



HANDS-ON WALKTHROUGH · SPLUNK HEC

Is every HEC endpoint up and receiving?

One tenant follows the HTTP Event Collector of a Splunk deployment — its endpoints, the pools they form behind a load balancer and the tokens senders use — from the telemetry Splunk already writes: nothing to install, nothing new to ingest.

Captured live on TrackMe 2.4.21 · two data centres, two heavy forwarders each, fed by Cribl Stream



HEC fails quietly: senders retry, data piles up elsewhere

A heavy forwarder that stops, a pipeline that backs up, a token sending to an index it may not use: the sender sees an error, Splunk logs it, nobody looks.

Endpoints that stop

A heavy forwarder down, a pool left with one member, telemetry stalled behind a blocked output.

"Server is busy"

Refused requests, back pressure along the ingestion pipeline, the stage that cannot keep up, indexer acknowledgment under pressure.

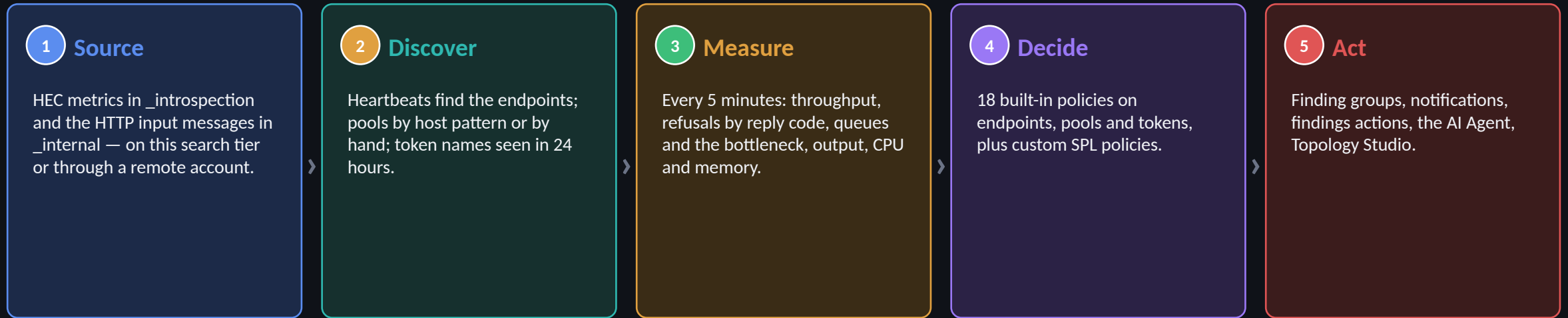
Tokens and the load balancer

Data rejected per token, a token that went silent, a pool member up but starved of traffic.

What a Splunk HEC tenant does

A dedicated tenant type for the HEC endpoints of one Splunk deployment. TrackMe reads the HEC telemetry Splunk already writes — its metrics in `_introspection`, its HTTP input, queue and output messages in `_internal` — with a few bounded searches every five minutes, and raises findings from eighteen built-in policies on endpoints, pools and tokens. It only reads: nothing is installed or changed on Splunk, and a token is only ever known by its name.

Splunk's own telemetry, read every five minutes, judged per pool



Pools judged as a whole

Endpoints behind one load balancer form a pool: TrackMe compares its members — members up against the expected size, the share of the traffic each one receives, their versions and settings — and judges its capacity and its traffic as a whole.

Traffic models learn what each pool and each token usually receives per hour of the week, from the history `_introspection` already keeps.

What we are going to do — one tenant, four parts

PART 1

Setup

- The Splunk HEC card
- Source, endpoints and pools
- The workspace and its strip

PART 2

Endpoints, pools & tokens

- The overview
- Endpoints and one endpoint up close
- A pool, the tokens

PART 3

Findings

- A heavy forwarder stops
- How it is evaluated
- Topology Studio, the AI Agent

PART 4

Operate

- Policies and token silence
- Traffic models and tuning
- Collection health, splunkd API

Every screen is the product as shipped, on two data centres of heavy forwarders.

01

PART 1 • SETUP

One source, two pools, one tenant

The Splunk HEC card, the source and its endpoints discovered from their heartbeats, the pools of each data centre, and the workspace.

A dedicated tenant — pick Splunk HEC

Create a TrackMe Virtual Tenant

Choose one component to enable for your new Virtual Tenant. You can configure details in the following steps.

Splunk Feeds Tracking

Monitor and track Splunk data feeds

- SPLK-DSM: data source monitoring
- SPLK-DHM: event endpoint tracking
- SPLK-MHM: metric endpoint tracking

SPLK-DSM

Select

Splunk Volume (Outliers)

Licensed volume drops and spikes per index

- Outliers on the licensed volume of every index
- Daily licensed volume and month projection
- Local or remote, history backfilled on day 1

Select

Splunk Fields Quality Monitoring

Assess and monitor data quality at scale

- Create sampling jobs for CIM and non-CIM
- Generate quality dictionaries
- Detect and alert on quality issues

Select

Splunk Workload

Track and manage scheduled report activities

- Discover and monitor schedules
- Detect anomalies and trends
- Optimize workload costs

Select

Splunk Flex Objects Tracking

Track entities from your "magic searches"

- Register searches producing objects
- Status and KPIs via simple conventions
- Workflow & metrics ingestion automated

Select

User Activity Monitoring

Investigate Splunk user activity and policy findings

- Local or remote environment
- Human and service account classification
- Evidence, collection gaps and investigation workflow

Select

Cribl Stream

Monitor the health and traffic of your Cribl Stream workspace

- Sources, destinations, pipelines, routes and workers
- Findings from health, traffic outliers and worker policies
- Computed from your Cribl metrics, no extra ingestion

Select

Cribl Edge

Monitor your Cribl Edge Fleets and Edge Nodes at any scale

- Node connectivity per Fleet, version and configuration drift
- Source and Destination health reported by the nodes, disk hotspots
- From the Cribl Leader API, no extra ingestion

Select

Splunk HEC

Monitor your Splunk HTTP Event Collector endpoints and pools

- Endpoints discovered automatically, grouped in pools
- Activity, refused requests, queues and tokens per endpoint
- From Splunk's own logging, no extra ingestion

Select

Close

1 The Splunk HEC card

Endpoints discovered automatically, grouped in pools; activity, refused requests, queues and tokens per endpoint.

2 From Splunk's own logging

No extra ingestion and nothing installed on the endpoints.

3 Four steps

Basics, Source & endpoints, Indexes & RBAC, Review.

Licence

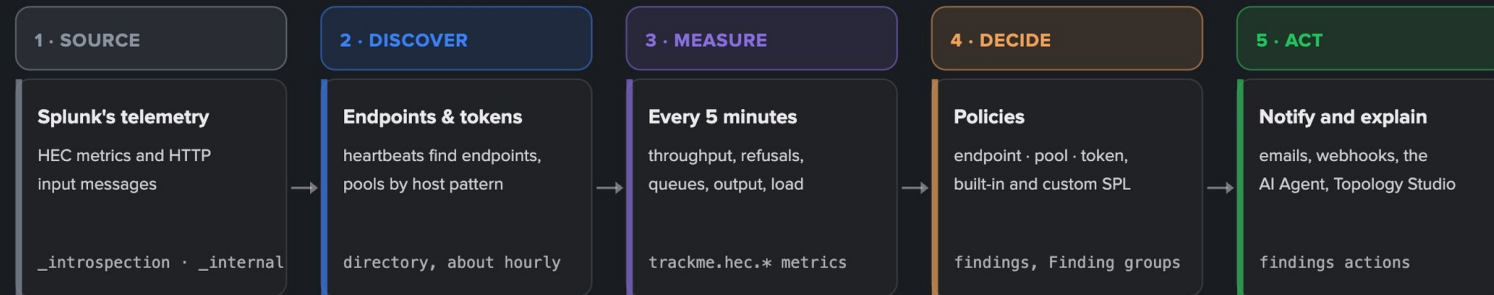
A valid Enterprise, Unlimited, Unlimited trial or Developer licence. Not available on the Foundation edition or the Foundation trial.

Virtual Tenants → Actions → Create a tracking tenant.

The first screen explains the tenant — with a diagram on demand

Splunk HEC — how it works

Splunk's own telemetry, read every 5 minutes: every endpoint, pool and token of the deployment, no agent and no new data.



1

Source

The HEC metrics and the HTTP input messages Splunk already writes.

2

Discover

Heartbeats find the endpoints; pools by host pattern; a directory about hourly.

3

Measure

Every 5 minutes: throughput, refusals, queues, output, load.

4

Decide and act

Policies on endpoints, pools and tokens; findings, emails, webhooks, the AI Agent.

5

Cost

A few bounded searches per window, whatever the number of endpoints.

Source & endpoints — found from their HEC heartbeats

Create a new TrackMe Virtual Tenant: Splunk HEC

1

2

3

4

BasicsSource & endpointsIndexes & RBACReview

TrackMe reads the HTTP Event Collector telemetry Splunk already logs (the HEC metrics of the introspection index and the splunkd and queue logs of the internal index) on this search tier, or on another deployment through a remote account. Nothing is installed on the endpoints and no token value is ever read: token names only.

Source

local (this search tier)

local: this search tier. A remote account: the deployment it reaches (its search tier must see the HEC endpoints' internal indexes).

Internal index

_internal

The index holding splunkd.log and metrics.log of the endpoints (default _internal).

Introspection index

_introspection

The index holding the HEC metrics of the endpoints (default _introspection).

