a free Blogger blog, a Namecheap domain, Cloudflare and Claude become a publishing network one person can run

THE BOOK · TECH SPECS · PROMPT LIBRARY

Marie-Soleil Seshat Landry

Founder, CEO and Spymaster · Marie Landry Spy Shop

EDITION OF OCTOBER 29, 2026

LandryIndustries.ca · MarieLandrySpyShop.com

About this book

The Viblogger System shows how a free Blogger blog becomes a multi-site publishing network that one person runs by conversation with Claude.

Every figure in it was checked against the live site, the Cloudflare account or a linked source on October 29, 2026.

By Marie-Soleil Seshat Landry, OSINT analyst and researcher · ORCID 0009-0008-5027-3337 · Marie Landry Spy Shop, a Landry Industries publication

> **AI NOTICE**
>
> This book was drafted with Claude, an AI assistant, and may contain errors. Check important facts against the linked primary sources before you rely on them.

HOW TO READ IT

Part I, The Book, explains why the stack works, how to build it and how to run it. **Part II, Tech Specs**, is the reference: inventory, DNS, data stores, limits and code patterns. **Part III, Prompt Library**, holds the starter prompt that clones the whole infrastructure, the 100 prompts behind marielandryspyshop.com, and 100 more you can adapt.

DEFINITION

Viblogger (vibe blogger): a publisher who builds, edits and runs a website by describing outcomes to an AI operator, which reads the live site, changes it, deploys and verifies, while the publisher keeps every decision.

© 2026 Marie-Soleil Seshat Landry. Marie Landry Spy Shop is a division of Landry Industries, Moncton, New Brunswick, Canada.

LandryIndustries.ca · MarieLandrySpyShop.com · ceo@marielandryspyshop.com

Marie Landry Spy Shop is an independent OSINT research publication. It is not a private investigation agency, police force or government intelligence service.

Screenshots of marielandryspyshop.com captured October 29, 2026. Product names belong to their owners.

Contents

PART I · THE BOOK

	Manifesto: Blogger is a backbone, not a relic	5
1	What a Viblogger is	6
2	The stack at a glance	7
3	Case study: marielandryspyshop.com	9
4	Foundation: Blogger, Namecheap and Cloudflare	13
5	The subdomain empire	15
6	Wiring Claude: Project, connectors and the Blogger API	17
7	Teaching your Viblogger to remember	19
8	The public file repository	21
9	The operating rhythm	23
10	Guardrails	25
11	The road ahead	27
	Sources	28

PART II · TECH SPECS

1	Inventory	30
2	DNS and routing	31
3	Data stores and AI services	33
4	Platform limits that shape the design	34
5	Pattern: the header Worker in front of Blogger	34
6	Pattern: the Blogger API bridge Worker	35
7	Deploy, verify and roll back	37
8	Security and standards baseline	37

PART III · PROMPT LIBRARY

1	The starter prompt: clone the infrastructure	40
2	100 prompts behind marielandryspyshop.com	42
3	100 more prompts to adapt	46

PART I

The Book

Why Blogger works as a backbone, how the stack fits together, how to build it, and how to run it every day.

MANIFESTO

Blogger is a backbone, not a relic

Our thesis: Blogger is a free, Google-hosted content database with a public API, and an edge network plus an AI operator turn it into a modern web platform. mari-elandryspyshop.com is the evidence.

On October 29, 2026 the site's header reports 3,129 blog posts, 2,903 library files, 31 subdomains, 117 AI assistants and 205 intel-feed sources, all run by one person.

The network grew fast once the operator arrived. The first Cloudflare Worker was created on June 18, 2026; by October 28 the blog and 31 subdomains shared one header, one security policy and one corrections system (Site Log). Twenty-five Workers carry it today, on Cloudflare's free website plan.

Why Blogger holds up as the core:

- **It costs nothing to host.** Google runs the servers and issues the HTTPS certificate for your custom domain (Blogger Help).
- **It is programmable.** The Blogger API v3 can list, search, insert, patch, publish and revert posts and pages (posts.insert reference).
- **It publishes feeds.** Every blog exposes post feeds that Workers read to build related-post blocks and widgets.
- **It stays independent of the front end.** A Worker in front of Blogger can restyle and extend every page; removing the Worker restores the plain blog instantly (Site Log, October 3, 2026).
- **Its gaps are fillable.** What Blogger lacks (subdomains, databases, search, AI) lives at the edge, beside it.

What Blogger does not do, stated plainly: the API has no endpoints for themes or Layout widgets, so those stay manual; Blogger escapes some HTML in ways that break custom code (Site Log, July 12, 2026); and its built-in look is dated. The Viblogger stack works around all three.

We believe this pairing can become a standard way to blog: free storage you own, an edge you program, and an operator you direct in plain words. The rest of this book shows how to build it and how to keep it honest.

CHAPTER 1

What a Viblogger is

A Viblogger directs outcomes; Claude does the operating.

You say "review my site and fix what's broken"; Claude reads the live pages and the current Worker code, proposes a short plan, makes the smallest change that works, deploys it, checks the result at desktop and phone widths, and logs it.

The split of roles is the whole method:

Role	Who	Owns
Editor-in-chief	You	Goals, priorities, approvals, anything irreversible, ethics calls
Operator	Claude in a Project	Reading, research, code edits, deploys, verification, logs
Platform	Blogger + Cloudflare	Storage, publishing, edge code, data, caching, security
Rulebook	Your governing documents	What may never be done, whatever the request

> **A REAL SESSION, OCTOBER 28 TO 29, 2026**
>
> Marie opened a chat with two words: "Lets work." Claude pulled the live homepage, the affiliate directory and the dossiers hub, and returned six findings ranked by urgency, among them affiliate links on the investigations hub, an empty NordVPN affiliate ID and three numbers that contradicted each other. Marie replied "Fix everything." Claude edited 18 Workers, recorded each previous version for rollback, verified the live pages, logged two corrections and handed back one manual Blogger step with the exact text to paste.

That exchange shows the four habits that make the system safe:

1. **Read before writing.** Every change starts from the live page and the current code, never from memory.
2. **Propose, then act.** Findings arrive as a numbered list you can accept whole ("fix everything") or in part ("do 1 and 3").
3. **Verify on the live site.** A deploy is not done until a fresh render confirms it.
4. **Leave a paper trail.** Corrections are dated on the page and on the Corrections page; notable changes go in the Site Log.

CHAPTER 2

The stack at a glance

Six layers, each doing one job.

A visitor's request passes through DNS to Cloudflare's edge, where a Worker either wraps a Blogger page or serves a subdomain on its own. Claude reaches every layer through connectors, and reaches Blogger itself through a Worker that holds the Google credentials.

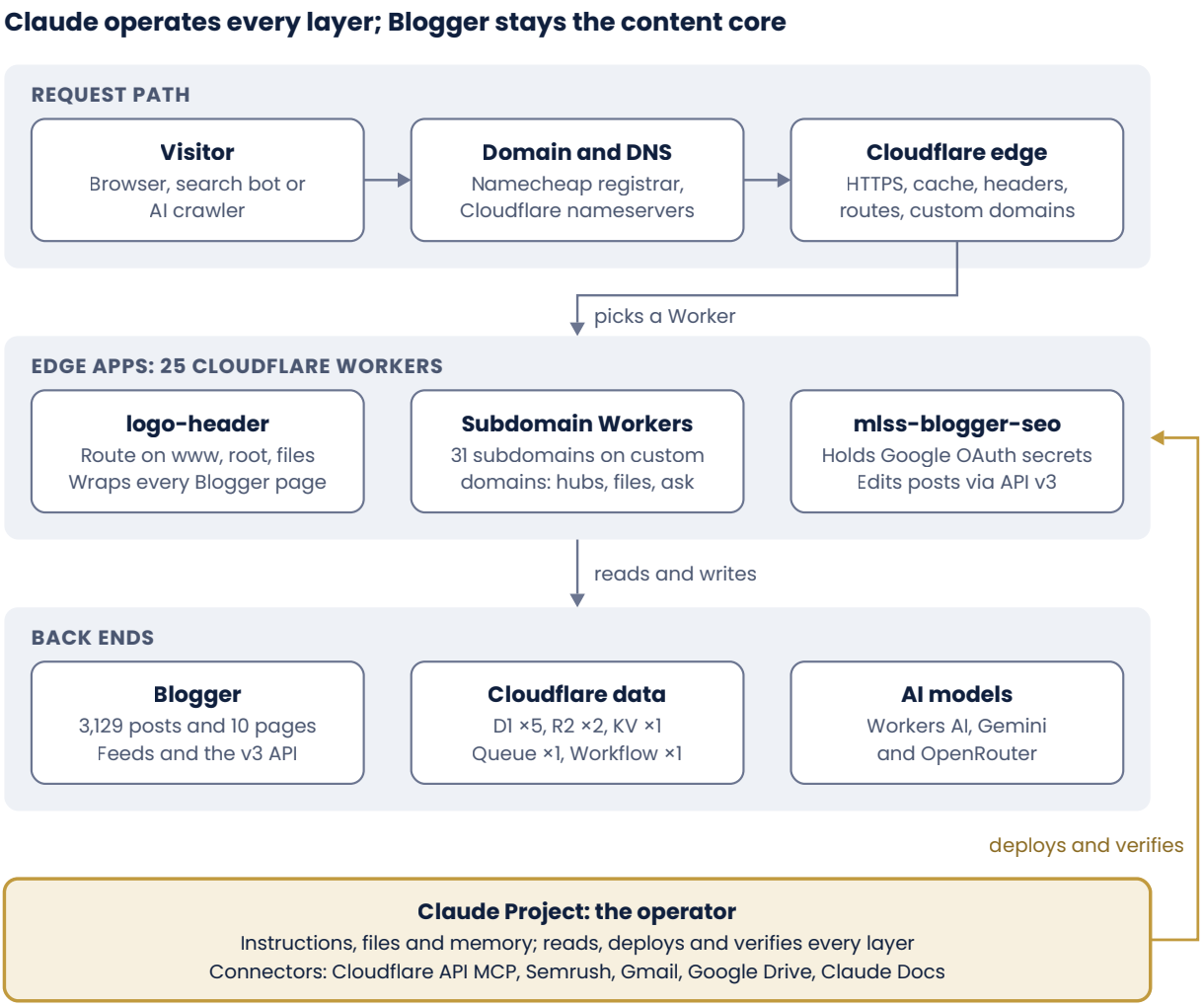


Figure 1. How the stack fits together: three tiers, nine components, one operator. Requests flow down from the visitor; Claude works from below, deploying and checking every layer through its connectors.

