VeriProof

Technical guide for people who build and run agents

The policy check is the easy part.



Business Owner	Governance Engineer	Compliance Officer	Developer	Administrator	Auditor
Value against AI spend, budgets, and risk acceptance	Rules, test datasets, Shadow, staged rollout	Framework coverage and tracked remediation	SDKs, adapters, gateway routes, MCP	Separation of duties, rule approvals, SSO, SCIM	Scoped engagements, signed packs, ledger checks

Getting everything else right separates pilots from production.

Exceptions need an answer from someone with authority, and the application has to report what it actually did before compliance can tie the result to a control.

VeriProof handles the check and everything below it.



The business control plane for AI agents

veriproof.app

Platform overview



An agent action passes through each layer below on its way from the first call to the audit.

Your agents

Connect, page 3

- Framework agents
- Custom-code agents
- Copilot Studio agents
- Computer-use agents
- MCP-connected assistants
- Automated workflows
- Payment agents
- API-connected agents

Enforcement points

Connect, pages 3–4

- **SDKs and adapters** — 6 core SDKs and 35 framework adapters
- **Agent Gateway** — Model, MCP, and agent-to-agent traffic
- **Service API and MCP tools** — Any platform, including computer use

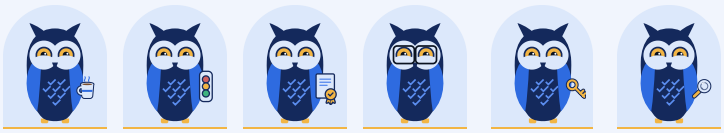
Rules

Enforce, page 5

- **Industry lexicons** — 14 industries, 476 typed fields
- **Rego rules** — Proven in Shadow before they enforce
- Allow
- Warn
- Transform
- Deny
- Escalate

People

Coordinate, pages 6–7



Six roles work from one shared record, and VeriProof hands work between them. Reviewers answer in Teams, Slack, ServiceNow, or email.

Frameworks

Comply, page 8

- **11 frameworks evaluated** — NIST AI RMF, SOC 2, HIPAA, ISO 42001, and more
- **Coverage and remediation** — Each gap gets an owner and a due date
- **Your GRC system** — Vanta, ServiceNow, ComplyAssistant

Record

Prove, page 9

- **Write-once repository** — Sealed with locked retention
- **Proof ledger** — Azure SQL Ledger or Solana
- **Signed evidence pack** — Proofs and a readable chronology
- **Standalone verifier** — No account or API key

Underneath it all

Run, pages 10–11

- **Where it runs** — Managed SaaS, or Private Enterprise in your own Azure
- **Who can act** — SSO, SCIM, service principals, and workload identity
- **How you drive it** — CLI, MCP server, Data access API, and your monitoring tools

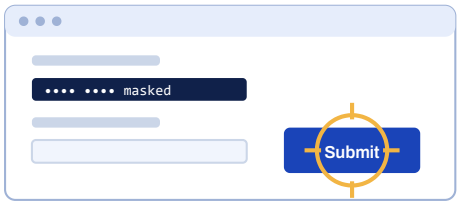
Connect any agent, however it was built.

Pick the path that suits each agent, and the same rules, reviews, and evidence apply.

```
await vp.govern(
  "claim.pay", claim,
  issue_payment,
  facts=facts)
```

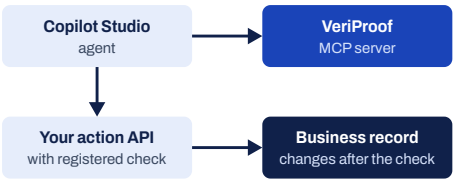
SDK and framework agents

Wrap the operation in `govern()` or add your framework's adapter, and each model call, tool call, or task is checked before it runs.



Computer-use agents

Browser and desktop agents check each click or entry before making it, and the record keeps a masked view of the screen behind it.



Microsoft Copilot Studio

Connect through MCP to find records and draft rules, and each confirmed business action is checked at the API that owns the record.

