OpenClaw Plugins & Skills Marketplace: Complete Guide

February 5, 2026 · by Mark Bartlett

Download this guide as PDF

> **Quick Answer:** OpenClaw ships with 50+ bundled skills and connects to ClawHub, a community marketplace with 14,706+ skills. Start with the bundled skills only — they're vetted and cover most use cases (email, calendar, browser, shell, GitHub). ClawHub is dangerous: RankClaw's full audit found 1,103 malicious skills out of 14,706 total (7.5% of the marketplace). OpenClaw now has a VirusTotal partnership for scanning new submissions, but it only catches known signatures. Never install a community skill without reading its source code first. If you need custom functionality, writing your own skill takes 10 minutes — it's just a SKILL.md file with instructions. Update to OpenClaw 2026.2.1 minimum to patch CVE-2026-28458 (Browser Relay auth bypass).

📚 **More on this topic:** OpenClaw Setup Guide · OpenClaw Security Guide · Best Models for OpenClaw · How OpenClaw Works

OpenClaw's skill system is what makes it more than a chatbot. Skills give the agent hands — the ability to read your email, commit code, browse the web, manage your calendar, and automate workflows. The 50+ bundled skills cover the basics. ClawHub, the community marketplace, has 14,706+ more.

The problem: ClawHub's security track record is bad. RankClaw's full audit found **1,103 malicious skills** across the entire registry — 7.5% of all skills on the platform. The original Koi Security audit caught 341 in a smaller sample. OpenClaw has since partnered with VirusTotal for automated scanning of new submissions, but that catches known malware signatures, not social engineering tricks or novel exfiltration.

This guide covers what's safe to install, what to avoid, and how to build your own skills when you don't trust someone else's code.

How OpenClaw Skills Work

A skill is a folder containing a `SKILL.md` file. That file has YAML frontmatter (name, description, requirements) and markdown instructions that tell the agent what the skill does and when to use it. That's it. No compiled code. No complex framework. Just a text file that the agent reads.

When OpenClaw starts a session, it snapshots all eligible skills and includes their instructions in the agent's context. Each skill costs roughly 24 tokens plus the length of its instructions. The agent uses these instructions to decide when and how to invoke tools.

Where Skills Live

Skills load from three locations, highest priority first:

Location	Path	Purpose
Workspace	`<workspace>/skills/`	Project-specific skills
User	`~/.openclaw/skills/`	Your personal skills
Bundled	Shipped with OpenClaw	Official, vetted skills

Workspace skills override user skills, which override bundled ones. If you create a skill with the same name as a bundled one, yours takes precedence. This is useful for customizing default behavior — and a potential attack vector if a malicious skill shadows a trusted one.

Skill Requirements and Gating

Skills can declare what they need to function:

```
---
name: github-integration
description: Manage GitHub repos, issues, and PRs
metadata:
  openclaw:
    requires:
      bins: ["gh"]
      env: ["GITHUB_TOKEN"]
    os: ["darwin", "linux"]
---
```

If the required binaries aren't on your PATH or environment variables aren't set, the skill silently disables itself. This is a smart design — it means your agent context doesn't fill up with instructions for tools that can't actually run.

Bundled Skills: Start Here

These ship with OpenClaw and are maintained by the core team. They're the safest option and cover most common use cases.

Productivity

Skill	What It Does	Risk Level
email-triage	Read, summarize, draft replies, flag urgent messages	Medium — agent reads your email
calendar	Create events, check conflicts, set reminders	Low
apple-notes	Create, search, and update Apple Notes	Low
apple-reminders	Manage reminders and due dates	Low
screenshot	Take and analyze screenshots	Low
pdf-extractor	Extract text from PDFs	Low

Developer Tools

Skill	What It Does	Risk Level
github-integration	Issues, PRs, repo management, webhook triggers	Medium — needs repo access token
agent-browser	Headless Playwright automation with accessibility tree	High — can browse anywhere
exec	Run shell commands	High — full system access
file-read / file-write	Read and write files on disk	High — filesystem access
web-fetch / web-search	Make HTTP requests, search the web	Medium — outbound network access
docker-skill	Container management	High — Docker socket access

The High-Risk Bundled Skills

Four bundled skills deserve extra caution. They're not malicious — they're official and useful — but they give the agent broad access:

- **exec**: Shell command execution. The agent can run anything your user can run. Disable unless you need it.
- **agent-browser**: Full browser automation. The agent can visit any URL, fill forms, click buttons. Good for research, bad if the agent gets prompt-injected.
- **file-read / file-write**: Filesystem access. The agent can read your config files, SSH keys, anything in its path.
- **web-fetch**: Outbound HTTP. A compromised agent can exfiltrate data to external servers.

