ON METERING, INCENTIVES AND THE TELEMETRY YOU DECIDED NOT TO COLLECT

How you are charged decides what you can see

A pricing model is not a commercial detail sitting outside the architecture. It is the mechanism by which a budget becomes a coverage gap, and it does that work quietly, in a spreadsheet, months before anyone notices.

IN BRIEF

Security platforms are metered on volume, endpoints, seats, assets, users and stores — often several at once. Each meter creates an incentive, and volume metering creates the worst one: it makes the security team the party that argues for less telemetry. This paper examines what each common meter rewards, why a single ingestion meter is the least distorting of the available options rather than a costless one, and what a buyer should demand of any meter — including ours.

1 · The conversation nobody records

A detection engineer proposes enabling command-line capture across the server estate. It would materially improve coverage of a technique family the team knows it is blind to. It would also roughly double daily ingestion, and ingestion is billed per gigabyte.

The proposal does not get refused. It gets deferred to the next budget cycle, then narrowed to a subset of hosts, then quietly dropped when the subset turns out to be the wrong hosts. Nobody made a decision to accept that risk. **The meter made it, and there is no record of it anywhere** — no risk acceptance, no exception, nothing an auditor could find.

This is the central objection to volume pricing, and it is not that it is expensive. It is that it puts the security function on the wrong side of the argument about visibility. A team whose costs rise with the completeness of its own telemetry will, over years and without anyone intending it, converge on collecting less than it needs.

FIGURE 1 · WHAT EACH METER REWARDS

`per GB/day`	Rewards collecting less. Filtering at the edge becomes a cost-saving project, and the filters are written by whoever is under budget pressure that quarter.
`per endpoint`	Rewards leaving assets unmanaged. The devices nobody wants to pay for are disproportionately the ones nobody is patching either.
`per seat`	Rewards shared logins. A control that charges per accountable human is a control that produces fewer accountable humans.
`per asset scanned`	Rewards narrowing scope. The estate's edges — test environments, acquisitions, forgotten subnets — are exactly where the scope shrinks first.
`per data store`	Rewards leaving stores unclassified — and an unclassified store is the one that appears in the breach notification.

Every meter distorts something. The question is not which is neutral — none is — but whether the distortion pushes toward or away from the behaviour the product exists to encourage.

2 · Six meters is worse than the sum of six

An estate assembled from six products carries six meters, and the cost is not six invoices. It is six negotiations, six renewal dates, six forecasting models, and six versions of the answer to “what does security cost”.

The operational consequence is subtler than the administrative one. When capabilities are separately metered, **turning one on is a procurement event**. A team that discovers mid-incident that it needs a capability it did not license does not enable it; it works around it, badly, and files a request. The delay between recognising a need and satisfying it is measured in weeks, and that interval is where the actual damage from fragmented licensing lands.

There is also a quieter effect on architecture. Separately metered capabilities are built to be separable, which means they are built with boundaries between them — and those boundaries are the seams that this series has spent eight papers describing as the source of the response gap. Commercial structure and technical structure are not independent.

> **The test for a bundle.** Ask whether enabling a capability you already pay for requires a conversation with anyone. If it does, the bundle is a discount structure rather than a single product — and the operational benefit being claimed does not exist.

3 · What a single ingestion meter actually costs

Metering one thing — the data a line ingests per day — and including every capability is the least distorting arrangement we have found. It is not distortion-free, and a paper that claimed otherwise would be doing exactly what Paper 08 objects to.

It still meters visibility. An ingestion meter is a volume meter wearing a better suit: including every capability removes the procurement incentive, but not the incentive to collect less. A team facing a spike will still think about what to stop sending.

It charges the noisy estate more. Two organisations of equal size and unequal hygiene pay unequal amounts, and the one with sprawl pays more. Arguably correct, and it will still feel unfair to the buyer who inherited the sprawl.

It makes a chatty source a budget event. One misconfigured logger can move the bill. The mitigation is per-source visibility and alerting on the meter itself, so a volume change is diagnosed rather than absorbed.

It is a bet on our own efficiency. If the cost of processing a gigabyte rises faster than our ability to process it cheaply, the model pressures the vendor rather than the customer. That is the intended direction, and it is worth stating that it is also the reason a vendor might abandon it later.

What the single meter does buy is the removal of the second-worst conversation in security operations: the one where a capability exists, is understood to be needed, and is not switched on because it sits in a different SKU. Enabling something becomes **a decision about your security rather than a procurement cycle** — and that is a real change in how a team behaves during the first hour of an incident.

FIGURE 2 · TWO WAYS TO NEED A CAPABILITY AT 02:00

SEPARATELY LICENSED

Recognise the need · confirm it is not licensed · raise a request · wait for an approval that is not urgent to anyone outside the incident · work around it meanwhile. The workaround becomes permanent.

INCLUDED, METERED ON INGESTION

Recognise the need · enable it · note that ingestion will rise · continue. The cost consequence is visible on the meter the following day, and reviewable.

The right-hand path still has a cost. It simply pays it after the incident rather than instead of responding to it.

4 · What to demand of any meter

Observable before the invoice. A meter you can only see in arrears is not a meter you can manage. Per-source, per-day, in the product.

Predictable under growth. You should be able to compute next year's cost from your own growth assumptions without asking the vendor. If a sizing figure requires a conversation, the model is not transparent.

Overage that does not stop the platform. A meter that halts ingestion at a threshold has converted a billing dispute into an outage of your visibility. Overage must degrade commercially, never operationally.

No meter on the security-critical path. Nothing that decides whether an event is *examined* should depend on a commercial threshold. Charge for volume if you must; never for judgement.

> **The unifying idea.** Every meter is an incentive pointed at your own team. Choose the one whose distortion you can live with, insist on seeing it daily, and never accept one that makes an outage the consequence of a budget disagreement.

5 · Questions worth asking, whoever you are buying from

01 How many separate meters will appear on our account, and on what renewal dates?

02 Name a capability we would have to buy separately after signing. If there is none, say so in writing.

03 Can I see the meter per source, per day, inside the product?

04 What happens at overage — a bill, a throttle, or a stop? Show me the behaviour, not the policy.

05 Can I compute next year's cost myself, from my own growth numbers, without calling you?

06 Does any commercial threshold affect whether an event is examined rather than merely stored?

07 If we add a noisy source by mistake, when do we find out — and from what?

08 What in your model would make you change it, and what would that mean for our renewal?

6 · Where this sits in the series

Paper 09 argued that the ingestion boundary decides what the platform can know. This paper is its commercial twin: the meter decides what reaches the boundary at all. Between them they bound everything the previous eight papers described — a closed loop cannot correlate across an estate whose telemetry was trimmed to fit a licence.

It is the least technical paper in the series and the one most likely to determine whether the rest of it matters in a given organisation. Architecture sets the ceiling; the meter sets where you actually operate.

> *No risk register records the coverage a pricing model talked you out of.*

EventLake

ABOUT THIS PAPER

Paper 10 in a series on the architecture of active defense, published by **EventLake Research**. Papers 01–09 cover the closed loop, explainable alerts, legitimacy, rule compilation, bounded autonomy, evidence, continuous trust, coverage honesty, and the integration plane. The series is deliberately vendor-neutral, and the questions in §5 are meant to be asked of us as readily as of anyone else.

EventLake meters each line on one thing — gigabytes ingested per day, or resolved checks on the verification line — with every application in the line included, per-source daily visibility, and overage that is a bill rather than a throttle. The sizing calculator is public and needs no conversation. **eventlake.com**