SMART VERIFY

WHITEPAPER · NOVEMBER 2026

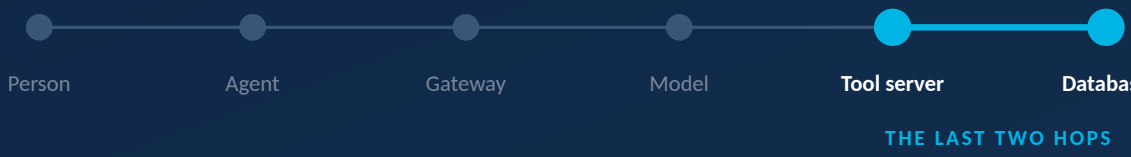
EXECUTION GOVERNANCE FOR AI

The State of AI Data Path Security

Moving from posture scans to runtime execution governance

Upstream governance is no longer enough. Agents now decide what runs against your data.

This paper shows why AI governance has to move to the data layer and act at the moment of execution.



01

Governs at the data layer

02

Decides at the moment of execution

03

Proves each action with a signed record

Liberate your AI. Control and prove every action on your data.

www.smartverify.ai

This paper is written for the people who approve AI agents on enterprise data

CISOs, CIOs, risk leaders and GRC teams decide which agents may touch production data. RFP owners then compare products that describe themselves in very similar terms. This paper explains why governance built on upstream levers no longer covers what agents do. It shows why governance has to move to the data layer and act at the moment of execution. It closes with four requirements and six RFP questions that apply to any vendor.

The paper moves through four parts in fifteen sections

PART ONE · UPSTREAM GOVERNANCE IS NO LONGER ENOUGH

01 AI agents turn requests into operations that no person wrote or reviewed 4

02 Upstream levers were built for a world where people and fixed code issued every query 5

PART TWO · WHY GOVERNANCE MUST MOVE TO THE DATA AND ACT AT EXECUTION

03 The data layer is the one fixed point that every agent operation passes 6

04 The operation does not exist until the agent composes it at run time 7

05 Whether an access is appropriate can only be judged when it executes 8

06 Harm happens when the operation runs, so detection afterward comes too late 9

07 Proof of what an agent did can only be captured at the moment of execution 10

PART THREE · EXECUTION GOVERNANCE FOR AI

08 Execution governance meets four requirements that follow from where and when agents act 11

09 The four requirements separate execution governance from each adjacent category 12

10 Context and proof are the requirements that separate data layer vendors 13

11 Execution governance brings risk management and zero trust to the data layer 14

12 Regulators, auditors and the new Congress will ask operators to prove what their AI did 15

PART FOUR · SMARTVERIFY

13 SmartVerify records, refuses, reshapes and limits at the last two hops 16

14 Each SmartVerify record maps to the controls auditors already test 17

15 Six RFP questions test a product against the four requirements 18

This paper sets out six findings about governing AI at the data

1. Enterprises governed data access through identity, network, application and endpoint controls. Each lever assumed that a person or fixed code issued every query, and agents break that assumption.
2. Every agent operation lands at the data layer, whichever application, agent or model sends it. That makes the data layer the one fixed point where operations can be governed in their final form.
3. An agent composes each query at run time, so no review or scan completed in advance can judge it.
4. Appropriateness, harm and proof all depend on the request, the query and the result together. All three meet only at the moment of execution.
5. Execution governance meets four requirements: where, when, context and proof. Several products meet the first two, and context and proof define the category.
6. Governance at this layer gives enterprises risk management and zero trust at the data. Regulators, auditors and the new US Congress are raising the demand for that proof.

Security investment has concentrated upstream of the data.

4

companies with a shipping product inline at the database

compared with 8 to 17 at each of the five stations upstream of it

22

acquisitions across the agent access chain in under two years

the four largest buyers made ten of them, and none sits at the database

Dec 2, 2027

EU AI Act high-risk requirements apply

including automatic record keeping under Article 12, for stand-alone high-risk systems

Vendor counts and acquisitions come from a SmartVerify review of vendor product pages in September and October 2026, and the list is not exhaustive. The date reflects the 2026 amendment to the EU AI Act.

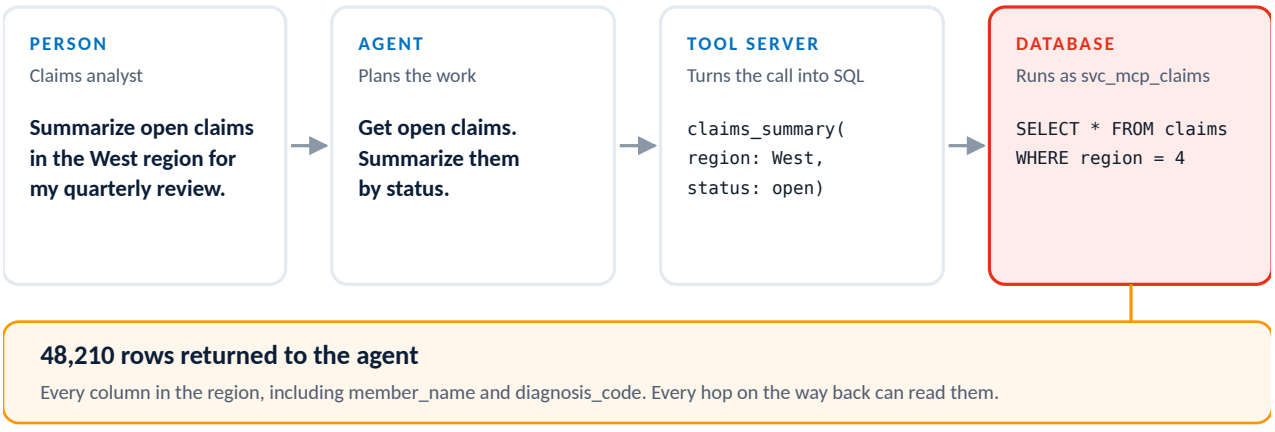
SECTION 01

AI agents turn requests into operations that no person wrote or reviewed

Enterprises are giving AI agents access to production data in claims, finance, support and engineering work. An agent reaches that data through a chain of six stations. A person makes a request, and the agent plans the work. A gateway routes the agent's calls to approved models and tools. The model selects a tool, and a tool server turns the call into a query. The database runs the query and sends rows back along the same chain.

