SMART VERIFY

WHITEPAPER

Assessing what your agents actually did

Why deployment topology determines your data security posture, and why a complete account needs observation at two boundaries rather than one.

Two organizations running the same tools, with the same policies, do not have the same coverage. The difference is how their agents reach data.

August 2026

smartverify.ai

About this paper

This paper is written for the security and compliance leaders who have to approve AI agents against production data, and for the architects who decide how those agents connect. It sets out the four ways agentic AI reaches data today, shows what each of those arrangements makes observable, and gives you a way to determine which records your own architecture is capable of producing before anyone asks for them. It assumes no prior familiarity with SmartVerify.

Contents

In summary

This paper sets out seven findings, and the sections that follow develop each one.

1. Deployment topology, not control inventory, determines what an organization can establish about its agents.

2. Four topologies are in common use. One of them supports a complete account of what an agent did.

3. A vendor-operated tool server yields the request and never the effect. A direct connection yields the effect and no request was ever stated.

4. The gap between what an agent asked for and what its operation actually read is the finding that matters, and it exists only where both boundaries are observed.

5. Controls at the prompt reduce how often a harmful request is made. They cannot bound what an allowed one does.

6. Coverage belongs per estate, with topology stated and gaps disclosed, rather than averaged into one number.

7. A control that closes the gap has to record, refuse, reshape and reason. Recording is the only one of the four that can be deployed without first deciding what to allow.

Terms used in this paper

Several ordinary words carry a specific meaning in the pages that follow.

Request

The tool call an agent makes. It names the capability the agent wants and the arguments it passes, for example a claims summary for one region and one plan year. It is not the user's prompt.

Effect

What then happened at the data store. The operation that actually ran, the fields it touched and the volume it returned.

Tool server

The component between the two. It receives a request, runs the operation and returns the result. Where it runs, and who operates it, is what separates the four routes in section 2.

Topology

Where those components run and who operates each one. The four routes in section 2 are the four topologies in common use.

Estate

One agent deployment with its own topology. Most organizations run several, which is why a single coverage figure across all of them answers nothing.

Interceptor

A component placed on a connection. It sees each message as it passes, records it, and can stop it before it continues.

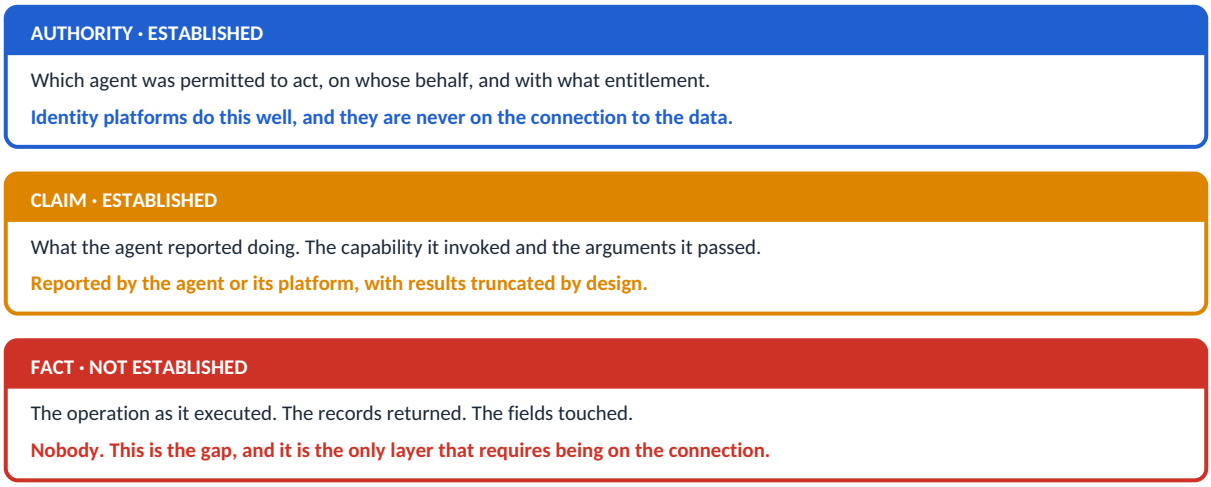
Label

The category attached to a field an operation touched, such as a payment identifier or a health identifier. A label describes the kind of data, never the value in it.

SECTION 01

Almost nothing observes an agent once it is inside your data perimeter

An AI agent acts with credentials you issued. Three things can be said about any access it makes, and only two of them are established today.