Model learning history

14 days

How far back the traffic models learn the HEC telemetry when the tenant is created (what the introspection index keeps), and the history a relearned model reads.

Endpoint host patterns

hec-pool-*

The endpoints this tenant monitors, as host names with the * and ? wildcards, comma separated (* for every endpoint).

Excluded hosts (optional)

Hosts never monitored, even when a pattern matches them.

HEC ENDPOINTS

4 endpoints over the last 24 hours

Discover again

Endpoint	Pool	Transport	Token names	Last heartbeat
hec-pool-hf1	Data centre 1 · by pattern	http	1	1 min ago
hec-pool-hf2	Data centre 1 · by pattern	http	2	1 min ago
hec-pool-hf3	Data centre 2 · by pattern	http	1	50s ago
hec-pool-hf4	Data centre 2 · by pattern	http	1	1 min ago

POOLS

3 pools

Use 1 proposed pool

Add pool

< Back

Next >

- 1

Source

local, this search tier, or a TrackMe remote account to another deployment.
- 2

Indexes

_internal and _introspection by default; locked once the tenant exists.
- 3

Model learning history

14 days by default: the traffic models learn the telemetry _introspection already keeps.
- 4

Host patterns

hec-pool-* here: the old forwarders of the deployment stay out.
- 5

HEC endpoints

Every endpoint that reported in 24 hours, its transport, token names and last heartbeat.

Pools — one per data centre, each of two heavy forwarders

Create a new TrackMe Virtual Tenant: Splunk HEC

×

hec-pool-hf1

Data centre 1 · by pattern

http

1

1 min ago

hec-pool-hf2

Data centre 1 · by pattern

http

2

1 min ago

hec-pool-hf3

Data centre 2 · by pattern

http

1

50s ago

hec-pool-hf4

Data centre 2 · by pattern

http

1

1 min ago

POOLS 3 pools

Use 1 proposed pool

+ Add pool

A pool groups the endpoints behind one load balancer or with one role: TrackMe compares its members (the share of traffic each receives, the members down) and evaluates its health as a whole. An endpoint belongs to the first pool whose patterns match it, unless you assign it to a pool by hand: click its pool in the endpoints above. A hand assignment overrides the patterns for that endpoint only.

Identifier	Name	Host patterns	Expected size	Replaceable	Endpoints now
dc1	Data centre 1	hec-pool-hf1, hec-pool-hf2	2	No	2 endpoints
dc2	Data centre 2	hec-pool-hf3, hec-pool-hf4	2	No	2 endpoints
default	Default	none: endpoints no other pool claims	Any	No	0 endpoints

Identifier: lowercase letters, digits, - and _ fixed once saved. Host patterns: host names with the * and ? wildcards, comma separated; the default pool needs none. Expected size (optional): how many endpoints the pool should have. Replaceable: its members are replaced by others (Splunk Cloud indexers, auto scaling hosts); a silent member is down only when no replacement brings the pool back to its expected size within the grace of the Endpoint availability policy.

Auto-join: a new endpoint that no pool claims joins the default pool

SPLUNKD API (OPTIONAL)

Optional: during each inventory, TrackMe reads the splunkd REST API of the endpoints these remote accounts reach (read only) to compare the members of a pool: their HEC tokens (names and settings, never a token value), their global HEC settings and version. Token consistency needs it; every other policy works without it. An account may list one URL per endpoint; each URL is matched to an endpoint by the names the instance reports. Splunk Cloud endpoints are never read. The tenant owner must hold one of the account's roles.

Remote accounts

No account

Remote accounts whose splunkd the inventory reads. None: the enrichment is off.

< Back

Next >

1

Endpoints and their pool
Each endpoint joins the first pool whose patterns match it; click its pool to assign it by hand.

2

Data centre 1 and 2
hec-pool-hf1/hf2 and hec-pool-hf3/hf4, expected size 2 each.

3

Proposed pools
One per host-name family, and one per Splunk Cloud stack.

4

Replaceable
For members replaced under new names: Splunk Cloud indexers, auto scaling.

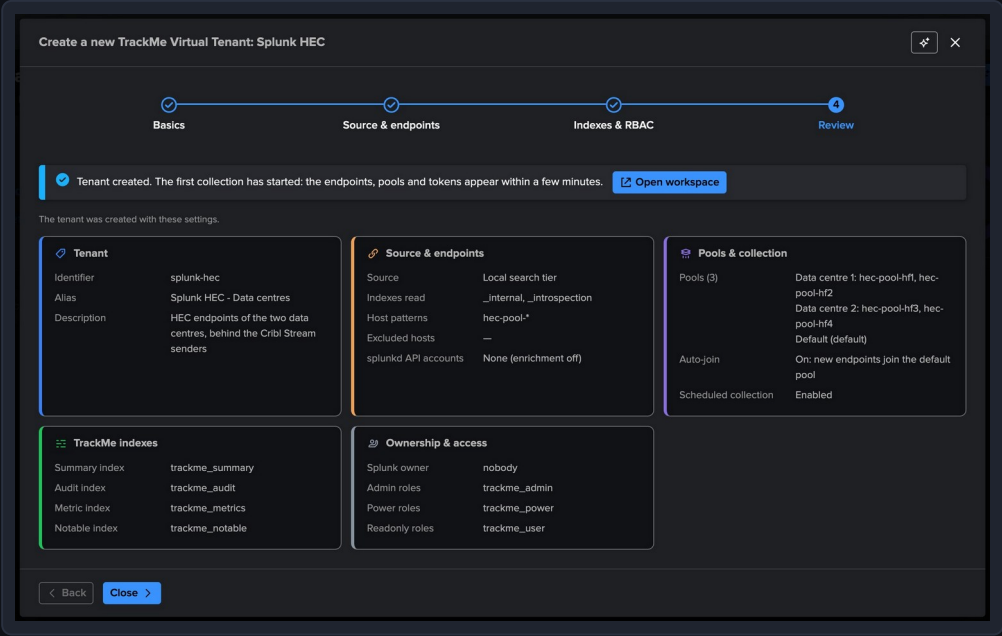
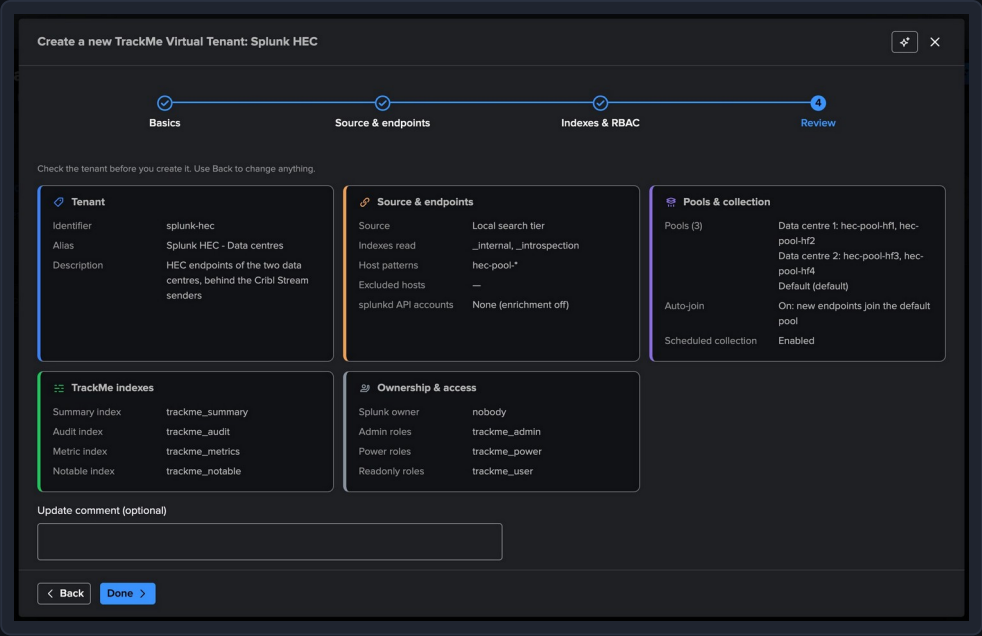
5

Auto-join
A new endpoint no pool claims joins the default pool.

TrackMe for Splunk · Hands-on: Splunk HEC

docs.trackme-solutions.com

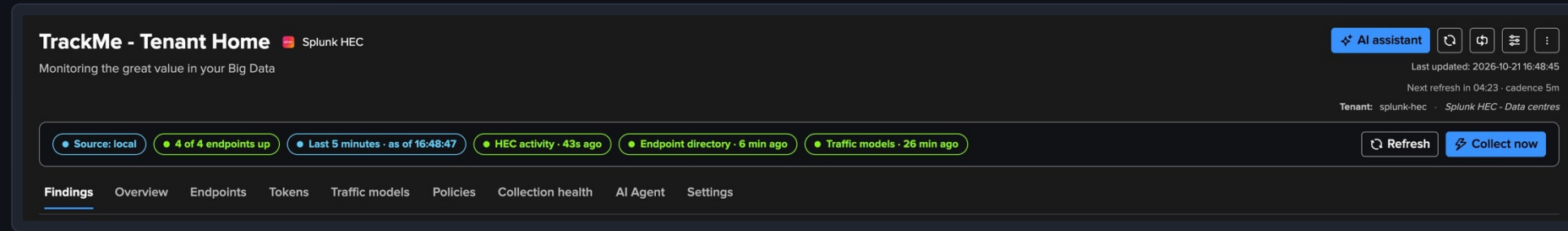
Review, then create — the first collection starts at once



- 1 The summary**
Source, indexes read, host patterns, pools, auto-join, splunkd API accounts.
- 2 Indexes and roles**
The tenant's TrackMe indexes, its owner and its roles.

