Ainary

AR-037 Confidence: 82%

AI Agent Security

Why Your Agent Infrastructure Is a Target

Your AI agent has the keys to your email, calendar, code repos, and bank accounts. On February 13, 2026, attackers stole their first agent identity. 341 malicious skills are live on the largest skill marketplace. This is the threat briefing you need to read before your next board meeting.

February 17, 2026
v1.0

Florian Ziesche · Ainary Ventures

"This finding marks a significant milestone in the evolution of infostealer behavior: the transition from stealing browser credentials to harvesting the 'souls' and identities of personal AI agents."

— Hudson Rock, February 16, 2026

CONTENTS

341

Malicious skills found on ClawHub

Koi Security / VirusTotal, Feb 2026

88%

Of orgs reporting AI agent security incidents

Gravitee State of AI Agent Security 2026

85%+

Prompt injection success rate vs. SOTA defenses

arXiv 2601.17548, Jan 2026

Feb 13

First documented agent identity theft in the wild

Hudson Rock / BleepingComputer, Feb 2026

1. Executive Summary

Section confidence: 85% · Based on primary source reporting from the past 14 days

AI agents are the new high-value targets. They hold credentials for every service they access, operate with minimal human oversight, and run on infrastructure that was never designed for adversarial conditions. The threat is no longer theoretical.

In the first two weeks of February 2026, three developments converged to make AI agent security an urgent priority:

1. **Supply chain poisoning at scale.** Security researchers at Koi Security identified 341 malicious skills on ClawHub, the largest marketplace for OpenClaw agent extensions. The majority belong to a campaign dubbed "ClawHavoc" that distributes Atomic Stealer malware through fake prerequisite instructions.[1]
2. **The first agent identity theft.** Hudson Rock documented the first in-the-wild case of infostealer malware (a Vidar variant) specifically exfiltrating an OpenClaw agent's configuration — including gateway tokens, private keys, SOUL.md personality files, and memory logs containing private messages and calendar events.[2]
3. **Enterprise exposure is systemic.** Gravitee's 2026 State of AI Agent Security report found that 88% of organizations have experienced confirmed or suspected AI agent security incidents. In healthcare, that figure is 92.7%.[3]

This report is not fear, uncertainty, and doubt. It's a practical threat assessment written by someone who runs these systems daily. We discovered our own security gaps — 44GB of unencrypted logs, autonomous crons modifying core identity files, no token rotation — in the process of writing it.

SO WHAT

If you're running an AI agent — personally or inside a fund — you are running an identity with broad access, no MFA, and a flat file for authentication. This report tells you what's being exploited, what's at risk, and exactly what to do about it.

2. The Attack Surface

Section confidence: 90% · Based on documented architecture and confirmed attack vectors

A typical AI agent installation — whether OpenClaw, a custom LangChain deployment, or an enterprise copilot — presents five distinct attack surfaces. Understanding them is prerequisite to defense.

Exhibit A — Agent Attack Surface Map

SURFACE	WHAT'S EXPOSED	ATTACK VECTOR	SEVERITY
Config files	API keys, gateway tokens, OAuth secrets in `.env`, `openclaw.json`	Infostealer file sweep, exposed instances	Critical
Gateway tokens	Authentication for remote agent control	Token theft enables remote command execution	Critical
Skill marketplace	Third-party code running in agent context	Malicious skills, typosquatting, supply chain	High
Memory files	MEMORY.md, daily logs, SOUL.md — private context	Exfiltration reveals schedule, contacts, decisions	High
Cron jobs	Automated actions with agent privileges	Prompt injection via ingested content, no human review	High
Device keys	`device.json` — public/private key pair	Stolen private key enables device impersonation	Critical

Source: Ainary analysis based on OpenClaw architecture documentation, Hudson Rock findings, Koi Security audit

The Fundamental Problem: Flat-File Identity

Most AI agent frameworks store credentials in plain-text configuration files on disk. OpenClaw stores its gateway token in `openclaw.json`, device keys in `device.json`, and behavioral identity in `soul.md`. There is no hardware security module, no encrypted keychain integration, no token rotation by default. The agent's entire identity — credentials, personality, memory — lives in a directory that any process with file-read access can copy.

This is analogous to storing your SSH private keys, browser passwords, and personal diary in a single unencrypted folder called `.secrets`. It was acceptable when these tools were experimental. It is not acceptable when they're managing email, calendar, financial data, and code repositories.

