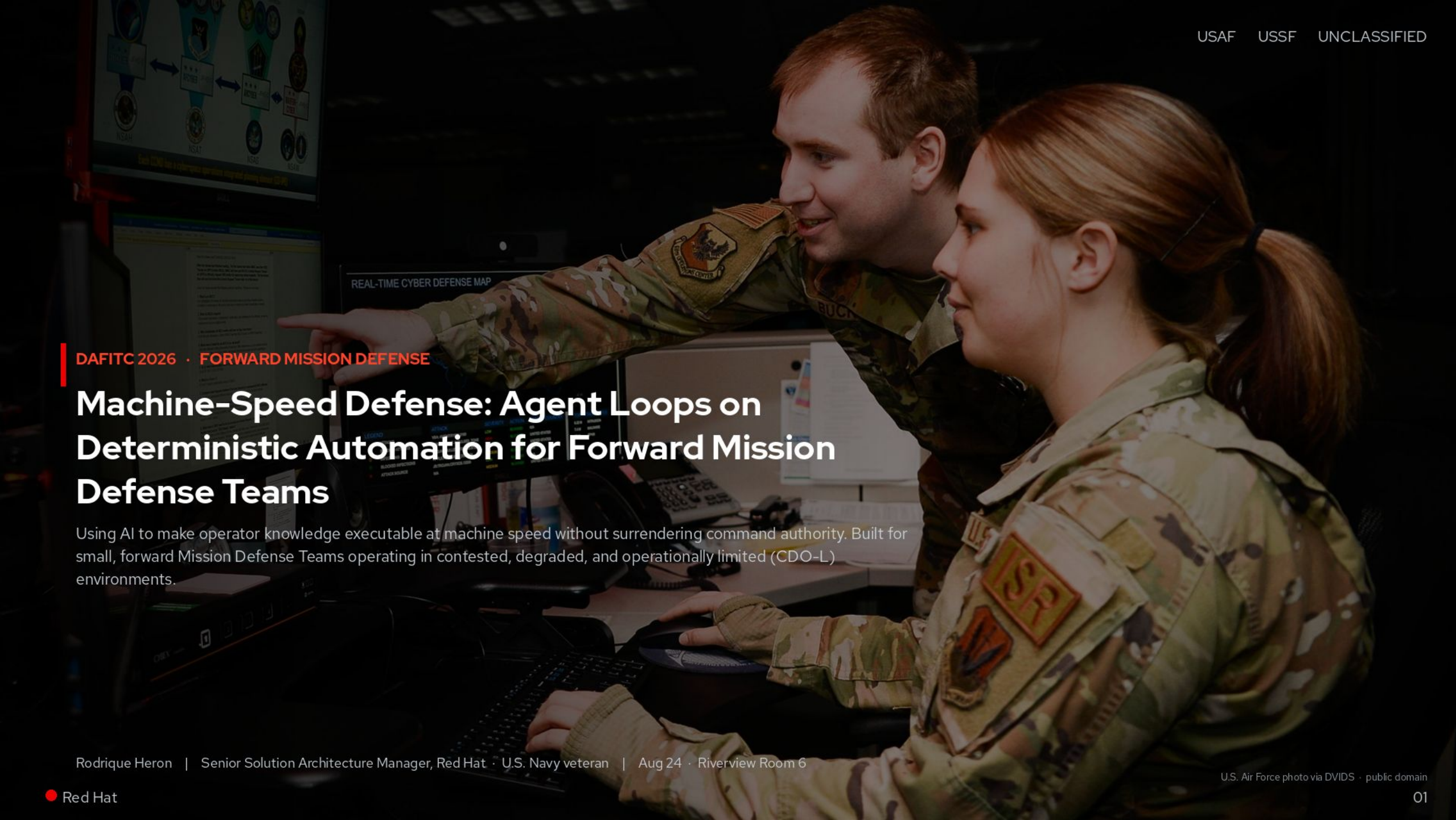


DAFITC 2026 · FORWARD MISSION DEFENSE

Machine-Speed Defense: Agent Loops on Deterministic Automation for Forward Mission Defense Teams

