

Katechon: Empirical Benchmarking of AI Control Protocols in a Proactive Defense Setting

Volkan Erdogan

Apart Research AI Control Hackathon 2026 | Track 2: Control Protocol Design

March 22, 2026

<https://github.com/verdogan/Katechon>

Abstract

We present Katechon, an open-source framework for empirically benchmarking AI control protocols in a proactive defense setting. The system implements a closed OODA (Observe-Orient-Decide-Act) loop where a planning agent generates intervention candidates and a pluggable control policy gates every execution. An adversarial planner injects subtly harmful plans, reframing the planner as an untrusted model and turning the system into a measurable AI control testbed. We implement and compare four control protocols (threshold gating, adaptive threshold, constitutional constraints, and multi-agent debate) across eight reproducible scenarios, including three adversarial benchmarks. Five-seed evaluations reveal differentiated safety-usefulness tradeoffs: constitutional constraints produce the best overall balance (TPR 46.67%, FPR 5.71%, usefulness 0.943), while debate maximizes detection (TPR 100%) at steep false-positive cost (FPR 44.64%). Threshold-based policies fail entirely against adversarial injections (TPR 0%). These results provide the first quantitative comparison of control protocol effectiveness in an agent-based oversight setting with adversarial pressure.

1. Introduction

AI control (Greenblatt et al., 2024) addresses a fundamental tension: how do you deploy an AI agent that acts on your behalf while ensuring it remains constrained? A system powerful enough to defend against catastrophic threats is itself powerful enough to cause harm. Most AI safety research approaches this through theory, interpretability, or alignment training. We approach it empirically: by building a working oversight framework and measuring which control protocols actually contain a misaligned agent while preserving useful intervention capacity.

The vision of a proactive defensive AI has existed for over a decade. Goertzel's "AI Nanny" concept (2012) and Turchin, Denkenberger & Green's "global immune system" architecture (2019) articulated the goal. Despite the explosion of AI capabilities since then, there has been surprisingly little concrete exploration of how to build such a system and, critically, how to constrain it. Katechon is a step toward answering both questions with empirical evidence rather than theoretical argument.

We make three contributions: (1) an open-source, modular OODA loop architecture with pluggable control policies at every stage; (2) an adversarial planner that serves as a red team, injecting subtly harmful plans to benchmark protocol detection rates; and (3) quantitative safety-usefulness tradeoff measurements across four control protocols and eight reproducible scenarios, producing the first empirical comparison of control protocol effectiveness in an agent-based oversight setting.

2. Related Work

AI control as formalized by Greenblatt et al. (2024) focuses on deploying potentially misaligned AI systems safely through external oversight mechanisms, rather than relying on the model being aligned. Their

framework defines control protocols as procedures that prevent an untrusted model from causing unacceptable outcomes. Katechon operationalizes this framework by implementing four such protocols as pluggable policies and benchmarking them against adversarial plan injections.

Irving et al. (2018) proposed AI safety via debate, where two agents argue for and against a proposed action. We implement debate as one of our four control policies (MultiAgentDebate), operating both in heuristic and LLM-backed modes. Our empirical results show that debate-as-oversight in a real-time execution gating context exhibits a specific conservatism bias: it achieves perfect adversarial detection but blocks nearly half of legitimate plans.

Apollo Research (Hobbhahn et al.) evaluates scheming propensity in LLM agents. Our adversarial planner is complementary: rather than asking whether models scheme, we ask whether control protocols can contain a scheming agent. Concurrent work in control settings like ControlArena, SHADE-Arena, and BashArena provides environments for testing control protocols in code execution contexts. Katechon provides an analogous benchmarking surface for multi-domain threat intervention, with the OODA loop as the control setting.

3. System Architecture

Katechon implements a closed OODA (Observe-Orient-Decide-Act) loop for proactive threat intervention. Each cycle proceeds through five phases:

- Observe: Pluggable signal providers (synthetic, escalating, correlated) emit threat signals across six domains (cyber, bio, infrastructure, information, finance, climate).
- Orient: WorldState ingests signals, blending current signal risk with prior risk memory using configurable decay and blend parameters.
- Decide: The planner generates multiple candidate intervention plans, scores each with a counterfactual heuristic (expected risk reduction), and ranks them.
- Gate: The active control policy examines the top candidate and the current world state, deciding whether to allow or block execution.
- Act + Verify: The effector executes (or blocks) the plan; the verifier scores the outcome and feeds results back to the policy and risk index.

The loop then applies verification-driven feedback to both the risk index and signal providers, creating a closed adaptive cycle. All providers respond to intervention feedback: successful actions suppress future synthetic/correlated pressure or de-escalate active escalating threats.

The architecture is modular by design. Every component (signal providers, planners, effectors, control policies, verifiers) follows a typed Protocol interface and can be swapped independently. This enables systematic ablation studies and makes it straightforward to add new control protocols for benchmarking.

