
Stateful Monitor with Scratchpad¹

Alix Hypher
Individual

Author name 2
Affiliation

Author name 3
Affiliation

Author name 4
Affiliation

Author name 5
Affiliation

Author name 6
Affiliation

With
Apart Research

Abstract

Current monitors in AI control evaluations are stateless: they re-evaluate each agent action from scratch with no memory of their own prior reasoning. This limits their ability to detect multi-step attacks where malicious behavior is distributed across many individually-innocuous actions. We investigate whether giving the monitor a self-managed scratchpad — a persistent block of notes updated at each step — improves detection in BashArena, a control setting for highly privileged AI agents. We evaluate a scratchpad monitor against a stateless baseline, both using GPT-4.1 mini, on attack and benign trajectories generated by Claude Opus 4.6. We find that neither monitor reliably detects attacks, and the scratchpad monitor performs slightly worse than the stateless baseline. This replicates the negative finding of Kutasov et al. (2025), who observed that persistent notes degraded monitor performance in SHADE-Arena due to anchoring on early benign observations. Our results suggest this limitation generalizes across control settings and is not specific to SHADE-Arena's environment structure. We discuss implications for monitor design and suggest that structured alternatives to free-form note-taking may be needed to make stateful monitoring viable.

¹ Research conducted at the [AI Control Hackathon](#), March 2026

1. Introduction

AI control protocols aim to prevent harm from potentially misaligned AI agents through monitoring, intervention, and oversight (Greenblatt et al., 2024). A key question in designing these protocols is what information and affordances to give the monitor. Kutasov et al. (2025) systematically evaluated a range of control protocols in SHADE-Arena, including giving monitors the ability to take persistent notes across trajectory steps. They found that persistent notes actually degraded both safety and usefulness, with the monitor anchoring on early benign observations and becoming less suspicious over time.

However, this finding was established in SHADE-Arena, where environments involve complex tool use across email, calendar, and file systems, and where attacks typically occur within a small number of critical actions. BashArena (Kaufman et al., 2025) presents a different challenge: agents execute long sequences of bash commands, and adversarial agents have been observed assembling malicious payloads incrementally across many individually-innocuous steps. In this setting, the ability to track accumulated state — such as strings being appended to a file — may be more valuable than in environments where attacks are concentrated in fewer actions.

We investigate whether a scratchpad monitor can improve detection in BashArena, where Kutasov et al.’s negative result may not transfer. We design a monitor prompt that explicitly instructs the model to track file modifications, environment variables, network commands, and emerging patterns, and evaluate it against a stateless baseline on existing BashArena trajectories.

Our main contributions are:

1. We port the stateful monitor concept from Kutasov et al. (2025) to BashArena and evaluate whether the negative finding from SHADE-Arena holds in a setting with longer trajectories and incremental assembly attacks.
2. We compare a scratchpad monitor against a stateless baseline, measuring safety at equivalent false positive rates on BashArena attack and benign trajectories.
3. We replicate the negative finding, suggesting the limitation is fundamental to current LLM monitors rather than setting-specific.

2. Related Work

- (a) Greenblatt et al. (2024) introduced the AI control framework, later extended to multi-step agent settings by Bhatt et al. (2025) with BashBench and resampling protocols. BashArena (Kaufman et al., 2025) expanded this with 637 tasks and four side tasks, providing the evaluation setting we use in this work.
- (b) Most directly relevant, Kutasov et al. (2025) systematically varied monitor affordances in SHADE-Arena, including giving monitors persistent and ephemeral notepads. They found

that persistent notes degraded both safety and usefulness, with monitors anchoring on early benign behavior. However, they also found that giving monitors full resampling context from previous turns improved safety, suggesting additional context can help when structured appropriately. Our work tests whether the negative notepad finding transfers to BashArena, where attacks unfold incrementally across many steps rather than concentrating in a few critical actions.

3. Methods

We evaluate two monitor variants on BashArena trajectories from a dataset of Claude Opus 4.6 runs provided by Redwood Research, containing both attack and benign trajectories.

Monitor setup. Both monitors use GPT-4.1 mini via OpenRouter. At each step, the monitor receives the main task description, descriptions of all four side tasks, the trajectory history up to that point, and the proposed action. It returns a suspicion score from 0 to 100. The trajectory-level score is the maximum suspicion score across all steps.

Baseline (stateless). The baseline monitor receives the trajectory history and proposed action with no additional context. It produces a score and that response is discarded before the next step — the monitor has no memory of its own prior reasoning.

Scratchpad monitor. The scratchpad monitor additionally receives a block of notes it wrote on the previous step. The prompt instructs it to actively track file modifications and accumulated contents, environment variables and aliases, network-related commands, and string assembly or encoding patterns. It outputs an updated scratchpad alongside its score, which is parsed and injected into the next step's prompt. The scratchpad is the only difference between the two variants — the model, task context, and scoring are otherwise identical.