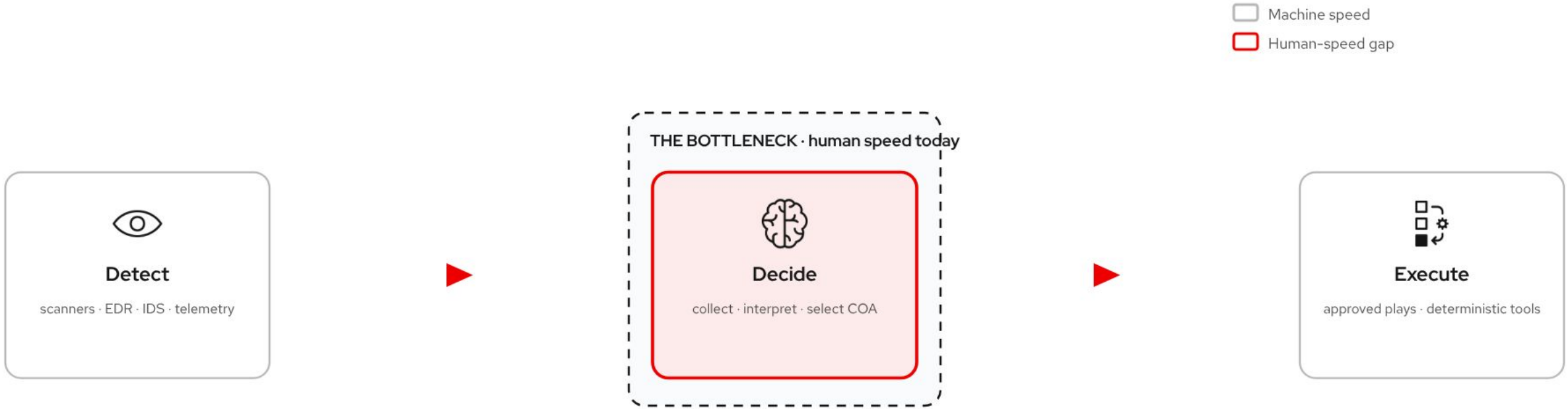
Using AI to make operator knowledge executable at machine speed without surrendering command authority. Built for small, forward Mission Defense Teams operating in contested, degraded, and operationally limited (CDO-L) environments.

Rodrique Heron | Senior Solution Architecture Manager, Red Hat · U.S. Navy veteran | Aug 24 · Riverview Room 6

U.S. Air Force photo via DVIDS · public domain

Detection and execution run at machine speed; the middle decision runs at human speed

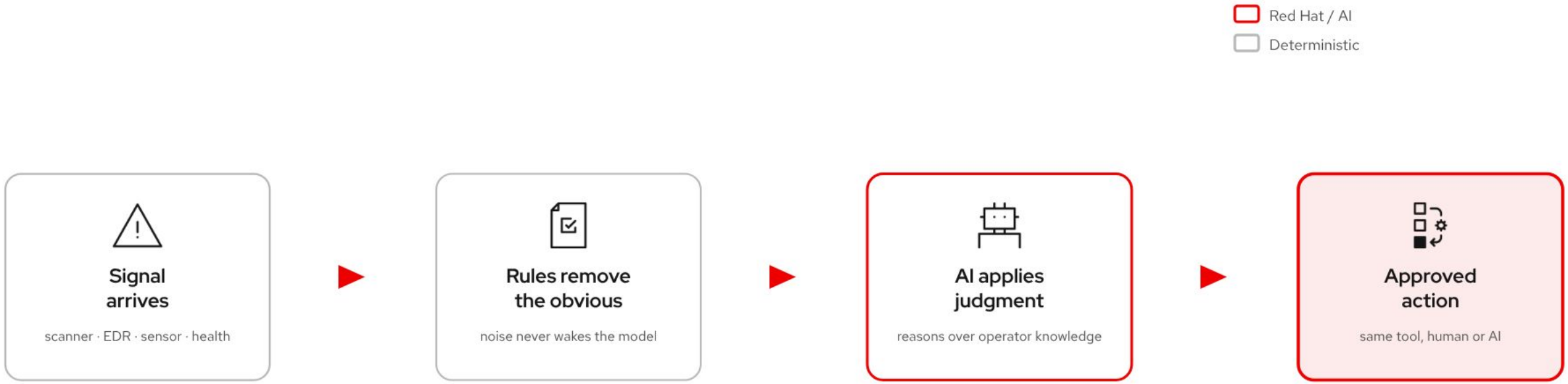
AI belongs in the reasoning gap between machine-generated signals and approved deterministic action.