- 3 Created**
Collections and reports are created and the first collection starts.
- 4 Within minutes**
Endpoints, pools and tokens appear in the workspace: Open workspace.

The workspace — nine tabs and a strip that sums it up



1 Source and endpoints up

local here; 4 of 4 endpoints up.

2 Last 5 minutes

The statistics of the last five minutes up to now, read on every refresh.

3 The three jobs

HEC activity, Endpoint directory and Traffic models with their last run.

4 Collect now

For the TrackMe administrators of the tenant.

5 Nine tabs

Findings, Overview, Endpoints, Tokens, Traffic models, Policies, Collection health, AI Agent, Settings.

02

PART 2 · ENDPOINTS, POOLS & TOKENS

The deployment at a glance, any endpoint up close

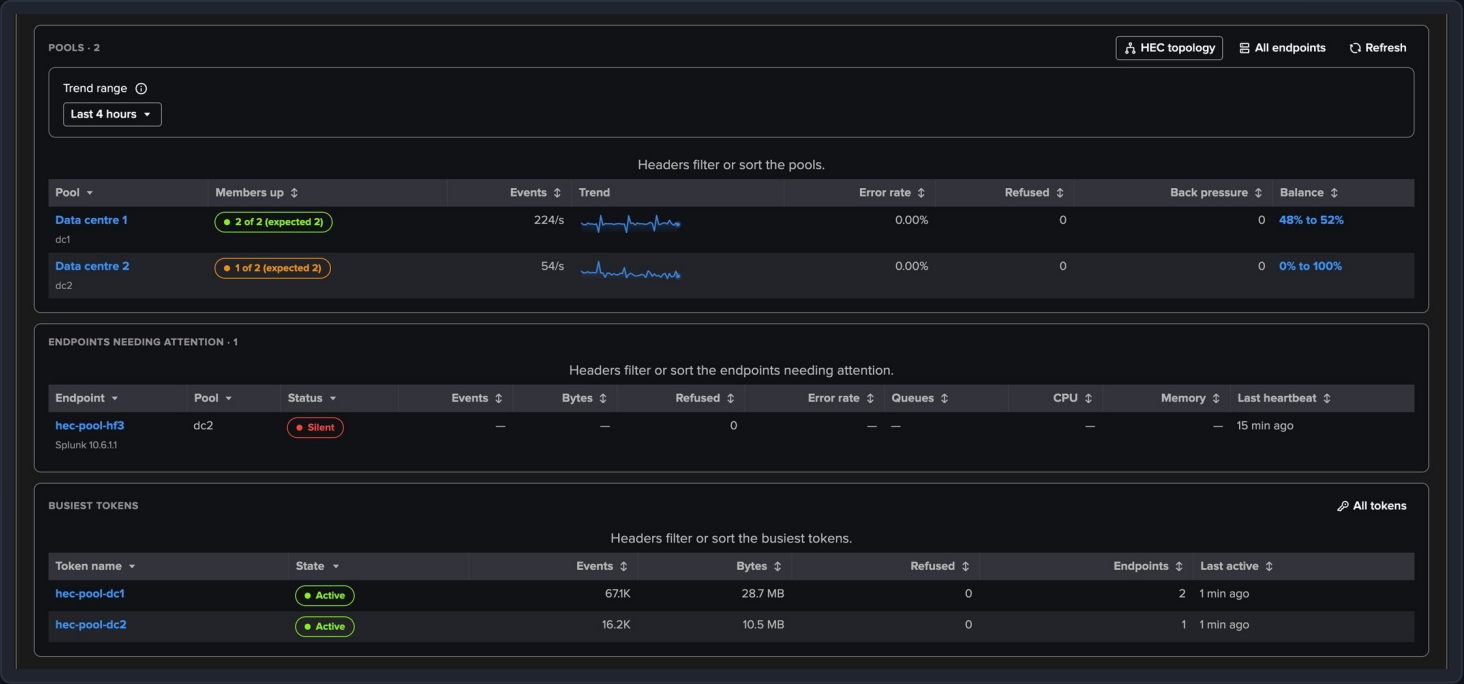
The overview, the endpoints and one endpoint with its ingestion pipeline, a pool and how it spreads its traffic, the tokens, and the maps each tab draws.

The overview — the HEC questions, answered at a glance



- 1 Endpoints up**
A heartbeat in the window.
- 2 Events, bytes, active tokens**
The traffic of the last five minutes.
- 3 Refused requests, server busy**
Load balancer probes excluded; HTTP 503 when HEC could not queue.
- 4 Back pressure**
Endpoints whose queues backed up.
- 5 Traffic over time**
Events, bytes, refusals and server busy over a range you choose.

Pools, endpoints needing attention, busiest tokens



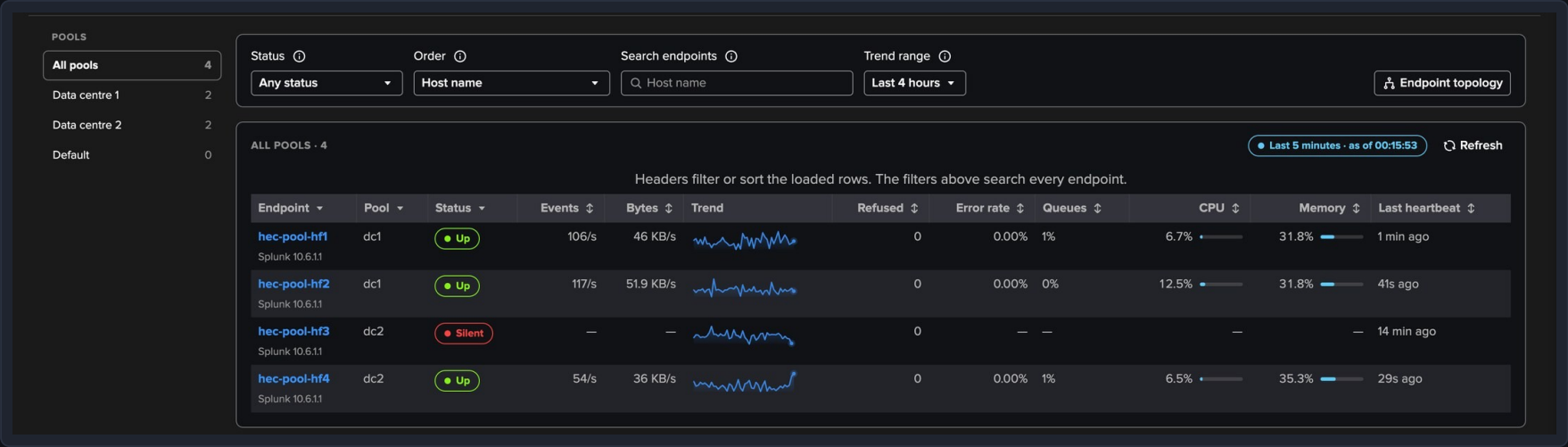
1 Pools
Members up against the expected size, traffic, refusals, back pressure, trend.

2 Balance
The smallest and largest member shares: 48 % to 52 % is an even split.

3 Needing attention
Not up, refusing, backed up, dropping oversized events, failing destinations.

4 Busiest tokens
By name, with the endpoints that received them.

Endpoints — by pool, filtered and paged by the server



- 1

Pools sidebar

All pools, one pool, or the discovered endpoints.
- 2

Status

Up, late, stalled or silent: hec-pool-hf3 is silent.
- 3

Order

Most events, most refusals, highest error rate, back pressure first.
- 4

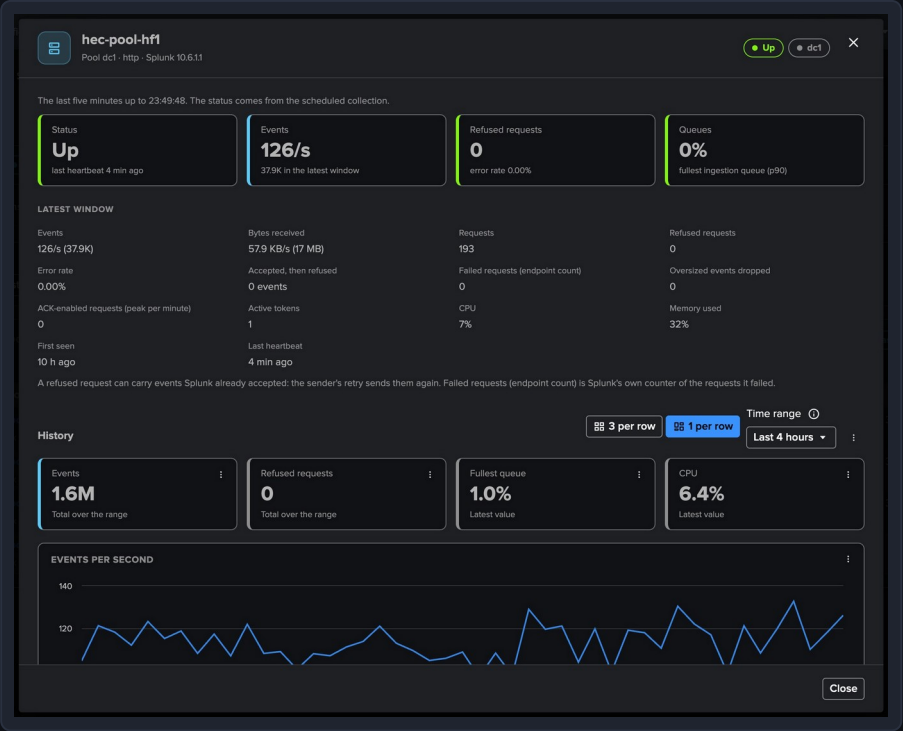
Per endpoint

Pool, traffic and its trend, refusals, error rate, fullest queue, last heartbeat.
- 5

