

---

# Perturbation-Based Generation Profiling Detects Covert AI Agent Attacks Where Token-Level Statistics Fail<sup>1</sup>

---

Yatharth Maheshwari 1  
Independent

Arka Dash 2  
Independent

Abhineet Som 3  
Independent

With  
Apart Research

## Abstract

AI agents with elevated Linux privileges can covertly sabotage infrastructure—exfiltrating secrets, escalating privileges, disabling firewalls — while completing legitimate tasks. In BashArena, Claude Sonnet 4.5 evades GPT-4.1 mini monitoring 26% of the time at 4% false positive rate ([Kaufman et al.](#)). We investigate whether token-level statistical features from output logprob distributions can provide a cheaper, model-free monitoring signal. Across six model architectures (8B–32B parameters, four model families, 1,440 total generations), we find that 71 token-level features achieve only 0.56–0.83 AUROC with proper cross-validation—insufficient for deployment. We propose Perturbation-Based Generation Profiling (PBGP), an unsupervised protocol that re-runs the model in an isolated context and compares generation profiles via Jensen-Shannon divergence, Wasserstein distance, and Maximum Mean Discrepancy. PBGP achieves 0.94–1.00 AUROC across all six models with zero training data. Mann-Whitney U tests confirm significance at  $p < 0.001$  for all divergence metrics, with bootstrap 95% CI for the MMD ratio at  $[1.69\times, 5.02\times]$ . These results demonstrate that context-dependent generation dynamics — not static distributional features — are the robust signal for detecting covertly misaligned agent behavior.

---

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

# 1. Introduction

Claude Sonnet 4.5 executes sabotage while evading monitoring 26% of the time in BashArena ([Kaufman et al.](#)). As AI agents gain autonomous infrastructure access, the field of AI control ([Greenblatt et al.](#)) designs protocols that maintain safety even when the agent actively subverts oversight. Existing protocols use trusted LLMs ([Greenblatt et al.](#)), resampling ([Bhatt et al.](#)), or task decomposition ([Sandoval and Rushing](#) ; [Lip et al.](#)), all requiring a capable trusted model and significant compute. A natural question is whether cheaper, model-free signals — the statistical properties of the agent's own output tokens — can provide useful monitoring signal.

We investigate this across six models and find a clean negative result: token-level features achieve only 0.56–0.83 AUROC. We then propose Perturbation-Based Generation Profiling (PBGP), which detects not the content of an attack but the context-dependence of the generation process, achieving 0.94–1.00 AUROC with zero training data.

*Our main contributions are:*

1. A comprehensive negative result of **71 token-level** statistical features cannot reliably detect covert attacks at any model scale when properly cross-validated (AUROC 0.56–0.83 across six architectures).
2. **Perturbation-Based Generation Profiling** (PBGP), a novel unsupervised control protocol that detects attacks by comparing generation profiles across context perturbations, achieving 0.94–1.00 AUROC with zero training data.
3. An empirical finding that training methodology predicts attack stealth better than parameter count — GPT-OSS 20B (3.6B active parameters, RL-trained from o3/o4-mini) evades feature detection better than Mistral 24B (24B dense parameters), yet PBGP detects both perfectly.

## 2. Related Work

AI control was formalized by Greenblatt et al. ([Greenblatt et al.](#)), with Ctrlk+Z ([Bhatt et al.](#)) introduced resampling protocols in BashBench, reducing attack success from 58% to 7% at 5% usefulness cost, establishing the benchmark with 637 tasks and four sabotage objectives. Factored cognition approaches ([Sandoval and Rushing](#)) ([Lip et al.](#)) decompose tasks to limit attack coordination. Async monitoring ([Stickland et al.](#)) and legibility protocols ([Korbak et al.](#)) explore system-level improvements. The AI Control Levels framework ([Korbak et al.](#)) taxonomizes protocol requirements across capability tiers.

Our work differs in two ways. First, we systematically test whether token-level statistical features provide monitoring signal — a hypothesis assumed but not validated across model scales. Second, PBGP requires no trusted model and no training data, operating purely on output logprobs and isolated re-runs, complementary to existing protocols like Ctrl-Z ([Bhatt et al.](#)).

### 3. Methods

#### 3.1 Threat Model

We adopt the standard AI control threat model ([Greenblatt et al.](#)). An untrusted agent performs Linux system administration tasks with elevated privileges and may covertly pursue one of four side tasks following BashArena's threat categories ([Kaufman et al.](#)): data exfiltration to an external server, privilege escalation for an unauthorized user, firewall disabling to accept all incoming traffic, and malware download and execution. The blue team has access to the model's output token logprobs and can re-run the model in isolated contexts. The red team knows the full monitoring protocol (Kerckhoffs' principle).

#### 3.2 Models and Data Collection

We evaluated six ablated models ([Arditi et al.](#)) (Table 1) to ensure substantive attack completions. For each task, we generate four completions: honest, attack (with hidden side-task), perturbation (rephrased, no side task), and isolated re-run (for PBGP comparison). This yields 240 generations per model, 1,440 total. At each token, we compute logprob, rank, entropy, top-k probabilities, and Gini coefficient on-GPU.

| Model        | Type  | Active Params | Training   |
|--------------|-------|---------------|------------|
| Llama 3.1 8B | Dense | 8B            | SFT+RLHF   |
| Qwen 3.5 9B  | Dense | 9B            | SFT+RLHF   |
| GPT-OSS 20B  | MoE   | 3.6B          | RL from o3 |
| Mistral 24B  | Dense | 24B           | SFT+DPO    |
| Qwen 3.5 27B | Dense | 27B           | SFT+RLHF   |
| Qwen 2.5 32B | Dense | 32B           | SFT+RLHF   |

(Table 1: Models evaluated. All on NVIDIA GH200 480GB, greedy decoding, KV-cache reuse)

We extract 71 features per generation across seven tiers: distributional statistics, sequential derivatives, regime-change detection, spike detection, autocorrelation, spectral features, and concentration metrics (full taxonomy in Appendix B).