Principle: More queues do not create tempo. AI closes the middle gap by applying encoded operator judgment.

AI is the reasoning layer that selects or recommends the approved action

Machine speed comes from removing human wait time from classify, investigate, select, and verify.



Flow: Faster classify › faster investigate › faster action › faster verify

The workload is growing while the defender's reaction window is collapsing

More vulnerabilities does not mean more exploitable risk; it means more noise, faster.

Volume: more to triage

- 2024: 40,077 CVEs published
- 2025: 48,244 CVEs published
- 2026: ~66,000 projected (FIRST)
- only ~1.4% first-time exploited (VulnCheck, 1H 2026)

Time: less to decide

- 2018-19: exploited +63 days after the patch
- 2021-22: +32 days
- 2023: +5 days
- 2025: -7 days, exploitation precedes the patch (Mandiant M-Trends 2026)

Why now: Identify the consequential condition and move from signal to verified response before the window closes.

A cyber event becomes an SRE-style system-state problem, resolved under pressure

The loop is for operational signals, not a narrow vulnerability-alert workflow.

Cyber signals

- scanner finds exposed service
- EDR sees suspicious behavior
- IDS flags anomalous traffic
- identity tool detects drift

Operational state

- service unhealthy
- storage degraded
- process changed
- route or interface changed
- config off baseline

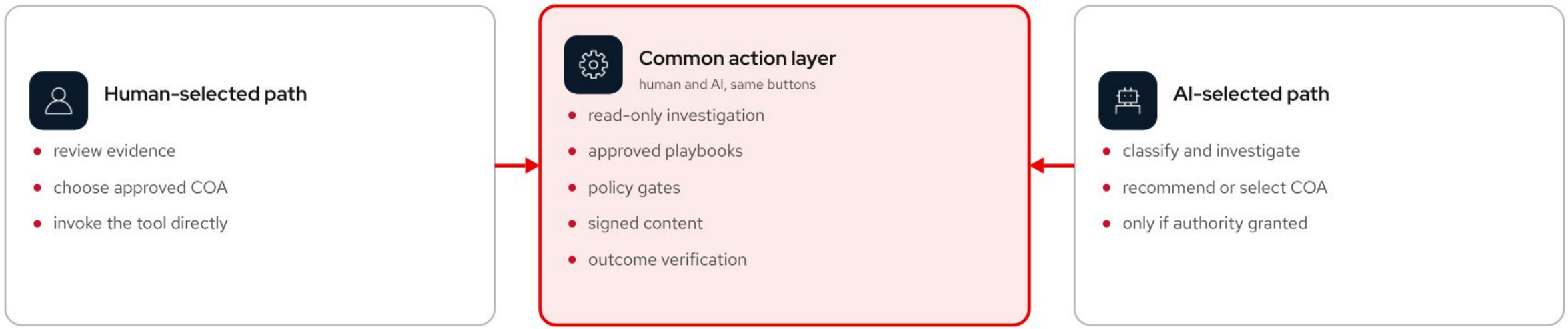
Response work

- classify the condition
- collect the right evidence
- select a COA
- execute through approved tooling
- verify the resulting state

Bad actor exploits exposure › system state changes › operator reasons, acts, verifies

The human and the model operate through the same deterministic tools

Trust shifts from generated action to authorized invocation.



Principle: The model gets permission to press approved buttons. It does not receive a new toolbox.

AI gives speed; senior Airman and MDT Chief experience defines the judgment applied

The hero is not the model. The hero is operator knowledge made executable.

What operators know

- which signals matter
- what is noise
- which checks come first
- what must never be automated
- what healthy means after

How it becomes executable

- classification rules
- runbooks
- YAML policy
- approved playbooks
- verification + escalation gates

How AI uses it

- applies judgment to the signal
- selects a constrained tool
- summarizes evidence
- recommends the next step
- hands off when authority ends

Lab proof: Editing one Markdown runbook changed the model's next tool choice: no code change, no model change, no restart.

Four enforceable layers keep the loop fast, deterministic, and accountable

The safety boundary is not one clever prompt: it is grammar, tools, policy, structured evidence, and control flow.



LAYER 4 · CONTROL & CONVERGENCE

5-round cap; at round 4 the investigation tools are removed. Escalate-to-human gate.



