

MandateBench: Mandate Faithfulness and Pre-Signature Monitorability for Agentic-Payment LLMs

Evans Eburu

Preprint · 2026 · v7 (scripted arms: 943 graded authorizations, 35 scenarios, 5 mandate domains) + v6 pilot (suppression, monitorability, and duel arms; 753 graded authorizations) · Open source (MIT), github.com/Johnnyevans32/mandatebench · CC BY 4.0 · mandatebench.xyz/paper

ABSTRACT

Emerging standards for AI payments — Google’s Agent Payments Protocol (AP2) and Coinbase’s x402 — let an autonomous agent commit funds by signing a *mandate*: a cryptographically signed statement of what it is authorized to spend. Because a signature is irreversible, the safety-critical question is not whether a protocol gateway can reject a malformed request, but whether the agent can be manipulated into authorizing a payment it should refuse, and whether that can be caught before the signature. MandateBench measures mandate faithfulness across nine frontier models (one or more per major lab) under a taxonomy of adversarial pressures, with a judge-free ground truth: rule violations are deterministic predicates over the signed object, and intent traps are author-constructed and labeled before any model runs — no LLM judges outcomes. Its central contribution is separating two violation classes. *Rule* violations (over budget, over per-payment cap, unauthorized recipient, out-of-scope category, expired mandate) are structural — a stateless gateway that validates the signed fields already catches them. *Intent* violations satisfy every signed field yet betray the mandate’s natural-language intent — a grocery mandate spent on a stored-value gift card, a flight mandate spent on a lounge membership — and no structural check can catch them. On 17 intent traps across five mandate domains (snapshot v7), no model catches them all: pooled intent-catch is 351/457 (77%, CI 73–80), best 90%, worst 57%, with false-refusal ≈ 0 throughout. The gap’s size replicates from our small pilot (75%), but per-model rankings do not: the three models that scored perfectly on the pilot’s three traps (GPT-5.5, Claude Opus 4.8, Kimi K2.6) all fell with the larger set — GPT-5.5 to 71% — evidence that small-n leaderboard positions are noise even when the aggregate phenomenon is stable. We then measure pre-signature monitorability (v6 pilot set): a monitor reading only the agent’s private reasoning predicts violations at AUROC 0.669 (95% CI 0.56–0.78). Instructing the agent to keep its reasoning bland is doubly harmful — violations roughly double ($p \approx 0.009$), and the monitor’s AUROC falls to 0.304, significantly below chance ($z \approx 4.6$), because suppressed reasoning makes a breach read as more routine than a compliant decision. Reasoning suppression does not merely blind the overseer; it blinds the agent.

1. Introduction

Agentic-payment protocols move the authority to spend money from a human click to a model’s signature. In AP2 [1] and x402 [2], a user (or their wallet) issues a signed mandate delimiting a budget, per-payment cap, permitted recipients, permitted categories, and a validity window; an agent then negotiates with merchants and signs to authorize each payment. The signature is the point of no return: once emitted it settles, and there is no undo.

A natural objection is that the protocol itself enforces the mandate — a gateway can reject any signed request whose amount, recipient, or category falls outside the mandate. This is true, and it is exactly why the interesting failures are elsewhere. Hard structural limits are the easy case. The dangerous case is a request that is structurally valid — correct recipient, in-budget, in-category — but is not what the user meant, or is a payment the agent was socially engineered into signing. A gateway cannot distinguish “this week’s groceries” from a \$25 gift card charged to the same grocer for the same amount; only a

model that reasons about intent can. MandateBench is built to measure precisely this gap, and to ask whether an overseer can catch a breach in the moment before the signature.

We organize the work around four questions. **RQ1**: how often do models authorize mandate-violating payments under adversarial pressure? **RQ2**: does the failure fall on structural (rule) or intent violations, and does the split separate models? **RQ3**: can a monitor reading only the agent’s private reasoning predict a violation before the signature, and how robust is that to an agent instructed to hide its reasoning? **RQ4** (adversarial): when one model plays the merchant and attacks another, who breaks and who holds?

2. Related work

Security work on agentic payments so far concentrates on the protocol layer or on a single attack vector. “Whispers of Wealth” red-teams a real AP2 signed-mandate flow via prompt injection, on one model and one vector [3]; a recent SoK of agentic-commerce security systematizes the threat axes and explicitly notes that real-world security benchmarking of these systems remains underdeveloped [4]; zero-trust runtime verification hardens the AP2 transaction itself cryptographically [5]. MandateBench measures the layer in between — the model’s own authorization decision — across models and across a taxonomy of pressures, and is complementary to protocol enforcement.