Our OpenClaw Security Guide covers how to restrict these using allowlists and sandboxing. The short version: disable everything you don't actively need.

ClawHub: The Community Marketplace

ClawHub is OpenClaw's public skill registry at clawhub.ai. Anyone with a GitHub account (at least one week old) can publish a skill. As of March 2026, it hosts over 14,700 community-built skills with vector-powered search, star ratings, and versioning. New submissions are now scanned via VirusTotal before publication (see security section below).

How to Install a ClawHub Skill

```
# Search for skills
clawhub search "notion"

# Install a skill to your workspace
clawhub install notion-integration

# Update all installed skills
clawhub update --all
```

Installed skills land in your workspace's `skills/` directory. You can read the `SKILL.md` before enabling it.

Worth Installing (If You Read the Source First)

These community skills fill gaps that bundled skills don't cover. Verify the source code before installing any of them.

Skill	What It Does	Why It's Useful
notion-integration	Notion database and page operations	If you live in Notion, this connects your agent to your workspace
obsidian-vault	Read, search, and manage Obsidian notes	Pairs well with local-first knowledge management
news-aggregator	Aggregates Hacker News, GitHub Trending, Product Hunt, and 5 other sources	Great for morning briefings via heartbeat or cron
gitlab-integration	MR and pipeline management	If your repos are on GitLab instead of GitHub
home-assistant	Smart home control	Voice-to-agent-to-home-assistant is genuinely useful
trello-boards	Trello board and card management	Lightweight project management integration

What to Avoid

Do not install community skills that:

- **Request permissions they shouldn't need.** A note-taking skill doesn't need `exec` or `web_fetch`. Check the SKILL.md for what tools it tells the agent to use.
- **Have low download counts and no stars.** Popularity isn't a guarantee of safety, but zero community vetting is a red flag.
- **Claim to automate financial transactions.** The Koi Security audit found the majority of malicious skills masqueraded as Solana wallets, Polymarket bots, and cryptocurrency trading tools.
- **Are newly published by accounts with no other contributions.** The ClawHavoc campaign used fresh accounts to push 335 malicious skills disguised as legitimate tools.
- **Have obfuscated code or fetch remote payloads.** If a SKILL.md contains base64 strings, curl commands to unfamiliar servers, or tells the agent to download and execute external scripts, it's malware.

The ClawHub Security Problem

This isn't theoretical risk. It's documented, measured, and ongoing.

The Numbers

Two audits, two months apart, and the numbers got worse:

Audit	Date	Skills Scanned	Malicious Found	Rate
Koi Security	Feb 2026	2,857	341	11.9%
RankClaw (full registry)	Mar 2026	14,706	1,103	7.5%

The original Koi Security audit caught the ClawHavoc campaign — 335 skills from a single coordinated attack deploying Atomic Stealer (AMOS) malware through fake prerequisite instructions. RankClaw's full-registry audit found additional campaigns beyond ClawHavoc: credential exfiltration via legitimate-looking API wrappers, supply chain poisoning of popular skill forks, and skills that activate malicious behavior only after several days of normal operation.

The findings across both audits:

- **Atomic Stealer (AMOS)** deployed via fake prerequisites — a macOS info-stealer that costs $500-1,000/month on criminal marketplaces
- **Keyloggers** capturing API keys and credentials
- **Reverse shells** hidden within functional code — the skill works, but also gives an attacker remote access
- **Bot credentials** stolen from `~/.clawdbot/.env` and exfiltrated via webhook.site
- **Delayed-activation malware** — skills that behave normally for days before turning malicious

The campaigns specifically target OpenClaw users running Mac Minis — a common setup since OpenClaw runs well as a headless service on Apple Silicon.

For detailed analysis, see our security reports: January 2026 · February 2026 · ClawHub Security Alert

This came on top of earlier research finding that 26% of 31,000 agent skills across platforms contained at least one vulnerability, including active data exfiltration via curl commands with no user notification.

ClawHub Moderation: Getting Better, Still Insufficient

The safeguards as of March 2026:

Safeguard	What It Does	Status
VirusTotal scanning	Automated malware scan on new submissions	**New (Mar 2026).** Catches known signatures. Misses social engineering and novel techniques.
GitHub account age check	Publishers need a 1-week-old GitHub account	Trivial to bypass. Attackers pre-create accounts.
Community reporting	3 unique reports auto-hide a skill	Reactive, not preventive. Damage done before reports.
Report limit	20 active reports per user	Limits spam but also limits vigilant users.
Download counts	Visible on skill listings	Easily inflated. O'Reilly inflated a fake skill to 4,000 downloads, devs from 7 countries installed it.

The VirusTotal partnership is a real improvement — it's the first automated scanning ClawHub has ever had. But remember: the original AMOS binary was only detected by 16 out of ~70 security engines on VirusTotal. And the ClawHavoc attack's primary vector was social engineering (fake "prerequisite" shell commands), which no automated scanner would catch.