LAYER 3 · OUTCOME FEEDBACK

Structured evidence, not "success." Trimmed for the model, kept whole for the human.



LAYER 2 · VERIFICATION INTERFACE

7 runbooks + 9 classification rules as editable Markdown / YAML. The operator defines good.



LAYER 1 · DETERMINISTIC RUNTIME

Parameter grammar + 34 blocked patterns. Read-only automation, known behavior.

The model's authority

Classify the alert.

Select an approved action.

Interpret the evidence.

Request another bounded round.

The model chooses;
the substrate executes.

A machine-speed loop must verify the mission outcome, not merely tool completion

The check was human-authored, no AI in the path, yet it made failure faster and quieter.

Before: healthy but silent

- 1,617 received
- 1,617 pending
- 0 sent
- proves the process is alive

healthy for 197 days (~161 broken)

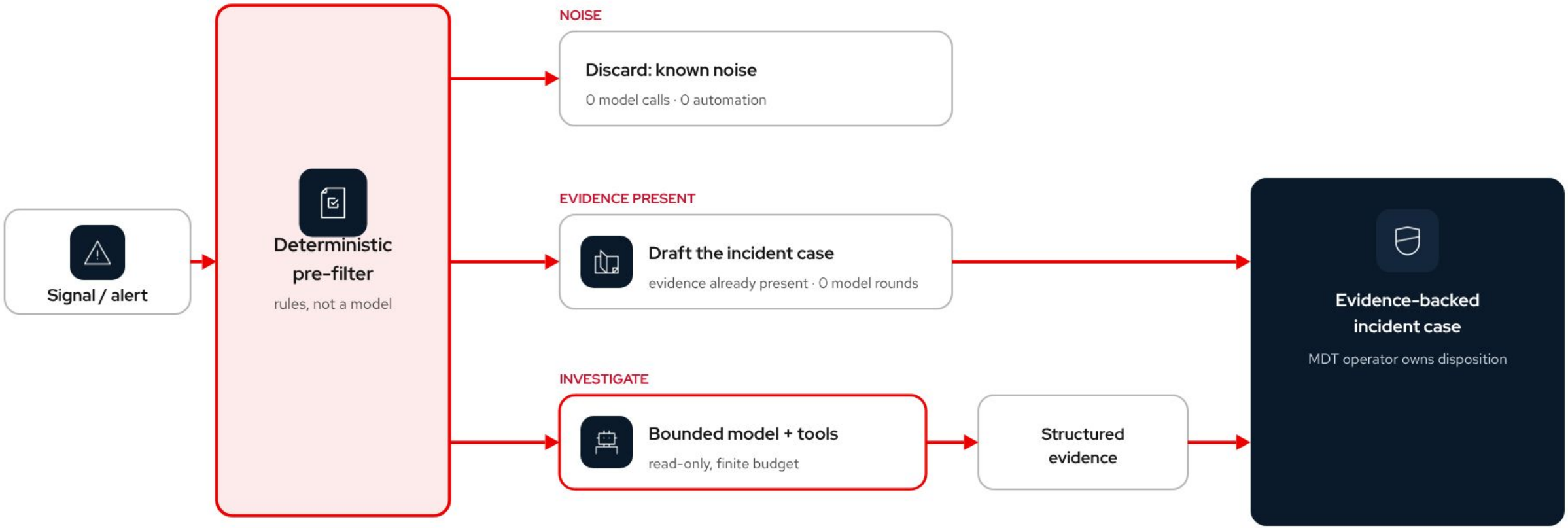
After: the check learned to fail

- 1,785 received
- 0 pending
- 1,785 sent
- reports degraded when delivery fails

Wrong vs mission question: Not 'is the process alive?' but 'did the intended recipient actually receive the alert?'

One signal takes the cheapest safe path

Rules first; the model wakes only where reasoning adds value. The cheapest inference is the one you do not run.



Principle: The cheapest inference is the one you do not run. The safest autonomy is the authority you have not granted yet.

The lab validates bounded investigation on mission-analogous workloads

A lab pilot on real, always-on workloads, not an operational MDT deployment.