Layer	Service	Job on marielandryspyshop.com
Domain	Namecheap	Registrar; nameservers point to Cloudflare
DNS and edge	Cloudflare, free website plan	DNS, HTTPS, caching, security headers, routing
Content core	Blogger	3,129 posts and 10 pages, with feeds and the v3 API
Edge apps	Cloudflare Workers (25)	Header and nav on every page, 31 subdomains, the library, search, AI help desk
Data	D1 (5), R2 (2), KV (1), Queues (1), Workflows (1)	Library metadata, news, search index, corrections, backups, upload events, edit jobs
AI	Workers AI, Gemini, OpenRouter, Claude	Help desk answers, library summaries, tagging, blog edits; Claude operates the whole stack

Counts are from the Cloudflare account and the live header on October 29, 2026. Part II, Tech Specs, lists every Worker, hostname, binding and record.

CHAPTER 3

Case study: marielandryspyshop.com

One Blogger blog wearing a Worker-built header, and five subdomains that are pure Workers.

The screenshots below were captured on October 29, 2026 with Cloudflare Browser Run, the same tool Claude uses to verify every deploy.

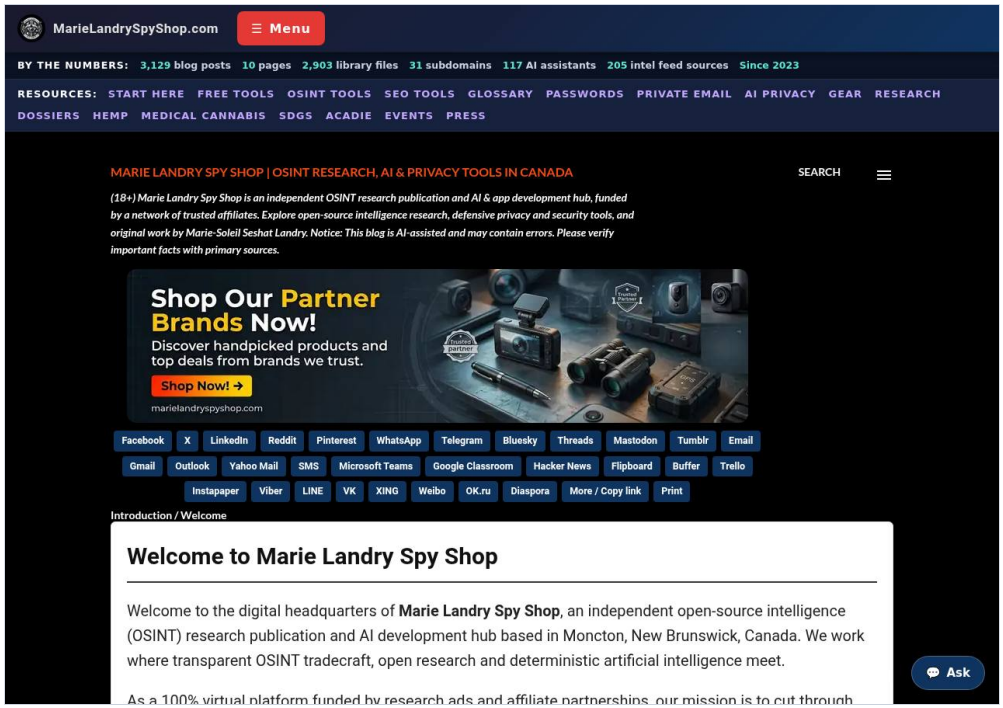


Figure 2. The Blogger homepage at 1280 px. Everything above the blog title (logo bar, statistics row, Resources row) is injected by the logo-header Worker; the Ask button at bottom right opens the AI help desk.

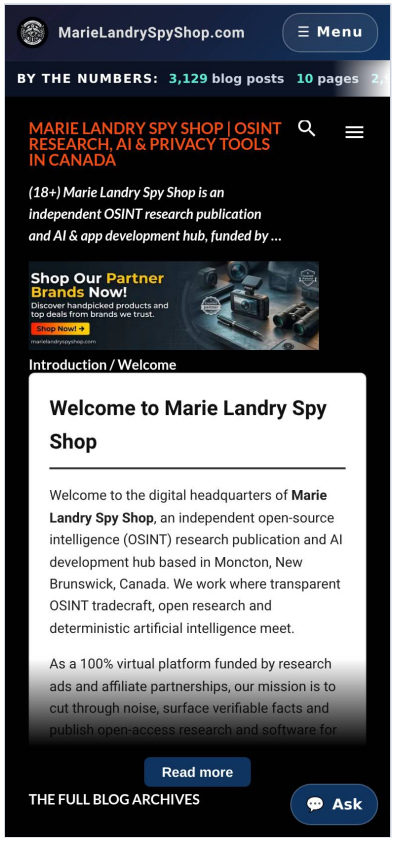


Figure 3. The same page at 390 px. On phones the header collapses to a Menu button and each row scrolls sideways (Site Log, October 21 and 24, 2026).

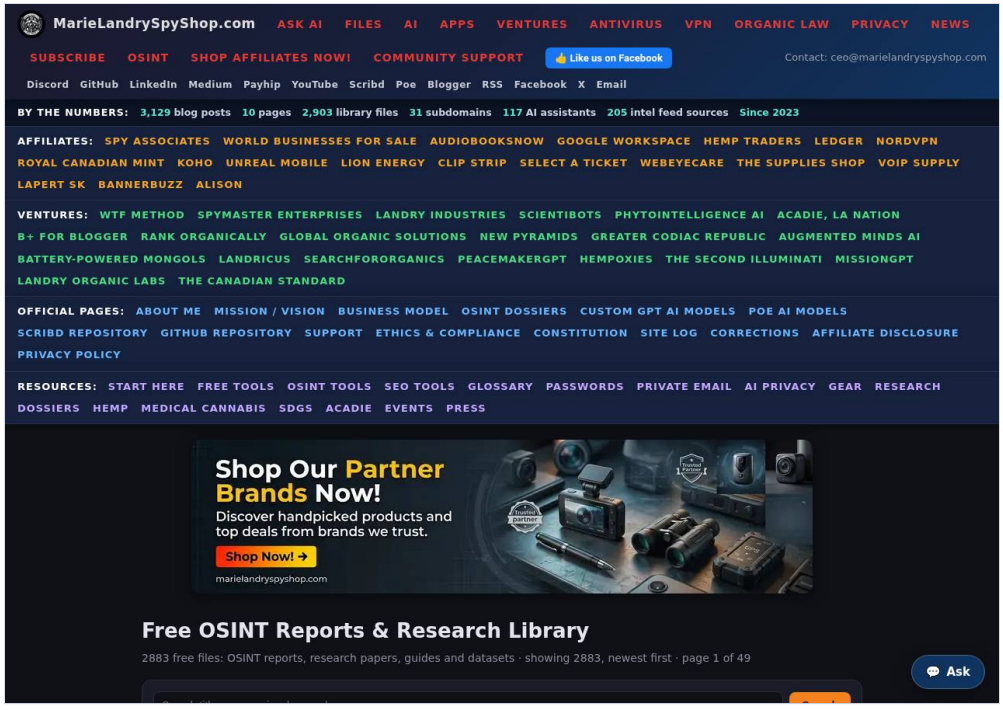


Figure 4. files.marielandryspyshop.com: a Worker serving 2,883 listed files from R2, with AI summaries and tags (Chapter 8).

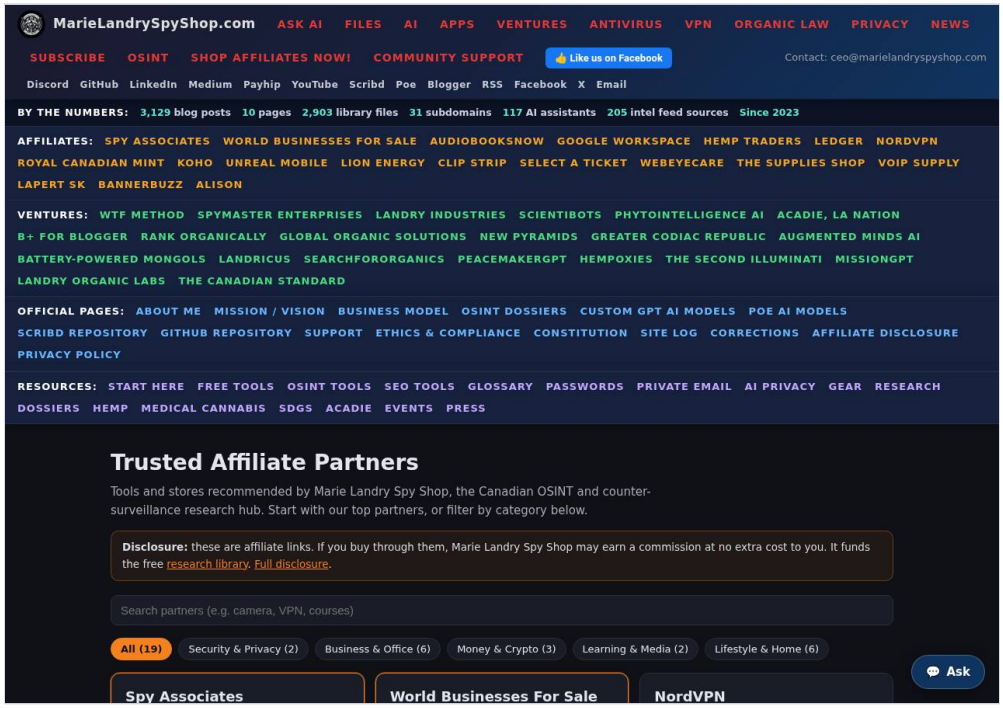


Figure 5. affiliates.marielandryspyshop.com: 19 partners, with the affiliate disclosure placed before the first link.