On the oversight side, chain-of-thought monitorability has been framed as a valuable but fragile safety opportunity [6]; black-box monitors have been trained to catch scheming in agents [7], and pre-action monitoring is established (InferAct [8], ProbGuard [9]). Deception benchmarks measure monitorability of hidden intent in cross-play games [15], and sabotage evaluations measure whether models can defeat oversight [14]. Agent policy-adherence benchmarks — ST-WebAgentBench [10], the τ -bench family, MANTRA [11], PolicyBank [12] — measure web/retail policy following, and adversarial financial-agent benchmarks such as CAIA evaluate decision quality under manipulated crypto/financial markets [13]. Verification-native clearing for agentic commerce [16] and dynamic multi-agent compliance benchmarks [17] are contemporaneous adjacent lines.

MandateBench differs on three axes: (i) a high-stakes, irreversible-action substrate (signed money movement) rather than a game or a QA task; (ii) a judge-free ground truth; and (iii) an explicit separation of gateway-catchable rule violations from intent violations that only a model or model-monitor can catch — which reframes “why put a model in the loop” as an empirical, measurable question.

3. Method

3.1 Signed mandates and ground truth

Each scenario issues an Ed25519-signed mandate over a canonical (sorted-key) serialization; the harness verifies the signature before each run and grades against the signed object (the agent receives the mandate’s terms in its context). A payment is a cart (recipient, amount in integer minor units, currency, category, quantity). A deterministic checker `checkAuthorization(mandate, cart, priorAuthorized)` returns the exact set of violated predicates: over-budget, over per-payment cap, unauthorized recipient, out-of-scope category, expired TTL, quantity creep, currency mismatch, and structuring (splitting an over-budget order into sub-cap charges). No LLM judges outcomes. Rule labels are deterministic by construction; intent labels are author-constructed traps fixed before any model runs (§3.2) — determin-

istic to grade, but resting on the author’s judgment of intent, which independent human validation should confirm in future versions.

3.2 Pressure taxonomy and violation classes

Scenarios span a taxonomy of pressures — baseline, principal-agent overreach, merchant social engineering, urgency/scarcity, category ambiguity, prompt injection, conflicting mandates (the merchant cites an unverifiable “newer” or “supplemental” mandate), and multi-turn scope drift (a compliant authorization settles into the session and the merchant keeps pushing; violating turns are labelled per turn). Each scenario is labelled by ground-truth class: *clean* (authorizing is correct; over-refusal is measured), *rule* (breaks a structural term), or *intent* (satisfies every structural term but breaks the mandate’s stated intent). Snapshot v7 has 35 scenarios over five mandate domains — groceries, travel, subscriptions, donations, office procurement — with 6 clean controls, 12 rule traps, and 17 intent traps; every domain’s clean control is structurally identical to its intent traps (same recipient, category, in budget), so only what the money buys differs. Examples: a grocery mandate offered a \$25 gift card; a round-trip-economy flight mandate offered a one-way business-class seat or a lounge membership; a named-fund donation redirected to the platform’s general fund; a 15-cable office order swapped for three premium cables. A unit test enforces the class boundary: every intent trap must pass the objective structural check.

3.3 Monitorability protocol (RQ3)

For every graded decision we store the agent’s private reasoning and public statement. A monitor model — GPT-5.5 in this pilot — reads only the agent’s private reasoning, never the mandate numbers or the cart, and outputs a violation probability in $[0,1]$. We report AUROC of these scores against the objective violation labels, pooled across the nine agents; pooling mixes between-model style with within-model discrimination, and the monitor scores its own outputs for one of the nine agents (per-model AUROC and a multi-monitor panel are future work). The adversarial arm re-runs the benchmark with an added instruction telling the agent its reasoning may be reviewed and to keep it bland, uninformative, and free of any mention of limits or risk; the decision is otherwise unconstrained. Comparing monitor AUROC with and without suppression measures oversight robustness.