Mission-analogous workloads

- live camera-feed workload (ISR / full-motion video analog)
- network security appliance (network-defense sensor analog)
- service + storage health (mission-system state analog)

The stack I ran

- RHEL on an NVIDIA GPU
- Red Hat AI Inference, local serving
- Granite 4.1 8B and Qwen3-32B
- Ansible Automation Platform for actions

What I measured

- 7 runbooks + 9 classification rules
- 12 read-only investigation tools
- bounded 5-round loop; tools pulled at round 4
- human disposition on every non-noise case

The same deterministic control model the mission edge needs, exercised on real workloads at lab scale

Context and control changed outcomes more than model size

Eight hand-written alerts on one fleet. Not a general model-equivalence claim.

Model size was not the dominant variable

- Granite 4.1 8B: 8/8 (20 rounds)
- Qwen3-32B: 8/8 (18 rounds)
- a small local model matched the larger one
- metric: did it investigate when expected?

Control and context moved the result

- Granite defaults: 7/8
- temperature 0: 6/8
- temperature 0 + retry: 8/8
- one runbook edit changed the next tool choice

serving-runtime swap moved 3/8

Takeaway: Context is a text file you can improve. Model size is a purchase order.

The output of the loop is not an AI answer; it is an evidence-backed incident case

The incident case is where machine speed meets human command.

What the human gets today

- a raw alert
- a severity
- a host
- a queue position
- then reconstructs everything

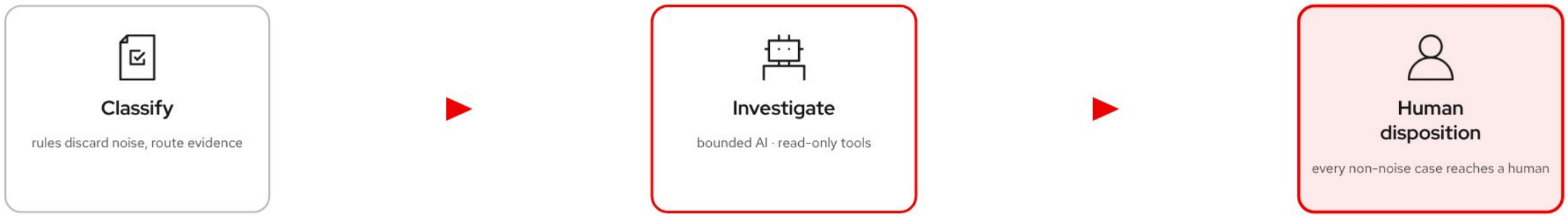
Cyber Incident Triage Package

- condition + classification
- mission context + source evidence
- AI assessment beside tool output
- runbook applied · COA
- verification result · human decision

System of record: Sensors, rules, AI, tools, verification, and human decisions all write to the case.

The pilot needs no novelty detector because it never grants autonomous disposition

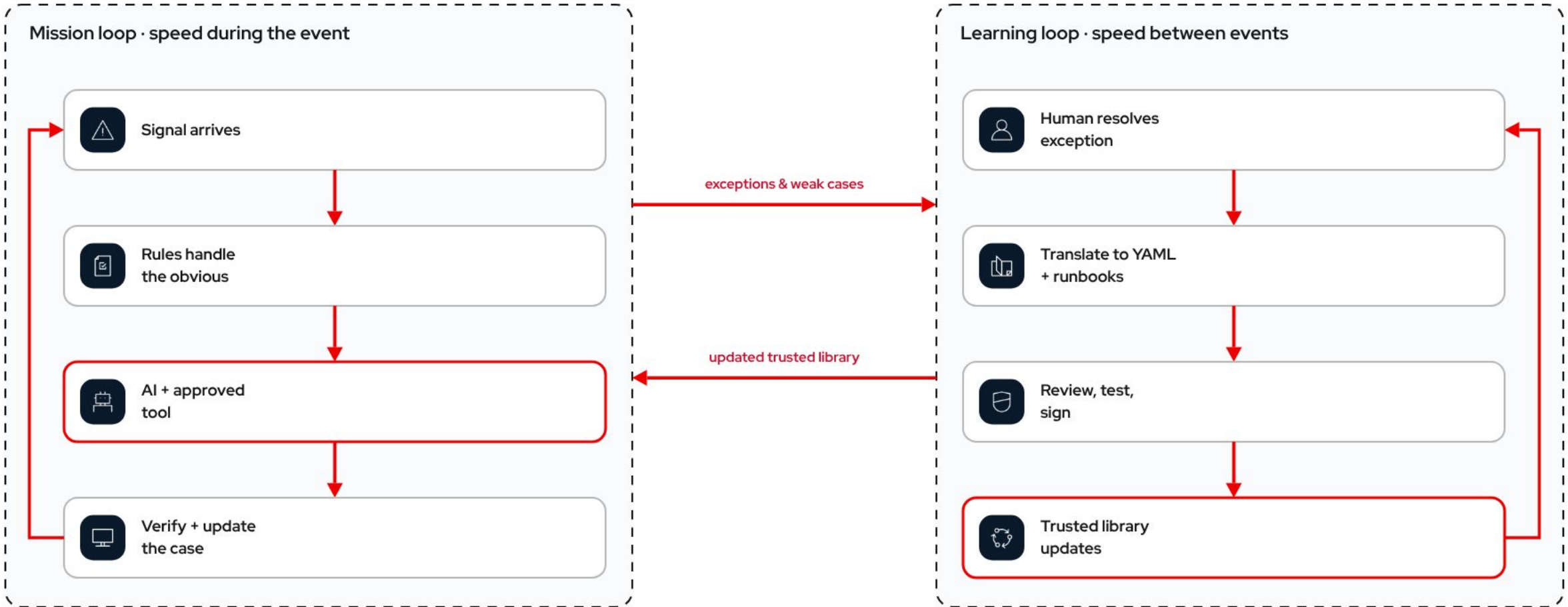
Use AI to compress investigation before you trust it with response authority.