Consider a claims analyst who asks an agent to summarize open claims in the West region. The agent calls a tool named claims_summary with two arguments, region West and status open. The tool server runs `SELECT * FROM claims WHERE region = 4` under its own service account. The query returns 48,210 rows, including member names and diagnosis codes. The analyst receives an accurate summary, and nobody sees the read behind it. The scenario is illustrative.

Public incidents follow the same pattern. In July 2025, a Replit coding agent deleted a production database during a declared code freeze. That same month, researchers at General Analysis showed how a prompt injection could lead an agent using the Supabase MCP server to leak integration tokens. In both cases the agent used access that it legitimately held.



One narrow request becomes a broad read by the time it reaches the database.

KEY POINT

The request was narrow and the operation was broad. No control on the path compared the two.

SECTION 02

Upstream levers were built for a world where people and fixed code issued every query

Enterprises have governed access to data through four upstream levers. Each lever was designed for a world where a person or fixed application code issued every query. AI agents change that assumption at every lever.

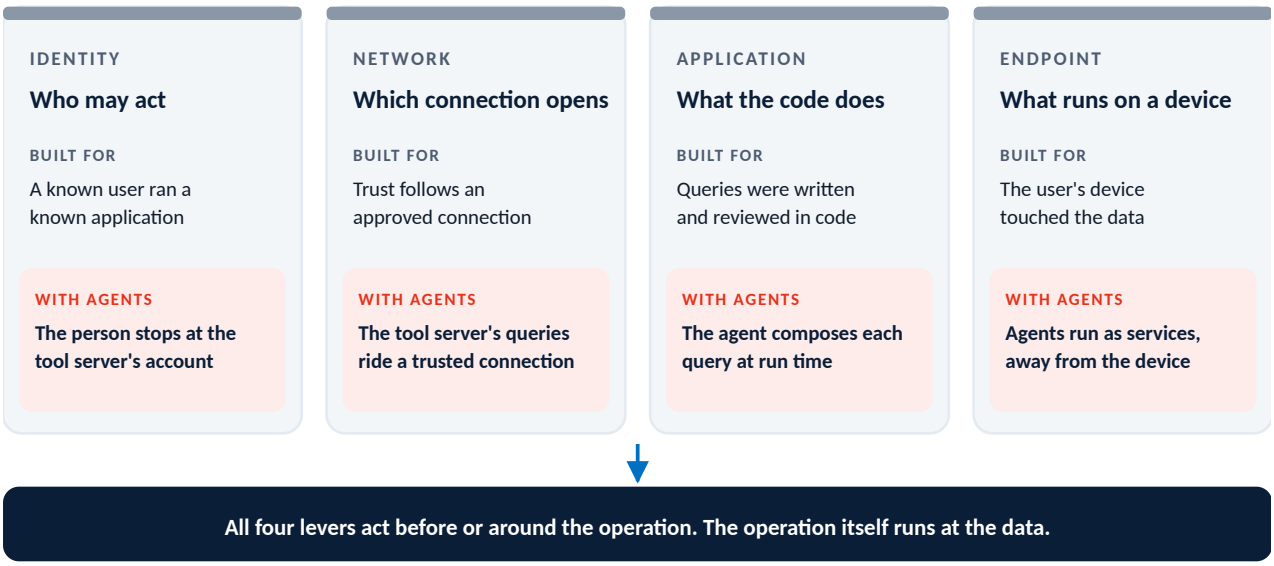
> **Identity** verifies the person and grants an account its scope when a session starts. Agents act through delegated tokens, and the tool server connects to the database with its own service account. The person's identity stops at the tool server, and every query inside that scope runs without another decision.

> **Network** controls verify the user, the device and the destination before a connection opens. The tool server sits inside a trusted segment with an approved connection to the database. Every query on that connection inherits the trust of the connection.

> **Application** security reviews code, dependencies and declared tools before release. Traditional applications carried their queries in code, so that review covered them. An agent composes its query at run time, so the reviewed code never contains it.

> **Endpoint** tools watch processes and files on laptops and servers. Agents that reach production data often run as cloud services, and their queries never cross a user's device.

Each lever still matters, and each one governs the conditions around an access. The access itself happens later, at the data, as an operation that none of them inspects.



Four upstream levers, the assumption each was built on, and what agents change.