> **CRITICAL FINDING**
>
> Hudson Rock confirmed that the stolen OpenClaw configuration included a "high-entropy gateway authentication token" that could enable remote connection to the victim's local agent instance or client impersonation in authenticated requests.[2]

3. ClawHub: 341 Malicious Skills

Section confidence: 90% · Based on Koi Security primary research, confirmed by VirusTotal and The Hacker News

ClawHub is the primary marketplace for OpenClaw skills — third-party extensions that give agents new capabilities. Think of it as an app store for AI agents. In early February 2026, Koi Security audited 2,857 skills on the platform and found **341 that were actively malicious**.[1]

The ClawHavoc Campaign

335 of the 341 malicious skills belong to a single campaign that Koi has codenamed "ClawHavoc." The attack pattern is consistent and effective:

1. **Typosquatting and impersonation.** Skills are named to mimic legitimate tools or the platform itself: `clawhub`, `clawhub1`, `clawhubb`, `clawhubcli`, `clawwhub`, `cllawhub`. Others impersonate popular categories: Solana wallet trackers, Polymarket bots, YouTube utilities, Google Workspace integrations.
2. **Professional-looking documentation.** Each skill has polished README files with credible-looking feature lists. A "Prerequisites" section instructs users to install an external dependency.
3. **Platform-specific payloads.** On Windows, users download `openclaw-agent.zip` from a GitHub repository (password-protected). On macOS, they paste a shell script from glot.io into Terminal. Both deliver Atomic Stealer (AMOS), a commodity infostealer available for $500–$1,000/month.[1]

```
User installs skill from ClawHub
→ Skill README says "install prerequisite"
→ Windows: download ZIP from GitHub → run .exe (trojan)
→ macOS: paste script into Terminal → obfuscated shell → fetches
Mach-O binary
→ Atomic Stealer harvests: API keys, credentials, browser data,
```

```
agent configs
→ Exfiltration to C2 at 91.92.242[.]30
```

Beyond ClawHavoc: The Outliers

Six additional malicious skills used distinct techniques:

- **Reverse shell backdoors** embedded in functional code (`better-polymarket`, `polymarket-all-in-one`) — the skill works as advertised while maintaining persistent access.[4]
- **Credential exfiltration via webhook** — the `rankaj` skill read bot credentials from `~/.clawdbot/.env` and posted them to webhook.site.[1]
- **Data poisoning** — skills that subtly alter agent behavior by injecting instructions into memory files.

Why This Happened: No Vetting

At the time of the Koi audit, ClawHub had **no pre-publication security review**. Anyone could publish a skill. There was no code scanning, no sandboxing, no behavioral analysis. The marketplace relied entirely on user trust.

OpenClaw has since announced integration with VirusTotal scanning — skills flagged as malicious are blocked from download, and active skills are re-scanned daily.[5] This is a meaningful improvement, but it is reactive: it catches known-bad signatures, not novel attack patterns.

> SO WHAT
>
> The agent skill supply chain has the same problems that plagued npm, PyPI, and browser extension stores — but with higher stakes. A malicious npm package steals secrets from a build pipeline. A malicious agent skill steals secrets from a system that has access to your email, calendar, bank, and code.

4. Infostealers Targeting Agent Configs

Section confidence: 88% · Based on Hudson Rock primary findings, confirmed by BleepingComputer, Feb 16, 2026

On February 16, 2026, Hudson Rock published what they described as "a significant milestone in the evolution of infostealer behavior": the first documented case of malware specifically exfiltrating an AI agent's operational identity.[2]

What Happened

A Vidar infostealer variant infected a victim's machine on February 13, 2026. The malware ran a broad file-stealing routine, scanning for directories and files containing keywords like "token" and "private key." The `.openclaw` configuration directory matched these patterns, and the following files were stolen:

Exhibit B — Stolen Agent Files

FILE	CONTENTS	IMPACT
`openclaw.json`	Email, workspace path, gateway authentication token	Remote agent control, client impersonation
`device.json`	Public and private key pair for device pairing/signing	Device impersonation, bypass "Safe Device" checks, access encrypted logs
`soul.md`	Agent behavioral identity, rules, escalation procedures	Social engineering leverage, identity cloning
`MEMORY.md`, `AGENTS.md`	Daily activity logs, private messages, calendar events, decisions	Full operational intelligence on the victim

Source: Hudson Rock, BleepingComputer, Feb 16, 2026

Hudson Rock's analysis concluded that the stolen data was sufficient to "potentially enable a full compromise of the victim's digital identity."[2]

Why This Matters More Than Browser Credential Theft

Traditional infostealers target browser cookies, saved passwords, and cryptocurrency wallets. Agent configuration theft is qualitatively different:

- **Aggregated access.** A single agent config can contain credentials for email, calendar, GitHub, Slack, financial services, and custom APIs — all in one directory.
- **Behavioral identity.** SOUL.md and memory files reveal not just what services the victim uses, but how they think, what decisions they're making, and who they're communicating with.
- **Persistent control.** A stolen gateway token doesn't just read data — it can issue commands to the agent, potentially sending emails, modifying code, or exfiltrating additional data through the agent's existing permissions.