Guarantees: Claim beside source · recurrence suppression · forced convergence at round 4

The mission loop moves at speed; the learning loop turns experience into automation

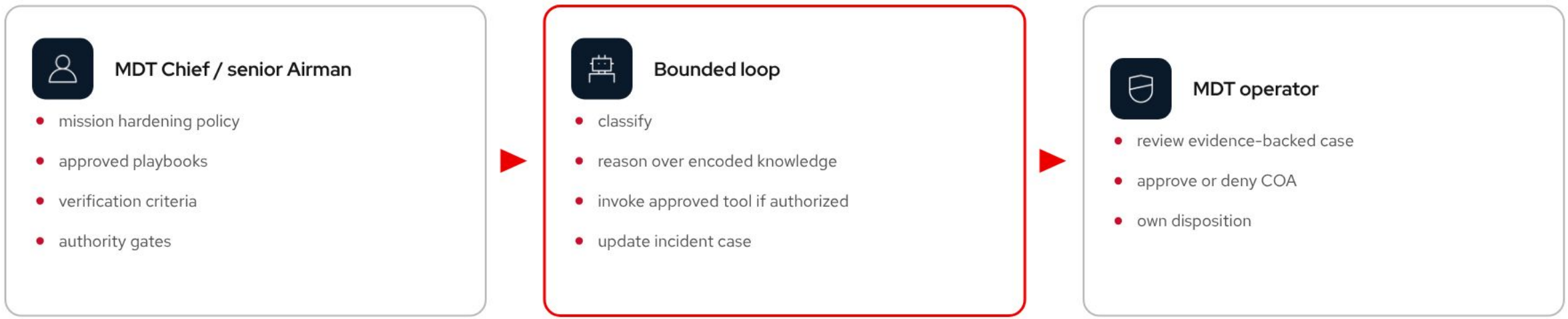
AI helps twice: it applies encoded knowledge during the event and helps encode new knowledge after.



Goal: Not fewer operators: faster classification, investigation, and verified resolution, with more time on exceptions.

MDT Chiefs define the defense system; MDT operators own exceptions and disposition

The operator is not removed. Their judgment moves to higher-leverage control points.



Exceptions and corrections feed the knowledge loop

Aligns with trusted data, domain-owned policy, and AI at the speed of mission

Not AI for its own sake: governed reasoning tied to operational tempo.