Authority and claim come from the agent reporting on itself. Fact requires being on the connection.

Three things can be said about any access. Two of them are established.

The harder question determines what an organization can actually produce when asked. Given how your agents are deployed, what can anyone establish, and what can nobody establish.

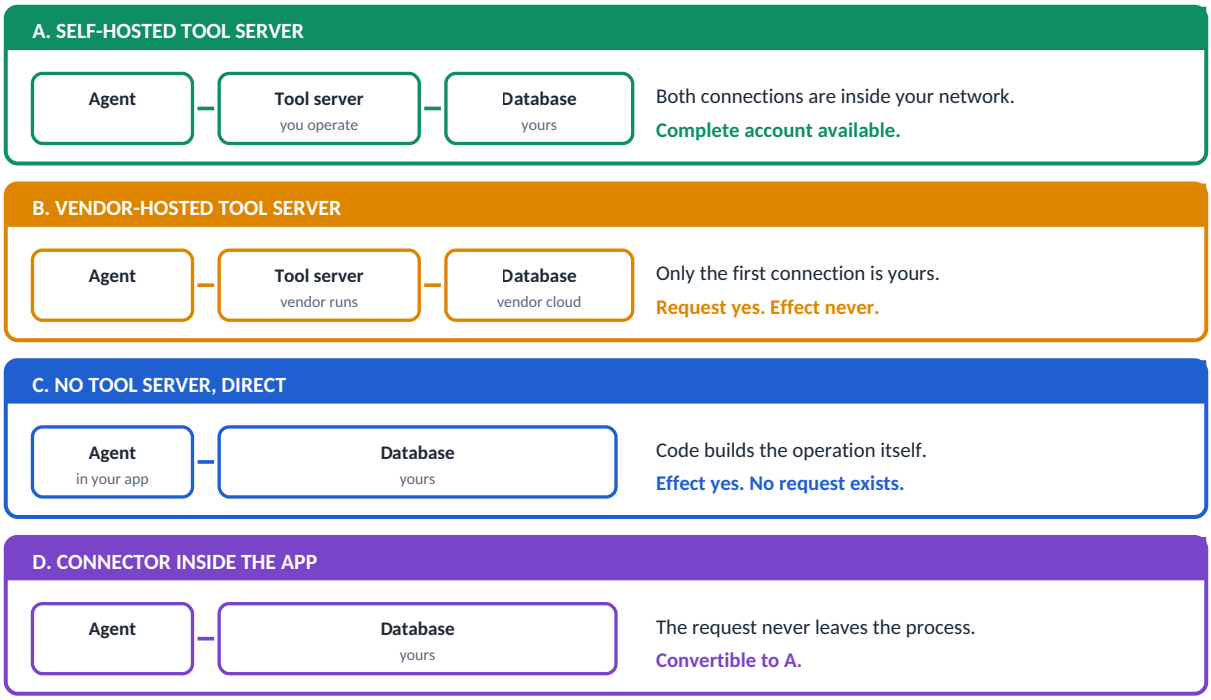
The completeness of your data security posture assessment is determined by your deployment topology, not by the number of controls you have bought.

An absence of records is not a neutral fact in an investigation. It reads as an absence of reasonable care.

SECTION 02

Agentic AI reaches data by four different routes

The routes are not equally observable, and the differences are structural rather than a matter of product maturity.



Most enterprises run two or three of these at once, and have not written down which is which.

Each of the four routes exposes a different part of the interaction.

Route D deserves separate attention because it is recoverable. The connector here is the database driver the application already links in, so the agent reaches the store from inside its own process. Moving that connector out of the process and behind a tool server you operate turns a partly observable estate into a fully observable one, usually without changing how the agent behaves.

