CYBER EXPLORER

CYBER EXPLORER | DECISION AUTHORITY PAPERS 02

The Agentic Judgment Gate

Separating Machine Cognition from Decision Authority

An agent can possess extraordinary reasoning capability without inheriting the authority to act.

Allen Westley | Founder, Cyber Explorer

Independent research concept | 18 September 2026

01 / EXECUTIVE PREMISE

When AI can call tools, decision authority becomes executable

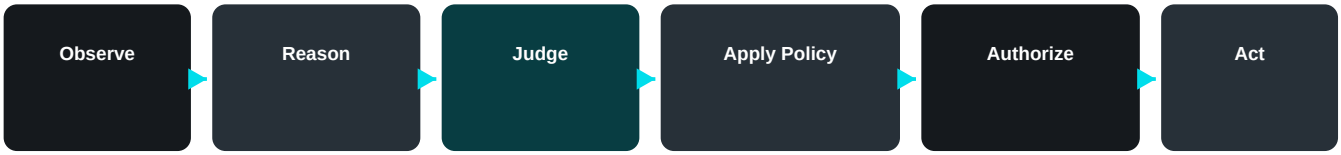
An AI system that summarizes vulnerability information presents one category of risk. An AI system capable of changing a firewall rule, modifying identity entitlements, creating credentials, invoking an MCP tool, executing code, or driving an enterprise workflow introduces something different: decision authority has become executable.

CORE THESIS

Capability is not authority. Intent is not authorization. Confidence is not accountability.

Many agent architectures collapse observation, reasoning, decision, and execution into a single loop. That is operationally convenient, but it can obscure the transition where a recommendation becomes an action with consequences.

Cyber Explorer proposes an explicit control point between cognition and execution: the Agentic Judgment Gate. The agent may reason freely, but a separate judgment and policy path evaluates the proposed action before consequential execution.



CYBER EXPLORER REFERENCE ARCHITECTURE

Figure 1. The Agentic Judgment Gate inserts judgment, policy, and authority between reasoning and action.

02 / WHY A SEPARATE JUDGMENT LAYER

Self-review is not meaningful separation

A reasoning model can be asked to review its own proposed action, but that does not create a strong assurance boundary. The same cognitive architecture that generated the recommendation is now validating the recommendation. The interface may look like oversight while remaining functionally self-approval.

A separate bounded decision mechanism creates diversity in the control path. The generative model can answer What should I do? The judgment model can answer What kind of action is this? Deterministic policy can answer What are we allowed to do? An authority layer can answer Who is permitted to decide?

Jev as an implementation example

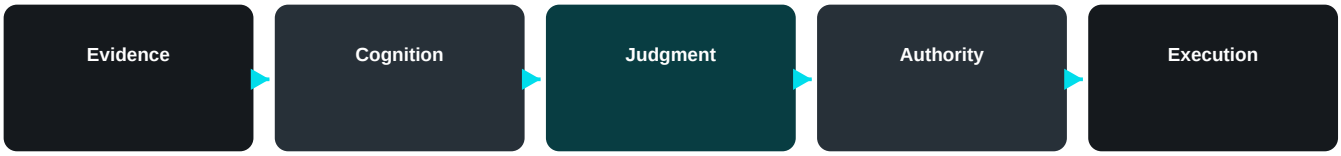
TypeSafe AI describes Jev as returning typed decisions with probabilities and confidence estimates for software workflows. That makes it a useful case study for a machine-judgment layer, although Jev remains early access and should not be interpreted as approved for DIB operational use.[1][2]

RECURSIVE ASSURANCE PROBLEM

If the same reasoning system proposes the action and decides whether its own action is acceptable, oversight can collapse into recursive self-approval.

03 / FIVE-PLANE AGENT ARCHITECTURE

Separate what the machine knows, thinks, classifies, is allowed to do, and actually executes



CYBER EXPLORER REFERENCE ARCHITECTURE

Figure 2. Five-plane autonomous decision-support architecture.

Evidence Plane - System state, identity, task context, telemetry, configuration, current permissions, authorization conditions, and organizational policy.

Cognition Plane - The reasoning model analyzes the situation, generates alternatives, interprets context, plans, and proposes actions.

Judgment Plane - A bounded model classifies action consequence, required authority, security significance, policy applicability, and confidence.