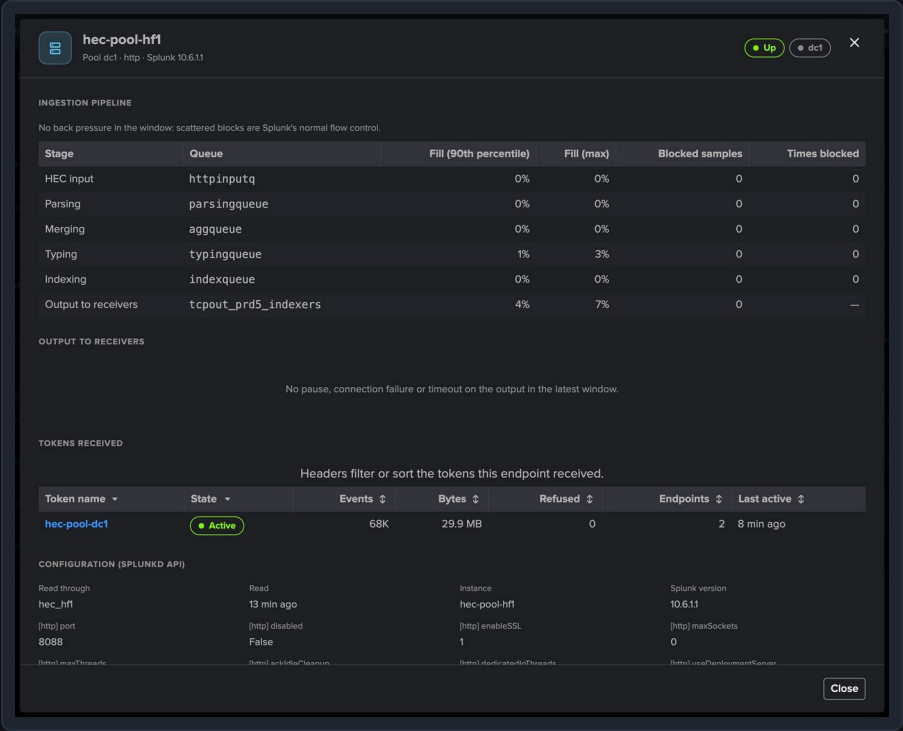
CPU and memory

Host-wide, with a bar that turns amber from 75 % and red from 90 %; a dash without telemetry.

One endpoint — its window, its pipeline, its output

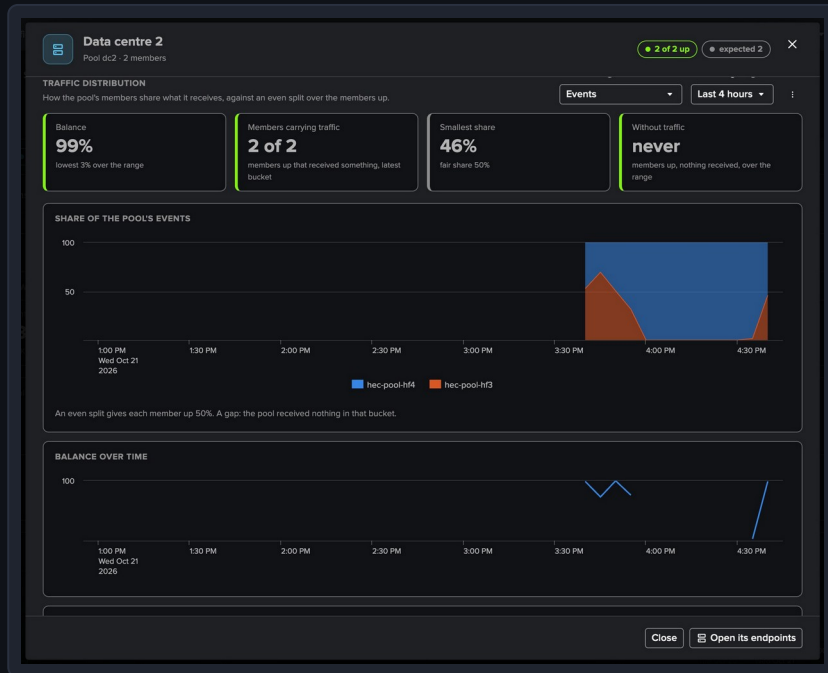


- 1 Latest window
Events, bytes, requests, refused, error rate, CPU, memory, last heartbeat.
- 2 Refused requests
By HEC reply code, in plain words.



- 3 Ingestion pipeline
Each queue of the HEC path, its fill, the bottleneck and how busy its thread was.
- 4 Output and history
Destinations and receivers; events, refusals, fullest queue and CPU over time.

A pool — is the load balancer spreading the traffic?



1

Balance

100 for an even split over the members up, 0 when one member takes everything.

2

Members carrying traffic

And the smallest share against the fair share.

3

Members up that received nothing

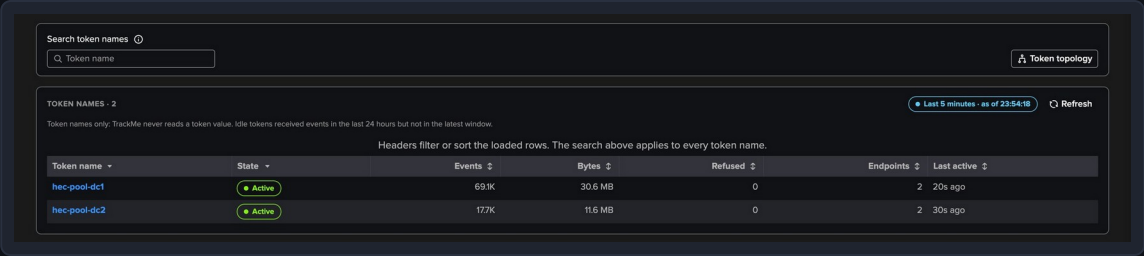
What Pool imbalance watches.

4

Share of each member over time

With the balance over time and the members table.

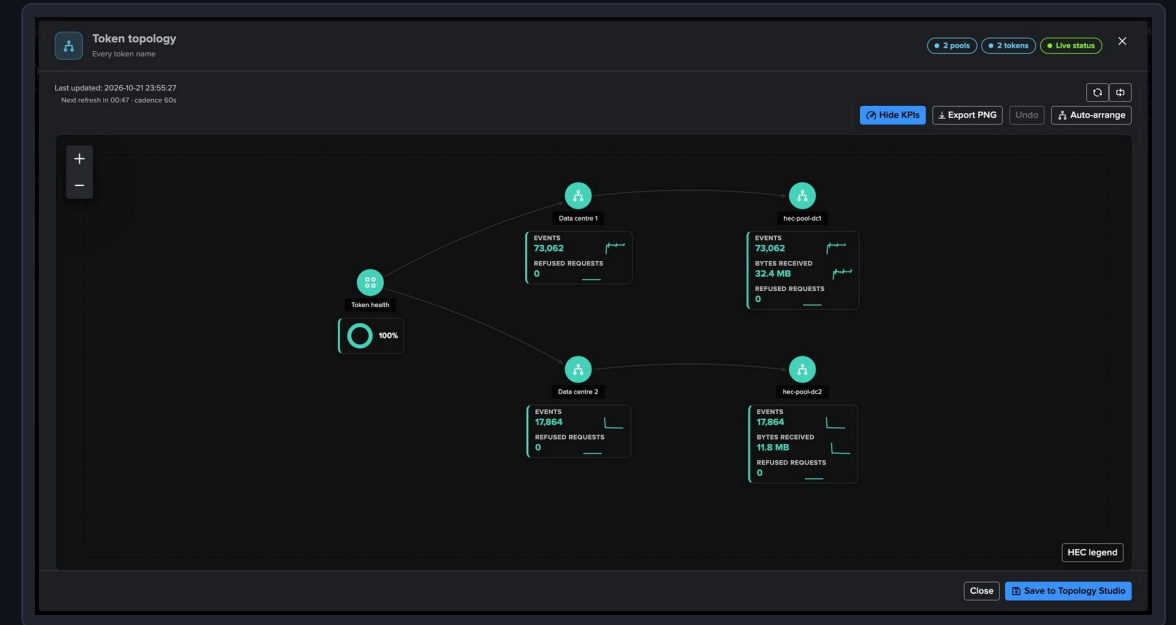
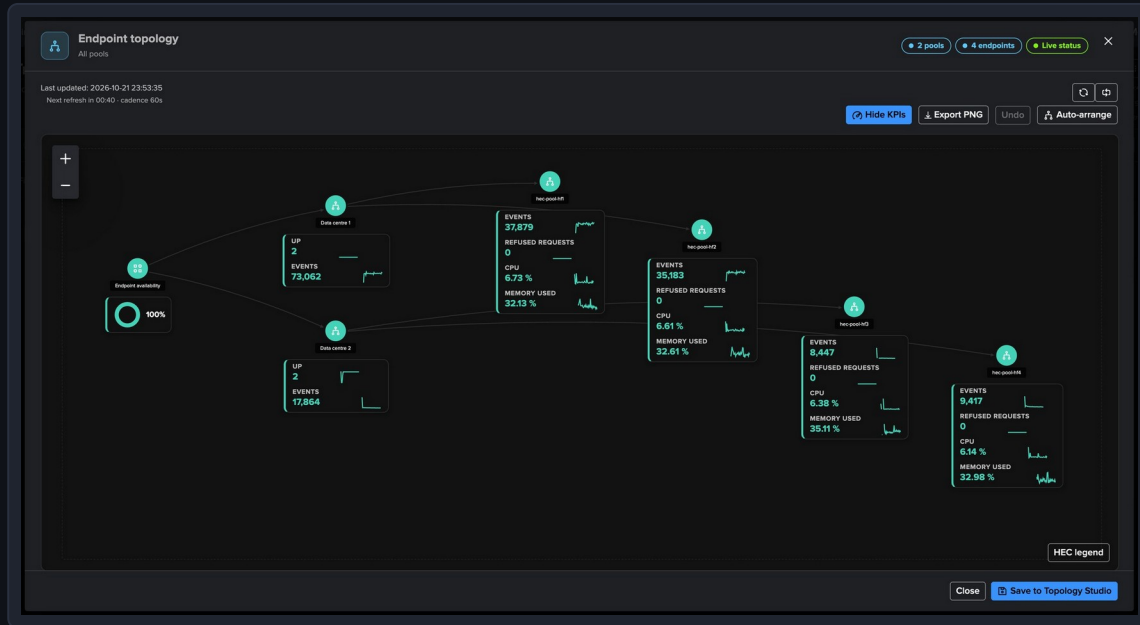
Tokens — known by name, never by value



