

Data your agents can reason over with no normalization project first

Most AI SOC projects stall on the data, not the agent

The usual fix is to write the parsing up front and keep it current as vendors change their formats. That is a data engineering project nobody staffed for, and it's usually where the AI use case dies. Realm closes that gap on the way in.

What lands in your lake

With Realm upstream, each event lands in Realm lake format, parsed and tagged, with the original log intact beside it. Ask an agent the same question of both.

Agent asks Which internal hosts talked to 203.0.113.45, and is it malicious?

What lands today

```
1,2026/09/18 14:02:11,013201009832,TRAFFIC,en
d,2561,2026/09/18 14:02:11,10.4.12.37,203.0.11
3.45,198.51.100.10,203.0.113.45,allow-web-out,
corp\jsmith,,ssl,vsys1,trust,untrust,ethernet
1/2,ethernet1/1,fwd-lake,418822,1,52144,443,41
```

Nothing to query. Parse it first.

What lands with Realm

```
raw      1,2026/09/18 14:02:11,0132...
```

```
parsed      src_ip 10.4.12.37
```

observables ip 203.0.113.45

```
realm_metadata source palo_alto
```

```
enrichments threat_intel tor_exit
```

10.4.12.37, user jsmith. Destination is a known Tor exit.

How Realm does it

Realm lake format

One record shape for every source. Whatever the vendor's log looks like coming in, what lands is the same JSON structure. It is not a storage engine or a schema you have to adopt, and it works with any lake destination.

Schema intelligence

No parser project for your team. Clarity AI reads each source, works out its schema, and picks out the observables. When a vendor changes its log format, Realm catches the drift before it breaks a parser.

Why an agent needs each block

raw

The original log, kept intact. A field can be checked against it, and you keep the evidence.

parsed

The log broken into fields, so the model reads the field it needs directly.

observables

IPs, users, and hosts, mapped to key OCSF observables. The fields an investigation pivots on.

realm_metadata

Where the event came from and when. An answer can show its provenance.

enrichments

Geo, ASN, and threat intelligence attached in stream. Context arrives with the event, not from a lookup mid-question.

Key outcomes

- **One shape across sources.** Agents work from the same record structure whatever the vendor, instead of one format per source.
- **A faster start.** Agents work on the data the day it lands, instead of waiting on a data engineering program.
- **The original is never lost.** Raw and parsed sit in the same record, which matters to the analyst and to the auditor.
- **Less compute in the lake.** Parsing and enrichment happen on the way in, not in the lake after the data lands.
- **An open format.** Open JSON with the raw log in every record, readable by any tool your team uses in the lake.
- **Your data engineers get their time back.** Every new source, vendor schema change, and broken parser stops being their job, so they work on what is unique to your business.

Your SIEM and your lake want opposite things

A pipeline that only moves data sends the same shape everywhere. Realm collects each source once and decides what each destination needs, so the same pipeline that cuts your SIEM bill feeds your lake data your agents can use. Realm feeds the lake and the agents you already chose. It does not become a lake or replace your agent, and what you build on top is yours.

Your SIEM

- **Needs** lean, parseable, detection-relevant data. Noise costs money.
- **Realm sends** the leanest input that still fires every detection.
- **You get** lower ingest cost, with no detection gaps.

Your lake

- **Needs** structure and context that join across sources, without owning the parsing.
- **Realm sends** every event in Realm lake format, with the raw event intact.
- **You get** data an agent can reason over the day it lands.

Global manufacturer

One of the largest Microsoft Sentinel deployments in the world.

3x

return per \$1 spent on Realm

1,000+

KQL rules replaced in 72 hours

>50%

reduction of Palo Alto logs

Their own agents now run on firewall data in their lake, with no data engineering project in between.

They came to Realm to cut a Sentinel bill, and chose Realm over other pipelines for reduction effectiveness and a modern, fully managed architecture. The same firewall data Realm trims for Sentinel now also lands in their lake in Realm lake format, off the same pipeline.

Watch an agent hunt in a lake it can read.

In 30 minutes, our security & AI experts run a live threat hunt on Realm lake format data. If you want to go further, we run it on a sample of your own logs next.

Book a session → realm.security/ai-threat-hunt

About Realm Security

Realm Security gives security teams independent control over their telemetry before it reaches the SIEM, data lake, or AI agent. The platform understands what each source produces and what each downstream system needs, then guides teams on what to retain, cut, route. Headquartered in Boston, Realm deploys in about a week and typically cuts SIEM ingestion costs by 50% or more.



www.realm.security

LinkedIn: [/company/realm-security](https://company/realm-security)

X: [@Realm_Security](https://twitter.com/Realm_Security)