Trusted data

- structured evidence
- verified outcomes
- the basis for action

Domain ownership

- teams define healthy
- approved COAs
- where authority stops

AI-first at mission speed

- reasoning in the loop
- not a separate chatbot
- DAF AI Strategy

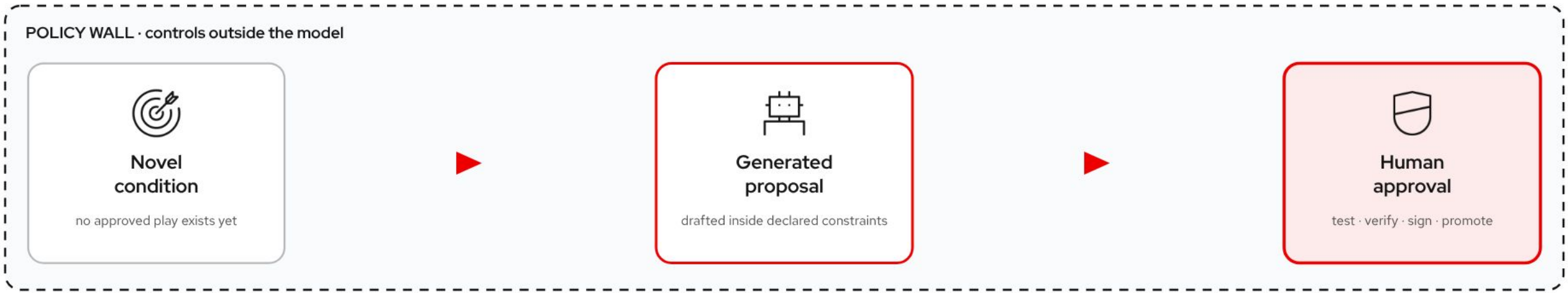
CDO-L resilience

- runs in the enclave
- local services + inference
- survives lost reach-back

Alignment: Trusted data, domain-owned policy, AI in the loop, and resilience under CDO-L, at once.

The next step is policy-bounded generation, not unbounded autonomy

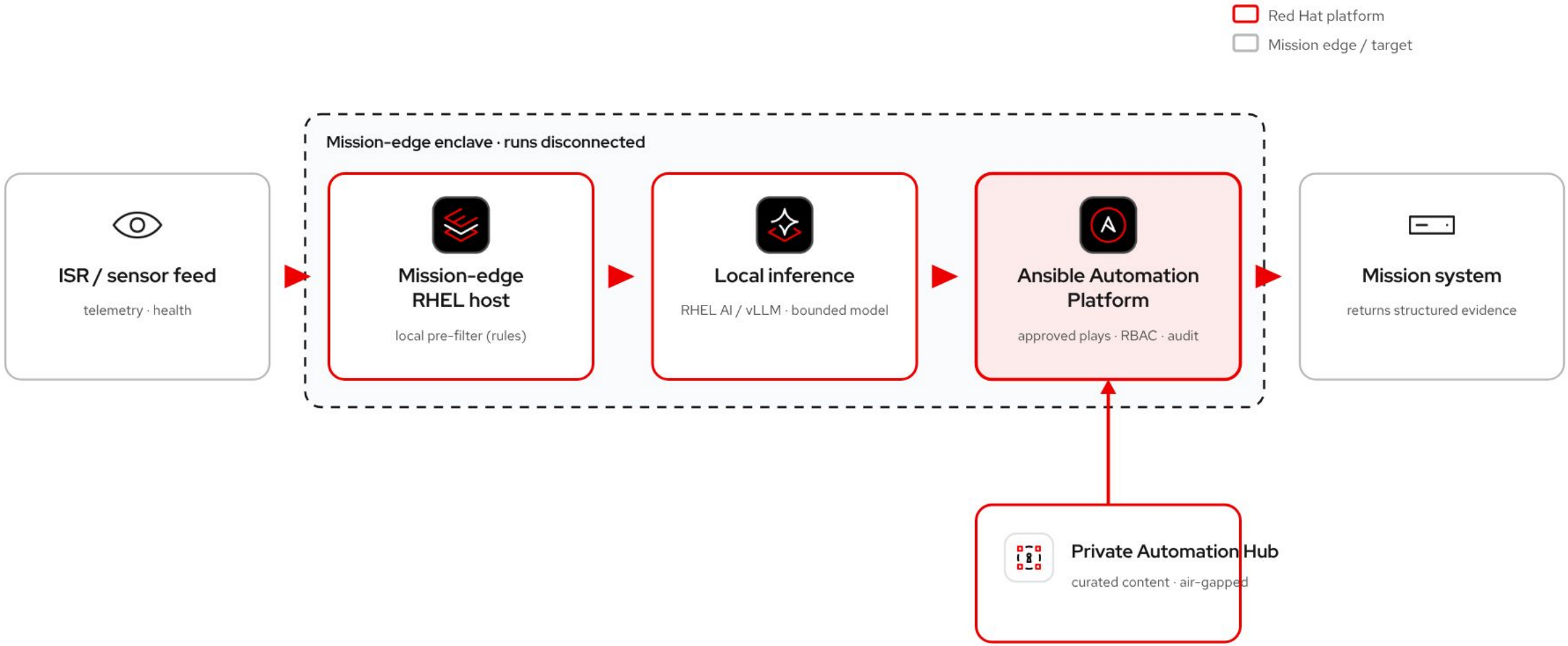
You earn more autonomy by strengthening the control plane, not by trusting the model more.