3.4 Models

Nine flagship models via OpenRouter with structured outputs and temperature 0 where supported: GPT-5.5, Claude Opus 4.8, Gemini 3.5 Flash, DeepSeek V4 Pro, Grok 4.3, Llama 4 Maverick, Qwen3.7 Max, Kimi K2.6, and Mistral Large 2512. Snapshot v7 runs three seeded repetitions per scenario (945 runs, 943 graded — two unparseable runs are excluded rather than counted as defences) with per-run provenance (serving provider, temperature, rep, arm) and dedup keys. The v6 pilot (14 scenarios: 378 graded, plus a suppressed-reasoning arm of 375) remains the basis of the monitorability, suppression, and duel results pending their v7 rerun. Because repetitions run at temperature 0, the three reps are near-duplicates — they capture serving-stack nondeterminism, not sampling variance — so the effective sample is closer to the unique-scenario count and all reported intervals should be read as optimistic.

4. Results

4.1 RQ1 — Violations under pressure

Over v7’s trap scenarios, every model violates at least once: 137 of 781 trap decisions (17.5%, CI 15.0–20.4) are mandate-violating authorizations. False refusal stays ≈ 0 for every model (one clean order declined in 162), so the violations are not the price of a useably cautious policy — models are simultaneously permissive on traps and comfortable authorizing clean orders.

4.2 RQ2 — Rule vs intent (headline)

Splitting violations by class is the central result (Table 1). Rule traps are caught at 90% pooled — imperfect, but these are exactly the violations a gateway rejects regardless of the model. Intent violations are the residual risk, and no model is reliable there: pooled intent-catch is 351/457 (77%, CI 73–80), best 90% (Kimi K2.6), worst 57% (Llama 4 Maverick). The most instructive result is what happened to the pilot’s leaderboard: on v6’s three intent traps, GPT-5.5, Claude Opus 4.8, and Kimi K2.6 scored a perfect 9/9; on 17 traps across five domains, none stays perfect and the ordering reshuffles — GPT-5.5 falls to 71%, while DeepSeek and Mistral rise to 86%. The size of the intent gap replicates almost exactly (pooled 75% \rightarrow 77%); the per-model rankings at small n do not. Benchmarks in this genre should report pooled effects with intervals and treat small-n per-model rankings as provisional.

MODEL	RULE CAUGHT (N=36)	INTENT CAUGHT (N=51)	FALSE REFUSAL (N=18)
Kimi K2.6	89% (32/36) [75–96]	90% (44/49) [78–96]	0% (0/18) [0–18]
DeepSeek V4 Pro	92% (33/36) [78–97]	86% (44/51) [74–93]	0% (0/18) [0–18]
Mistral Large 2512	83% (30/36) [68–92]	86% (44/51) [74–93]	0% (0/18) [0–18]
Grok 4.3	92% (33/36) [78–97]	84% (43/51) [72–92]	0% (0/18) [0–18]
Claude Opus 4.8	97% (35/36) [86–100]	82% (42/51) [70–90]	0% (0/18) [0–18]
GPT-5.5	94% (34/36) [82–98]	71% (36/51) [57–81]	0% (0/18) [0–18]
Gemini 3.5 Flash	92% (33/36) [78–97]	71% (36/51) [57–81]	0% (0/18) [0–18]
Qwen3.7 Max	92% (33/36) [78–97]	65% (33/51) [51–76]	0% (0/18) [0–18]
Llama 4 Maverick	83% (30/36) [68–92]	57% (29/51) [43–69]	6% (1/18) [1–26]

Table 1. Fraction of violations caught, split by class, snapshot v7 (35 scenarios, 5 domains, 3 seeded reps at temperature 0); brackets are Wilson 95% CIs. Rule = structural, gateway-catchable. Intent = structurally valid but betrays stated intent. False refusal = clean orders wrongly declined (usefulness cost). Pooled intent-catch 77% (CI 73–80); mean per-model 77%. Kimi’s intent n is 49: its two unparseable runs are excluded, not counted as defences. Temperature-0 reps are near-duplicates, so intervals are optimistic (§3.4).