MCP-connected assistants
Proposed actions, held to the signed-in person's own access.

Automated workflows
The refund, account change, or disclosure, before the step runs.

Payment agents
AP2, pay.sh, x402, and stablecoins, before the wallet or provider moves money.

Any other platform
The service API, or the Agent Gateway on the next page.

What happens on every governed call

1. **Attach facts** — The application passes typed business facts, so rules never have to read prompt text.
2. **Evaluate first** — The check runs before the operation, so a stopped run never reaches your code.
3. **Apply the verdict** — The call continues, continues corrected, waits for a person, or stops.
4. **Report the outcome** — The application reports what happened, and the evidence goes to the record.

6 core SDKs and 35 adapters, and every adapter enforces

Language	Adapters	Frameworks and libraries
Python	8	Anthropic SDK, CrewAI, Google ADK, LangGraph, LlamaIndex, Microsoft Agent Framework, OpenAI Agents, Pydantic AI
TypeScript	11	Anthropic SDK, Express, Fastify, Genkit, Google GenAI, LangChain.js, LlamaIndexTS, Mastra, Next.js, OpenAI Agents, Vercel AI SDK
.NET	5	AutoGen, Microsoft Agent Framework, OpenAI .NET, Prompty, Semantic Kernel
Java	6	LangChain4j, Micronaut LangChain4j, OpenAI Java, Quarkus LangChain4j, Semantic Kernel Java, Spring AI
Go	3	Anthropic Go, LangChainGo, OpenAI Go
Rust	2	async-openai, Rig

Put the gateway in the path when an agent can't take an SDK.



Running beside your applications as a signed container, the Agent Gateway applies the same rules, reviews, and evidence to model, MCP, and agent-to-agent traffic.



Controls that need no classifier

Approved destinations
Each route goes to one approved place and nowhere else.

Approved models
A request has to name a model on that route's approved list.

Credential holding
Provider keys stay in the gateway and never reach your agents.

Pinned tool catalogs
A tool that changes is blocked until someone reviews it.

Argument and result validation
Calls and results have to match the schemas you pinned.

Action classes
You decide which tools read, write, send, or run code, and an upstream server doesn't get a say.

Work limits
Each request is capped on size, time, tokens, and spend.

A locked-down container
The image is pinned by digest and runs read-only with every capability dropped.

How it behaves

Detectors feed your rules
Prompt Shields, PII detection, and local scanners report what they find, your rules make the call, and a route that requires a detector stops without one.

Streaming
Protected output waits until every check passes, while interactive text streams right away and holds back any tool call until it's authorized.

Agent-to-agent handoffs
Each delegation carries limits tied to the person who started the work, and an agent passing work along can only narrow them.

When the network drops
Active work continues on its signed lease, evidence waits in a local journal, and a confirmed stop stays in place.

Built to deploy
Run it on AKS or Azure Container Apps, and use the published capability matrix to see what each protocol enforces.

Rules in your industry's terms go live only once they've proven themselves.



A vocabulary for your industry

A rule learns little from a tool call, but a typed field can tell it this claim is `approved_partial_payment`. VeriProof ships 14 industry lexicons with 476 typed fields, built on standards such as ACORD, HL7 FHIR, and ISO 20022.

`ClaimAdjudicationOutcome`

Values `approved_full_payment, denied_policy_exclusion, denied_fraud_suspected, and 18 more`

A rule in the guided builder

Runs	Before action
When	Payment Amount is greater than 10,000
Outcome	**Escalate**

The tests saved with it

12,500	**Escalate**
10,000	Allow
8,000	Allow

Each drafted rule comes with positive, negative, and boundary tests, and `veriproof policy test` replays them from your CI pipeline.