- 1 Every token seen in 24 hours
The pools and endpoints that received it, events, bytes, refusals.
- 2 The token drilldown
Its state, its last 24 hours, its traffic over time.

- 3 Endpoints involved
Each endpoint's share of the token's events against an even split.
- 4 Errors
Reply codes in plain words, the index refused, the client addresses.

Each tab draws its own map — one click, live



1 HEC topology

From the Overview: every pool and its endpoints, with the availability donut.

2 Endpoint topology

The endpoints the tab selects, with events, refusals, CPU and memory.

3 Token topology

Each token under the pools that received it, with its traffic.

4 Save to Topology Studio

A view like any other: share it, edit it, add Topology Alerts.

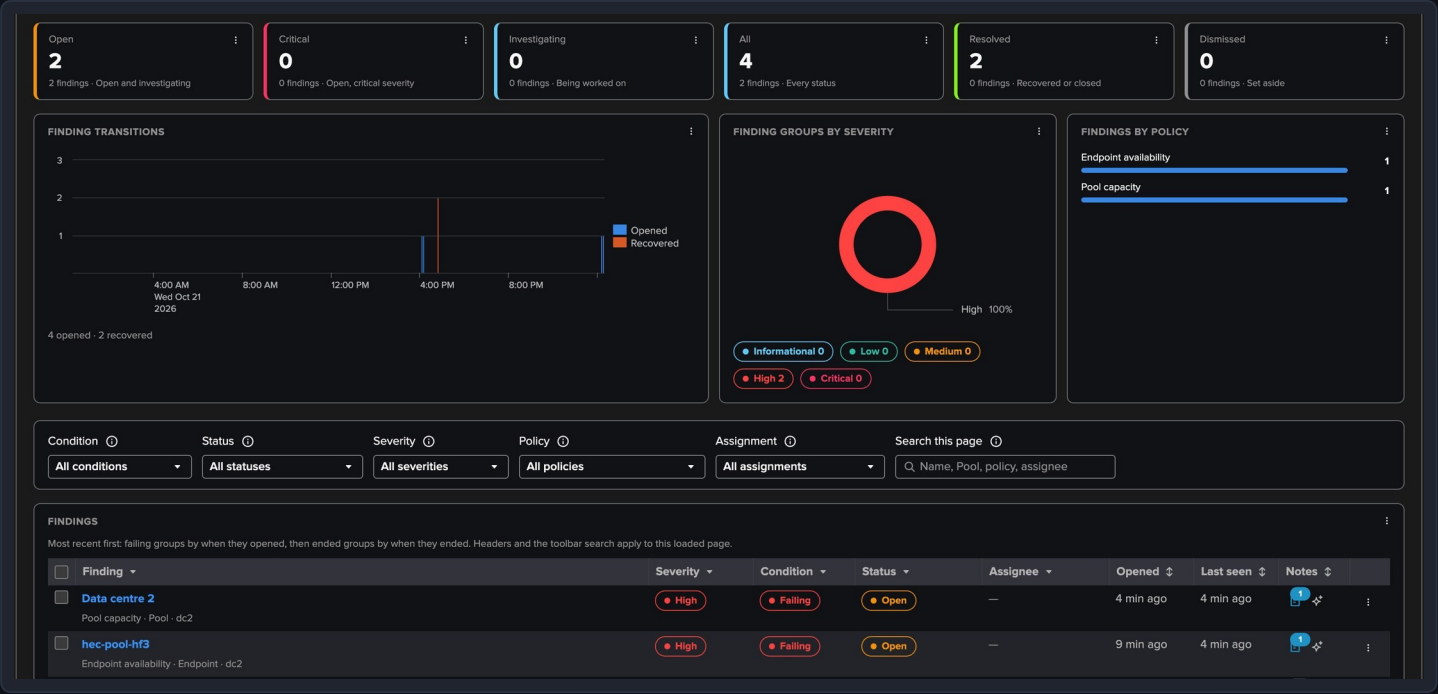
03

PART 3 • FINDINGS

A heavy forwarder stops — what opens, and why

hec-pool-hf3 stopped on purpose: Endpoint availability on the endpoint, Pool capacity on its pool, how each is evaluated, the topology and the AI Agent.

The Findings tab — two findings, one cause



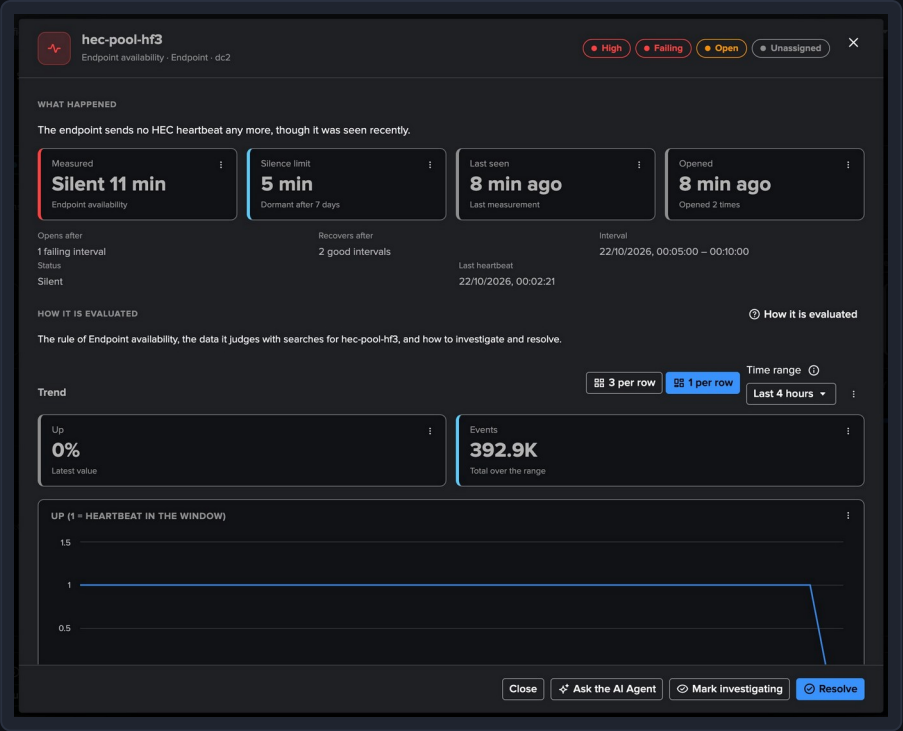
1 **Open, critical, investigating**
The counters of the tenant's findings.

2 **Transitions and severities**
Opened and recovered over time, Finding groups by severity, findings by policy.

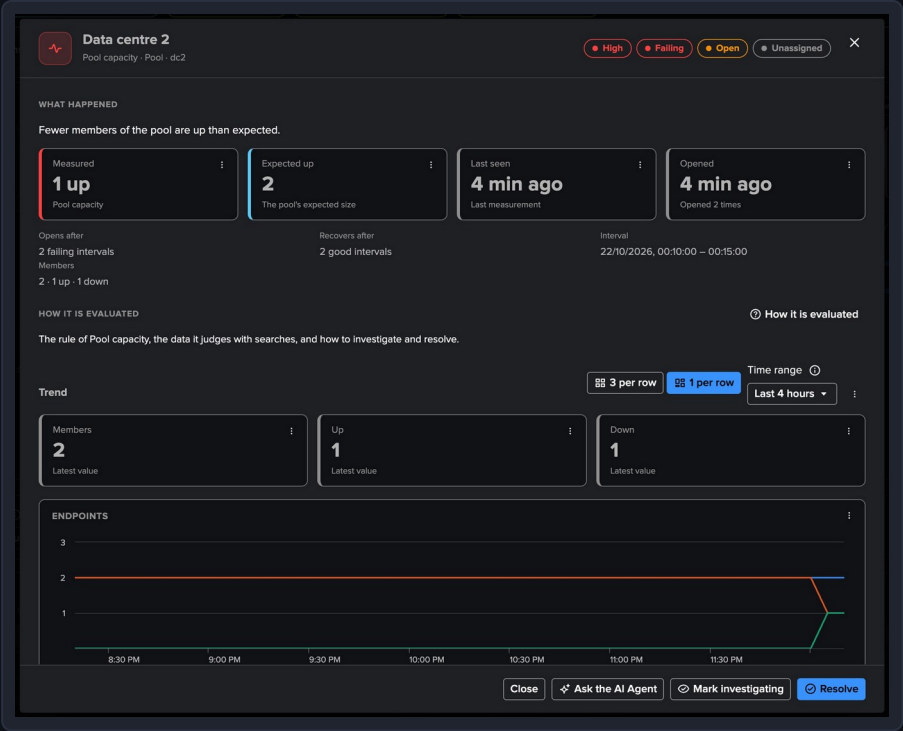
3 **Endpoint availability**
hec-pool-hf3, High: silent for longer than 5 minutes.