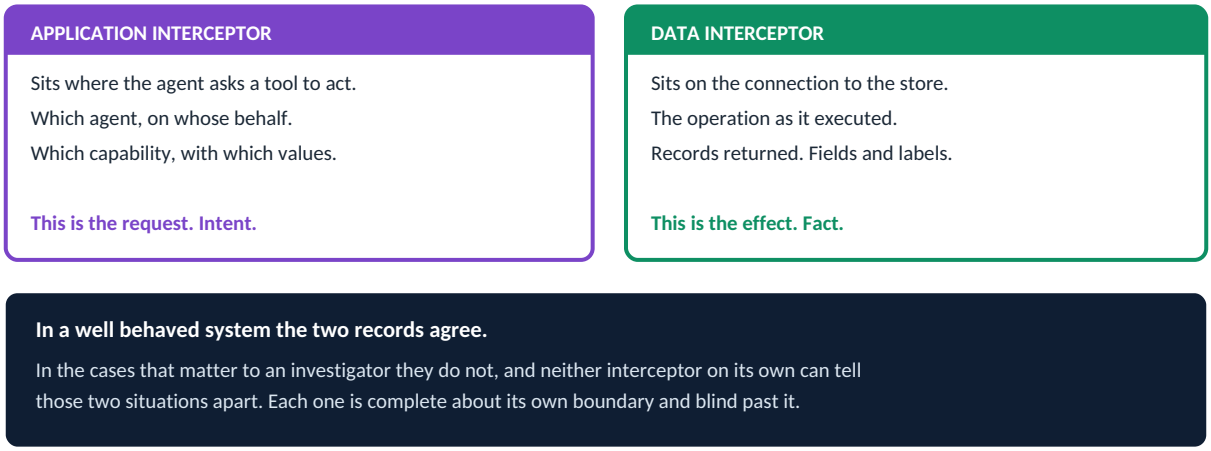
KEY POINT

For most organizations, converting an in-process connector is the single highest value change available.

SECTION 03

SmartVerify implements two interceptors for a complete picture

An interceptor is a component placed on a connection, which sees each message as it passes and can record it or stop it before it continues. One sits at the application boundary, where an agent asks a capability to act, and it sees the request. The other sits on the connection to the data store, and it sees the effect.



A request describes what was meant to happen. An effect describes what happened. Holding both is the only way to know which case you are looking at.

With both records in hand, the questions an auditor asks become answerable in the same form every time. Which agent, acting on whose behalf, asked for what, and what the operation then read.

SECTION 04

Your deployment topology determines what anyone can establish

The table below is the assessment. Its absence from posture documentation is why two organizations with identical control inventories carry very different exposure.

TOPOLOGY	Who acted	Request	Effect	Volume	Agreement	Prevention
A Self-hosted	✓	✓	✓	✓	✓	Both
B Vendor-hosted	✓	✓	✗	✗	✗	Request
C Direct	✓	None	✓	✓	n/a	Data
D In-process	✓	Hidden	✓	✓	n/a	Data

Agreement means the request and the effect can be placed side by side and checked against each other.
Prevention names the boundary at which an operation can still be stopped before it runs.

Vendor-hosted gives you the request and never the effect. Direct gives you the effect and there was no request.

One topology lets you compare them.

What an organization can establish depends on its topology.

One further constraint sits underneath all of it. Most applications connect to a data store with a single service credential, so the store sees one identity for every agent that uses it. The actor is identifiable at the application boundary and anonymous at the data boundary, which is why attribution requires both.

KEY POINT

A posture assessment that does not record deployment topology is an estimate rather than an assessment.

SECTION 05

The gap between what was asked and what ran is the finding

An agent is asked for a summary of claims in one region for one plan year. To produce it, the operation reads 40,247 rows across 31 fields, two of which are health identifiers. The summary that comes back is correct and the analyst gets the right number.

The problem is the distance between those two facts. A summary needs totals. This operation read the rows behind the totals, including fields the summary never used, so data the task did not need was read anyway. That is a finding under any data minimization rule, and it is also what sets the size of an incident.

The request, the operation and the gap between them.

The gap is that neither half of the comparison sits with the other. The agent platform holds the request and truncates the result, so it shows what was asked and not what was read. Monitoring at the database holds the rows read and never saw the request, so it cannot say whether reading them was reasonable. Both layers are accurate, and neither can produce the comparison.