14 industry lexicons

- Banking & financial services
- Mortgage & loan processing
- Insurance: P&C, life & annuity
- Healthcare & life sciences
- Telecom & IT operations
- Manufacturing, supply chain & logistics
- Higher education
- Energy & utilities
- Government & public sector
- Legal & regulatory technology
- HR & workforce management
- GRC, ESG & enterprise risk
- Professional services & consulting
- Agentic payments & stablecoins

Plus a shared governance vocabulary

An engine you can reason about

Rules you write in the guided builder or directly in Rego compile to WebAssembly, and the same signed bundle runs in VeriProof and on the Agent Gateway.

An action whose evaluation fails doesn't run, and no setting can change that.

Five outcomes, strongest first

Deny	Stop
Escalate	Wait for a person with authority
Transform	Continue with a corrected version
Warn	Continue with a warning
Allow	Continue as submitted

A release path for every rule

1 Draft — Describe the intent, or start from a control or industry template.

2 Test — Check its verdicts against sample cases.

3 Shadow — Compare proposed results with real activity before the rule goes live.

4 Approve — Get sign-off from up to two independent approvers.

5 Enforce — Roll out by application or group, and any edit to an active rule sends it back to Shadow.

And a way to stop. Emergency stops halt a fleet or a whole tenant from the portal or API, and each stop is recorded.

Follow one payout through every team that answers for it.



An insurance agent proposes a payout above the automatic limit, and ten steps later the money has moved and an auditor can check how it happened.

Claim	Proposed payout	Limit for automatic payouts
CLM-20931	$18,400	$10,000

Before the agent runs



Governance Engineer

1 Writes the rule
Sets payouts over $10,000 to wait for a person, and tests the rule on sample cases and in Shadow.



Developer

2 Sends the missing fact
A fix request asks for `payment_amount` and closes when the data arrives.



Administrator

3 Approves the tested version
An Administrator or Compliance Officer signs off before it reaches production.

When the agent acts



Agent and VeriProof

4 Agent proposes the payout
`govern("claim.pay")` sends $18,400 on claim CLM-20931.

5 Verdict: Escalate
The rule matches, so the action waits.

A Reviewer decides on page 7

One connected record 1 2 3 4 5 6 7 8 9 10

Every role on one record

Each role works the same records from a view built for its job, and one person can hold several roles.

Handoffs the system runs

When work passes to another team, VeriProof routes it to the right person and keeps it open until it's done.

Long after the money moves, an auditor can still check who approved it.



Nobody has to chase a thread to learn what happened, because VeriProof carries the work from one team to the next.

From page 6: the payout is waiting

When the agent acts



Reviewer

6 Lowers the amount

Checks the facts on the Teams card and approves $15,000 instead of $18,400.



Agent and VeriProof

7 The action runs

A signed outcome comes back, and the app pays the approved $15,000 and reports it.

After it acts



Compliance Officer

8 Tracks the evidence

The decision counts toward mapped controls, and any failed control becomes a fix with an owner and due date.



Business Owner

9 Decides on the risk

When Compliance proposes accepting a risk, the Business Owner decides, and approved risks come back for review.



Auditor

10 Verifies the record

Works in an engagement scoped to these applications and dates, and checks the pack.

One connected record 1 2 3 4 5 6 7 8 9 10

Approval for the exact action

An approval covers only the action the reviewer saw. A review that stalls goes to the people who answer for that application, and if time runs out, the action doesn't run.

Wherever the reviewer answers

Reviews reach people in Slack, Microsoft Teams, ServiceNow, email, and the VeriProof Inbox, and the decision lands in the same record wherever they make it.

Agent activity becomes framework evidence as it runs.



Your Compliance Officer sees coverage and gaps for each application's frameworks, worked out from the decisions, reviews, and outcomes your agents record as they run.