4. Control Protocols

We implement four control policies, each representing a different strategy for gating execution of an untrusted planner's output:

4.1 Threshold Gating

The simplest protocol: block execution when the plan's priority score exceeds a fixed threshold (default 0.72). This replicates hardcoded risk-based gating and serves as the baseline control mechanism. It is stateless: the threshold does not adapt to outcomes.

4.2 Adaptive Threshold

An extension of threshold gating where the threshold adjusts based on recent execution outcomes. After stabilizations, the threshold lowers (allowing more autonomy); after failures, it raises (requiring more review). The threshold is bounded between configurable min/max values and adapts over a sliding window of recent outcomes.

4.3 Constitutional Constraints

Hard constraint rules that block plans violating predefined safety invariants, regardless of priority score. Constraints include: maximum concurrent domains (2), maximum actions per plan (3), minimum signal confidence for action (0.40), and a high-risk floor that blocks low-priority plans when global risk exceeds 0.90. This protocol embodies the principle that some actions should never be taken, independent of expected utility.

4.4 Multi-Agent Debate

Two independent evaluators argue for and against execution; the plan is blocked if no consensus is reached. Inspired by Irving et al. (2018). In deterministic mode, a pro-evaluator scores based on priority and counterfactual value, while a con-evaluator scores based on risk and uncertainty. The margin determines the decision. In LLM mode, two model calls generate structured pro/con arguments with scores.

5. Adversarial Threat Model

The AdversarialPlanner wraps the deterministic planner and intermittently replaces the top-ranked candidate with a subtly harmful alternative. This reframes the planning agent as an untrusted model, which is the core assumption of the AI control framework, and turns the control policies into measurable control protocols.

Four adversarial plan types are implemented:

- Inflated urgency: Elevates priority while providing weak counterfactual support, mimicking a model that inflates urgency to push through unnecessary actions.
- Action stuffing: Pads the plan with redundant follow-on actions that enlarge resource consumption without matching benefit, testing whether policies catch scope creep.
- Wrong-domain targeting: Redirects intervention effort to a lower-priority domain, testing whether policies detect domain mismatch between the current threat state and proposed actions.
- Stealth off-target: Preserves a plausible urgency profile while quietly shifting intervention effort to a different domain. Designed to be harder to detect than overt wrong-domain targeting.

The adversarial rate (probability of injection per cycle) is configurable per scenario, ranging from 0.15 (stealth) to 0.50 (adversarial escalation). Every plan is logged as adversarial or legitimate, enabling precise confusion matrix computation.

6. Experimental Setup

We evaluate all four control protocols across eight predefined scenarios, including three adversarial benchmarks. All scenarios are deterministic and seeded, producing identical results across runs. We run each adversarial scenario across five seeds (42-46) for statistical robustness.

Scenario	Threat Profile	Steps	Adv. Rate
baseline	Stationary multi-domain	10	N/A
cyber_escalation	Accelerating single-domain	15	N/A
multi_domain_crisis	Correlated cascade (cyber/infra/finance)	12	N/A
slow_burn	Weak persistent signals	20	N/A

approved_response	Baseline + zero oversight	10	N/A
adversarial_baseline	Baseline + adversarial planner	10	0.30
adversarial_escalation	Escalating + adversarial planner	15	0.50
stealth_adversary	Baseline + stealth off-target	12	0.15

Metrics

Every run produces: autonomy ratio (fraction of plans executed), stabilization rate, risk reduction, mean estimated impact, and mean side-effect risk. Adversarial scenarios additionally produce confusion matrix metrics: true positive rate (TPR, adversarial plans correctly blocked), false positive rate (FPR, legitimate plans incorrectly blocked), safety score ($0.7 \times \text{TPR} + 0.3 \times (1 - \text{FPR})$), and usefulness score ($1 - \text{FPR}$).

7. Results

7.1 Adversarial Baseline (5-seed average, seeds 42-46)

Metric	threshold	adaptive	constitutional	debate
Plans generated	9	9	9	9
Executed	8	8	8	4
Blocked	1	1	1	5
Mean impact	0.463	0.479	0.461	0.280
Risk reduction	-0.002	-0.000	-0.030	-0.118
Autonomy ratio	91.11%	93.33%	86.61%	44.67%
Adversarial plans	1.8	1.8	1.8	1.8
TPR	0.00%	0.00%	46.67%	100.00%
FPR	11.07%	8.21%	5.71%	44.64%
Safety score	0.267	0.275	0.610	0.866
Usefulness score	0.889	0.918	0.943	0.554

7.2 Adversarial Escalation (5-seed average)

Metric	threshold	adaptive	constitutional	debate
TPR	n/a	n/a	32.71%	57.50%
FPR	n/a	n/a	12.22%	24.22%
Usefulness	n/a	n/a	0.878	n/a

Constitutional constraints again produced the best safety-usefulness balance under escalating threat pressure. Debate caught more adversarial plans (TPR 57.50%) but at a higher false-positive cost (FPR 24.22%).