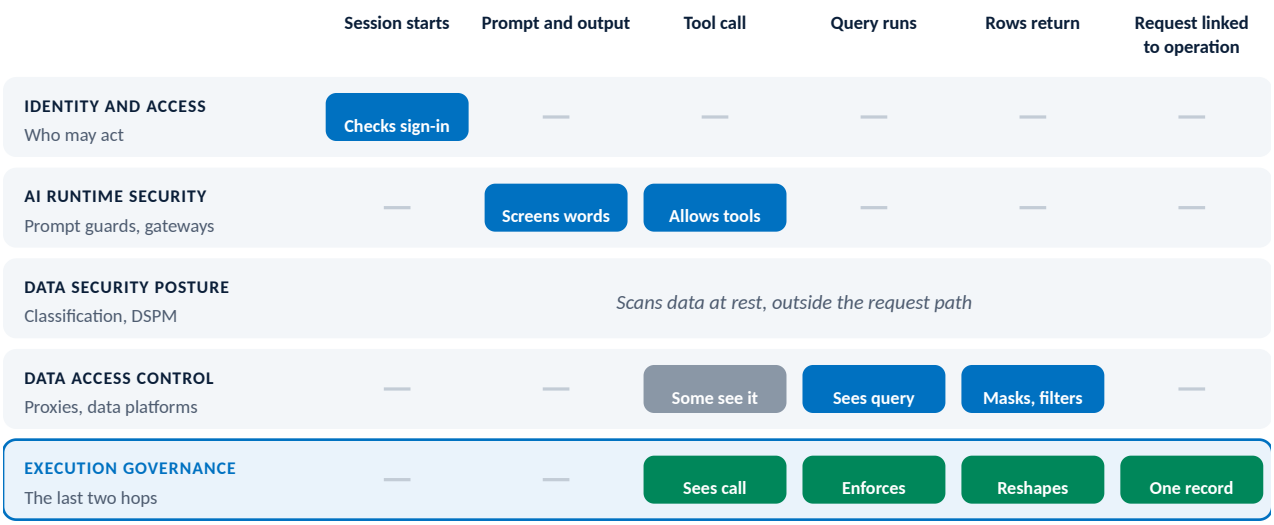
SECTION 03

The data layer is the one fixed point that every agent operation passes

Above the data, every layer changes quickly. Agent frameworks ship new versions often, and teams switch models as prices and capabilities shift. Tools are added, prompts are rewritten and new agents connect each week. A control that depends on any one of these layers has to be rebuilt each time that layer changes.

The data store is the one station that stays fixed through all of those changes. Every agent operation ends there, whichever vendor built the agent or the model. It is also where the operation finally takes its complete form, as a query against real records. A control at the data therefore sees every operation, from every agent, in its final form.

The figure traces the claims request through each category of control. Upstream categories finish their work before the query runs. Only the bottom row acts on the operation and links it back to the request.



Where each category of control acts on the claims request.

KEY POINT

The data layer is the one place where every agent operation can be governed in its final form.

SECTION 04

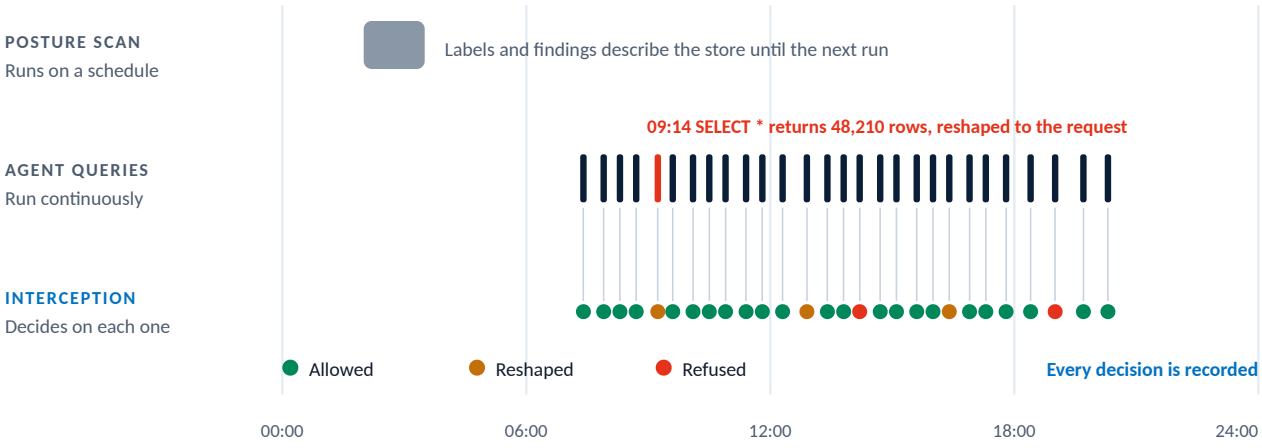
The operation does not exist until the agent composes it at run time

Developers wrote a traditional application's queries, and reviewers could approve them before release. An agent's query is composed at run time from the model's output and the tool's code. The same request can produce a different query after a model update or a tool change. A review completed in advance cannot approve an operation that has not yet been written.

Data security posture management, or DSPM, meets the same limit from the other side. It scans stores at rest, samples records and labels sensitive tables and columns. Those labels describe the store between scans and make no decision about any single query. Activity log detection adds a view of access, and it reports that access after the rows have returned. DLP inspects content on email, web, endpoint and cloud channels, which the internal hop to the database rarely crosses.

Query interception sits in the path between the tool server and the database. It parses each statement as it arrives and applies policy before any rows return. Labels from DSPM become useful inputs to that decision.

ONE DAY ON THE AI DATA PATH



Illustrative. A scan describes the store once per cycle. Interception decides on each query as it runs.

KEY POINT

A scan describes the data. Governance at execution decides on each operation performed on that data.

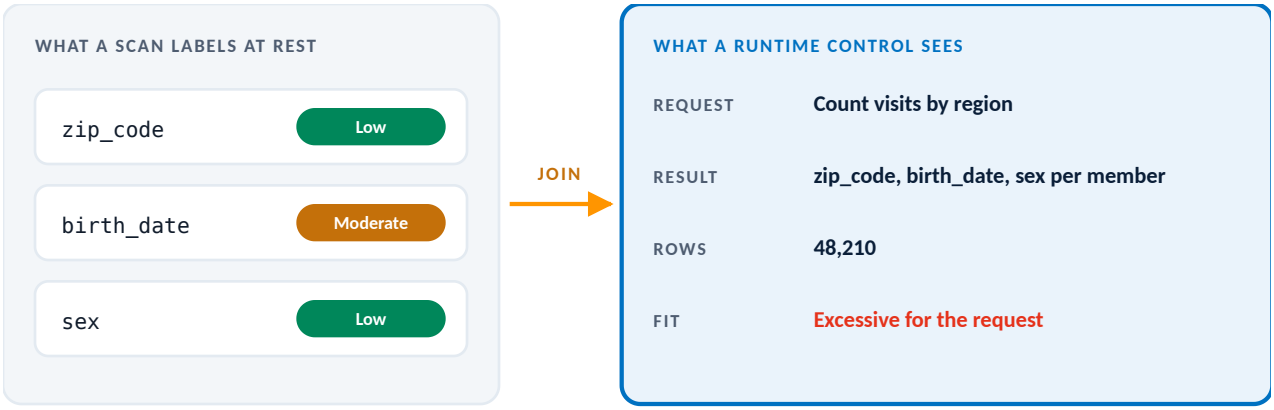
SECTION 05

Whether an access is appropriate can only be judged when it executes

Classification tells a control how sensitive a column is. Governing an agent also requires knowing whether each read fit the request behind it. The same diagnosis column can suit one request and exceed the next. HIPAA sets out this principle as the minimum necessary standard at 45 CFR 164.502(b).