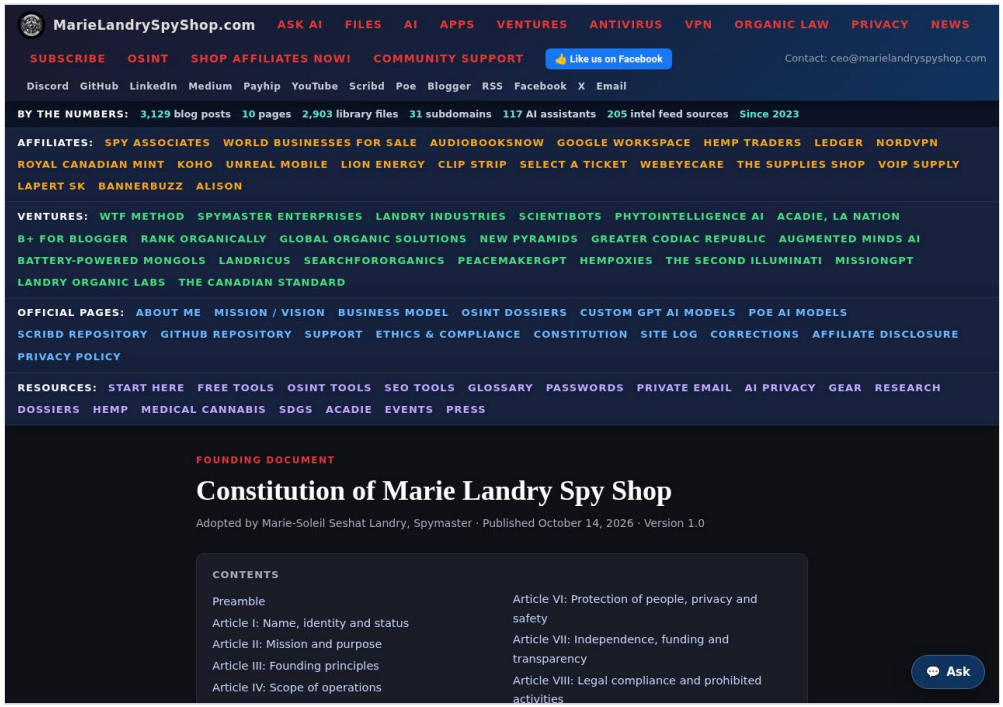


Figure 6. constitution.marielandryspyshop.com: the rulebook the operator must follow, published as its own subdomain.

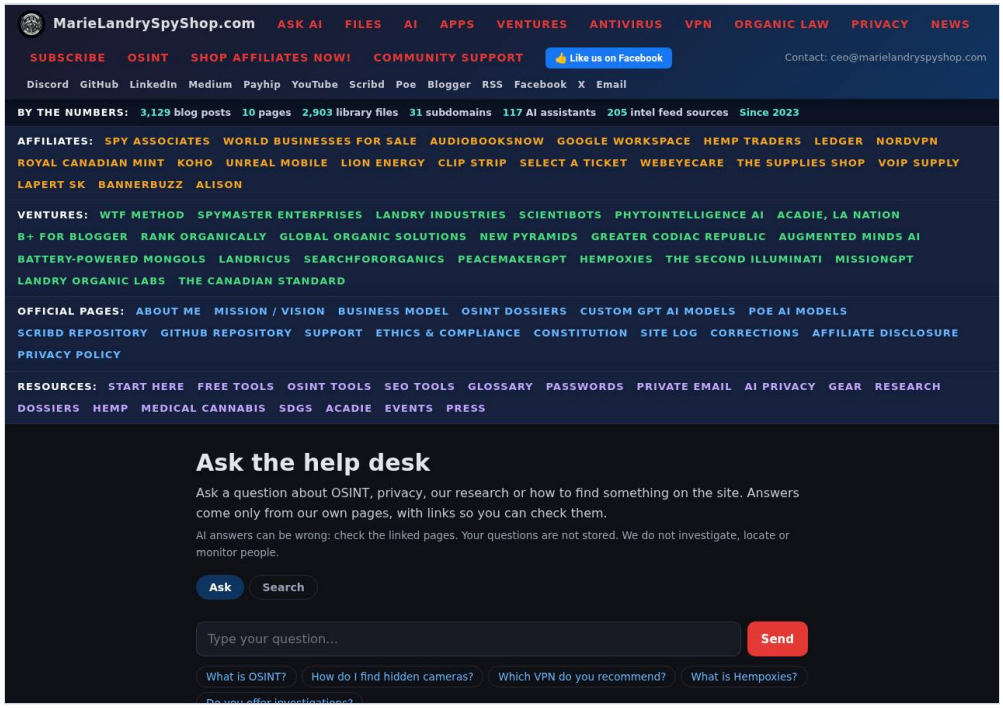


Figure 7. ask.marielandryspyshop.com: answers come only from the site's own pages, on Workers AI.

Milestones, newest first (from the Site Log):

Date	Milestone
2026-10-28	Corrections page, AI help desk, SEO hub; one header and one security policy across the blog and 31 subdomains
2026-10-27	Private Blogger API editing system: 91 high-value posts tightened, each backed up
2026-10-24	Edge caching cuts repeat blog loads from about 250 to 400 ms to about 20 to 40 ms
2026-10-22	Fifteen subdomains launched in one day, including Start Here, dossiers, press, hemp, SDGs, Acadie and events
2026-10-21	OSINT hub, antivirus and VPN rankings, privacy guide, News Matrix, Site Log
2026-10-14	Constitution published; mSpy removed as a partner
2026-10-13	Affiliates, AI and apps subdomains; paid investigation offers removed
2026-10-03	Custom header Worker placed in front of Blogger
2026-09-24	Research library with AI summaries goes live
2026-06-18	First Cloudflare Worker
2023-02-10	Blog launches on Blogger

CHAPTER 4

Foundation: Blogger, Namecheap and Cloudflare

The foundation takes an afternoon of clicks and up to two days of DNS waiting. Do it once, by hand; after this, Claude does the building.

1. **Start the blog.** Create a Blogger blog and publish a first post, so the blog has content to serve.
2. **Register the domain at Namecheap.** Any registrar works; the steps below match Namecheap's dashboard.
3. **Add the domain to Cloudflare** on the free plan. Cloudflare gives you two nameservers (marielandryspyshop.com uses charles.ns.cloudflare.com and demi.ns.cloudflare.com).
4. **Point Namecheap at Cloudflare.** In Namecheap, open **Domain List**, click **Manage** next to the domain, set **Nameservers** to **Custom**, paste both Cloudflare nameservers and save with the check mark. Then click **Check nameservers** in Cloudflare and wait for **Active**. Namecheap says propagation can take 24 to 48 hours (Namecheap).
5. **Ask Blogger for its records.** In Blogger, go to **Settings › Publishing › Custom domain**, enter www.yourdomain.com and save. Blogger answers with two CNAME records: one for www pointing to ghs.google.com, and a security CNAME unique to your blog (Blogger Help).
6. **Create the records in Cloudflare DNS.** Add both CNAMEs, wait at least an hour, then save the custom domain in Blogger again.
7. **Catch the bare domain.** Add four A records for the root (216.239.32.21, 216.239.34.21, 216.239.36.21, 216.239.38.21) and turn on **Redirect domain** in Blogger's Publishing settings. If you use CAA records, allow letsencrypt.org or Blogger cannot issue its certificate.
8. **Put the first Worker in front.** Create a Worker that fetches each Blogger page and adds your header, then attach it as a **route** on www.yourdomain.com/* (a route runs in front of an existing origin; a custom domain makes the Worker the origin) (Cloudflare).

The resulting records on marielandryspyshop.com:

Type	Name	Target	Proxy
CNAME	www	ghs.google.com	Proxied
A (×4)	@	216.239.32.21, .34.21, .36.21, .38.21	Proxied
AAAA	each subdomain	100:: (placeholder)	Proxied
MX (×3)	@	Cloudflare Email Routing	DNS only
TXT	@, _dmarc, DKIM	Verification, SPF, DMARC, DKIM	DNS only

> **INTELLIGENCE GAP**
>
> Blogger's help page does not say whether its records may run behind Cloudflare's proxy. On marielandryspyshop.com they do, and HTTPS works today. If certificate setup stalls on a new blog, switch the www and root records to DNS only until Blogger confirms, then turn the proxy back on.

Hand-off line. Once the custom domain loads over HTTPS, open your Claude Project and say: "Read my site and my Cloudflare account, then tell me what you see."

CHAPTER 5

The subdomain empire

Every subdomain is a Worker attached as a custom domain, so the Worker is the origin and Cloudflare manages the DNS record and certificate.

One Worker can serve many hostnames: 31 subdomains run on far fewer Workers (Cloudflare).

Worker	Serves	Pattern
logo-header	www, the bare domain, files (as routes)	Wraps pages from another origin
hubs-site	14 hubs: dossiers, guide, research, press, gear, passwords, email, ai-tools, hemp, sdg, acadie, events, 420, seo	One codebase, a page per hostname
mlss-hubs	glossary, tools	Shared reference pages
alias-redirects	docs, shop	Short addresses that forward
r2-file-index	files	Library app over R2, D1 and KV
One Worker each	affiliates, ai, apps, ask, antivirus, vpn, constitution, corrections, intel, log, news, organiclaw, osint, privacy, ventures	Standalone sites

The files hostname shows a useful trick: it is a custom domain on r2-file-index *and* a route on logo-header. A route runs first and can fetch the custom-domain Worker behind it, so two Workers chain on one request (Cloudflare).

The shared-block contract. The header bar, statistics row, Affiliates row, Ventures row, Official Pages row, Resources row, legal notice and footer appear on every site. Each Worker carries its own copy, so a change to one block means editing every Worker that holds it, then verifying each. On October 29, 2026 one nav fix touched 18 Workers. Tell Claude to list the Workers it touched every time.

Living inside the free tier. The whole network runs on Cloudflare's free plans. The limits that matter (Cloudflare):

Limit (Workers Free)	Value
Requests	100,000 per day, reset at midnight UTC
CPU time	10 ms per request
Subrequests	50 external per request
Workers per account	100
Cron Triggers per account	5

The network hit its free daily D1 database limits on October 22, 2026 and News Matrix paused until midnight UTC. The fix was a shared cache refreshed every 30 minutes and calmer schedules (Site Log). Edge caching does the rest: blog pages are cached for 5 to 15 minutes and library pages for 10 to 60 minutes, with a week-long backup copy if Blogger is down.

