

Lesson 12 — Audit, Logging and Monitoring AI Usage

You cannot govern AI usage you cannot see. But the wrong kind of visibility — dumping every prompt into a log bucket — builds a new breach target bigger than the original risk. This lesson is about logging the right things.

Audience: Security + Leads

AI OpSec

Audit & Monitoring

Notes: The goal is metadata you can investigate with, not content you have to defend.

What you will learn

- **What to log** — the metadata set that makes incidents investigable
 - **What not to log** — the content that turns your audit trail into a breach target
 - **How to detect policy violations** — "AI was used" vs. "AI was misused"
 - Outcome: you can answer "who used what model, on what data, under which policy" without standing up a prompt lake
-

The metadata-over-content principle

- **Log that it happened** — user X called model Y at time T on data class C, outcome was allow/block
 - **Log the classification** — the DLP tag, the policy that matched, the decision
 - **Do not log the content** — raw prompts and raw responses become a data-at-rest liability the moment they land in a log store
 - Every byte of prompt you keep is a byte an attacker, a regulator, or a subpoena can ask for later
-

What to capture per AI interaction

- **User** — identity, team, role
 - **Tool** — which client: chat UI, CLI, IDE plugin, agent
 - **Model** — provider and model name/version
 - **Timestamp** — start and end
 - **Prompt length** — token or byte count, not the text
 - **Data classification tags** — PII, secrets, source code, customer data
 - **Outcome** — sent / blocked / redacted
 - **Policy version** — which rules were in force at the time
-

What NOT to capture

- **Raw prompt text** — it will contain secrets, PII, and internal plans you did not anticipate
 - **Raw responses** — the model repeats and sometimes amplifies whatever was in the prompt
 - **Attached files** — your audit log should not become a mirror of every document anyone ever uploaded to an AI
 - **Chain-of-thought snippets** — same content risk, plus they are often not meaningful out of context
 - If you would not store it in plaintext in a CRM, do not store it in your AI audit log
-

Enterprise admin consoles

- **ChatGPT Enterprise** — admin API exposes workspace-level audit events
 - **Claude Team / Enterprise** — audit log export for user, tool, and policy events
 - **Gemini for Workspace** — surfaces AI events through the standard Google Workspace audit log
 - Wire them into your SIEM on day one. Do not wait for an incident to discover you never enabled the feed.
 - These feeds are already metadata-only. That is exactly the shape you want.
-

Detection rules: "AI was used"

- **Traffic to provider domains** — `api.openai.com`, `api.anthropic.com`, `generativelanguage.googleapis.com`, etc., from endpoints and egress proxies
 - **CLI agent telemetry** — process-start events for known AI CLIs, outbound connections from those PIDs
 - **IDE plugin pings** — known copilots and AI assistants beaconing home from developer laptops
 - This tier is about knowing *where* AI is in the environment. It is the baseline before any misuse detection is meaningful.
-

Detection rules: "AI was MISUSED"

- **Unapproved provider** — traffic to any AI API that is not on the sanctioned list
 - **High-volume uploads** — a user suddenly sending ten times their normal token count, or multi-MB attachments
 - **Off-hours spikes** — heavy AI usage at 03:00 from an account that never works nights
 - **DLP classification hits** — a sanctioned provider call whose classifier flagged PII, secrets, or customer data
 - Priority order: *unapproved provider > volume anomalies > content-class hits*
-

Retention policy for AI audit logs

- **Long enough to investigate** — 90 to 180 days for metadata covers most incident timelines
 - **Short enough to not become a breach target** — indefinite retention turns your audit store into the next disclosure
 - **Different classes, different windows** — policy decisions and block events can live longer; high-volume detail can age out sooner
 - **Content retention is zero** — there is nothing to retain because nothing was logged in the first place
-

Privacy tradeoff

- Over-logging prompts is not "extra safety" — it is an **insider-risk** and **privacy** liability of its own
 - Staff treat AI like scratch paper. It contains half-formed ideas, personal context, HR situations, draft resignations
 - A prompt archive is a **surveillance-grade record** of what people were thinking. Treat it accordingly — or better, do not build it
 - "Log everything" sounds responsible. It is the opposite of responsible.
-

Rules to remember

- Log metadata, not content. Never stand up a prompt lake.
- Capture: user, tool, model, timestamp, length, classification, outcome, policy version.
- Pipe enterprise admin console audits into your SIEM on day one.
- Detection rule priority: unapproved provider > volume anomalies > content-class hits.
- Retain AI audit logs 90–180 days. Longer is a liability.
- Over-logging prompts is its own privacy and insider-risk problem.