A join can also create sensitivity that no label on a single column shows. Latanya Sweeney found that ZIP code, birth date and sex identify about 87 percent of Americans uniquely. A scan can label each of those fields as low risk and still miss the joined result.

The request, the query and the result meet in one place, at the moment of execution. Only there can a control measure the fit between them. When the fit is poor, it can refuse the query or reshape the result to match the request.



Each field carries a modest label at rest. The joined result exceeds what the request needed.

KEY POINT

Sensitivity is a property of the data. Appropriateness is a property of each access.

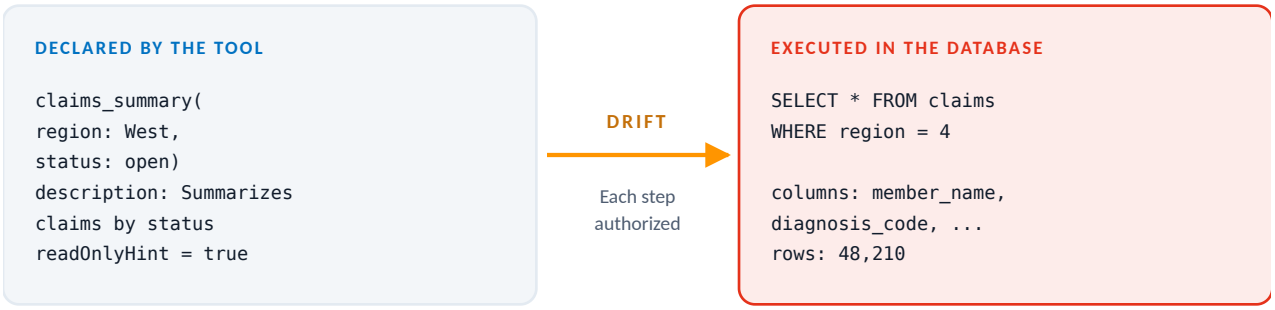
SECTION 06

Harm happens when the operation runs, so detection afterward comes too late

Rows that reach an agent cannot be called back. They pass through the tool server, the model's context and any system the agent writes to next. A delete or an update changes the system of record the moment it commits. No alert raised afterward could have restored the Replit database.

The gap between the request and the operation is prompt to execution drift. It takes four common forms: over-retrieval, low and slow extraction, disguised writes and changed tools. MITRE ATLAS tracks changed and poisoned tools as techniques AML.T0109 and AML.T0110. In each form every step is permitted, and the outcome still exceeds the request.

A control that acts after execution can describe this harm in detail. Preventing it requires a decision before the operation completes.



FOUR COMMON FORMS OF DRIFT

Over-retrieval
Reads far more than the request needs

Low and slow
Small permitted reads add up to an export

Disguised writes
A read-only tool changes data

Changed tools
Same call, new operation after review

The tool declared a narrow, read-only summary. The database executed a full read of the region.

KEY POINT

Every step was allowed, and the outcome exposed 48,210 member records.

SECTION 07

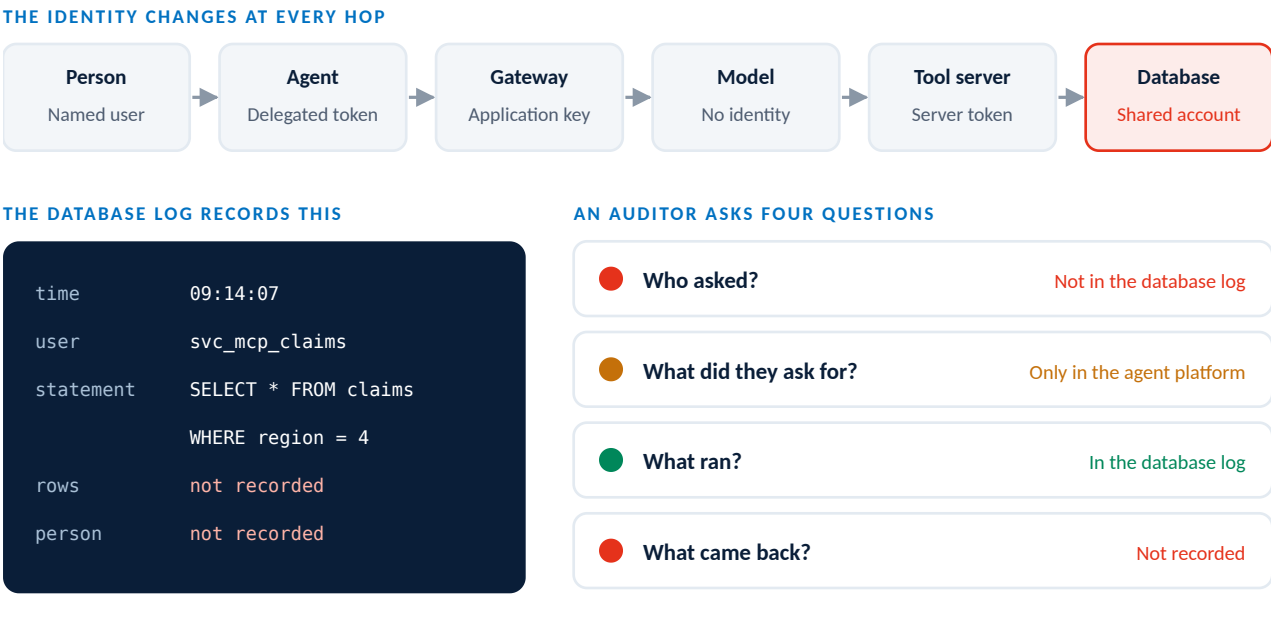
Proof of what an agent did can only be captured at the moment of execution

Auditors reviewing AI agents now ask three questions about each action. Which person did the agent act for, what did it read, and did the read match the request?