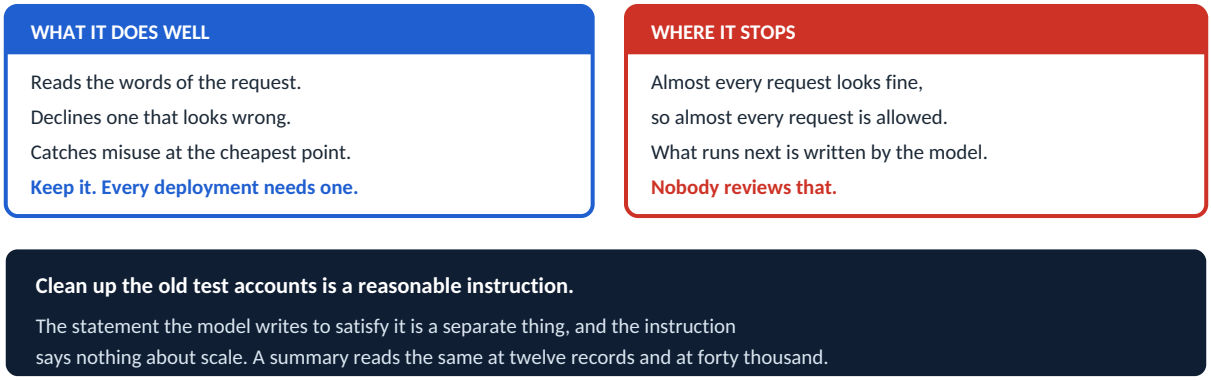
KEY POINT

A reasonable request and a routine operation can still be a finding side by side. Nothing that sees only one of them can produce it.

SECTION 06

Controls at the prompt cannot bound what an allowed request does

Controls that evaluate a request before a model acts are necessary and every serious deployment should have one. They are not sufficient, and the reason concerns the requests they allow rather than the ones they stop.



Controls at the prompt stop before the operation begins.

One control reduces how often a harmful request is made. The other bounds what one can do when it gets through, and records what it did.

SECTION 07

The same incident produces three different answers

The difference between topologies is easiest to see in the situation every organization is preparing for.

An agent invoked a reporting capability. A large volume of sensitive records was read. It surfaces a week later.

TOPOLOGY A	TOPOLOGY B	TOPOLOGY C
Which agent, for which service.	Which agent, for which service.	Which agent, by connection.
It asked for a regional summary.	It asked for a regional summary.	No request was stated.
40,247 records returned.	Volume unknown.	40,247 records returned.
Two sensitive fields.	Fields unknown.	Two sensitive fields.
Scope is bounded.	**Scope cannot be bounded.**	**Purpose is unknown.**

Where scope cannot be bounded, counsel assumes the widest plausible exposure.

The breach notification then follows that assumption, which is frequently the difference between contacting a defined set of individuals and contacting everyone.

Nothing separates these three but a decision about where the connector runs.

One incident reaches three different conclusions in three estates.