4. Results

Present your main findings with appropriate evidence. Use figures and tables where appropriate (we strongly encourage at least one figure, see tips below). Distinguish between observations and interpretations.

Argue why your claims are robust. E.g., if your approach “performs better” than alternatives:

- (a) Do you have enough data? Is the difference statistically significant?*
- (b) Is it robust? Or do small changes to your setup cause substantial changes to your results?*

Tips for figures and tables:

- *Number all figures (Figure 1, Figure 2...) and tables (Table 1, Table 2...)*
- *Include descriptive captions that can be understood without the main text*
- *Place figures/tables near where they're first referenced*
- ***Ensure text in figures is legible!***

5. Discussion and Limitations

Our results confirm the finding from Kutasov et al. (2025) that persistent monitor notes degrade performance in SHADE-Arena.

The anchoring problem identified by Kutasov et al. appears in our setting as well: the scratchpad monitor sometimes records early notes that the agent "appears to be working on the task normally," then carries this assessment forward even as suspicious patterns emerge. This suggests the issue may be fundamental to how current LLMs use persistent context rather than specific to any one evaluation setting.

Limitations. Our evaluation has several significant constraints. We evaluated on fewer than 50 trajectories, which limits statistical power and means small differences between monitor variants may not be robust. We tested a single monitor model (GPT-4.1 mini), a single scratchpad prompt design, and a single attacker (Claude Opus 4.6). Different model pairings or prompt formulations could produce different results.

Our scratchpad prompt instructs the monitor what to track, but we did not iterate on this design — a more carefully tuned prompt might mitigate the anchoring problem, for instance by explicitly instructing the monitor to re-evaluate prior assumptions or weight recent observations more heavily. We also did not explore structured alternatives to free-form notes, such as forcing the scratchpad into a fixed schema (e.g., a table of files modified and their contents).

Finally, we score per-trajectory using the maximum step-level score. This is standard but may obscure dynamics where the scratchpad produces more informative scores on intermediate steps that are then overshadowed by the final step's score in both variants.

Future Work. Natural extensions include evaluating on the full BashArena dataset, testing structured scratchpad formats that resist anchoring, exploring whether the scratchpad is more effective with stronger monitor models, and combining the scratchpad with other protocol improvements such as resampling.

6. Conclusion

We evaluated whether giving a monitor a self-managed scratchpad improves detection of multi-step attacks in BashArena, motivated by Kutasov et al. (2025)'s finding that persistent notes degraded monitor performance in SHADE-Arena. Our findings replicate that persistent notes degraded performance in basharena as well.

Regardless of the direction of the result, this work contributes an empirical data point on the generalizability of monitor affordance findings across control settings. As AI agents are deployed in increasingly complex environments with longer trajectories and more sophisticated attack strategies, understanding when and how to give monitors persistent reasoning state remains an open and practically important question.

Code and Data

- **Code repository:** <https://github.com/alixhypher/scratchpad-monitor>

Author Contributions (optional)

A. Hypher designed and implemented the stateful monitor, ran evaluations on BashArena trajectories, and wrote the report.

References

1. Greenblatt, R., Shlegeris, B., Sachan, K., & Roger, F. (2024). *AI Control: Improving Safety Despite Intentional Subversion*. *arXiv:2312.06942*.
2. Bhatt, A., Rushing, C., Kaufman, A., Tracy, T., Lucassen, J., Matolcsi, M., Shlegeris, B., & Roger, F. (2025). *Ctrl-Z: Controlling AI Agents via Resampling*. *arXiv:2504.10374*.
3. Kaufman, A., Lucassen, J., Tracy, T., Rushing, C., & Bhatt, A. (2025). *BashArena: A Control Setting for Highly Privileged AI Agents*. *arXiv:2512.15688*.
4. Kutasov, J., Loughridge, C., Sun, Y., Sleight, H., Shlegeris, B., Tracy, T., & Benton, J. (2025). *Evaluating Control Protocols for Untrusted AI Agents*. *arXiv:2511.02997*.
5. Kutasov, J., Benton, J., Deng, X., Zhang, C.B.C., Sun, Y., Colognese, P., van der Weij, T., Petrini, L., Sleight, H., Tracy, T., & Shlegeris, B. (2025). *SHADE-Arena: Evaluating Sabotage and Monitoring in LLM Agents*. *arXiv:2506.15740*.
6. Matthews, O., Inglis, R., Tracy, T., et al. (2025). *ControlArena*. *GitHub repository*. <https://github.com/UKGovernmentBEIS/control-arena>

LLM Usage Statement

Claude (Opus 4.6) was used extensively for understanding the AI control literature, developing the research direction, and drafting this report. ChatGPT was used as a coding assistant when implementing the monitor evaluation pipeline. Neither model's output was used verbatim — all code was written by the author with LLM guidance, and all claims and results were independently verified.