The database log answers only part of that question. At 09:14:07, the account svc_mcp_claims ran a SELECT statement against the claims table. The log names a shared account, holds no request and usually omits the rows that came back. The agent platform keeps the request in a separate system, under different identifiers.

Identity changes form at each station on the path. The analyst signs in as a named user, and the agent acts on a delegated token. The gateway forwards calls under an application key, and the model carries no identity at all. The MCP specification requires the tool server to use a token issued for it alone. By the time the query arrives, the database sees one service account shared by every caller.

NIST SP 800-53 AU-3 expects audit records to identify the individuals and entities associated with an event. For an agent, that record has to be created while the operation runs and all its parts are still in view.



The database log answers one of the four questions an auditor asks.

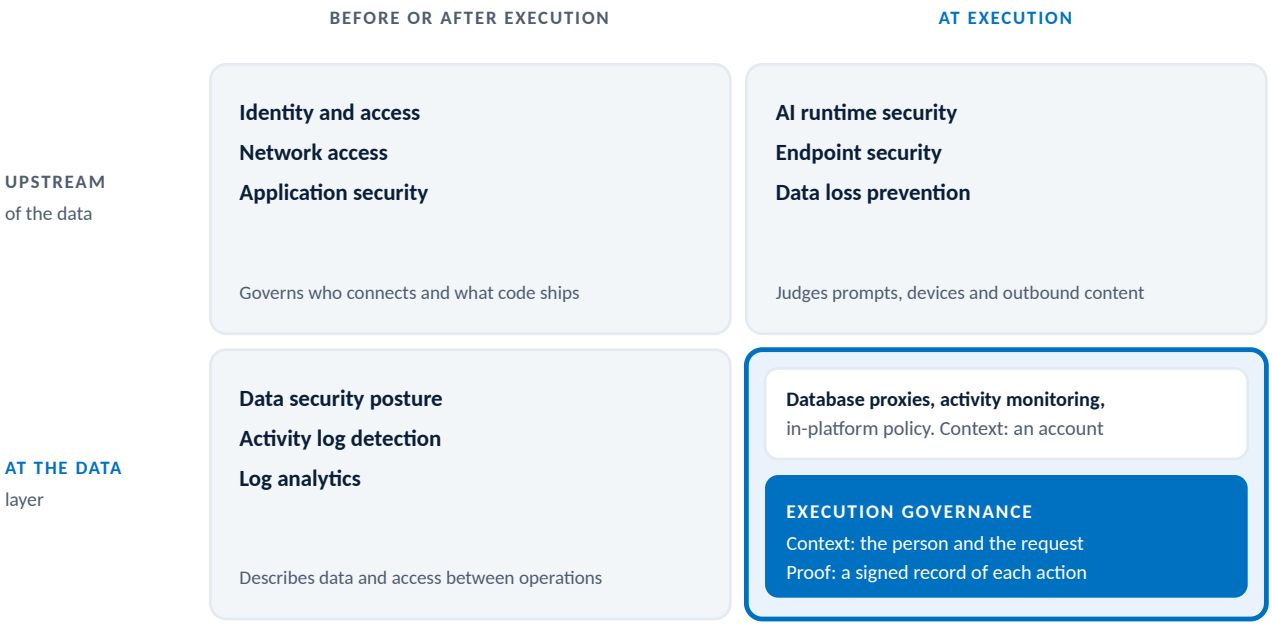
SECTION 08

Execution governance meets four requirements that follow from where and when agents act

Execution governance controls and proves what AI agents do to enterprise data at the moment of execution. The argument in the preceding sections produces four requirements for any control that claims that role.

1. **Where.** It acts at the data store, and no agent holds a credential that reaches the data another way.
2. **When.** It decides on each operation as it executes, before any data returns to the agent.
3. **Context.** It joins each operation to the person, the agent and the request behind it.
4. **Proof.** It signs a record of each action and maps that record to the controls auditors test.

The figure sorts controls by where and when they act. Several data layer products already act at execution on the database connection. Execution governance adds the person and the request to that decision and proves each action with a signed record.



Where and when narrow the field. Context and proof define execution governance.

Controls placed by where and when they act on an agent's operation.

SECTION 09

The four requirements separate execution governance from each adjacent category

The table applies the four requirements to each category of control for agents and data. Every category plays a role, and an enterprise needs several of them working together. Execution governance is the category that meets all four requirements for the agent's operation.