Recorded decisions and reviews → Controls evaluated automatically → Framework coverage and gaps → **Remediation and GRC delivery**

11 frameworks evaluated automatically

NIST AI RMF	Model versions, policy settings, outcomes, and review decisions
SOC 2	Six criteria, from attributed and verified activity
HIPAA	Security Rule evidence from access grants, archived sessions, and recorded reviews
PCI DSS	Records for the payment workflows you identify
NIST CSF 2.0	Monitoring, event analysis, and access permissions
FDA SaMD	Workflow cases, experiment summaries, and changes with their linked retests
Agentic Trust Controls	Vanta's open set of 65 controls, with evidence for VeriProof's part of 24
ISO/IEC 42001	Six requirements, from recorded activity and reviewed organization evidence
ISO/IEC 27001	Logging and monitoring activities
GDPR	Seven requirements, from declared data categories, access, and rule checks
EU AI Act	Four Article 26 deployer obligations

Canadian clients also get the decision records that OSFI Guideline E-23 and Quebec Law 25 call for.

Targets for each application

Set a target score per framework, and see where each application stands for any period.

Evidence from your organization

Controls that depend on your organization take reviewed evidence alongside recorded activity.

Gaps with an owner

A failed control becomes a fix with an owner and a due date, and it stays open until the fix lands.

Assessments that hold their period

A saved assessment seals that period's activity, so a later report shows what was true at the time.

Delivered to your GRC system

Vanta
Access records, remediation tasks, and evidence reports.

ServiceNow
Evidence requests, governance records, and remediation.

ComplyAssistant
Approved reports, filed under the right document types.

Every decision leaves evidence an auditor can verify independently.



The record of each decision journey is sealed and anchored to your proof ledger, and anyone you give the signed pack to can check it from a terminal.

1 Record — Policy decisions, authorized reviews, and reported outcomes for each decision journey.

2 Seal — Write-once storage, with encryption and locked retention.

3 Anchor — SHA-256 fingerprints, written to your proof ledger.

4 Export — Proofs and a readable chronology, in a signed evidence pack.

5 Verify — Anyone with the pack and public key runs the standalone verifier, with no account or API key.

Choose the proof ledger that fits

	Azure SQL Ledger	Solana
Best for	Enterprises that keep verification private	Web3 and stablecoin programs that want a public reference
How someone checks	With read-only access to the ledger	By inspecting the public transaction
What leaves your environment	Nothing reaches a public network	Only fingerprints, never decision content or personal data
Available on	Managed SaaS and Private Enterprise	Managed SaaS

Inside the evidence pack

The pack is a ZIP your auditor can open and read. Its file inventory is signed with ECDSA P-256, and every file in it is fingerprinted with SHA-256.

```
chronology.pdf                          The readable chronology
manifest.json                           What this export includes
sessions/                               Each recorded session
policy-evaluation-decisions.json        Each policy decision
integrity-proof.json                    Signature and ledger proofs
verification-scope.json                 What the proofs cover
README.txt                              How to verify the pack
```

Verify from a terminal

On Windows, Linux, or macOS, the verifier confirms that

- the pack holds exactly the files that were signed
- every file matches its signed hash and length
- each session's inclusion path leads to its anchor
- the anchor is on your proof ledger

```
$ veriproof-verify --pack evidence-pack.zip \
    --trusted-evidence-public-key \
    evidence-signing-public.pem

verified
```

Only `verified` exits with 0, so a pipeline can gate on the result.

Run it as a service, or entirely inside your own Azure.



Both options run the same product and differ only in where it lives and who operates it.

Where each piece lives

Managed SaaS

VeriProof's Azure
- Role portals and APIs
- Policy and review services
- Evidence repository
- Azure SQL Ledger or Solana

Your environment
- Your agents with SDKs or the Agent Gateway

Private Enterprise

Your Azure subscription
- Role portals and APIs
- Policy and review services
- Database and key vault
- Evidence repository
- Azure SQL Ledger
- Your model deployments
- Your agents with SDKs or the Agent Gateway

	Managed SaaS	Private Enterprise
Who operates it	VeriProof	Your team
Where it runs	VeriProof's Azure environment	Your Azure subscription, separate from SaaS
Proof ledger	Azure SQL Ledger by default, or Solana	Azure SQL Ledger in your subscription
AI assistance	Managed AI analysis included	Your Microsoft Foundry or Azure OpenAI models
Keys and secrets	Held by VeriProof inside the service	In your Key Vault, under your policy
VeriProof staff access	Role-based, with every staff action logged	None, standing or temporary
Azure infrastructure	Included	Billed by Azure to your subscription
How you get it	Start free at veriproof.app/start	Directly from VeriProof, installed in your subscription

Read the source

The SDK and platform source is available on request under the VeriProof license, to inspect for a security review or, on Private Enterprise, to modify and run.