CHAPTER 6

Wiring Claude: Project, connectors and the Blogger API

Run your Viblogger inside a Claude Project, not a loose chat.

A Project keeps its own chats, its own instructions and its own files, so every new conversation starts already knowing your site, your rules and your brand (Claude Help). Projects are available on every plan; the free plan allows five.

Set up the Project in four moves:

1. **Name and describe it.** Marie's is "MarieLandrySpyShop.com / Viblogger Assistant", with a description of the business model and revenue target.
2. **Write the instructions.** Role, what the site is and is not, the non-negotiable rules, the site map, the connectors, the editing loop, writing and SEO standards, voice and visual style. The starter prompt in Part III is a ready template.
3. **Add the files.** Governing documents first (Marie's Project holds the Constitution as a PDF), then the knowledge pack in Chapter 7.
4. **Connect the tools.** The connectors below turn Claude from an adviser into an operator.

Connector	What Claude does with it
Cloudflare (API MCP server)	Reads and deploys Workers, edits DNS, manages D1, R2, KV and Queues, renders live pages with Browser Run
Semrush	Keyword, competitor, backlink and site-audit data behind SEO choices
Gmail	Drafts partner outreach for your review; sends only when told
Google Drive	Research files and the public library folders
Claude Docs	Long drafts, governing documents and books like this one

The Cloudflare server reaches the whole Cloudflare API through two tools, search and execute (plus a docs search), in about 1,000 tokens: Claude writes short JavaScript that finds the right endpoint and calls it inside a sandbox (Cloudflare). That is why one chat can edit 18 Workers. MCP itself is the open standard Anthropic introduced on November 25, 2024 for connecting AI tools to data and software (Anthropic).

Reaching Blogger through a Worker. There is no Blogger connector in the list, and none is needed. Marie's private mlss-blogger-seo Worker holds the Google OAuth client and tokens as encrypted secrets, asks for the Blogger scopes (blogger and blogger.readonly), and calls the Blogger API v3 on Claude's behalf. Claude never sees the Google credentials; it deploys and drives the Worker through Cloudflare. The Worker:

- checks a Cloudflare Access sign-in against a list of admin emails before it does anything;
- runs edit jobs as a Cloudflare Workflow, with a cron that checks the queue every minute;
- backs up each post to R2 before any change, and keeps a rollback endpoint;
- asks Gemini for careful edits and rejects any that change a link or image, drop a disclaimer, cut too much or add unsupported claims (Site Log, October 27, 2026);
- notifies Bing through IndexNow after each change.

The API can insert, patch, publish and revert posts and pages (Blogger API). It has no endpoints for themes or Layout widgets, so Claude still hands those back as manual steps, in order, with the exact text to paste.

CHAPTER 7

Teaching your Viblogger to remember

Claude remembers through three layers and checks a fourth, the live site; each fact belongs in exactly one of them.

Put rules in the instructions, reference material in Project files, and the small personal details you mention along the way in memory.

Layer	Holds	How it gets there
Project instructions	Rules, role, workflow, standards	You write them once and edit as rules change
Project files	Governing documents, brand kit, partner list, links, plans	You upload them; Claude reads them on demand
Memory	Preferences, people, ongoing work, decisions	Claude saves topics as you chat, or when you say "remember that…"
The live site	Current counts, partner list, page copy	Claude reads it before every change

Each Project keeps its own memory, alongside your account-wide memory. You can view and edit it under **Settings › Your Claude › Manage**, ask Claude to change or forget something, or isolate a Project by turning off **Use account memory** on its page. Memory skips sensitive topics unless you opt in, and never stores ID or financial account numbers (Claude Help).

The Viblogger knowledge pack. Upload these as Project files, one topic per file, and date each one:

File	Contents on marielandryspyshop.com
Governing documents	Constitution v1.0, Ethics and Operational Compliance Policy, Affiliate Disclosure, Privacy Policy
Brand kit	Navy #0F1F3D and gold #C19A3E; Cinzel, EB Garamond, Poppins and Lora; logo files; the standard byline
Affiliate register	All 19 partners with tracking links, category, disclosure wording and status (Spy Associates first)
Social and profile links	Facebook, X, LinkedIn, Discord, GitHub, Medium, Payhip, YouTube, Scribd, Poe, Blogger RSS, ORCID
Architecture map	Every Worker, hostname, binding and shared block (copy Part II, Tech Specs)
Business plan	Revenue model, targets, ventures and what the site will never sell
Workflow playbook	The editing loop, the correction procedure, the manual Blogger steps
Subjects and voice	Favourite topics, banned claims, house terms, Canadian spelling

Keep one source of truth. When a file and the live site disagree, the live site wins and the file gets updated. Partner counts and lists change; that is why the instructions tell Claude to check affiliates.marielandryspyshop.com before naming or counting partners.

Never put API tokens, passwords or affiliate account logins in Project files or memory. Secrets belong in Worker secrets, where code can use them and nobody can read them back.

CHAPTER 8

The public file repository

files.marielandryspyshop.com turns a storage bucket into a searchable library where every document gets its own search-optimized page, written and tagged by AI with no manual step.

You upload a file; the pipeline gives it a title, summary, keywords, FAQ, related files, share buttons and embed code (Site Log, September 24, 2026).

How the pipeline runs:

1. **Upload.** A file lands in the R2 bucket named spyshop.
2. **Event.** An R2 event notification for object-create puts a message on the spyshop-uploads queue (Cloudflare).
3. **Autotag.** The r2-autotagger Worker consumes the queue, skips summary and metadata files, and asks a language model (Llama 3.2 3B Instruct through OpenRouter) for a title, summary and keywords, which it writes to the file_metadata table in D1.
4. **Publish.** The r2-file-index Worker lists the bucket, joins the D1 metadata, and renders the library, a page per file, a sitemap and a statistics page that separates people from bots. Gemini writes the longer landing-page text for each document.
5. **Cache.** Library pages are cached at the edge for 10 to 60 minutes, so most visits never touch the database; view counts still update (Site Log, October 22, 2026).

Self-repair built in. The index Worker hides duplicates and points each one at its main copy, keeps a hidden quarantine area for files under review, and turns off ads and search indexing on adult-themed documents. On October 28, 2026, 276 duplicate files were moved out of the library and every old address now redirects to the copy that stays.

Build your own in five prompts:

1. "Create an R2 bucket for my public library and a D1 table for title, summary, keywords and views."
2. "Create a queue and an R2 event notification so every new upload lands on it."
3. "Write a consumer Worker that tags each new file with an AI summary and keywords, and stores them in D1."
4. "Build a library Worker on files.mydomain.com with search, filters, a page per file, a sitemap and edge caching, using my site's header."
5. "Find duplicates and files with weak summaries, show me the list, then fix what I approve."

Keep private material out of a public bucket. The library withdrew personal email printouts on October 24, 2026; a quarantine prefix and a review step before upload prevent that class of mistake.

CHAPTER 9

The operating rhythm

Open a fresh chat in the Project for each piece of work, and start with a question rather than an instruction.

"What should I work on next?" lets Claude read the live site, the Site Log and its memory, and come back with a ranked list you can accept in one line.

The five conversations that run the site:

Conversation	Opening line	What comes back
Triage	"What should I work on next?"	Three to six findings, ranked, each with the fix and its cost
Red team	"Red team my blog for weaknesses and strengthen it."	Broken links, contradictions, legal and trust risks, security gaps, with exact locations
Build	"Create a subpage about X" or "Add share buttons to the homepage."	A one-line plan, the change, a live check, the manual steps
Write	"Write a relevant blog post about X."	A sourced draft with title, meta description, slug and internal links
Clean up	"Find low-quality content and deslopify it."	A list of weak posts, proposed rewrites, then edits you approve

Accepting suggestions. Answer a ranked list with "fix everything", "do 1 and 3" or "skip 2, it's intentional". Claude does only what you approved, then reports what changed, where it is live, what it verified and what is left for you in Blogger.

Auto-fixing and debugging. Paste an error or describe a symptom ("the Ask button vanished on phones"). Claude renders the page, reads the Worker, finds the cause, makes the smallest fix and re-renders to prove it. When a fix is a factual correction, it goes on the page and on the Corrections page with the date.

A sustainable cadence:

When	Prompt
Every session	"What changed since last time, and what should I do next?"
Weekly	"Run a full red-team pass and a broken-link check across every subdomain."
Weekly	"Pull Semrush positions and tell me which three pages to improve."
Monthly	"Audit counts, partner lists and legal notices for contradictions."
Monthly	"Review the Site Log and suggest what to build next."

Keep chats short and focused. A new chat per task keeps context clean; memory and the Project files carry what matters forward.

CHAPTER 10

Guardrails

An operator that can deploy to 31 subdomains in one reply needs hard limits, written down where it reads them every time.

Speed is the stack's strength; the guardrails are what make that speed safe to use.

Rules the operator never bends:

- **The rulebook wins.** On marielandryspyshop.com the Constitution prevails over every request. When a request conflicts, Claude names the article and offers a compliant version, for example a counter-surveillance guide instead of a monitoring-app review (Article VII.6).
- **Ask before anything irreversible:** deleting a Worker, page, subdomain or partner, and any DNS change.
- **Secrets stay secret.** API tokens and OAuth credentials live only in Worker secrets, never in page code, Project files or memory.
- **Back up, then change.** The Blogger editing Worker copies each post to R2 before touching it; Worker deploys record the previous version so any change can be rolled back.
- **Disclose.** AI-assisted pages carry an error notice; affiliate links carry a plain-words disclosure next to them; investigative reports carry no ads or affiliate links.
- **Never fabricate.** No invented figures, quotes or links. What cannot be verified is labelled as unverified or logged as an Intelligence Gap.
- **Respect the meters.** Workers Free allows 100,000 requests a day and 10 ms of CPU per request; Browser Run on the free plan allows 10 minutes of browser time a day and one quick action every 10 seconds (Cloudflare). Claude spaces its checks to match.