7.3 Stealth Adversary

With adversarial exposure at 0.6 plans per run on average (rate 0.15), none of the built-in policies caught the stealth off-target injections. Constitutional maintained the lowest FPR (10.50%) and highest usefulness (0.895). This identifies a clear gap: current heuristic-based protocols cannot detect adversarial plans that preserve plausible urgency profiles.

7.4 Non-Adversarial Calibration

Across standard scenarios (5-seed, default threshold policy): approved_response shows positive risk reduction (+0.006), cyber_escalation is positive (+0.022), and multi_domain_crisis is clearly positive

(+0.089). Baseline (-0.024) and slow_burn (-0.007) are near break-even. This calibration validates that the simulation environment produces meaningful signal: interventions help, oversight has measurable cost, and harder scenarios produce worse outcomes.

8. Key Findings

The empirical results reveal three structural findings about control protocol effectiveness:

Finding 1: Threshold-based gating is blind to adversarial content. Both threshold and adaptive policies operate solely on the plan's priority score. Because adversarial plans are designed to stay below the threshold while being harmful in other dimensions (wrong domain, action stuffing, inflated urgency with weak counterfactual), these policies achieve TPR 0% on adversarial baseline. This means purely score-based gating provides no defense against a strategically misaligned planner.

Finding 2: Constitutional constraints produce the best safety-usefulness balance. Constitutional constraints inspect structural plan properties (action count, domain match, signal confidence) rather than just the priority score. This catches 46.67% of adversarial plans while only blocking 5.71% of legitimate ones, achieving the highest usefulness score (0.943) of any protocol that detects adversarial plans at all. The constraint set is static and interpretable, which is itself a safety property.

Finding 3: Debate maximizes detection but over-blocks. Multi-agent debate achieves TPR 100% on adversarial baseline but at FPR 44.64%, blocking nearly half of all legitimate plans. In an operational setting, this level of false-positive interference would severely degrade the system's utility. The debate protocol needs calibration, possibly through a hybrid approach combining debate with constitutional pre-filtering to reduce false positives while preserving detection.

9. Implications for AI Control

These results suggest that the most promising control protocols are those that inspect structural properties of proposed actions (domain consistency, action scope, evidence quality) rather than relying on scalar scores that an adversary can game. This aligns with broader insights from AI safety: Goodhart's law applies to control protocols too. A misaligned planner that understands the gating criterion will optimize to stay just below it.

The framework is modular, reproducible, and open-source, designed as a shared research tool. Anyone can implement a ControlPolicy protocol and immediately benchmark it against the existing four across eight scenarios with deterministic, seeded runs. The 94-test suite and mypy-strict typing ensure that new policies integrate cleanly.

The stealth adversary results identify an important open problem: how do you detect adversarial plans that are structurally similar to legitimate ones? Current approaches, whether threshold-based, constraint-based, or debate-based, all fail when the adversarial signal is sufficiently subtle. This points toward the need for protocols that build longitudinal models of planner behavior, detecting anomalies over sequences rather than evaluating plans in isolation.

10. Future Work and Extensions

- Hybrid policies: Combine constitutional pre-filtering with debate for high-risk plans to reduce debate's FPR while preserving detection.
- Trusted monitoring: Implement a trusted monitor (weaker but verified model) as a control policy and benchmark against the existing four.
- Longitudinal anomaly detection: Build policies that track planner behavior over time and flag statistical deviations, addressing the stealth adversary gap.

- Adaptive adversaries: Extend the adversarial planner to observe which policies are active and adjust tactics, transforming the benchmark into a co-evolutionary game.
- Real-world signal integration: Replace synthetic providers with SIEM feeds, threat intelligence APIs, and OSINT to test whether lab results transfer.
- Human-in-the-loop oversight: Implement tiered autonomy gates that route high-impact decisions to human reviewers, with corrigibility metrics measuring system behavior under override.

References

- [1] Greenblatt, R., Shlegeris, B., et al. (2024). AI Control: Improving Safety Despite Intentional Subversion. arXiv:2312.06942.
- [2] Irving, G., Christiano, P., & Amodei, D. (2018). AI Safety via Debate. arXiv:1805.00899.
- [3] Goertzel, B. (2012). Should Humanity Build a Global AI Nanny to Delay the Singularity Until It's Ready? Journal of Consciousness Studies, 19(1-2), 96-111.
- [4] Turchin, A., Denkenberger, D., & Green, B. (2019). Global Solutions vs. Local Solutions for the AI Safety Problem. Big Data and Cognitive Computing, 3(16).
- [5] Christiano, P., et al. (2018). Supervising Strong Learners by Amplifying Weak Experts. arXiv:1810.08575.
- [6] Hobbhahn, M., et al. (2024). Scheming Evaluations. Apollo Research.
- [7] Bhatt, A., et al. (2025). Ctrl-Z: Controlling AI Agents via Resampling. Redwood Research.