SOLUTION BRIEF

Runtime Control For Teams Evaluating AI Agents

Pipelock sits between AI agents and the internet and blocks secret leaks, unsafe tool traffic, and prompt-injection responses.

Open-source core · 31,000+ tests · >90% statement coverage · 242 public cases · Single-file Go binary

Why Teams Are Evaluating Agent Egress Controls Now

- Agents now have credentials, tool access, and outbound network reach.
- Model-side guardrails do not see what happens after the model call: MCP tool traffic, HTTP requests, WebSocket frames, or CONNECT tunnels.
- Security teams still need something concrete to evaluate: where enforcement happens, how it fails, and what evidence survives incident review.

What You Actually Deploy

Pipelock is a self-hosted Go binary that sits on the traffic path as a local forward proxy, an MCP wrapper, a Kubernetes companion proxy, or a verdict service for control-plane workflows.

```
Agent (secrets, no network) → Pipelock (inspection + policy) → Internet / MCP / APIs
```

- The agent keeps the credentials.
- Pipelock sees the traffic.
- No SaaS account. No telemetry to a vendor.
- Scanning and policy happen before the request leaves or before tool output re-enters the agent.

What Evaluators Usually Need To Verify

- **Visibility across live traffic** for HTTP bodies, WebSocket frames, and MCP descriptions, arguments, and responses.
- **Boundary enforcement** for DLP, prompt injection, SSRF, and MCP tool policy before traffic leaves.
- **Evidence that survives incident review** through flight recorder logs, Audit Packet v0, signed assessments, posture verification, and mediator-signed action receipts.

65
DLP patterns

29
injection patterns

Fail-Closed By Default

Timeouts, parse errors, and ambiguous state default to block.

Agent-External Attestation

Pipelock emits receipts and Audit Packet evidence from outside the agent trust boundary.

Not self-attested logs from inside the agent process.

Defense In Depth

Sandbox isolates execution
Guardrails shape model output
Pipelock inspects the traffic boundary

Runtime Coverage

- 11-layer scanner pipeline
- 65 DLP patterns and 34 injection patterns
- Request bodies, WebSocket frames, and MCP traffic
- SSRF, metadata, DNS rebinding, tool policy, and rug-pull detection

Deployment Paths

Mode	Where It Fits	Why Teams Use It
Forward proxy	Workstations and servers	Inspect outbound HTTP/HTTPS with optional TLS interception.
MCP proxy	Agents using stdio, HTTP, or WebSocket MCP	Inspect tool descriptions, arguments, responses, and rug-pulls.
Kubernetes companion proxy	Cluster workloads	Generate a companion proxy topology plus NetworkPolicy around the workload.
Host containment	Local and CI agent runs	Install, verify, and remove host network containment around agent processes.
Verdict API	Control plane and CI	Ask for a policy verdict without moving the full traffic path.
Process sandbox	Higher-risk local or server-side runs	Add OS-level containment alongside the traffic boundary.

For the full capability inventory, see pipelab.org/pipelock. The enterprise brief focuses on deployment fit and evidence, not every CLI detail.

Evidence And Auditability

Beyond the proxy decision itself, evaluators get several distinct artifacts:

- **Flight recorder.** Hash-chained audit log of scanner decisions, policy actions, and session events.
- **Audit Packet v0.** Portable run evidence with a public JSON Schema and standalone `pipelock-verifier` CLI.
- **Signed assessments.** `pipelock assess` generates signed HTML and JSON reports with framework mappings.
- **Mediator-signed action receipts.** Optional per-decision records signed from outside the agent trust boundary.
- **Posture verification.** `pipelock doctor`, `pipelock posture verify`, and `pipelock verify-install` separate configured controls from enforceable controls.
- **Public proof surfaces.** Public benchmark results, public docs, public mappings, and published release artifacts.

Pipelock Conductor (Enterprise)

Run Pipelock free on every agent; the Conductor governs the whole fleet from one place.

- **One policy everywhere.** No agent quietly rolls back to an older one.
- **A signed record from every agent.** Check it offline with the free binary.
- **A multi-person kill switch.** Stop the fleet, or one agent; takes more than one operator.
- **Run it for years.** Enroll and retire agents; back up and restore the Conductor's data.

Detection and enforcement stay free; the Conductor sells coordination and governance. Gated on the fleet license. pipelab.org/enterprise.

Serious Engineering, Not A Demo

- 31,000+ tests across scanner, proxy, MCP, and policy paths
- >90% statement coverage
- Public corpus plus adversarial release checks against the production binary
- Cosign signatures, SLSA provenance, SBOMs, and cross-platform validation
- Single static binary, 27 direct Go dependencies

Typical Evaluation Scenarios

- Teams deploying AI coding agents internally
- Enterprises adopting MCP-based tooling across multiple teams
- Security teams reviewing agent egress risk as part of broader governance work
- Control-plane builders who need a verdict service via API

Operator Checks

pipelock doctor reports whether configured protections are enforceable in the current deployment.

pipelock verify-install checks that scanning and containment are wired correctly.

pipelock posture verify turns signed posture capsules into CI-friendly gates with distinct exit codes.

Start An Evaluation

Review the enterprise page, Assess demo reports, and public benchmark first.

If you want to talk deployment shape, logging, or policy fit, use the evaluation contact below.

pipelab.org/enterprise
pipelab.org/assess
pipelab.org/gauntlet
luckypipe@pipelab.org

© 2026 PipeLab