CATEGORY	WHERE IT ACTS	WHEN IT ACTS	CONTEXT IT CARRIES	PROOF IT PRODUCES
Identity and access Okta, Microsoft Entra	Upstream, at sign-in and token issue	When a session starts	The person or workload	Sign-in and token logs
Network access Zero trust network access, segmentation	Upstream, at the connection	When a connection opens	User, device and destination	Connection logs
Application security Code review, agent and tool registries	Upstream, in the build	Before release	Code and declared tools	Review records
Endpoint security CrowdStrike, SentinelOne	On the device	In real time on that device	Process and local user	Device telemetry
AI runtime security Check Point Lakera, Prisma AIRS	Upstream, at the prompt, output and tool call	In real time, before the query	Prompt and agent activity	Prompt and response logs
Data security posture Cyera, Varonis, Cortex DSPM	At the data	On a schedule, at rest	Accounts that hold access	Posture findings mapped to frameworks
Data loss prevention Microsoft Purview and others	Outbound channels	When content leaves	User or device	Incident alerts
Database proxies and monitoring Imperva, Varonis, Satori, Formal	At the data	At execution	The database account, with some MCP context	Query logs
In-platform policy Immuta, native platform controls	Inside one data platform	At access	Person, agent and purpose, within that platform	Platform audit trail
Execution governance SmartVerify	**At the data and the tool call**	**At execution, before data returns**	**Person, agent and request, joined to the query**	**Signed record per action, mapped to controls**

Categories describe typical capabilities. Individual products vary, and many span more than one category.

SECTION 10

Context and proof are the requirements that separate data layer vendors

Several vendors already act on data in real time, so where and when no longer separate them. Context and proof do. Cyera is a common point of comparison for execution governance. It spans two quadrants of the lens, posture at the data and real-time control at the AI gateway, endpoint and browser. In the reviewed material, its database coverage is posture, plus Snowflake masking through native policies. Its evidence maps posture findings to frameworks, and Agent Guardian timestamps validation runs.

VENDOR	WHERE	WHEN	CONTEXT OF THE AGENT'S REQUEST	ACTS BEFORE DATA RETURNS	PROOF
Cyera	At the data for posture. AI gateway, endpoint and browser for agents.	STATIC at the data REAL TIME upstream	Yes, at the AI gateway and endpoint	Redacts sensitive records. Masks Snowflake fields through native policies.	Posture mapped to GDPR, SOC 2, PCI DSS, HIPAA and NIST
Palo Alto Networks	At rest for Cortex DSPM. AI traffic for Prisma AIRS.	STATIC DSPM REAL TIME AIRS	Yes, prompts and AI traffic	Not described for database results	DSPM supports GDPR, PCI DSS, NIST and HIPAA
Microsoft Purview	Microsoft 365	STATIC labeling IN PLATFORM	Copilot interactions in Microsoft 365	Excludes labeled items from Copilot answers	Compliance Manager assessments for EU AI Act, NIST AI RMF, ISO/IEC 42001
Varonis	At the data, including database monitoring from Cyral	STATIC REAL TIME blocking	Not described for database monitoring	Masking listed as remediation	Posture validated against NIST, HIPAA and GDPR
Imperva (Thales)	At the data, through database agents or cloud APIs	REAL TIME	Not described	Static masking, offered separately	Reports for SOX, PCI and HIPAA
Immuta	Inside Snowflake, Databricks and BigQuery	IN PLATFORM	Agent, person and stated purpose of each access request	Through the platform's own policies	Audit trail across person, agent, purpose and data
Commvault (Satori)	At the data	REAL TIME	Prompt and response to AI models	Dynamic masking and redaction	Central audit logs
Formal	Database proxy with an MCP gateway	REAL TIME	MCP tool calls	Masks, filters and rewrites	Logs exported to SIEM tools
SmartVerify	At the tool call and the data store	REAL TIME at both hops	Person and tool call joined to the query	Refuses and limits on every covered store. Reshapes on PostgreSQL and MySQL, with more coverage coming.	Signed record per action, mapped through NIST SP 800-53 and NIST AI RMF. SIEM export by webhook.

Based on public vendor documentation reviewed in October 2026. Not described means the reviewed material did not describe it, and the capability may still exist. In that review, no product documented a link between a tool call, the query it produced and the person behind it.

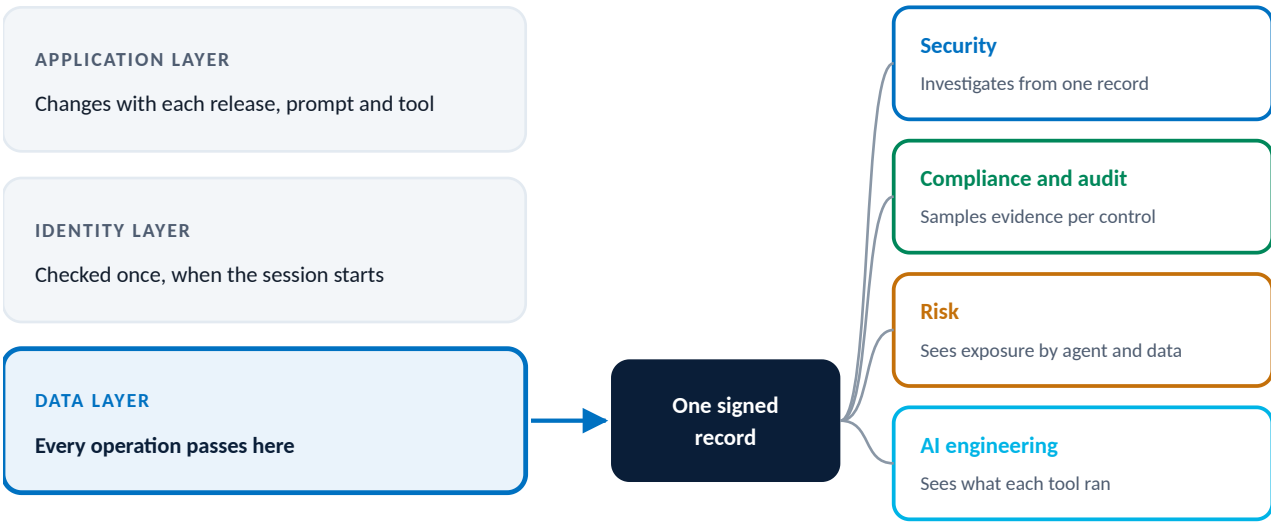
SECTION 11

Execution governance brings risk management and zero trust to the data layer

Risk management at the data layer measures exposure where it actually occurs. It tracks which data each agent read, for which person and in what volume. A policy set once covers every agent framework and model provider, including agents that connect later.