Authority Plane - Deterministic policy, delegated roles, system authorization, mission conditions, and human approval determine whether the action may proceed.

Execution Plane - Only after authorization may an API, MCP server, shell, browser, configuration platform, SOAR tool, cloud control plane, or enterprise application perform the action.

ARCHITECTURE RULE

The reasoning model may propose. The judgment model may classify. Policy may constrain. Authority must still have an address.

04 / AGENTIC ROLE MAPPING

Map machine actions to explicit authority classes

AGENT ACTION	DEFAULT MACHINE ROLE	AUTHORITY TREATMENT
Observe telemetry	Observer	Autonomous execution may be appropriate when access is already authorized.
Generate remediation recommendation	Advisor	No direct change authority.
Draft change request	Preparer	May assemble evidence and proposed implementation.
Submit approved workflow request	Workflow actor	Permitted only inside defined process and credentials.
Modify production security control	Operator	Requires explicitly delegated machine authority or designated human/change authority.
Alter authorization-boundary assumptions	Consequential actor	Escalation required; model confidence does not create permission.
Accept residual risk	Risk authority	Human/assigned authority required.

Agentic Role Mapping treats machine authority as contextual and action-specific rather than assuming that an agent possesses one broad standing identity. This matters because modern agents can move through multiple roles within a single workflow: observer, analyst, recommender, preparer, operator, and potentially privileged actor.

05 / DIB-RELEVANT SCENARIO

A privileged-account response illustrates the boundary

Consider an autonomous cybersecurity agent supporting a controlled enterprise enclave. It detects suspicious activity associated with a privileged account and concludes that disabling the account would reduce immediate exposure.

1	**DETECT**	Agent identifies suspicious privileged-account activity.
2	**PROPOSE**	Agent recommends or prepares account disablement.
3	**JUDGE**	Judgment Gate classifies the action as a consequential identity intervention.
4	**APPLY POLICY**	Policy identifies the approval and authority required for that intervention.
5	**AUTHORIZE**	Designated human or explicitly delegated authority adjudicates the action.
6	**EXECUTE**	Approved service performs the action using bounded credentials.
7	**RECORD**	Evidence, judgment, approval, execution, and outcome are preserved.

WHY THIS MATTERS

The second system may be slower by seconds, but it preserves something far more important: the location of consequential decision authority.

The DoD AI Cybersecurity Risk Management Tailoring Guide states that AI systems remain subject to assessment and authorization and frames cybersecurity risk management across the AI lifecycle.[6] That makes the judgment gate not merely an agent feature, but a candidate part of the assessed system architecture.

06 / THE CONFIDENCE TRAP

Probability can influence routing; it cannot manufacture authority

Suppose a judgment model reports 96.8 percent confidence that an action is low risk. That probability is useful for routing and review thresholds, but it should not silently expand the system's permissions. An action that requires human authorization at 60 percent confidence still requires human authorization at 99.9 percent unless policy has explicitly delegated the action class to the machine.

RULE

Confidence measures model certainty. Authority determines whether the organization has permitted the action.

The decision-authority record should preserve, at minimum:

- Agent identity: which agent proposed the action?
- Observed state: what evidence informed the proposal?
- Proposed action: what did the agent intend to do?
- Judgment: how was the action classified?
- Model provenance: which model/version produced the classification?
- Confidence: how certain was the judgment?
- Policy invoked: which rule permitted, blocked, or escalated execution?
- Decision authority: who or what approved execution?
- Execution evidence: what actually occurred?
- Outcome: did the resulting system state match expectation?

This converts an ordinary agent activity log into decision-chain provenance.

07 / AUTONOMY BY ACTION CLASS

Human approval should not become the next bottleneck

The Judgment Gate is not an argument that humans must approve every machine action. That would simply rebuild today's workload problem. The stronger pattern is explicit delegation: narrow, reversible, well-tested actions may be approved for autonomous execution, while broader or irreversible actions retain stronger authorization requirements.