4 **Pool capacity**
Data centre 2, High: 1 of 2 members up.

The investigation — the decision in plain words

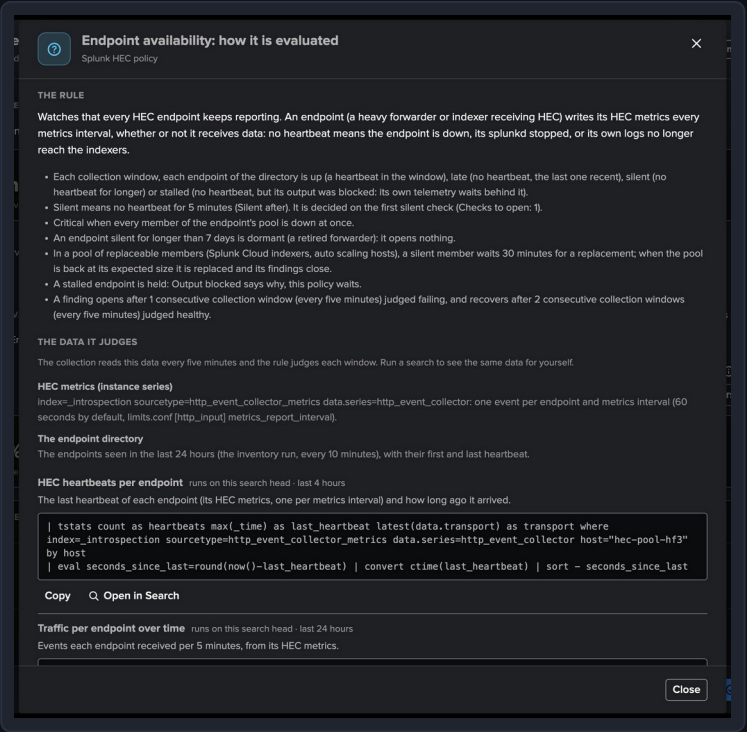


- 1 What happened
The endpoint sends no HEC heartbeat any more, though it was seen recently.
- 2 Measured against what
Silent 11 minutes against a 5-minute limit; 1 up against an expected 2.



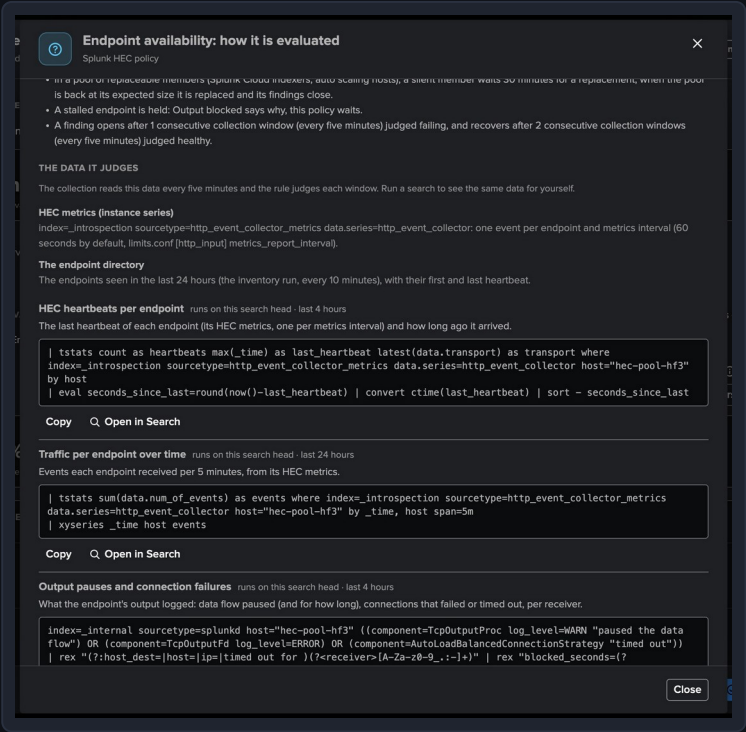
- 3 Opens and recovers
The checks to open and to recover, the interval judged.
- 4 Trend and actions
The KPIs the policy judges; Ask the AI Agent, Mark investigating, Resolve.

How it is evaluated — the rule, and the searches to check it



1 **The rule**
What the policy decides with its current settings.

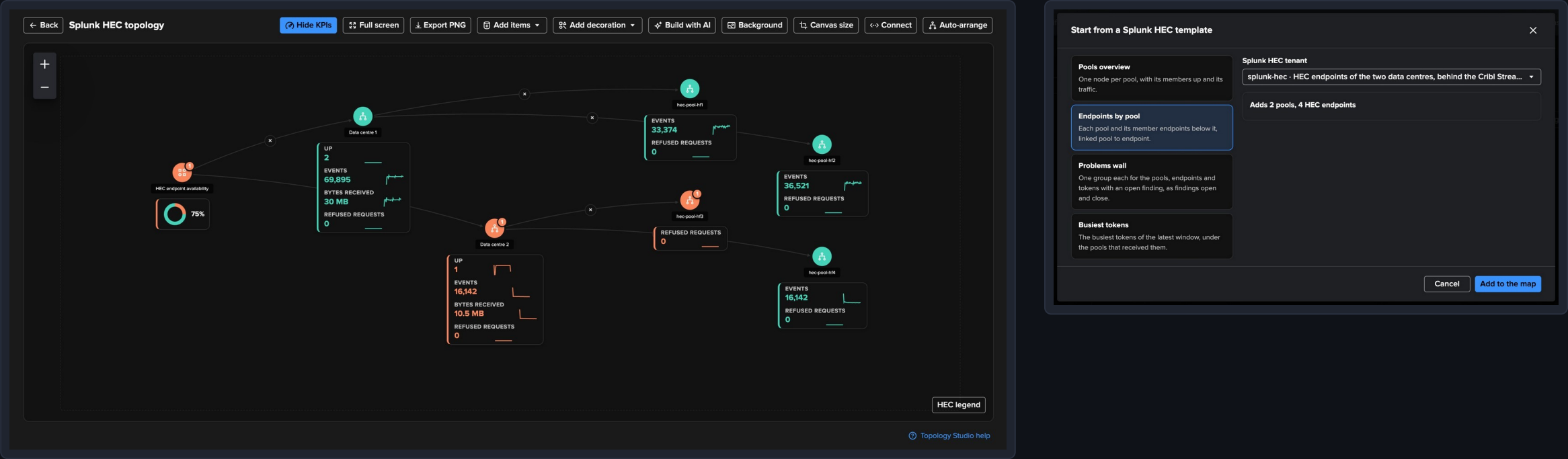
2 **The data it judges**
The telemetry each decision reads.



3 **Ready-to-run searches**
Narrowed to hec-pool-hf3, with Open in Search and Copy.

4 **Investigate and resolve**
Causes, resolutions and related policies.

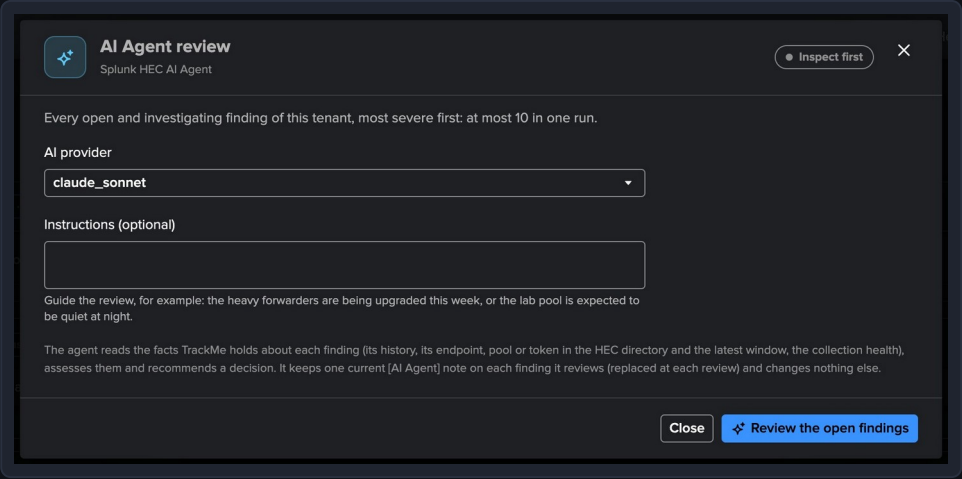
The data centres on a map — generated, saved, live



- 1 HEC topology, saved**
Generated from the Overview, saved to Topology Studio.
- 2 Live status**
Data centre 2 and hec-pool-hf3 orange with their finding badges; availability at 75 %.