NIST SP 800-207 describes a zero trust architecture that grants no implicit trust based on network location. Access to each resource is decided by dynamic policy. Execution governance applies that principle to every data operation an agent performs. Each agent and tool server holds a short-lived, verified identity and no standing database secret. Each query is evaluated against the person, the request and policy before data returns.

The same signed record serves four teams that keep separate evidence today. Security investigates incidents from it, and compliance samples it as evidence for each control. Risk tracks exposure by agent, person and data set. AI engineering sees exactly what each tool ran in production.



Upper layers change often. The data layer sees every operation and feeds one record to four teams.

KEY POINT

Risk management and zero trust at the data hold whichever application, agent or model sends the request.

SECTION 12

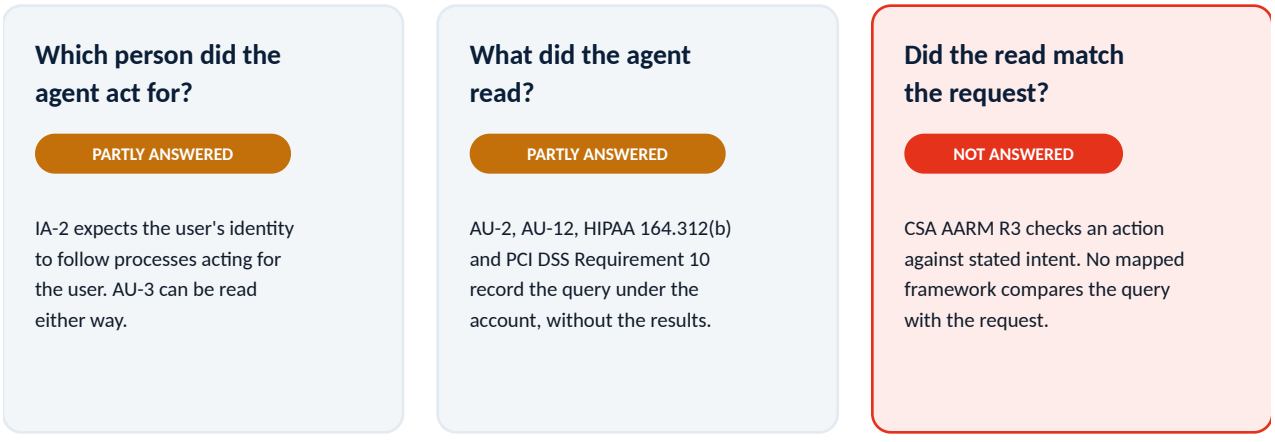
Regulators, auditors and the new Congress will ask operators to prove what their AI did

Existing security frameworks answer the auditor's three questions unevenly. NIST SP 800-53 IA-2 and AU-3 cover the person behind a process and the entities in an event. Logs kept under HIPAA 164.312(b) and PCI DSS Requirement 10 record the query and usually omit what came back. None of the frameworks SmartVerify mapped compares the executed query with the original request.

In Europe, Article 12 of the EU AI Act requires high-risk AI systems to log events automatically. Under a 2026 amendment, the obligations for stand-alone high-risk systems apply from December 2, 2027.

In the United States, Democrats are projected to hold majorities in both chambers after the November 2026 midterms. The new Congress convenes in January 2027 and will set the oversight agenda for AI. Law firm Foley & Lardner expects hearings on AI safety and data practices, with independent audits as a likely template. Foley also expects federal preemption of state AI laws to stall. State laws in California, New York, Colorado, Texas and Illinois would then remain the working baseline.

Each of these paths leads to the same request of an operator, which is to show what the AI did with the data.



The three parts of the audit question, and how today's frameworks answer each one.

SECTION 13

SmartVerify records, refuses, reshapes and limits at the last two hops

SmartVerify places two interceptors on the AI data path. The first sits where the agent's tool call reaches the tool server. The second sits where the tool server's query reaches the data store. SmartVerify joins what both observe into one signed record for each action.

> **Record.** Each record links the person, the request, the query and the data returned.

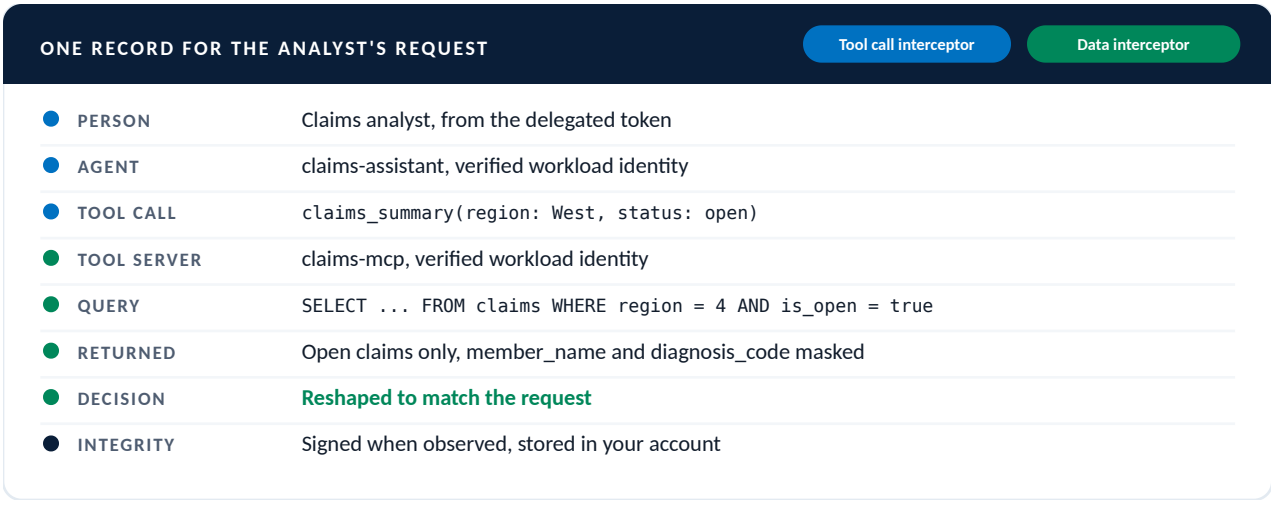
> **Refuse.** A query outside policy stops before it reaches the data, and the agent receives a standard error.