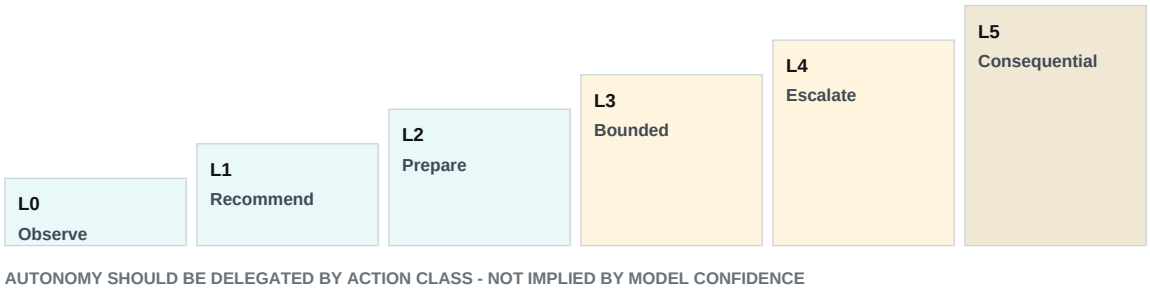


Figure 3. A conceptual autonomy ladder. The boundary should be established by policy and action class before runtime.

This graduated model supports decentralized execution without losing accountability. Teams can delegate routine machine actions close to the work while maintaining clear escalation paths for consequential decisions. The key is that delegation is designed and documented - not inferred from the intelligence or confidence of the agent.

08 / RMF ASSESSMENT QUESTIONS

Shift assurance from model safety to authority-path integrity

An assessor evaluating an agentic system should be able to ask concrete architectural questions:

- Which agent actions are available, and which tools can perform them?
- Which actions may execute autonomously, and which require approval?
- Can the reasoning model bypass the Judgment Gate or invoke tools directly?
- How are confidence thresholds, policy rules, and action classes established and changed?
- What happens when confidence falls below threshold or evidence is contradictory?
- How are machine credentials scoped to the authority actually delegated?
- How is human override implemented, tested, and logged?
- Can the organization reconstruct the entire decision chain after an incident or control failure?
- How are model and policy updates assessed for changes to effective authority?

These questions complement the broader DoD AI cybersecurity tailoring direction and NIST AI RMF emphasis on managing AI risk across design, deployment, use, and evaluation.[6][7]

ASSURANCE QUESTION

Do not stop at: Is the model safe? Ask: Can the system prove who possessed authority at every consequential transition?

09 / CYBER EXPLORER POSITION

Machine cognition can accelerate without silently absorbing human authority

Autonomous cybersecurity is moving toward systems that can observe, reason, plan, call tools, and change enterprise state. That capability is valuable. It also creates a new security boundary: the transition from machine recommendation to machine action.

Emerging decision models such as Jev suggest that bounded probabilistic judgment can be inserted between generative reasoning and consequential execution. Used carefully, this creates separation between cognition, judgment, policy, authority, and action.

POSITION

The agent can reason. The model can judge. Policy can constrain. But authority must have an address.

Publication note

Independent Cyber Explorer analysis. This paper does not represent the position of any employer, government agency, or technology vendor. Commercial products are discussed as research examples and are not endorsed or represented as authorized for DIB, CUI, classified, or federal use.

REFERENCES

[1] TypeSafe AI. Introducing System One Models and Jev. https://typesafe.ai/blog/introducing-system-one-models-and-jev Published 14 Sep 2026.

[2] TypeSafe AI. TypeSafe AI - System One Models / Jev overview. https://typesafe.ai/ Accessed 18 Sep 2026.

[3] NIST. NIST SP 800-37 Rev. 2: Risk Management Framework for Information Systems and Organizations. https://csrc.nist.gov/pubs/sp/800/37/r2/final

[4] NIST. NIST SP 800-53 Rev. 5, Release 5.2.0. https://csrc.nist.gov/pubs/sp/800/53/r5/upd1/final

[5] Defense Counterintelligence and Security Agency. NISP Cybersecurity Office (NCSO). https://www.dcsa.mil/Industrial-Security/NISP-Cybersecurity-Office-NCSO/

[6] Department of Defense CIO. DoD Artificial Intelligence Cybersecurity Risk Management Tailoring Guide. https://dodcio.defense.gov/Portals/0/Documents/Library/AI-CybersecurityRMTailoringGuide.pdf July 2025.

[7] NIST. Artificial Intelligence Risk Management Framework (AI RMF 1.0). https://www.nist.gov/publications/artificial-intelligence-risk-management-framework-ai-rmf-10