- **No rotation by default.** Unlike browser sessions that expire, agent tokens and device keys persist indefinitely unless manually rotated.

> EMERGING THREAT
>
> Hudson Rock expects infostealers to develop "more targeted mechanisms for AI agents" as adoption grows. The current theft was opportunistic — the malware swept for keywords and happened to find agent configs. Purpose-built agent-targeting malware is the logical next step.[2]

The infostealers.com analysis went further, documenting how stolen memory files contained VPN credentials, corporate portal access details, and meeting notes — operational context that transforms a credential theft into a full intelligence operation.[6]

5. The Autonomous Agent Problem

Section confidence: 80% · Based on enterprise surveys and documented failure patterns

The security challenges above — malicious skills, credential theft — are exacerbated by a design philosophy that prizes autonomy over oversight. Modern AI agents are explicitly built to act without human intervention. That's the feature. It's also the vulnerability.

Cron Jobs: Automation Without Guardrails

AI agents routinely run scheduled tasks (cron jobs) that execute with full agent permissions. Common patterns include:

- Morning briefings that read email and calendar data
- Nightly memory distillation that reads and writes to identity files
- Automated code review and PR creation
- Content publication to social media
- Financial data aggregation and reporting

These crons run in isolated sessions — they have no awareness of decisions made in the main conversation. This creates what the community calls the **"Dory Problem"**: you cancel something in chat, but the scheduled cron fires anyway because it doesn't know.[7]

Prompt Injection at Scale

When autonomous agents ingest external content — emails, web pages, webhook payloads, documents — every piece of content is a potential prompt injection vector. A research paper from January 2026 analyzed 78 studies and found that **attack success rates against state-of-the-art defenses exceed 85% when adaptive strategies are employed**.[8]

Forbes reported on the practical implications: AI agents with authority to process payments could be manipulated by instructions hidden in invoice attachments.

Agents that manage email could be redirected by prompt injections embedded in incoming messages.[9]

The Governance Gap

Microsoft's Cyber Pulse AI Security Report found that most organizations lack visibility into their agent populations — "unsupervised or ungoverned AI agents can compound risks in the enterprise, threatening security, business continuity, and reputation."[10]

CyberArk's 2026 analysis warned: "The more autonomous and interconnected these AI agents become, the larger the attack surface they create. By 2026, we won't just be experimenting with AI agents; we'll be relying on them."[11]

> SO WHAT
>
> Autonomy and security are in direct tension. Every permission you grant an agent to act independently is a permission an attacker inherits if the agent is compromised. The question isn't whether to use agents — it's how to bound their autonomous authority.

6. Case Study: Our Own Security Gaps

Section confidence: 95% · Based on direct observation of our own infrastructure

We wrote this report while running the systems it describes. In the process, we audited our own OpenClaw deployment and found security gaps that mirror every vulnerability discussed above. We're publishing them because transparency is more useful than pretending we had it figured out.

Finding 1: Capability Evolver Autonomously Modifying SOUL.md

We had a cron job called "Capability Evolver" that ran weekly. Its purpose was to analyze agent performance and update SOUL.md — the file that defines the agent's behavioral identity — to improve capability. In effect, **the agent was autonomously rewriting its own operating instructions** without human review.

The security implication: if a prompt injection altered the agent's behavior during the week, the Capability Evolver could cement that alteration into the permanent identity file. We've since added a rule that cron jobs MUST NOT modify SOUL.md, AGENTS.md, or MEMORY.md autonomously.

Finding 2: 44GB of Unencrypted Logs

Agent session transcripts accumulate rapidly. Our workspace contained approximately 44GB of unencrypted log data — including full conversation transcripts, daily memory files, API responses, and tool outputs. This data contained email contents, calendar details, financial discussions, and access credentials mentioned in conversation.

An infostealer targeting the `.openclaw` directory would have access to years of operational context.

Finding 3: No Token Rotation

Our gateway token had never been rotated since initial setup. Device keys were generated once and persisted indefinitely. There was no alerting for unusual

access patterns and no mechanism to invalidate compromised tokens without full redeployment.