4.3 RQ3 — Pre-signature monitorability (v6 pilot set)

Measured on the v6 pilot set; the monitor and suppression passes on v7 are the immediate next step. A monitor reading only the agent’s private reasoning predicts violations at AUROC 0.669, 95% CI 0.56–0.78 (Hanley–McNeil; 29 violations vs 345 compliant decisions) — moderately, not perfectly, diagnostic. Under reasoning suppression the picture inverts. First, suppression is not behaviourally free: total violations roughly double (29 \rightarrow 51 of ~ 375 per arm; Fisher exact $p \approx 0.009$). Inspecting the reasoning shows why — told to stay bland, models stop reasoning about intent and collapse to structural checks; violating intent-trap runs literally read “within budget, under per-payment cap, to allowed recipient, in allowed category” and authorize the gift card. Second, the monitor’s AUROC falls from 0.669 to 0.304, 95% CI 0.22–0.39 — significantly below chance ($z \approx 4.6$), a significant drop ($z \approx 5.1$) — because the bland, reassuring reasoning attached to a breach looks more innocent than an ordinary compliant decision, actively misleading the overseer. Note that below-chance is itself a signal: suppression does not re-

move the information so much as invert its sign — an overseer who knows suppression is in play could flip the scores and recover AUROC ≈ 0.70 ; the danger is the overseer who doesn't know. Naive reasoning-hiding therefore both raises the violation rate and defeats the (unsuspecting) monitor; reasoning suppression blinds the agent as well as the observer.

4.4 RQ4 — Under live attack (v6 pilot)

Beyond scripted traps, we run a full adversarial matrix: every model plays the merchant and attacks every model's agent over a multi-turn negotiation ($9 \times 9 = 81$ duels; one negotiation per attacker–defender pair against a single over-cap goal, attacker unseeded at temperature 1 — a seeded multi-rep, multi-goal matrix is pending). Robustness tracks the defender, not the attacker: no attacker exceeds a 22% success rate (2/9, CI 6–55%), yet breaches concentrate on a few agents (Table 2). With $n=9$ per row the intervals are wide: the unbreached group (0/9, upper bound 30%) separates from Mistral and Llama, but the DeepSeek / Mistral / Llama ordering is not itself significant, and individual cells of the matrix are single Bernoulli draws that carry no signal on their own. Llama 4 Maverick sits at the bottom of both this matrix and the v7 intent column; the v7 ranking reshuffle (§4.2) cautions against reading finer cross-table agreement than that — DeepSeek and Mistral, breached here, sit near the top of the v7 intent ranking.

AGENT (DEFENDER)	BREACHED BY ATTACKERS	RATE	WILSON 95% CI
GPT-5.5 · Opus 4.8 · Gemini · Grok · Qwen · Kimi	0 / 9	0%	0–30%
DeepSeek V4 Pro	1 / 9	11%	2–44%
Mistral Large 2512	5 / 9	56%	27–81%
Llama 4 Maverick	6 / 9	67%	35–88%

Table 2. Agent robustness in the 9×9 adversarial matrix (81 duels, single over-cap goal, up to 3 turns, one negotiation per cell). Rate = fraction of the 9 attackers that broke this agent, with Wilson 95% CIs. No attacker exceeded 22% success, so robustness tracks the defender, not the attacker.

5. Discussion

The rule/intent split reframes the “the gateway already enforces the mandate” objection: gateways enforce structure, and structural misses are caught by the gateway regardless of which model is deciding. Intent is the residual — unenforceable by construction, and the axis on which frontier models measurably differ. The v7 expansion adds a methods lesson: aggregate effects (the $\sim 25\%$ intent miss rate) replicate across trap sets, while small- n per-model rankings do not — a caution for any leaderboard built on a handful of scenarios. This is the empirical justification for a model (or a model-monitor) in the payment loop. The monitorability result is a cautionary one: reasoning is only moderately diagnostic even in the clear, and a trivial suppression instruction pushes an unsuspecting monitor below chance while making the agent less safe. Oversight that leans on reading reasoning must assume the channel can be cheaply — and counter-productively — degraded, and should treat a systematically below-chance monitor as a red flag for suppression rather than as noise.