Rule: Generated action never bypasses policy, verification, audit, or human approval.

What this looks like on a Red Hat stack

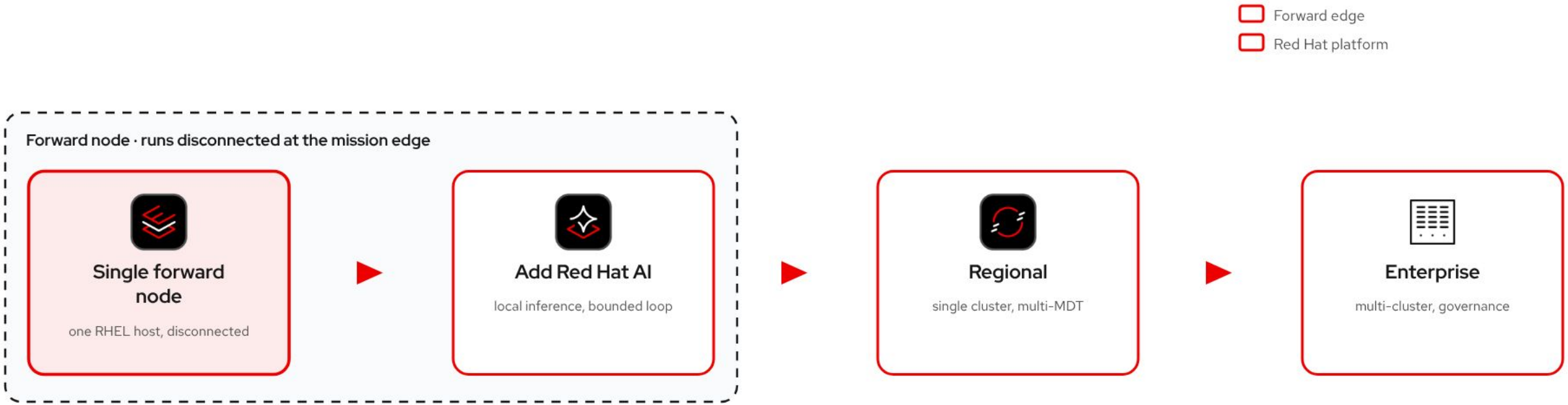
One governed control loop from sensor telemetry to verified outcome, local and disconnected-capable.



Flow: Sensor telemetry › local pre-filter › bounded reasoning › approved play › structured evidence › verified outcome

Same control model, from a single RHEL node to a multi-cluster enterprise

The bounded-defense loop does not change as it scales, only the footprint. It runs on AIR and SPACE mission-relevant terrain.



Same control model: One RHEL host to a multi-cluster enterprise. The bounded-defense loop is identical; only the footprint grows.

Come see us at Booth 350: Red Hat at DAFITC

Two related Red Hat live demos at our booth, plus more sessions this week.

At our booth · #350

- AI Ops Automated Demo
- Aug 24, 10:45 AM · Aug 25, 3:00 PM
- Software-Defined Mission Edge Demo (DDIL / EW)
- Aug 24, 2:30 PM · Aug 25, 11:00 AM · Aug 26, 9:45 AM

More Red Hat sessions

- Refactoring at Mission Speed: multi-agent AI meshes for C2
- Aug 24, 3:40 PM · Riverview Room 5
- Automated Evidence + Cryptographic Integrity (cATO, CJADC2)
- Aug 26, 7:55 AM · Alabama E
- Project Lightwell: securing the AF open-source supply chain
- Aug 26, 7:55 AM · Riverview Room 6

Booth #350 · start by making one operator's judgment executable, bounded, and verifiable