Finding 4: Overly Broad Skill Permissions

Installed skills had access to the full agent context — there was no permission scoping, no sandboxing, and no distinction between a skill that summarizes YouTube videos and one that reads your email.

> **WHAT WE CHANGED**
>
> - Added explicit SOUL.md protection: `Cron jobs MUST NOT modify SOUL.md, AGENTS.md, or MEMORY.md autonomously`
> - Scheduled log rotation and archival with encryption
> - Implemented token rotation schedule
> - Audited all installed skills against the Koi Security findings
> - Added pre-flight checks: every cron verifies a `DECISIONS.md` file before taking external action

7. Defense Playbook

Section confidence: 82% · Mitigations are practical and tested; effectiveness against novel attacks is uncertain

Security is not a binary state. The goal is to raise the cost of attack above the value of your assets. Here's what works today, organized by operator type.

For Solo Operators (Running Personal Agents)

1. Audit Your Skills

Review every installed skill against the Koi Security findings. Remove anything you didn't explicitly choose. Check for typosquatting variants.

2. Encrypt Your Config

Move secrets out of plain-text files. Use macOS Keychain, Linux secret-tool, or a password manager CLI. At minimum, `chmod 600` your `.openclaw` directory.

3. Rotate Tokens Monthly

Set a calendar reminder. Regenerate gateway tokens and API keys. If you can't remember when you last rotated, rotate now.

4. Protect Identity Files

SOUL.md and MEMORY.md should be read-only for automated processes. Only human-initiated sessions should write to identity files.

5. Scope Cron Permissions

Crons should run in isolated sessions with minimal permissions. Add pre-flight checks against a `DECISIONS.md` file. Never auto-commit to git without review.

6. Manage Log Volume

Implement log rotation. Encrypt archives. Delete transcripts older than your retention policy requires. 44GB of plaintext logs is a liability, not an asset.

For Enterprise / Fund Operations

7. Agent Inventory

Know every agent running in your organization. Microsoft's research shows most enterprises lack visibility into their agent population.[10] You can't secure what you can't see.

8. Network Segmentation

Agents should not run on workstations with access to production systems. Isolate agent infrastructure. Use separate credentials with minimal scope.

9. Human-in-the-Loop for High-Stakes

Any action involving financial transactions, code deployment, external communication, or data access over a defined threshold should require human approval.

10. Prompt Injection Defense

Treat all external content as untrusted. Mark it explicitly. Add injection resistance instructions to system prompts. Monitor for behavioral anomalies.

11. Supply Chain Governance

Maintain an approved list of agent skills/plugins. Review code before deployment. Pin versions. Monitor for updates that introduce new permissions.

12. Incident Response Plan

Know what to do when an agent is compromised: token revocation procedures, affected service identification, log preservation, communication protocols.

REALITY CHECK

No defense is complete. Prompt injection success rates exceed 85% against SOTA defenses when adaptive attacks are used.[8] The goal is layered defense that makes exploitation expensive and detection likely — not perfection.

8. Recommendations for Fund Managers

Section confidence: 78% · Based on threat landscape analysis; LP-facing implications involve judgment calls

AI agents are entering fund operations rapidly — deal sourcing, portfolio monitoring, LP communications, compliance workflows. The security implications are directly material to fund governance and LP obligations.

Why Agent Security = Fund Security

- **Data exposure risk.** An agent with access to deal flow data, LP communications, or portfolio company financials represents a single point of compromise for the fund's most sensitive information.
- **Regulatory implications.** Autonomous agents making decisions about data handling, communications, or financial operations create compliance surface area that most fund legal frameworks haven't addressed.
- **Reputational risk.** A compromised agent sending emails on behalf of a fund manager — or leaking LP data through an unaudited skill — is an existential reputational event.

Minimum Standards for Fund Agent Deployments

AREA	MINIMUM STANDARD	BEST PRACTICE
Credential Management	Encrypted storage, monthly rotation	HSM-backed, automatic rotation, zero standing privileges
Skill/Plugin Governance	Approved list only, code review before install	Sandboxed execution, behavioral monitoring, version pinning
Autonomy Boundaries	Human approval for external communications and financial actions	Tiered autonomy with dynamic trust scoring
Data Classification	Agent access scoped to data classification level	Per-task credential issuance, audit logging
Incident Response	Token revocation procedure documented and tested	Automated anomaly detection, automatic containment
LP Disclosure	Agent usage disclosed in operational risk section	Agent security audit results shared with LP advisory committee

LP-FACING IMPLICATION

Funds that adopt AI agents without security governance are accepting unquantified operational risk. The question LPs should be asking is not "are you using AI?" but "what are your agent security controls?" Funds that can answer this question credibly will have a governance advantage in the next fundraise cycle.