Red-team findings from writing this book (October 29, 2026). Capturing the screenshots and reading the Workers surfaced five items for Marie's decision:

1. **Stale figure in the Site Log.** The October 28 entry "Homepage widget and counts corrected" on log.marielandryspyshop.com still says the AI hub lists more than 100 of "over 250 AI models". The October 29 correction removed "over 250" as unverifiable. Fix: edit that entry in the log-site Worker.
2. **Three library counts.** The header says 2,903 library files, the library page heading says 2,883, and the September 24 Site Log entry says "Today it holds 3,190 files". The first two likely differ by duplicates the library hides. Fix: say which number counts what, and date the log figure.
3. **Partner banner on the library.** files.marielandryspyshop.com shows the "Shop Our Partner Brands Now!" banner directly above "Free OSINT Reports & Research Library". The same reasoning that removed the banner from the dossiers hub on October 29 (Articles IV.1 and VII.2) may apply here.

4. **Banner imagery.** The partner banner shows a pen, compact cameras and a GPS-style unit. Images like these can read as covert-recording gear, which the defensive-only rule avoids promoting. Check the image against the products it advertises.
5. **Phone header.** At 390 px the statistics row runs past the right edge. It scrolls sideways by design; a fade or arrow would tell readers there is more.

CHAPTER 11

The road ahead

The Viblogger stack lowers the cost of a serious publication to a domain name and good judgment.

The parts are open: Blogger's API, Cloudflare's edge, and MCP, which Anthropic donated to the Linux Foundation's Agentic AI Foundation in December 2025 (Anthropic). Anyone can build on them, and no single vendor owns the method.

Next steps for marielandryspyshop.com:

- **One shared-block source.** Move the header, rows and footer into one shared module or KV entry, so a nav change is one edit instead of 18.
- **French editions** of the key pages: Start Here, the Constitution, privacy and the library home.
- **A Viblogger template** others can clone: the starter prompt in Part III, a sanitized Worker set and this book.
- **A paid Workers plan when the meters bind.** It lifts CPU time to 5 minutes per request and removes the daily request cap (Cloudflare).
- **Measured growth.** Tie each new hub to a Semrush keyword and track it monthly.

On its own, Blogger looks like a simple blog host. With an edge in front and an operator beside you, it becomes the quiet database under an empire, and the empire stays yours: your domain, your words, your rules.

REFERENCE

Sources

Pages and systems opened on October 29, 2026.

- Marie Landry Spy Shop homepage · https://www.marielandryspyshop.com/ (rendered with Cloudflare Browser Run)
- Site Log · https://log.marielandryspyshop.com/ (entries read from the log-site Worker)
- Trusted Affiliate Partners · https://affiliates.marielandryspyshop.com/
- Free OSINT Reports & Research Library · https://files.marielandryspyshop.com/
- Constitution of Marie Landry Spy Shop · https://constitution.marielandryspyshop.com/
- Ask the help desk · https://ask.marielandryspyshop.com/
- Marie Landry Spy Shop Cloudflare account: Workers, routes, custom domains, DNS records, D1, R2, KV, Queues and Workflows (read through the Cloudflare API MCP server)
- Blogger Help: Set up a custom domain · https://support.google.com/blogger/answer/1233387?hl=en
- Blogger API v3: posts.insert · https://developers.google.cn/blogger/docs/3.0/reference/posts/insert?hl=en
- Namecheap: How to set up DNS records for your domain in a Cloudflare account · https://www.namecheap.com/support/knowledgebase/article.aspx/9607/2210/how-to-set-up-dns-records-for-your-domain-in-a-cloudflare-account
- Cloudflare Workers: Limits · https://developers.cloudflare.com/workers/platform/limits/
- Cloudflare Workers: Custom Domains · https://developers.cloudflare.com/workers/configuration/routing/custom-domains/
- Cloudflare Workers: Routes · https://developers.cloudflare.com/workers/configuration/routing/routes/
- Cloudflare Browser Run: Limits · https://developers.cloudflare.com/browser-run/limits/
- Cloudflare: MCP servers for Cloudflare · https://developers.cloudflare.com/agents/model-context-protocol/cloudflare/servers-for-cloudflare/
- Cloudflare R2: Summarize PDF files on upload with event notifications · https://developers.cloudflare.com/r2/tutorials/summarize-pdf/
- Claude Help: What are projects? · https://support.claude.com/en/articles/9517075-what-are-projects
- Claude Help: Chat search and memory · https://support.claude.com/en/articles/11817273-use-claude-s-chat-search-and-memory-to-build-on-previous-context
- Anthropic: Introducing the Model Context Protocol · https://www.anthropic.com/news/model-context-protocol
- Anthropic: Donating MCP to the Agentic AI Foundation · https://www.anthropic.com/news/donating-the-model-context-protocol-and-establishing-of-the-agentic-ai-foundation

PART II

Tech Specs

The reference for the stack as deployed on marielandryspyshop.com, read from the live Cloudflare account on October 29, 2026.

SECTION 1

Inventory

25 Workers serve the bare domain, www and 34 hostnames (31 subdomains plus intel, docs and shop). Three Workers have no hostname.

Worker	Hostnames	Role
logo-header	Routes: marielandryspyshop.com/*, www/*, files/*	Wraps Blogger and the library with the shared header, rows and footer
hubs-site	dossiers, guide, research, press, gear, passwords, email, ai-tools, hemp, sdg, acadie, events, 420, seo	Topic hubs from one codebase
mlss-hubs	glossary, tools	Glossary and free tools
r2-file-index	files	Research library over R2, D1 and KV
r2-autotagger	none (queue consumer)	Tags new uploads with AI title, summary and keywords
ask-site	ask	AI help desk, full-text search, weekly digest
intel-feed	intel	IntelMatrix aggregator for the homepage feed
news-site	news	News Matrix headlines
corrections-site	corrections	Corrections register
log-site	log	Site Log
affiliates-site, ai-site, apps-site, antivirus-site, vpn-site, constitution-site, ventures-site, osint-site, privacy-site, organiclaw-site	one hostname each	Standalone sites
alias-redirects	docs, shop	Forward to the library and the affiliate directory
mlss-blogger-seo	workers.dev only, admin sign-in	Blogger API bridge and editing jobs
marie-landry-seo, googledrive, shrill-glitter-8f23	none	Unattached; review before any cleanup

Notable bindings:

Worker	Bindings	Schedule
mlss-blogger-seo	R2 BACKUPS; D1 DB and SEARCH; Workflow SEO_WF; secrets for Google OAuth, Gemini keys, session and token keys; Access audience and team domain; admin emails; IndexNow key; publish lock	Every minute
r2-file-index	R2 SPYSHOP_BUCKET; D1 FILE_VIEWS_DB; KV LIB_KV; Gemini key and model	Has a scheduled handler
r2-autotagger	R2 SPYSHOP_BUCKET; D1 FILE_VIEWS_DB; Queue SPYSHOP_UPLOADS; OpenRouter key	Queue-driven
ask-site	Workers AI; D1 CORR and SEARCH; rate limiter	Mondays 12:15 UTC
hubs-site	D1 FILES and NEWS	None
intel-feed	KV	Every 3 minutes
corrections-site	D1 DB	None

SECTION 2

DNS AND ROUTING

Zone: marielandryspyshop.com on Cloudflare's free website plan, status active, nameservers charles.ns.cloudflare.com and demi.ns.cloudflare.com, original registrar Namecheap.

Record	Name	Content	Proxy	Purpose
A ×4	@	216.239.32.21, 216.239.34.21, 216.239.36.21, 216.239.38.21	Proxied	Blogger bare-domain redirect
CNAME	www	ghs.google.com	Proxied	Blogger origin
AAAA	each Worker subdomain	100::	Proxied	Originless placeholder
MX ×3	@	route1, route2, route3.mx.cloudflare.net	DNS only	Cloudflare Email Routing
TXT	@	Google site verification; SPF	DNS only	Ownership, mail auth
TXT	_dmarc	v=DMARC1; p=none	DNS only	Mail policy (monitor only)
TXT, CNAME	DKIM selectors	Cloudflare and Mailchimp keys	DNS only	Mail signing

Routing rules:

- A Worker that **wraps an origin** (Blogger) attaches as a **route** with a trailing /*, on a proxied record.
- A Worker that **is the origin** attaches as a **custom domain**; Cloudflare creates the record and certificate. Custom domains need an exact hostname match and cannot be created on a hostname that already has a CNAME (Cloudflare).
- A route and a custom domain can share a hostname: the route runs first and can fetch the custom-domain Worker, which is how files gets the shared header.
- Same-zone fetches between Workers succeed when the target is a custom domain; a route target needs a service binding.

DMARC is set to p=none, which only monitors. Moving to quarantine is a candidate hardening step once reports show clean mail.

SECTION 3

Data stores and AI services

Store	Name	Created	Holds
D1	mlss-search	2026-10-29	Full-text search index for the help desk and site search
D1	mlss-corrections	2026-10-28	Corrections register (corrections table)
D1	mlss-blogger-seo	2026-10-24	Blog editing jobs, configuration and history
D1	mlss-news	2026-10-22	News Matrix headlines and sources
D1	file_views	2026-09-29	Library metadata (file_metadata) and view counts
R2	mlss-blog-backups	2026-10-24	Post backups taken before every edit
R2	spyshop	2026-09-23	Public research library files
KV	mlss-cache	—	Shared cache for feeds and lists
Queue	spyshop-uploads	—	R2 upload events for the autotagger
Workflow	mlss-blogger-seo	—	Durable, step-by-step editing jobs

AI service	Model	Used by	Job
Workers AI	@cf/google/gemma-4-26b-a4b-it	ask-site	Answers questions from the site's own pages
Google Gemini API	gemini-3.5-flash-lite (configured)	r2-file-index	Long landing-page text per library document
Google Gemini API	gemini-3.1-flash-lite (in code)	mlss-blogger-seo	Careful post edits under guardrails
OpenRouter	meta-llama/llama-3.2-3b-instruct:free	r2-autotagger	Title, summary and keywords for new uploads
Claude	via Claude app	the Project	Operator: plans, codes, deploys, verifies, writes

Model names are read from Worker code and settings. Check current model availability and pricing with each provider before you copy them; free model tiers can change.