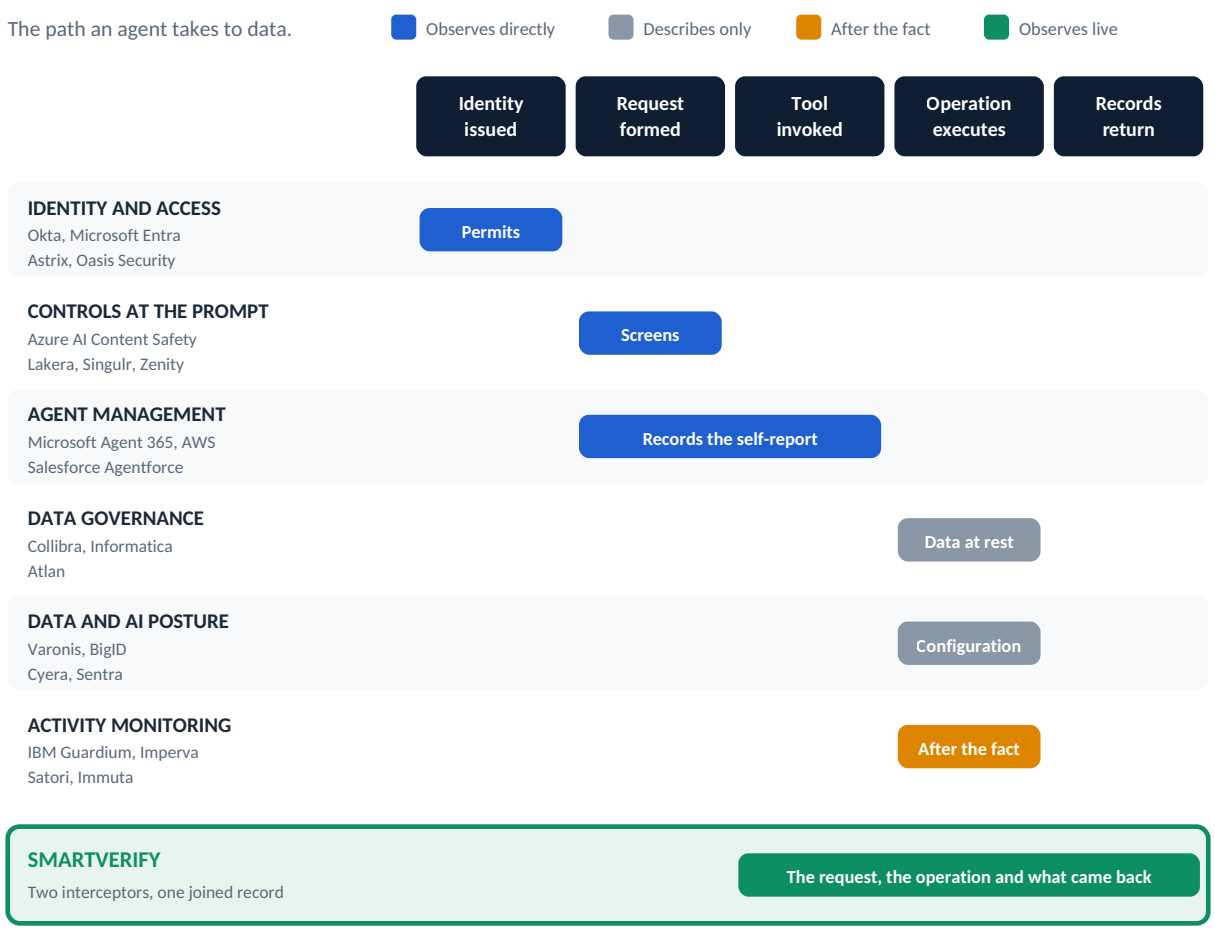
KEY POINT

The first estate and the third cost the same to operate. The difference between them was an architecture decision, and it is recoverable in advance rather than afterwards.

SECTION 08

No existing category observes the operation as it happens

Each category below does something an observation layer does not, and a mature deployment has several of them. The boundary described is a property of the category rather than of any product within it.



Three categories describe the last stages. None of them watches the operation happen.

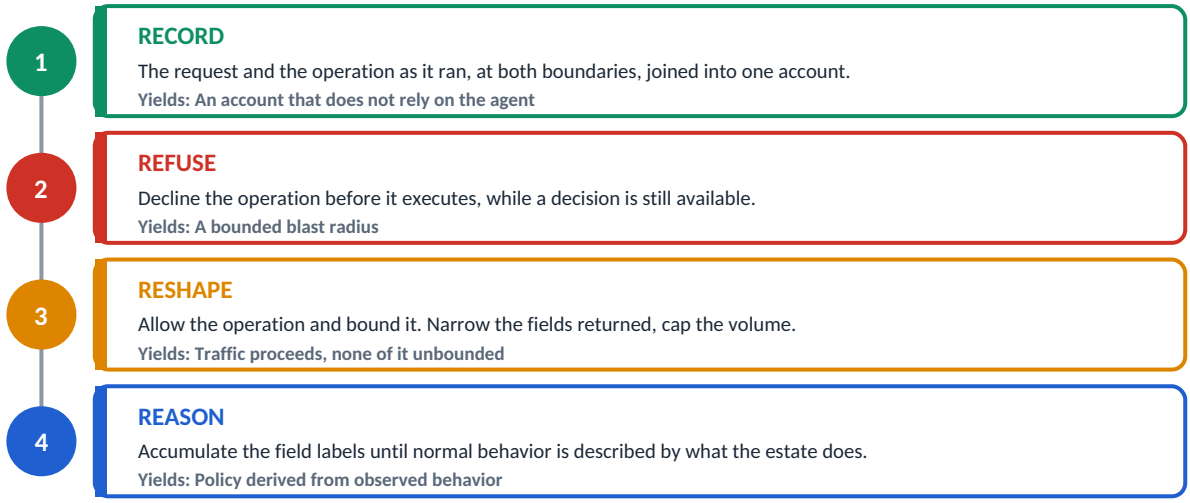
Each category covers one stage of the flow and stops there.

SECTION 09

A control at the interaction has to do four things

Anything that closes the gap has to be present on the connection while the operation runs. Once something is there, four capabilities follow in order.

Each capability depends on the one above it, and the order is also the order in which an organization adopts them.



Only the first can be deployed without a decision about what to allow.

Four capabilities, each depending on the one before it.