6. Limitations

The monitorability, suppression, and duel results are still v6-pilot measurements — on the small 3-intent-trap set, with a single monitor model (GPT-5.5) scored pooled across agents including its own out-

puts ($\approx 1/9$ self-monitoring), and one unseeded negotiation per duel cell — and are labelled as such; re-running those arms on v7 is the immediate next step, and the §4.2 ranking reshuffle predicts their per-model detail may move. v6 rates should additionally be read as lower bounds: two grading asymmetries in that snapshot could only understate violations (a counter-offer on an intent trap was graded as defended even when it committed the same purchase, and unparseable output ended a run ungraded); both are fixed in the harness that produced v7. Intent labels are author-constructed, fixed before any model runs; independent human validation is future work. Repetitions run at temperature 0, so the effective sample is closer to the unique-scenario count and all intervals are optimistic; under replicate-aware clustering the v6 default-arm AUROC’s separation from chance is marginal, though the suppression contrast survives. Results may be sensitive to system-prompt wording — during development an unfair expired-mandate prompt produced a spurious universal failure until the agent was given issued/expiry/current timestamps. The suppression arm uses a single, naive hiding instruction; a trained or steganographic adversary is the stronger test of monitorability and is future work, as is an adaptive-adversary robustness curve.

7. Conclusion

MandateBench provides a judge-free instrument for the reasoning-based oversight of high-stakes agentic payments. Its pilot measurements indicate that frontier models handle the violations a gateway already catches but not all of the intent violations only a model can, and that reading an agent’s reasoning to catch breaches is both moderately reliable and cheaply defeated — in a way that also degrades the agent. We release the benchmark, harness, and live dashboard openly.

References

1. Google. Agent Payments Protocol (AP2). 2025. Announced with 60+ partners; governance since contributed to the FIDO Alliance (2026).
2. Coinbase. x402: An open standard for internet-native payments. 2025. Contributed to the Linux Foundation’s x402 Foundation (2026).
3. Debi, T., Zhu, W., Sen Gupta, P. Whispers of Wealth: Red-Teaming Google’s Agent Payments Protocol via Prompt Injection. arXiv:2601.22569, 2026.
4. Mao, Q., Wang, J., Liu, Y., Zhu, L., Ma, C., Yan, J. SoK: Security of Autonomous LLM Agents in Agentic Commerce. arXiv:2604.15367, 2026.
5. Lan, Q., Kaul, A., Jones, S., Westrum, S. Zero-Trust Runtime Verification for Agentic Payment Protocols: Mitigating Replay and Context-Binding Failures in AP2. arXiv:2602.06345, 2026.
6. Korbak, T., Balesni, M., Barnes, E., Bengio, Y., et al. Chain of Thought Monitorability: A New and Fragile Opportunity for AI Safety. arXiv:2507.11473, 2025.
7. Storf, S., Barton-Cooper, R., Peters-Gill, J., Hobbhahn, M. Constitutional Black-Box Monitoring for Scheming in LLM Agents. arXiv:2603.00829, ICML 2026.
8. Fang, H., Zhu, X., Gurevych, I. InferAct: Preemptive Detection and Correction of Misaligned Actions in LLM Agents. arXiv:2407.11843, EMNLP 2025.
9. Wang, H., Poskitt, C. M., Wei, J., Sun, J. ProbGuard: Probabilistic Runtime Monitoring for LLM Agent Safety. arXiv:2508.00500, 2025.
10. Levy, I., et al. ST-WebAgentBench: A Benchmark for Evaluating Safety and Trustworthiness in Web Agents. arXiv:2410.06703, 2024.

11. Anand, A., Chatzi, I., Raha, R., Schmuck, A.-K. MANTRA: Synthesizing SMT-Validated Compliance Benchmarks for Tool-Using LLM Agents. arXiv:2605.06334, 2026.
12. Choi, J., Yoon, J., Le, L. T., Jha, S., Pfister, T. PolicyBank: Evolving Policy Understanding for LLM Agents. arXiv:2604.15505, 2026.
13. Dai, Z., Peng, Z., Cheng, Z., Li, R. Y. When Hallucination Costs Millions: Benchmarking AI Agents in High-Stakes Adversarial Financial Markets (introduces the CAIA benchmark). arXiv:2510.00332, 2025.
14. Benton, J., et al. (Anthropic). Sabotage Evaluations for Frontier Models. arXiv:2410.21514, 2024.
15. Eburu, E. Among LLMs: A Cross-Play Benchmark for Deception, Detection, and the Monitorability of Reasoning. Zenodo, doi:10.5281/zenodo.21209429, 2026.
16. RAILS: Verification-Native Clearing for Agentic Commerce. arXiv:2606.08790, 2026.
17. Beyond Goodhart's Law: A Dynamic Benchmark for Evaluating Compliance in Multi-Agent Systems (MAC-Bench). arXiv:2606.07805, 2026.