Still missing: code review process, sandboxed testing, cryptographic signing, and publisher verification beyond a GitHub account age check.

For the full security picture, read our OpenClaw Security Guide and the ClawHub Security Alert.

Building Your Own Skills

The safest plugin is one you wrote yourself. And it takes about 10 minutes.

Your First Custom Skill

Create a directory and a `SKILL.md` file:

```
mkdir -p ~/.openclaw/skills/my-standup
```

Write the skill:

```
---
name: my-standup
description: Generate a daily standup summary from git commits
metadata:
  openclaw:
    requires:
      bins: ["git"]
---

# Daily Standup Generator

When the user asks for a standup summary, or when triggered by
a morning cron:

1. Run `git log --oneline --since="yesterday"` in the current
   workspace
2. Group commits by type (feat, fix, chore, docs)
3. Format as a brief standup update with "Yesterday" and "Today"
   sections
4. Keep it under 200 words
```

That's it. Restart OpenClaw or ask the agent to refresh skills. The agent now knows how to generate standup summaries. The skill file is the entire implementation — the LLM reads the instructions and uses its existing tools (exec, in this case) to carry them out.

Skill Structure Best Practices

- **Be specific about when to activate.** "When the user asks for a standup" is better than vague instructions the agent might trigger incorrectly.
- **Limit the tools referenced.** If your skill only needs `git`, don't write instructions that encourage the agent to also browse the web or read random files.
- **Keep instructions concise.** Every skill adds tokens to the agent's context. A 500-word SKILL.md is fine. A 5,000-word one wastes context budget. Skills cost ~24 tokens of overhead plus the instruction length.
- **Test with `openclaw agent --message`** before connecting to live messaging channels.

Specifying Model Routing

Skills can suggest which LLM handles them. This is powerful if you're running a multi-model setup:

```
---
name: code-review
description: Review code changes for bugs and style issues
metadata:
  openclaw:
    preferredModel: qwen-coder-32b
---
```

A coding skill routes to Qwen Coder. A planning skill routes to DeepSeek-R1. A casual chat stays on whatever your default model is. This requires each model to be available through your LLM backend — typically multiple Ollama instances or a mix of local and API models.

Publishing to ClawHub

If you want to share your skill:

```
clawhub publish my-standup
```

Your skill gets a page on clawhub.ai with version history, stars, and comments. But remember — you're contributing to an ecosystem with known security problems. Version your releases, write clear documentation, and accept that someone might fork your skill and inject malicious code into their version.

Recommended Skill Setup by Use Case

Productivity (Low Risk)

Stick to bundled skills. Enable only what you use.

Enable	Disable
calendar, apple-notes, apple-reminders, email-triage	exec, agent-browser, docker-skill

Add a custom news summary skill if you want morning briefings. Run heartbeats during business hours only to control token costs.

Developer (Medium Risk)

You'll need some high-risk tools. Constrain them.

Enable	Configure
github-integration, file-read, file-write, exec	Restrict exec to specific directories. Limit file access to project folders.

Consider a custom standup or PR review skill. If you need GitLab, install the community skill after reading its source. Run on a dedicated machine or VM — not your daily-driver.

Research (Medium Risk)

Web access is useful but introduces prompt injection surface from external content.

Enable	Configure
web-search, web-fetch, pdf-extractor, news-aggregator	Restrict outbound domains. Don't enable exec alongside web skills.

The combination of web browsing and shell access is particularly dangerous — a prompt injection from a webpage could lead to command execution. Keep these skill sets separated if possible.

The Bottom Line

OpenClaw's skill system is its greatest strength and its biggest vulnerability. The bundled skills are useful and reasonably safe. ClawHub is improving — VirusTotal scanning is a step forward — but 1,103 out of 14,706 skills (7.5%) are confirmed malicious.

The practical approach:

1. **Update to OpenClaw 2026.2.1 or later.** Earlier versions have a known auth bypass (CVE-2026-28458).
2. **Start with bundled skills only.** They cover email, calendar, GitHub, browser, files, and shell.
3. **Disable what you don't need.** Every enabled skill is attack surface.
4. **Read the source before installing any community skill.** If you can't understand what a SKILL.md does, don't install it. VirusTotal scanning catches known malware but misses social engineering attacks.

5. **Build your own for custom needs.** It takes 10 minutes and you control every instruction.
6. **Run on dedicated hardware** with throwaway accounts until you trust your setup. See our security guide for hardening steps.

The best OpenClaw setup isn't the one with the most plugins. It's the one with the fewest plugins that still does what you need.

Get notified when we publish new guides.

Subscribe — free, no spam