Recording changes nothing about how an agent behaves, so it can go on real traffic in weeks rather than quarters, and it is the prerequisite for the rest, since a refusal without a record is an outage nobody can explain. Refusal follows once the records show something worth refusing. Reshaping follows once refusal proves too blunt for the traffic that matters, and it is the common case rather than the exception, because most agent operations do not need to be stopped. They need to be bounded. Reasoning is not a separate purchase. It is what an accumulated record becomes.

KEY POINT

A control that only records is an audit. A control that only refuses is a firewall in front of a database. An assessment needs all four, and none of the four is available to something that is not on the connection.

SECTION 10

Compliance coverage should be reported for each estate separately

Each control is graded as fully evidenced, partially evidenced or a gap, with the counts shown rather than a single percentage asserted. Topology is the dimension usually missing.

CONTROL FAMILY	A	B	C	D
Accounting for a non-human actor	FULL	PART	FULL	FULL
Scale of what was reached	FULL	GAP	FULL	FULL
Sensitivity of fields involved	FULL	GAP	FULL	FULL
Purpose, as stated	FULL	FULL	GAP	GAP
Purpose matched to effect	FULL	GAP	GAP	GAP
Integrity of the record	FULL	FULL	FULL	FULL
Preventive control	FULL	PART	FULL	FULL

Integrity is the only family that is complete everywhere. Purpose and effect diverge in opposite directions.

The remedy for a gap in column B is a change of topology, not a better tool.

Coverage is graded by control family and by topology.

Very few organizations sit in one topology. A single figure across a mixed estate cannot answer the auditor's first follow up question, which is always which systems it applies to. Report one row per estate, grade by control family, and record any part of an estate that nothing can observe as an accepted risk with a named owner.

KEY POINT

A disclosed gap with a named owner and a remediation path is evidence of a working control process. The same gap found by somebody else is evidence of its absence.

SECTION 11

Six questions reveal what your organization can establish today

None of these requires a vendor to answer.

1	Does each agent reach data through a capability you operate, or one your vendor operates?
2	Where the vendor operates it, can you state what it did on your data?
3	Where there is no tool server, is the database driver inside your application process?
4	For one named agent on one named day, can you produce the operations and the volume?
5	For any agent that can change data, is the governing rule written down?
6	Does your posture documentation record which topology each estate is in?

If question four cannot be answered today, that is the finding.

These six questions need no vendor to answer.

Being unable to state which operations a named agent ran on a named day is the condition every framework in this area is concerned with, and it is the first question in any investigation.

SECTION 12

Deployment requires only a configuration change

Both interceptors are configured at the client rather than in application code. For a data connection this is a connection string. For an application boundary it is an endpoint setting. Neither requires a change to the agent, the framework or the data store.

Observation and enforcement run inside the customer's own account. Raw request and response content, and the audit record in full, remain there. Labels, shapes, counts, decisions and identifiers are the only things that leave.

A field holding a patient name crosses the boundary as a label identifying it as such, never as a value.

Three boundaries define the limits, and each is a property of where computation happens.

- **Intent.** The record establishes what was asked, what happened, and the distance between them. Whether an access was appropriate is a judgment for a person with the surrounding context.
- **A vendor environment.** Where the tool server is operated by your data platform, what it did on your data is observable only to that platform.
- **Prompts and model internals.** Nothing observed here speaks to prompt manipulation, model behavior or model supply chain.

SECTION 13

What to do next

You are being asked to accept a risk whose size nobody in the organization can currently state. That is not solved by a control whose output is a configuration assessment.

Three actions follow, and none of them requires a purchase.

- **Record the topology of every agent estate.** It determines what can ever be established, and it is probably written down nowhere today.
- **Convert the estates that are convertible.** Moving an in-process connector behind a tool server you operate is usually cheap and changes coverage more than any product will.
- **Name an owner for every part nothing can observe.** Where a vendor operates the tool server, that segment is an accepted risk rather than a gap waiting on tooling.

> KEY POINT
>
> **Then ask for the record. For one named agent, on one named day, the operations it ran and the volume it returned. If that cannot be produced, everything else is provisional.**

A fixed scope assessment on one estate, observing only, on real traffic, is the usual starting point. The deliverable is a report on your own agents in two parts. What was refused, and what was recorded but not refused. It states the share of traffic that could not be classified, because a coverage figure a reader cannot see is not a coverage figure.