SECTION 4

Platform limits that shape the design

Limit	Workers Free	Workers Paid	Design response
Requests	100,000 per day	No limit	Edge cache blog pages 5 to 15 min, library 10 to 60 min
CPU time	10 ms per request	5 min (default 30 s)	Stream HTML with HTMLRewriter; precompute in cron or Workflows
Subrequests	50 external per request	10,000 default	Batch feed reads; cache shared lists
Workers per account	100	500	One codebase serves many hostnames (hubs-site serves 14)
Cron Triggers	5 per account	250	Few, calm schedules
Browser Run time	10 min per day	10 h per month included	Space verification renders
Browser Run quick actions	1 every 10 s	10 per second	Wait between screenshots

Sources: Workers limits, Browser Run limits. The Blogger API also enforces Google API quotas; a quotaExceeded error means pausing the job, not retrying in a loop.

SECTION 5

Pattern: the header Worker in front of Blogger

Attach as a route on www.example.com/* and on example.com/*. Blogger stays the origin; the Worker streams each HTML page through HTMLRewriter, adds the shared header and security headers, and passes everything else through untouched.

A MINIMAL, GENERIC VERSION

```
// Route: www.example.com/*  (Blogger is the origin behind the proxied www CNAME)
const HEADER = `<div id="site-logo-bar">...logo, menu, stats row, nav rows...</div>`;
const STYLE = `<style>#site-logo-bar{background:#0F1F3D;color:#fff}</style>`;

export default {
  async fetch(request) {
    const res = await fetch(request);                       // ask Blogger for the page
    const type = res.headers.get("content-type") || "";
    if (!type.includes("text/html")) return res;           // feeds, images, CSS pass through

    const html = new HTMLRewriter()
      .on("head", { element(el) { el.append(STYLE, { html: true }); } })
      .on("body", { element(el) { el.prepend(HEADER, { html: true }); } })
      .transform(res);

    const headers = new Headers(html.headers);
    headers.set("Strict-Transport-Security", "max-age=15552000; includeSubDomains");
    headers.set("X-Content-Type-Options", "nosniff");
    headers.set("Referrer-Policy", "strict-origin-when-cross-origin");
    return new Response(html.body, { status: html.status, statusText: html.statusText, headers });
  },
};
```

Remove the route and the plain Blogger site returns at once; that makes the pattern safe to iterate on.

Shared-block contract. Give every shared block a stable id and keep the same markup in every Worker. On marielandryspyshop.com the ids include mlss-logo-bar, mlss-stats-row, mlss-aff-row, mlss-ven-row, mlss-footer-grid and mlss-ask. Verification checks for those ids on each rendered page.

SECTION 6

Pattern: the Blogger API bridge Worker

The bridge lets Claude change posts without ever holding Google credentials. It is a private Worker that owns the OAuth tokens and exposes a small job API that only an admin can reach.

Part	Spec on marielandryspyshop.com
API	Blogger API v3 at https://www.googleapis.com/blogger/v3
Scopes	https://www.googleapis.com/auth/blogger and blogger.readonly
Auth flow	/oauth/start and /oauth/callback; client ID and secret stored as Worker secrets; a separate TOKEN_KEY secret for stored tokens
Access	Cloudflare Access sign-in checked against an admin email list; /login and /logout for the session
Job API	/api/jobs, /api/posts, /api/status, /api/config, /api/events, /api/duplicates, /api/retry-failed, /api/rollback, /api/export/manifest, /health
Engine	A Cloudflare Workflow runs each job step by step: load job, select posts, check, fix, cite, dedupe, notify IndexNow, finish
Safety	Backup to R2 before each change; publish lock; rollback per post; edits rejected if they alter links or images, drop disclaimers, cut too much or add unsupported claims
Trigger	Cron every minute picks up queued work

Blogger API v3 methods the bridge can use (reference):

Resource	Methods
Posts	list, get, search, insert, delete, getByPath, patch, update, publish, revert
Pages	list, get, delete, insert, patch, update
Comments	list, get, approve, delete, listByBlog, markAsSpam, removeContent
Blogs	get, getByUrl, listByUser
Users, BlogUserInfos, PageViews, PostUserInfos	get (and list for PostUserInfos)

Not in the API: themes, template HTML, Layout gadgets and blog settings. Those remain manual Blogger steps.

To build your own: create a Google Cloud OAuth client for a web application with your Worker's /oauth/callback as the redirect URI, enable the Blogger API, store the client ID and secret with `wrangler secret put`, protect the Worker with Cloudflare Access, and start with read-only jobs before enabling writes.

SECTION 7

Deploy, verify and roll back

Every change follows the same seven steps, and Claude reports each one.

1. **Read.** Render the live page and pull the current Worker source (GET /accounts/{account_id}/workers/scripts/{name}/content/v2). Never rebuild a Worker from memory.
2. **Plan in one line:** what changes, on which Workers, and anything that cannot be undone.
3. **Edit by exact match.** Replace a quoted string that must occur a known number of times; abort the whole deploy if any count differs.
4. **Record the previous version** from the Worker's deployments list, so rollback is one call.
5. **Deploy** (PUT .../workers/scripts/{name}/content), then re-read the source and confirm it matches what was sent.
6. **Verify live** with Browser Run at 1280 px and 390 px: check the changed text, the shared-block ids and that nothing else moved. A cache-busting query string avoids reading a stale edge copy.
7. **Log.** Factual fixes get a dated note on the page and a row in the Corrections register; notable changes get a Site Log entry; Blogger-only steps go back to you in order.

Rollback. Redeploy the recorded version, re-verify, and log the rollback. For Blogger posts, restore from the R2 backup through the bridge's rollback endpoint.

SECTION 8

Security and standards baseline

Applied to the blog and all 31 subdomains on October 28, 2026 (Site Log).

Control	Setting
HSTS	HTTPS only, six months, all subdomains
Content sniffing	X-Content-Type-Options: nosniff
Referrer	A strict referrer policy
Permissions-Policy	Blocks camera, microphone, location, payment and USB
security.txt	RFC 9116 contact file on every subdomain
Crawl and AI files	robots.txt, sitemaps, ads.txt and llms.txt checked on every host
Page hygiene	Canonical links, language tags and structured data checked on every host
Secrets	Only in Worker secrets; never in page code, Project files or memory
Admin tools	Behind Cloudflare Access with an admin email list
Mail	SPF, DKIM and DMARC published; DMARC at p=none

Open hardening items: move DMARC from p=none toward quarantine once reports are clean; review the three unattached Workers; turn on two-factor sign-in for the Cloudflare, Google and Namecheap accounts if it is not already on, since each one controls the whole site.

PART III

Prompt Library

Copy, adapt and paste. Prompts work best inside a Claude Project that already holds your instructions and files.

SECTION 1

The starter prompt: clone the infrastructure

Two pastes rebuild the marielandryspyshop.com architecture for your own brand: Part A goes in your Claude Project's instructions, Part B is the first message of a new chat in that Project.

Replace every {{PLACEHOLDER}}. The prompt clones the architecture, never the brand: no name, logo, text, partner links or design from marielandryspyshop.com.

Before you paste: a Blogger blog serving your custom domain over HTTPS (Chapter 4); the domain active on Cloudflare; the Cloudflare connector enabled in Claude (Semrush, Gmail, Google Drive and Claude Docs are optional); and your governing documents, or a request to draft them.

PART A · PROJECT INSTRUCTIONS

```
You are my Viblogger: the editorial, research and web-operations assistant for {{BRAND}} ({{DOMAIN}}),
{{ONE-LINE DESCRIPTION}}, run by {{YOUR NAME}}.

STACK
- Content core: Blogger at www.{{DOMAIN}} (posts, pages, feeds, Blogger API v3).
- Domain at {{REGISTRAR}}; DNS, HTTPS and edge on Cloudflare (free plan unless I say otherwise).
- Edge apps: Cloudflare Workers. A header Worker runs as a route on www.{{DOMAIN}}/* and {{DOMAIN}}/*.
Every subdomain is a Worker on a custom domain carrying the same header, nav rows and footer.
- Data: D1, R2, KV, Queues and Workflows as needed. AI: Workers AI or another model I approve.
- Blogger API access goes through a private bridge Worker that holds the Google OAuth secrets behind
Cloudflare Access. You never handle my Google credentials.

WHO WE ARE AND ARE NOT
- We are: {{WHAT THE SITE IS}}. Audience: {{AUDIENCE}}.
- We are not: {{WHAT THE SITE MUST NEVER CLAIM OR SELL}}.

NON-NEGOTIABLE RULES
1. My governing documents ({{LIST}}) prevail. If a request conflicts, say so, name the rule and offer a
compliant version.
2. Verify every fact, figure, date, law and product detail with a live search or by opening the page. Never
fabricate citations, quotes, links or statistics; label what you cannot verify.
3. Disclose: AI-assisted pages carry an error notice; affiliate links carry a plain-words disclosure next
to them.
4. Never put API tokens or secrets in page code, Project files or memory.
5. Ask before deleting a Worker, page, subdomain or partner, and before any DNS change.
6. {{YOUR OWN RED LINES}}

EDITING LOOP (every change)
1. Read the live page and pull the current Worker code first. Never rebuild from memory.
2. State the plan in one line: what changes, on which Workers, what can't be undone.
3. Make the smallest change; keep the existing structure, CSS, header, rows and footer.
4. Shared blocks live on many Workers: update every Worker that carries one, and list them.
5. Record the previous version, deploy, then verify the live page at 1280 px and 390 px with Browser Run.
6. Log it: dated corrections on the page and on the Corrections page; notable changes in the Site Log.
7. Hand back Blogger-only steps (theme, Layout widgets) in order, with exact clicks and the text to paste.

STANDARDS
```

```
- Writing: lead with the point; short paragraphs; plain words; {{SPELLING}}; byline "{{BYLINE}}"; end
research posts with a Sources list of pages you opened.
- SEO: one primary keyword per page in the title, H1, meta description, first paragraph and slug; about 60-
character titles and 155-character descriptions; canonical, Open Graph and Twitter tags; structured data
where it fits; link each new page to its hub and older related posts to it.
- Brand: colours {{HEX CODES}}; fonts {{FONTS}}; read the live CSS before styling anything new.
- Affiliates: affiliates.{{DOMAIN}} is the source of truth; partners must be lawful and fit my principles;
outreach by email only, drafted for my review.

HOW TO RESPOND
- Lead with the result. After site changes: what changed, where it is live, what you verified, what is left
for me.
- Flag contradictions across the site (counts, partner lists, broken links, outdated notices) with exact
locations.
- If a request is ambiguous and costly to redo, ask one clear question; otherwise make a sensible call and
say what you assumed.
- Reply in the language I write in.
```