Identity and access

Sign in with Microsoft Entra ID or Okta, provision with SCIM, and scope access down to one application or audit engagement.

Fleet controls

Preview a change across the fleet, roll it out in stages, or stop everything in an emergency, and set a monthly AI budget for each application with an alert level.

Drive it from a terminal, an assistant, or your own systems.



Your team can script the routine work, hand investigation to an assistant, and feed records to your own systems, with the same permissions behind each one.

The CLI

```
veriproof doctor
veriproof apps list
veriproof policy test \
  --application $APP \
  --junit rules.xml
veriproof sessions export \
  $SESSION --output pack.zip
veriproof verify pack.zip
```

Script your setup, run policy regression tests, and export sessions and evidence.

The MCP server

```
get_my_context
search_sessions
get_session_integrity_record
list_alerts
submit_review_decision

# 40 tools, scoped to
# the person asking
```

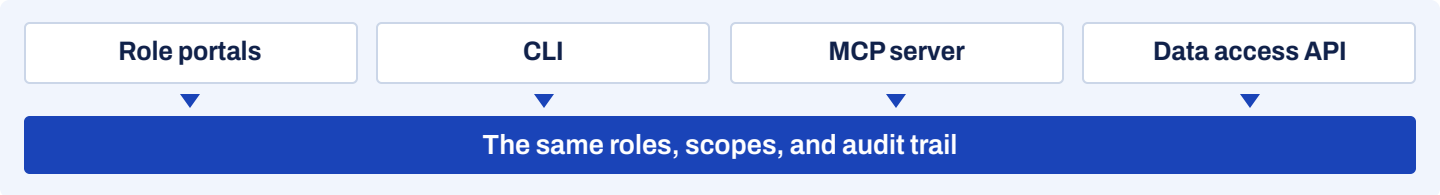
Investigate sessions and draft rules from your assistant, and anything it changes asks for confirmation and goes on the audit trail.

The Data access API

```
data.json
report.pdf
manifest.json

# ECDSA P-256 signature
# daily or weekly snapshots
# every read is logged
# OpenAPI contract
```

Pull inventories, oversight records, and assessments into your own systems, and read session evidence on demand.



Connects to the systems you run

Human review	VeriProof Inbox, Microsoft Teams, Slack, ServiceNow, and email.
Observability	Policy decisions appear in your OpenTelemetry and Langfuse traces, and the VeriProof app for Splunk adds dashboards and seven alerts on each agent's value and AI spend.
Your own systems	A signed webhook reaches your system whenever a review starts, changes, or finishes.
GRC	Vanta, ServiceNow, and ComplyAssistant.
Agent management	Microsoft Agent 365, with VeriProof adding business-rule checks and human review.
Your guardrails	Keep the guardrail you trust, and send its verdict along with the policy check.

VeriProof

Technical guide

Try it on one agent.

1

Connect an agent you run.

Add an SDK or adapter, or point it at a gateway route.

2

Put one rule in Shadow.

Compare its proposed verdicts with real activity before you enforce it.

3

Follow a decision through.

See it from each role's view, then export the pack and verify it.



Agents know the work.
VeriProof knows your business.

Read the docs

SDK references, the API, and the capability matrix.

docs.veriproof.app

Talk to us

Ask about a pilot or a security review.

veriproof.app/contact

Start free for 60 days

veriproof.app/start/

Or see an agent at work first at veriproof.app/demo/agent-at-work

Product names belong to their owners. © 2026 RJR Labs LLC. VP-T01, Nov 2026.