9. Methodology & Sources

Overall report confidence: 82%

Research Process

This report was compiled on February 17, 2026, using the following sources and methods:

- **Primary security research:** Koi Security's ClawHub audit (341 malicious skills), Hudson Rock's infostealer analysis, VirusTotal blog post on weaponized skills
- **Enterprise surveys:** Gravitee State of AI Agent Security 2026, Microsoft Cyber Pulse AI Security Report, CyberArk 2026 identity risk analysis
- **Academic research:** arXiv papers on prompt injection in agentic systems (2601.17548, ScienceDirect S2405959525001997)
- **Industry reporting:** The Hacker News, BleepingComputer, Forbes, eSecurity Planet, Stellar Cyber, Zenity threat landscape report
- **Direct observation:** Audit of our own OpenClaw deployment and operational security practices

Confidence Scoring

Each section includes a confidence score reflecting source quality and claim verifiability:

- **90%+** — Claims backed by primary research, confirmed by multiple independent sources
- **80–89%** — Claims backed by credible primary sources with limited independent confirmation
- **70–79%** — Claims involve interpretation or extrapolation from available data
- **<70%** — Speculative or forward-looking claims, flagged explicitly

Limitations

- This report focuses primarily on the OpenClaw ecosystem. Many findings generalize to other agent frameworks, but specific vulnerability details may differ.
- Enterprise survey data (Gravitee, Microsoft) reflects self-reported incidents, which may under- or over-count actual events.
- The prompt injection success rate (85%+) is from academic research using adaptive attacks; real-world success rates may vary based on deployment specifics.
- Our case study (Section 6) reflects a solo operator deployment; enterprise environments will have different risk profiles.

References

[1] Koi Security. "ClawHavoc: 341 Malicious ClawdBot Skills Found by the Bot They Were Targeting." koi.ai/blog, Feb 2026. Confirmed by The Hacker News, eSecurity Planet, VirusTotal Blog.

[2] Hudson Rock. "Infostealer Steals OpenClaw AI Agent Configuration Files and Gateway Tokens." hudsonrock.com/blog, Feb 16, 2026. Confirmed by BleepingComputer, TechNadu, Tech Edu Byte.

[3] Gravitee. "State of AI Agent Security 2026 Report: When Adoption Outpaces Control." gravitee.io/blog, Feb 2026.

[4] eSecurity Planet. "Hundreds of Malicious Skills Found in OpenClaw's ClawHub." esecurityplanet.com, Feb 2026.

[5] The Hacker News. "OpenClaw Integrates VirusTotal Scanning to Detect Malicious ClawHub Skills." thehackernews.com, Feb 2026.

[6] InfoStealers. "AI Agents' Most Downloaded Skill Is Discovered to Be an Infostealer." infostealers.com, Feb 2026.

[7] Ainary internal research. "OpenClaw Research — 2026-02-17." Community patterns analysis.

[8] arXiv 2601.17548. "Prompt Injection Attacks on Agentic Coding Assistants: A Systematic Analysis." Jan 2026.

[9] Marr, Bernard. "When AI Agents Turn Against You: The Prompt Injection Threat Every Business Leader Must Understand." Forbes, Jan 28, 2026.

[10] Microsoft. "Cyber Pulse: An AI Security Report." microsoft.com/security, 2026.

[11] CyberArk. "AI Agents and Identity Risks: How Security Will Shift in 2026." cyberark.com/blog, Dec 2025.

[12] VirusTotal Blog. "From Automation to Infection: How OpenClaw AI Agent Skills Are Being Weaponized." blog.virustotal.com, Feb 2026.

[13] Stellar Cyber. "Top Agentic AI Security Threats in 2026." stellarcyber.ai, Dec 2025.

[14] Zenity. "AI Agent Security: 2026 Threat Landscape Report." zenity.io, 2026.

[15] USCS Institute. "What is AI Agent Security Plan 2026? Threats and Strategies Explained." uscsinstitute.org, 2026.

About the Author

Florian Ziesche is the founder of Ainary Ventures, focused on AI strategy, research, and implementation. He builds and operates the agent infrastructure he writes about — including the systems analyzed in this report's case study. This report reflects practitioner experience, not theoretical analysis.

AI strategy · research · implementation. By someone who built the systems first.

Ainary

AI Strategy · Research · Implementation

For briefings, advisory, or to discuss how this applies to your fund or organization:

ainaryventures.com/contact

AR-037 · February 2026 · v1.0