PART B · KICKOFF MESSAGE

```
Clone the Viblogger infrastructure of marielandryspyshop.com for {{DOMAIN}}, under my own brand. Do not
copy that site's name, logo, text, partner links or design; build everything original for {{BRAND}}.

Work in phases. At the end of each phase, show me what you built, what you verified and what comes next,
and wait for my "go".

Phase 0, inventory: read my live site and my Cloudflare account (zone, DNS records, Workers, routes, custom
domains, D1, R2, KV). Report what exists, what is missing and anything risky. Change nothing.
Phase 1, header: build a Worker on www.{{DOMAIN}}/* and {{DOMAIN}}/* that keeps Blogger as the origin and
injects a logo bar with a Menu button on phones, a live statistics row, Affiliates, Projects, Official
Pages and Resources rows (placeholders where I have no content yet), an AI-assistance notice and security
headers. Show phone and desktop screenshots before and after.
Phase 2, governing pages: draft a charter or constitution, an ethics and compliance policy, an affiliate
disclosure and a privacy policy for my review. Publish each approved one as its own subdomain Worker and
add it to the Official Pages row.
Phase 3, hubs: create one hub Worker that serves several topic subdomains from one codebase, starting with
{{HUB LIST}}. Each hub gets an introduction, sections, related posts from my Blogger feeds and the shared
header.
Phase 4, library: create an R2 bucket, a D1 table, a queue with an R2 upload notification, an autotagger
Worker (AI title, summary and keywords) and a library Worker at files.{{DOMAIN}} with search, filters, a
page per file, a sitemap, a statistics page and edge caching.
Phase 5, help desk: build ask.{{DOMAIN}} that answers only from my own pages with links, refuses requests
outside my rules, offers full-text search and is rate-limited.
Phase 6, transparency: build log.{{DOMAIN}} (a dated Site Log starting with today's work) and corrections.
{{DOMAIN}} (a corrections register with a way to report errors).
Phase 7, Blogger API bridge: walk me through creating a Google OAuth client and enabling the Blogger API,
then build a private bridge Worker behind Cloudflare Access: read-only jobs first, an R2 backup before any
write, rollback, and edit guardrails (no link, image or disclaimer changes; no unsupported claims).
Phase 8, hardening and SEO: apply HSTS, nosniff, a referrer policy, a permissions policy and security.txt
on every host; check robots files, sitemaps, llms.txt, canonical links and structured data; cache Blogger
pages at the edge; then red-team the whole network and give me a ranked fix list.

Constraints: stay within Cloudflare's free limits unless I approve otherwise, and warn me as we approach
them; keep every shared block identical across Workers; never put secrets in code; ask before DNS changes
or deletions. Start Phase 0 now.
```

SECTION 2

100 PROMPTS BEHIND MARIELANDRYSPYSHOP.COM

These are the requests that built the site. The first nine are in Marie's own words. The other 91 are reconstructed from dated entries in the Site Log and the October 29, 2026 session: the work is real and dated; the wording is a faithful rebuild, not a transcript.

IN YOUR OWN WORDS

1. "Lets work"
2. "Fix everything"
3. "Review my site and optimize everything"
4. "Write a relevant blog post"
5. "Add social media sharing buttons to homepage"
6. "Create a sub page about this topic X"
7. "Look for low quality content and deslopify it"
8. "What to work on next?"
9. "Red team my blog for weaknesses and strengthen it"

HEADER, FOOTER AND SITE-WIDE FEATURES

10. JUN 18 "Create my first Cloudflare Worker for marielandryspyshop.com."
11. JUL 2 "Build a site-wide footer index: monthly archive, top 50 tags and posts, Zenodo records, Scribd documents, sister blogs, AI models and affiliates, with Organization and WebSite structured data."
12. JUL 12 "The ↗ arrows in my footer show as raw code. Fix them so Blogger stops escaping them."
13. AUG 17 "Run a full SEO audit of the whole site and list every problem you find."
14. OCT 3 "Put a Worker in front of Blogger that adds my logo, main navigation, social links, an Affiliates row and an Official Pages row to every page."
15. OCT "Add structured data to every post, tied to my ORCID: author, dates and breadcrumbs."
16. OCT "Add a table of contents, reading time, related posts and an author box to every post."
17. OCT "Convert images to WebP with lazy loading and alt text."
18. OCT "Add an llms.txt for AI answer engines and allow AI crawlers in robots.txt."
19. OCT "Add an SEO footer with an FAQ."
20. OCT "Cut tracking down to Google Analytics and show a cookie banner to EU visitors."
21. OCT 21 "Redesign the header: red links for Antivirus, VPN and Organic Law, amber, green and blue rows on dark blue, and rows you swipe sideways on phones."
22. OCT 21 "Add a cite-or-link box at the end of every post and page."

23. OCT 22 "Add a fourth Resources row in purple that links every new resource site."
24. OCT 24 "On phones, make the header compact with a Menu button, and make article text larger."
25. OCT 27 "Close every blog page's footer with a founder's letter that walks readers through the site."
26. OCT 28 "Give the blog and every subdomain one header: the same name, the same menu and a live statistics row."
27. OCT 28 "Add an Ask AI button to every page."

SUBDOMAINS

28. OCT 13 "Create affiliates.marielandryspyshop.com with all my partners in one directory, and a banner across the network."
29. OCT 13 "Create ai.marielandryspyshop.com linking all my custom AI models on Poe and ChatGPT."
30. OCT 13 "Create apps.marielandryspyshop.com to showcase my GitHub apps under MLSpyShop."
31. OCT 14 "Publish my Constitution at constitution.marielandryspyshop.com."
32. OCT 17 "Create ventures.marielandryspyshop.com for all Landry Industries ventures, and add a Ventures row to the header."
33. OCT 21 "Make docs. and shop. forward to the library and the affiliate directory."
34. OCT 21 "Build an OSINT Research Hub at osint. with guides, tools, a glossary and case studies."
35. OCT 21 "Build The Antivirus Page: rank antivirus products with an introduction, comparison table and FAQ."
36. OCT 21 "Build The VPN Page: an independent VPN ranking with no paid placements."
37. OCT 21 "Create organiclaw. for the Universal Declaration of Organic Rights: its laws, its articles, its history and a document library."
38. OCT 21 "Create a public Site Log at log. that records every update since 2023."
39. OCT 21 "Build The Privacy Page: quick wins, a checklist that saves progress, vetted tools, phone settings, how to spot trackers and stalkerware, Canadian privacy rights and a PIPEDA request letter."
40. OCT 21 "Build News Matrix at news.: live headlines from thousands of RSS feeds in 20 categories, with search, an OPML download, RSS per category and a JSON API."
41. OCT 22 "Build free browser-based OSINT tools at tools. and an OSINT glossary at glossary."
42. OCT 22 "Create a Hemp and Hempoxies research hub at hemp."
43. OCT 22 "Create a Sustainable Development Goals tracker at sdg. with a page for each of the 17 goals."
44. OCT 22 "Create research. as an index of all my research, linked to the library."
45. OCT 22 "Create dossiers. for free public-interest investigations, with a clear method and disclaimer."
46. OCT 22 "Create a press kit at press.: biographies, key facts, the logo and media contact."
47. OCT 22 "Rank password managers at passwords., with no paid placements."
48. OCT 22 "Rank private email providers at email."
49. OCT 22 "Rank AI assistants for privacy at ai-tools."
50. OCT 22 "Build a counter-surveillance gear guide at gear. that explains Canadian law."
51. OCT 22 "Create a bilingual English and French guide to Acadie and New Brunswick at acadie."
52. OCT 22 "Create an events calendar at events. for OSINT, privacy and security events."

53. OCT 22 "Create a Start Here guide at guide. for new visitors, and open the Resources row with a Start Here link."
54. OCT 22 "Create a medical cannabis research hub at 420."
55. OCT 28 "Create a Corrections page at corrections. with our policy and a way for readers to report errors."
56. OCT 28 "Build an AI help desk at ask. that answers only from my own pages and refuses requests to find or track people."
57. OCT 28 "Add full-text search of every post and a weekly digest with an RSS feed."
58. OCT 28 "Build an Organic SEO Hub at seo. that ranks free and paid SEO tools, with no affiliate links."

RESEARCH LIBRARY

59. SEP 23 "Create an R2 bucket to host my research library."
60. SEP 24 "Turn that bucket into files.marielandryspyshop.com: a page per document with an AI summary, keywords, FAQ, related files, share buttons and embed code."
61. SEP 24 "Add search, filters, sorting, a sitemap and a public statistics page that separates people from bots."
62. SEP 29 "Create a D1 database for each file's title, summary, keywords and views."
63. OCT 1 "Tag every new upload automatically with an AI title, summary and keywords."
64. OCT 22 "Cache library pages at the edge so most visits never touch the database."
65. OCT 24 "Turn off ads and search indexing on adult-themed library documents."
66. OCT 24 "Withdraw personal email printouts from the public library."
67. OCT 24 "Point duplicate documents at one main version for search engines."
68. OCT 28 "Move duplicate files out of the library and redirect every old address to the copy that stays."

FEEDS, SPEED AND FREE-TIER LIMITS

69. OCT 21 "Rebuild the IntelMatrix feed on my own aggregator, refresh it every 3 minutes and replace broken sources."
70. OCT 28 "Grow the Global Intel Feed and drop every source that fails a full check."
71. OCT 22 "We hit the free D1 limits. Move live headlines and lists to a shared cache and calm the schedules."
72. OCT 22 "Make News Matrix fall back to its last saved pages if the database is unavailable."
73. OCT 24 "Cache blog pages at the edge and serve a backup copy if Blogger goes down."
74. OCT 24 "Turn on Always Online and Speed Brain, and load 80 headlines in the homepage intel box instead of 500."