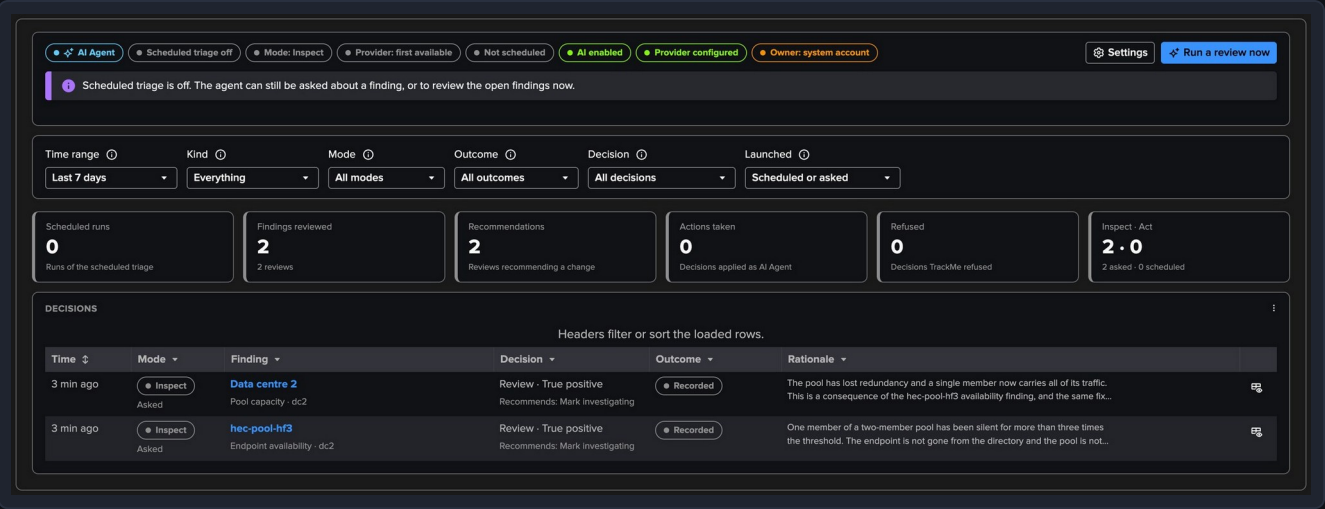
- 3 Sparklines on the nodes**
Members up, events, bytes and refusals per pool and endpoint.
- 4 Four templates**
Pools overview, Endpoints by pool, Problems wall, Busiest tokens.

The Splunk HEC AI Agent — reviews the findings like an operator



1 **Run a review now**
Every open finding, most severe first; optional instructions.

2 **Inspect first**
It assesses and recommends; only Act applies what the tenant allows.



3 **Its decisions**
True positive for both findings, recommending Mark investigating.

4 **One [AI Agent] note**
Kept current on each finding it reviews.

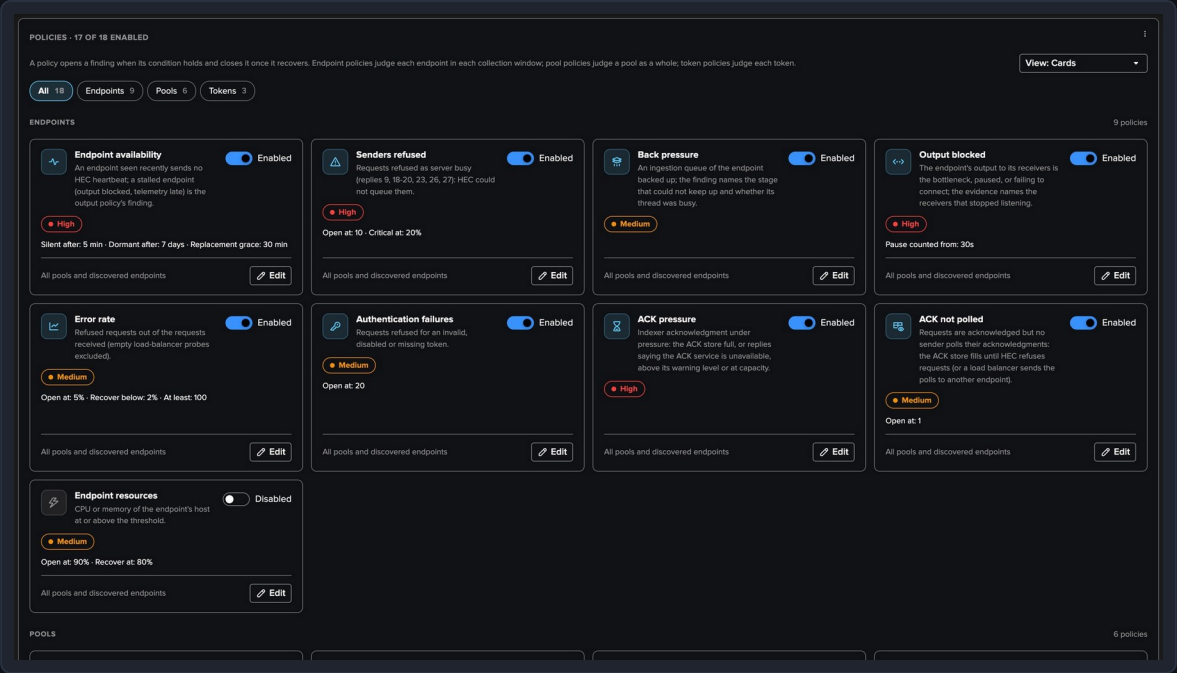
04

PART 4 • OPERATE

Policies, models and the collection

Eighteen built-in policies, the tokens that must keep sending, the traffic models and their tuning, collection health and the splunkd API.

Eighteen built-in policies — endpoints, pools, tokens



1 **Endpoints (9)**
Availability, senders refused, back pressure, output blocked, error rate, authentication, ACK.

2 **Pools (6)**
Capacity, imbalance, traffic drop, traffic outlier, token consistency, configuration drift.

3 **Tokens (3)**
Rejected data, token silence, token traffic outlier.

4 **17 on out of the box**
Endpoint resources is off: it watches the host, not HEC.

5 **Editor**
Severity, checks to open and recover, conditions, pools, bank holidays.

Token silence — the tokens that must keep sending

**Token silence**

Tokens - A selected token sent no events for longer than the silence period.

● Enabled

● High

● Unsaved changes

×

TOKENS

Tokens to watch

hec-pool-dc1 X

hec-pool-dc2 X

Type a token name or a pattern (* any characters, ? one), case-sensitive, 100 at most; the list offers the tokens seen in the last 24 hours. A name not seen sending in the last 24 hours is silent. Longer or changing lists belong in a lookup table below.

Lookup tables

No lookup table.

 Add a lookup table

Each row of the field is a token name (or a pattern with * and ?), case-sensitive; up to 100,000 names each, read again every 15 minutes. A lookup that cannot be read keeps its last names and is reported in Collection health.

 Test selection

Runs as a search: reads the lookups now and matches the tokens seen in the last 24 hours. Saves nothing.

●

 2 names (2 exact, 0 with * or ?) select 2 tokens: 2 sending, 0 silent for more than the silence period, 0 not seen sending in the last 24 hours (silent at once).

Each source and what it gave.

Source	Status	Names	Seen in 24 h	Ignored
Typed names	—	2	2	—

Selected tokens (up to 50, not seen first)

Token	State	Last active	Events 24 h	Selected by
hec-pool-dc1	● Sending	21/10/2026, 15:50:02	539.5K	hec-pool-dc1
hec-pool-dc2	● Sending	21/10/2026, 15:50:17	3M	hec-pool-dc2