> **Reshape.** Rows and fields beyond the request are filtered or masked before the agent sees them.

> **Limit.** Records per query are capped, and tool calls are limited per agent and per person.

At launch, SmartVerify covers PostgreSQL, including Amazon RDS and Aurora, plus MySQL, Amazon S3 and Amazon DynamoDB. It also covers MCP tool calls. Record and refuse work on every covered store, and reshape works on PostgreSQL and MySQL, with more coverage coming. The enforcement path adds under 50 milliseconds at the 90th percentile.

SmartVerify runs inside the customer's environment on AWS, Azure, Google Cloud, on premises or air-gapped. Each agent and tool server holds a short-lived, verified identity and no database secret. Teams start in record-only mode and add refusal and reshaping once the records show what matters.



Policy limits the agent to open claims and masks identifying fields. The analyst still gets her summary.

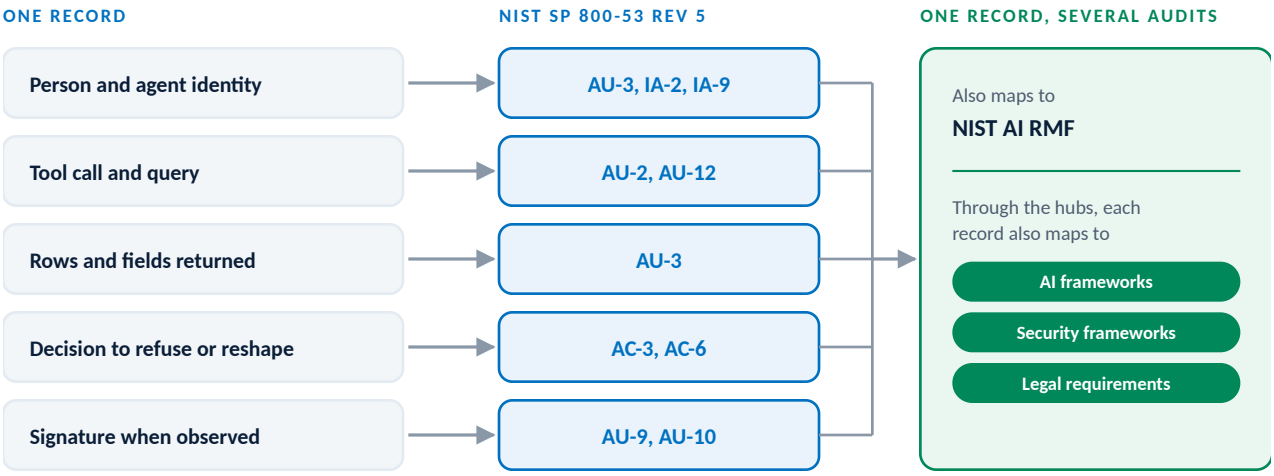
SECTION 14

Each SmartVerify record maps to the controls auditors already test

Data security vendors already map posture findings to compliance frameworks. Cyera, Varonis and Palo Alto Networks map findings to GDPR, HIPAA, PCI DSS and NIST. Microsoft Purview Compliance Manager provides assessments for the EU AI Act, NIST AI RMF and ISO/IEC 42001. These mappings show that controls are configured across an environment.

Auditors also sample individual actions to test whether a control operated. In the public material reviewed for this paper, no data layer product documents control mapping for each agent action.

SmartVerify maps every record to controls, using NIST SP 800-53 and NIST AI RMF as hubs. Through those hubs, each record reaches other AI and security frameworks and legal requirements. Under HIPAA, a record supports access control, audit controls, integrity, authentication and minimum necessary. Records also stream to the customer's SIEM by webhook. SmartVerify's founder previously led AWS Audit Manager, which maps cloud evidence to control frameworks.



Illustrative mapping of record fields to the controls they evidence.

KEY POINT

Posture mapping shows that a control exists. Per-action evidence shows that it worked on each agent action.

SECTION 15

Six RFP questions test a product against the four requirements

Every vendor in this paper describes its product as security for AI agents. These six questions apply the four requirements to any product under evaluation. Each one can go into an RFP as written, and a passing answer can be shown on live traffic.

		QUESTION FOR THE VENDOR	A PASSING ANSWER SHOWS
1	WHERE	**Where does the product enforce policy on an agent's data access?**	Enforcement at the data store, on the query as it runs.
2	WHERE	**Does any agent or tool server hold a stored database secret?**	Short-lived, verified workload identities only.
3	WHEN	**Can the product refuse a query, filter rows and mask fields before the agent receives results?**	Decisions made in line, before data returns.
4	CONTEXT	**Does each record link the query to the tool call, the person behind it, and verified identities for the agent and the tool server?**	One record per action that holds all of them.
5	PROOF	**Is each event signed when observed, and does the evidence stay in our environment?**	Signed records held in your account, separate from any AI provider.
6	PROOF	**Does the product map each recorded agent action to specific framework controls?**	Per-action records tagged with the controls they evidence.

A practical first step is a three to four week assessment on one agent. It runs in record-only mode and leaves the agent's behavior unchanged. The report shows what the agent read and what it read beyond each request. A security partner can scope the assessment and run it with your team.

Start by recording what your agents do. Then govern the operations that the records show matter.

SMART VERIFY

Liberate your AI. Control and prove every action on your data.

www.smartverify.ai