CONTENT AND SEO

75. SEP 21 "Write The Master Guide to OSINT."
76. SEP 21 "Write a guide to HUMINT in the modern age."
77. OCT 21 "Write Good Spy, Bad Spy: why OSINT is the legal side of intelligence."
78. OCT 22 "Fact-check every claim on the password, email, AI privacy and gear pages against live sources."

79. OCT 24 "Review every post against six months of Search Console data and hide the ones Google declined from search."
80. OCT 24 "Point Blogger's duplicate copies at their original posts, and hide comment feeds from search."
81. OCT 24 "Add large preview images and image-preview permission for Google Discover."
82. OCT 27 "Build a private editing system that backs up every post, asks Gemini for careful edits and rejects any edit that changes a link or adds unsupported claims."
83. OCT 27 "Tighten the posts that matter most for search, on their original addresses, with rollback."
84. OCT 28 "Replace links that point at Google search pages with direct links to the source."
85. OCT 28 "Shorten titles that are too long for search results, and add search descriptions to edited posts."
86. OCT 28 "Notify Bing of every change."
87. OCT 28 "Add a Sources and further reading section to every post, in the post's language, using only Wikipedia and Crossref."
88. OCT 28 "Expand the OSINT tools directory to 147 ranked tools, leaving out tools built to track private people."
89. OCT 28 "Take duplicate posts and the outdated privacy policy offline, and redirect their addresses."

COMPLIANCE AND TRUST

90. OCT 13 "Remove every offer of paid OSINT reports from the site."
91. OCT 14 "Remove mSpy everywhere and present Spy Associates for counter-surveillance gear."
92. OCT 14 "Add a Constitution link to the Official Pages row."
93. OCT "Reposition the site as an independent OSINT research publication, and add a legal notice on every page about New Brunswick's Private Investigators and Security Services Act."
94. OCT 28 "Rewrite the Gemini-SpyShop sidebar box as a free AI help desk, not services we no longer offer."
95. OCT 28 "Send HSTS, nosniff, a strict referrer policy and a permissions policy from every subdomain, and add security.txt."
96. OCT 28 "Check robots files, sitemaps, ads.txt, llms.txt, canonical links, language tags and structured data on every host."
97. OCT 29 "Remove the affiliate row and banner from the dossiers hub and the audit post."
98. OCT 29 "Fix the empty NordVPN affiliate ID."
99. OCT 29 "Put Spy Associates first everywhere, with a #1 partner badge."
100. OCT 29 "Make the AI-assistant count, the intel feed count and the footer's legal status agree with the Constitution, then log the corrections."

SECTION 3

100 MORE PROMPTS TO ADAPT

Swap the [brackets] for your own topics. Each prompt assumes the Project already holds your rules, so Claude will flag anything that breaks them.

DIRECTION AND PLANNING

1. "Read my Site Log and memory, then tell me the three highest-value things to build this week."
2. "What is the single biggest weakness of my site today, and how would you fix it in one session?"
3. "Map every page and subdomain I have, and show me the gaps a first-time visitor would notice."
4. "Draft a 30-day roadmap from my business plan, one deliverable per week, each tied to a metric."
5. "Compare my site with [competitor] and list what they do better that fits my rules."
6. "Which of my hubs earns its keep? Rank them by traffic, links and effort to maintain."
7. "Turn my last ten Site Log entries into a progress note for my readers."
8. "If I could only keep five subdomains, which five and why?"
9. "Find every promise my site makes to readers and check that each one is still true."
10. "Plan a launch for [new section]: pages, links, announcement post and the Site Log entry."

AUDITS AND RED-TEAMING

11. "Red team my homepage as a sceptical journalist would. What would you quote against me?"
12. "Check every outbound link on [subdomain] and fix or remove the broken ones."
13. "Find numbers that disagree across pages (counts, dates, prices) and show me each location."
14. "Audit every page for missing AI and affiliate disclosures."
15. "Check that no investigative page carries ads or affiliate links."
16. "Render my top ten pages at phone width and list anything that overflows, overlaps or is hard to tap."
17. "Run an accessibility pass: contrast, alt text, heading order, link names."
18. "List every page that makes a legal or medical claim and check each against a primary source."
19. "Find pages that still mention [retired product, partner or service] and propose fixes."
20. "Look for anything that could expose a private person, and remove it."

BUILDING PAGES AND SUBDOMAINS

21. "Create [topic].mydomain.com as a hub with the shared header, an introduction and links to my related posts."
22. "Add a new page to hubs-site for [topic], matching the existing hub layout."
23. "Build a comparison page that ranks [product category] with criteria, a table and an FAQ, no paid placements."
24. "Make a glossary page for [field] from terms already used on my site, linked to the posts that use them."

25. "Create a Start Here path for [reader type] through my existing content."
26. "Build an events page for [field] with dates verified from the organisers' own sites."
27. "Create a bilingual English and French version of [page], with hreflang tags."
28. "Add an FAQ section with FAQPage structured data to [page]."
29. "Make a short-address redirect: [alias].mydomain.com to [destination]."
30. "Build a simple free tool for [task] that runs in the browser and stores nothing."

DESIGN AND EXPERIENCE

31. "Add social sharing buttons under every post, in my house colours."
32. "Make the header lighter on phones without removing any links."
33. "Give every hub a consistent hero: title, one-line promise, three entry links."
34. "Add a reading-progress bar to long posts."
35. "Design a dark-mode stylesheet for my Worker pages that matches the brand."
36. "Add breadcrumbs to every subdomain page, with BreadcrumbList structured data."
37. "Replace any stock-looking image on [page] with a described placeholder for me to fill."
38. "Make my footer scannable in five seconds; propose a new grouping first."
39. "Add a back-to-top button and anchor links to every heading on long pages."
40. "Show me before-and-after screenshots for any design change before you deploy it."

WRITING NEW CONTENT

41. "Write a relevant blog post on [topic] for my audience, with sources I can check."
42. "Turn [library document] into a reader-friendly post that links back to it."
43. "Write a beginner's guide to [method], with one worked example from public sources."
44. "Draft a monthly roundup of what changed on the site, for readers and for search."
45. "Write a myth-versus-fact post on [topic], each fact tied to a primary source."
46. "Write a French version of [post] for Acadian readers, not a literal translation."
47. "Draft three titles and meta descriptions for [post] and tell me which wins and why."
48. "Write the press release for [launch], plain and factual, for my review."
49. "Create a checklist post for [task] that readers can print."
50. "Write an explainer on [law or rule] that quotes the statute's section numbers and links the official text."

QUALITY AND DESLOPIFY

51. "Find my twenty weakest posts and propose what to rewrite, merge, redirect or hide."
52. "Deslopify [post]: remove filler, hedges and repetition without changing facts or links."
53. "Find posts that repeat the same boilerplate paragraph and propose one shared version."
54. "Check [post] for claims without sources, and add sources or soften the claims."

55. "Merge [post A] and [post B] into one stronger post and redirect the weaker address."
56. "Rewrite headings on [post] so a skimmer understands it from the headings alone."
57. "Find posts with titles over 60 characters and propose shorter ones."
58. "Mark AI-assisted posts that lack the error notice, and add it."
59. "Find posts whose dates make them read as current when they are not, and add dated update notes."
60. "Check every quote on [page] against its source and fix any that drifted."

SEARCH AND GROWTH

61. "Use Semrush to find ten keywords I can rank for with my existing content."
62. "Pick one primary keyword for each of my hubs and check title, H1, description, first paragraph and slug."
63. "Find my pages that rank 11 to 20 and suggest one improvement each."
64. "Build an internal-linking plan: every new page linked from its hub and three older posts."
65. "Check my sitemaps and robots files on every subdomain and fix what search engines can't reach."
66. "Find which of my pages get backlinks and propose a linkable asset like them."
67. "Write structured data for [page] and validate it."
68. "List pages that cannibalise each other for the same keyword and propose a fix."
69. "Prepare an IndexNow notification for every page we changed today."
70. "Track [keyword] monthly and tell me when it moves."

AFFILIATES AND REVENUE

71. "Check every affiliate link on the site still resolves and still carries my ID."
72. "Is [company] a fit for my affiliate rules? Show me the evidence for and against."
73. "Draft an outreach email to [network] asking for a written reply only, for my review."
74. "Add [partner] to the affiliate directory and the shared Affiliates row on every Worker."
75. "Build a ranked page for [defensive product category] where affiliates sit beside non-affiliates on merit."
76. "Find pages with high traffic and no relevant, rule-compliant partner, and suggest one."
77. "Write the plain-words disclosure for [partner] to sit next to its links."
78. "Draft a digital product listing for [guide] on my store, with honest claims only."
79. "Check that no affiliate link appears on any investigative report or its hub."
80. "Summarise which partners appear where across the network, in one table."

DATA, LIBRARY AND AUTOMATION

81. "Upload [file] to the library and check its AI summary and tags are accurate."
82. "Find library files with weak or missing summaries and regenerate them."
83. "Find duplicate files in the library and redirect them to the best copy."
84. "Add a filter to the library for [category]."

85. "Show me how close each Worker is to the free-tier limits this week."
86. "Move [repeated database read] into a cache and show me the saving."
87. "Add a fallback so [subdomain] shows its last saved page if its database fails."
88. "Back up every post to R2 now, and tell me how to restore one."
89. "Schedule a weekly broken-link check and send me a summary."
90. "Build a JSON feed of my newest posts and library files for other sites to use."

MEMORY, WORKFLOW AND GOVERNANCE

91. "Remember that [preference], and use it from now on."
92. "Forget [outdated fact] and check that nothing on the site still depends on it."
93. "Read my Project files and tell me which ones are out of date."
94. "Turn today's work into a Site Log entry and a dated correction where needed."
95. "Summarise everything you know about my affiliate setup, then tell me what is missing."
96. "Draft an amendment to my Constitution for [change], with the reason, for my review."
97. "Write a checklist I can paste into future chats for [recurring task]."
98. "Review this chat and suggest one rule to add to the Project instructions."
99. "List every manual Blogger step still outstanding, in order."
100. "Prepare a handover note so a fresh chat can continue this work tomorrow."

Set in Cinzel, EB Garamond, Lora and Poppins. Rendered from HTML to PDF with WeasyPrint.

LandryIndustries.ca · MarieLandrySpyShop.com