Close

Save policy

- 1

Tokens to watch

Names or patterns typed in the policy: hec-pool-dc1 and hec-pool-dc2.
- 2

Lookup tables

Up to 20, local or remote, Exact or Wildcard, re-read every 15 minutes.
- 3

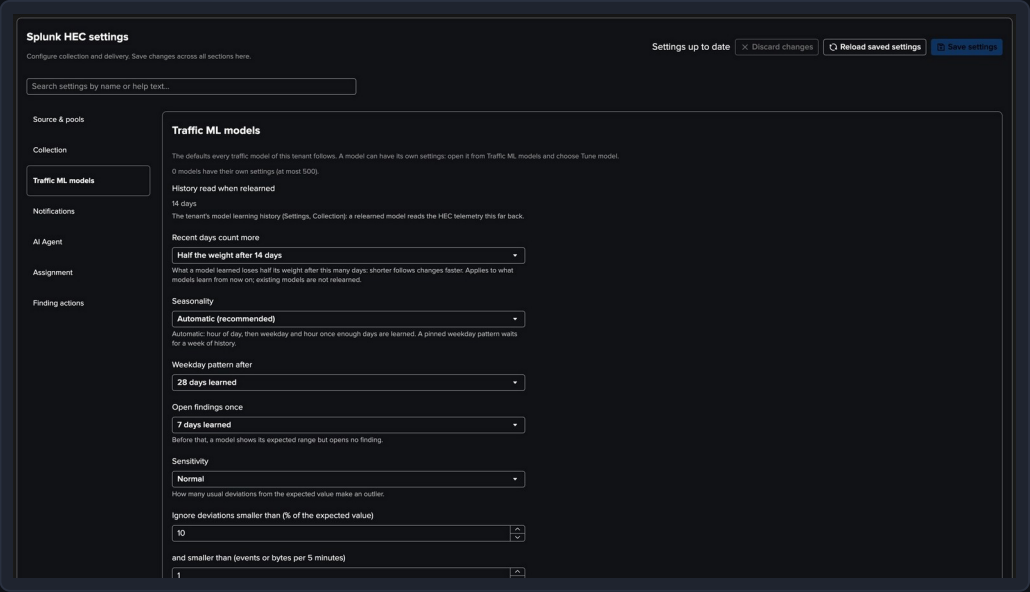
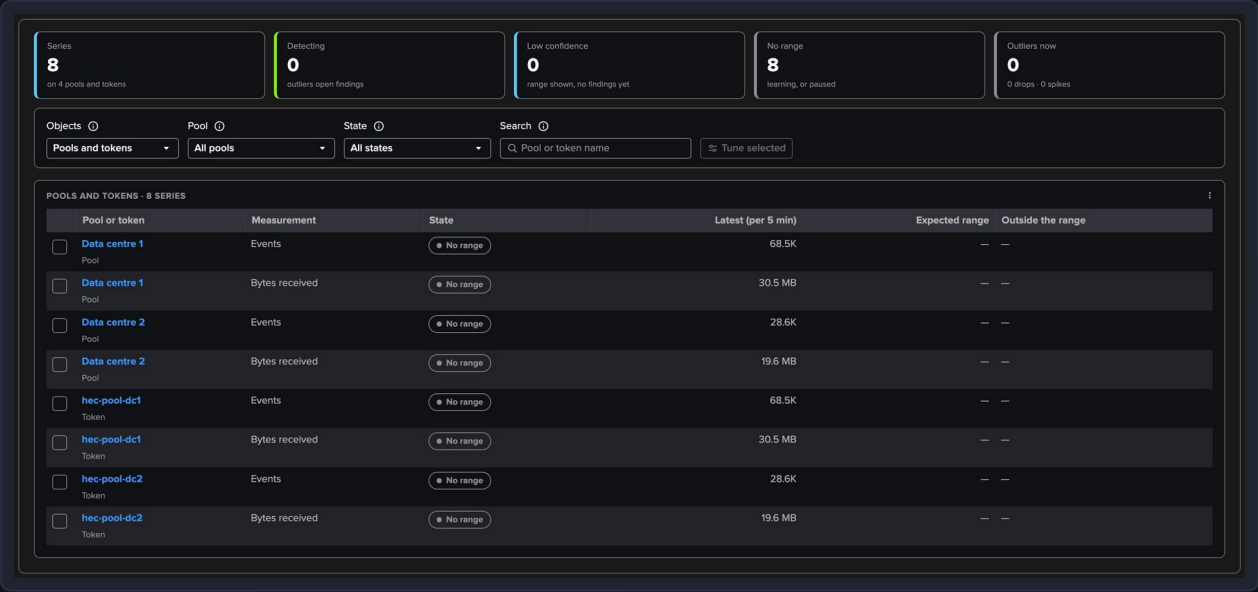
Silent at once

A name selected exactly but not seen sending in 24 hours.
- 4

Test selection

Runs as a search, reads every lookup now; saves nothing.

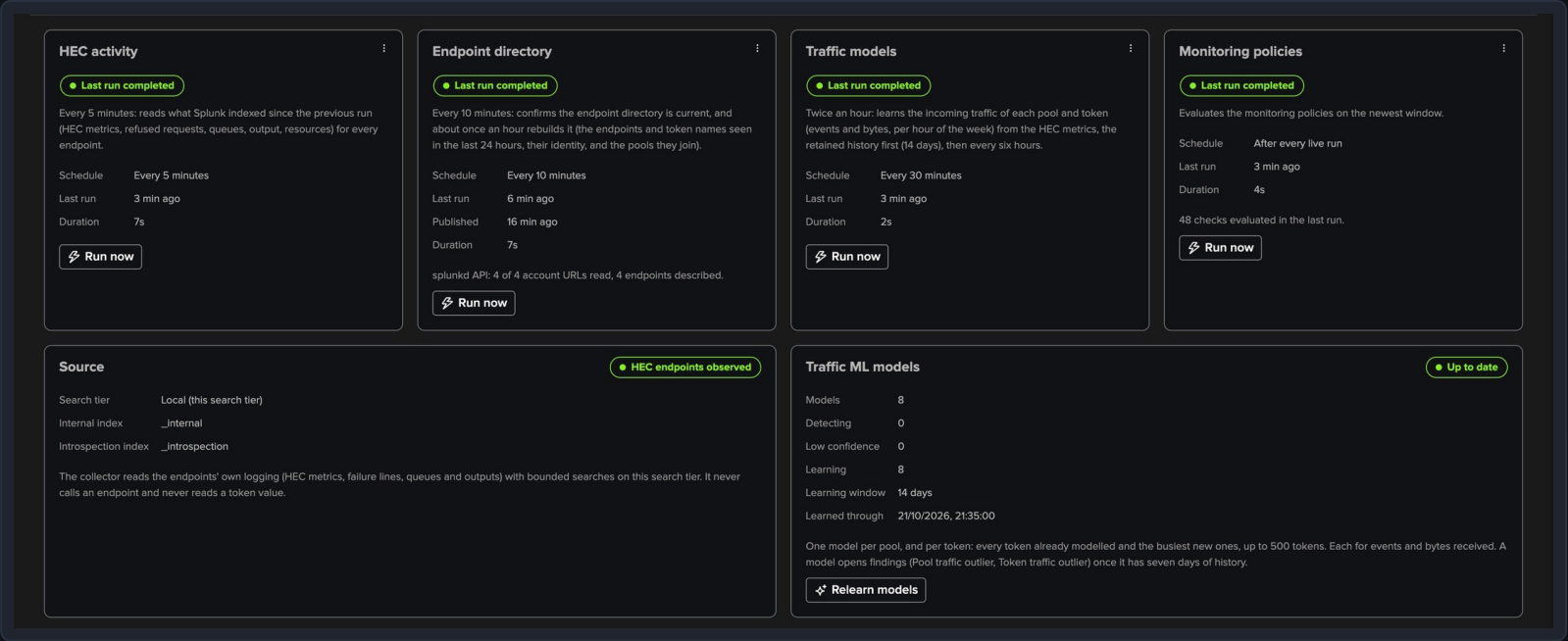
Traffic models — what each pool and token usually receives



- 1 One model per pool and token
Events and bytes received, per hour of the week; 500 tokens, busiest first.
- 2 Learned from _introspection
The Model learning history on the first runs, then every six hours.

- 3 Expected range on the charts
Pool and token histories and the traffic outlier findings.
- 4 Tuning and Not an anomaly
Defaults in Settings; Tune model per model; excluded periods.

Collection health — every job, the source and the models



- 1

One card per job
HEC activity, Endpoint directory, Traffic models, Monitoring policies.
- 2

Schedule and last run
With its outcome, the next run and Run now.
- 3

Run budgets
Each run keeps to its interval less 30 seconds and ends before the next.
- 4

Source and models
The indexes read, HEC endpoints observed, models and their confidence.

The splunkd API — optional, read only, configuration compared

hec-pool-hf1

Pool dc1 • http • Splunk 10.6.11

Up

dc1

×

Parsing	parsingqueue	0%	0%	0	0
Merging	aggqueue	1%	1%	0	0
Typing	typingqueue	1%	1%	0	0
Indexing	indexqueue	0%	0%	0	0
Output to receivers	tcpout_prd5_indexers	2%	21%	0	—

OUTPUT TO RECEIVERS

No pause, connection failure or timeout on the output in the latest window.

TOKENS RECEIVED

Headers filter or sort the tokens this endpoint received.

Token name ▾	State ▾	Events ↕	Bytes ↕	Refused ↕	Endpoints ↕	Last active ↕
hec-pool-dc1	Active	68K	29.6 MB	0	2	6 min ago

CONFIGURATION (SPLUNKD API)

Read through	Read	Instance	Splunk version
hec_hf1	3 min ago	hec-pool-hf1	10.6.11
[http] port	[http] disabled	[http] enableSSL	[http] maxSockets
8088	False	1	0
[http] maxThreads	[http] ackIdleCleanup	[http] dedicatedIoThreads	[http] useDeploymentServer
0	true	2	0

Token	Status	Index	Allowed indexes	ACK	Output group
hec-pool-dc1	Enabled	main	Any	Off	—

Read only, during each inventory: TrackMe never reads a token value and never changes a setting.

Close

- 1

Remote accounts on the management port

Up to 20, one URL per endpoint.
- 2

What is read

server/info, the global HEC settings, the HEC tokens by name.
- 3

Token value dropped

While the answer is parsed: never stored, logged or returned.
- 4

Two policies

Token consistency and the HEC settings of Configuration drift.
- 5

Never Splunk Cloud

Its indexers have no customer-reachable management port.

One tenant, every HEC endpoint, pool and token

● Set up

a tenant on the local search tier, two pools, the endpoints found from their heartbeats

● Followed

endpoints, pools and tokens, one endpoint's pipeline, a pool's balance

● Judged

an endpoint silent and its pool short of a member, explained and reviewed

● Operated

eighteen policies, token silence, traffic models, collection health, the splunkd API

What made it work

Splunk's own telemetry

No agent, no new data, no credential to the endpoints: the HEC metrics and logs Splunk already writes.

Pools judged as a whole

Members up against the expected size, shares of the traffic, versions and settings compared.

Bounded at any size

A few searches per five-minute window, whatever the number of endpoints and tokens.