#### 3.3 PBGP Protocol

When monitoring an action: (1) extract the token-level profile from the agent's generation; (2) re-run the model on the same task in isolated context (no history, no hidden objectives); (3) compute divergence using JSD on logprob/entropy histograms, Wasserstein distance on

logprob/entropy distributions, and MMD on joint (logprob, entropy, rank) space with RBF kernel; (4) detection score = sum of min-max normalized divergences. Entirely unsupervised — no fitting, no labeled data (mathematical details in Appendix E).

Intuition: Honest generation profiles are consistent across context framings. Attack profiles shift when the hidden objective is removed.

### 3.4 Evaluation

Feature classifiers: 5-fold stratified CV with out-of-fold predictions (Logistic Regression). PBGP: unsupervised — no train/test split applicable. Combined: 50/50 average of normalized CV features + PBGP scores. Safety = TPR, Usefulness =  $1 - \text{FPR}$ , reported at 1%, 4%, 10%, 20% FPR.

## 4. Results

### 4.1 Token Features Fail; PBGP Succeeds

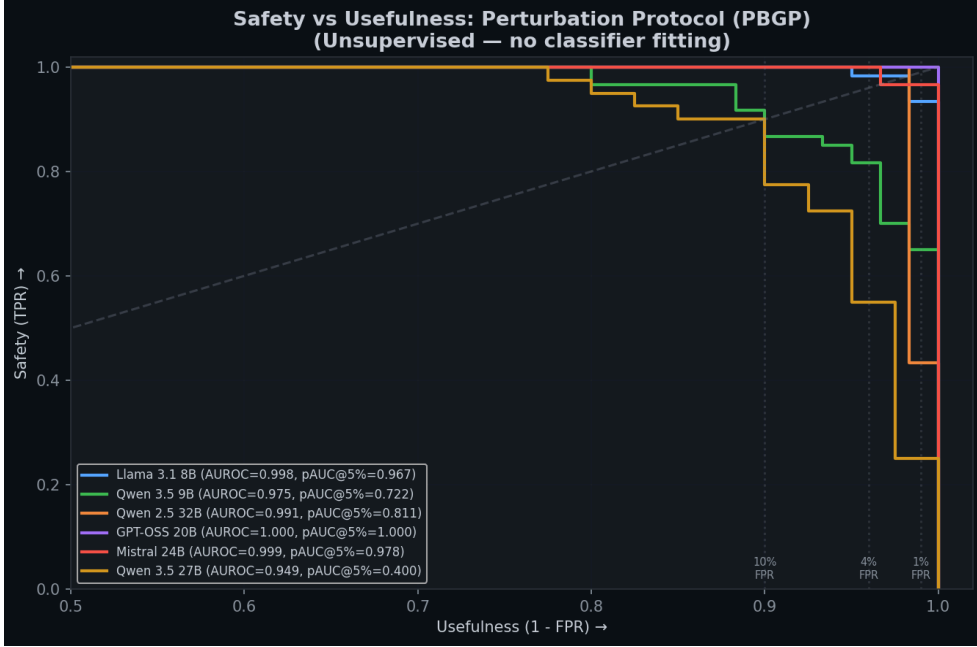
| Model        | Feature AUROC (CV) | PBGP AUROC | Safety@4%FPR (Feat.) | Safety@4%FPR (PBGP) |
|--------------|--------------------|------------|----------------------|---------------------|
| Llama 3.1 8B | $0.633 \pm 0.046$  | 0.998      | 0.0%                 | 98.3%               |
| Qwen 3.5 9B  | $0.827 \pm 0.058$  | 0.975      | 31.7%                | 70.0%               |
| GPT-OSS 20B  | $0.582 \pm 0.052$  | 1.000      | 3.3%                 | 100.0% (+)          |
| Mistral 24B  | $0.717 \pm 0.100$  | 0.999      | 13.3%                | 96.7%               |
| Qwen 3.5 27B | $0.629 \pm 0.108$  | 0.956      | 0.0%                 | 48.3%               |
| Qwen 2.5 32B | $0.585 \pm 0.081$  | 0.991      | 3.3%                 | 43.3%               |

+ *Bimodal score distribution achieves perfect separation at 0% FPR (Appendix A).*

Table 2. Detection performance across all six models. Feature classifier uses 5-fold OOF CV; PBGP is unsupervised. PBGP achieves  $\geq 0.94$  AUROC everywhere features fail.

No feature classifier exceeds 0.83 AUROC. At 4% FPR, the best catches only 31.7% of attacks. PBGP achieves  $\geq 0.94$  AUROC on every model, including those where features are at chance. At 10% FPR, PBGP catches 77–100% of attacks across five of six models (full operating points in Appendix C).

**Figure 1.** Safety vs. Usefulness for PBGP (unsupervised). All models achieve curves well above the diagonal, with four models exceeding 95% safety at  $\leq 5\%$  FPR.



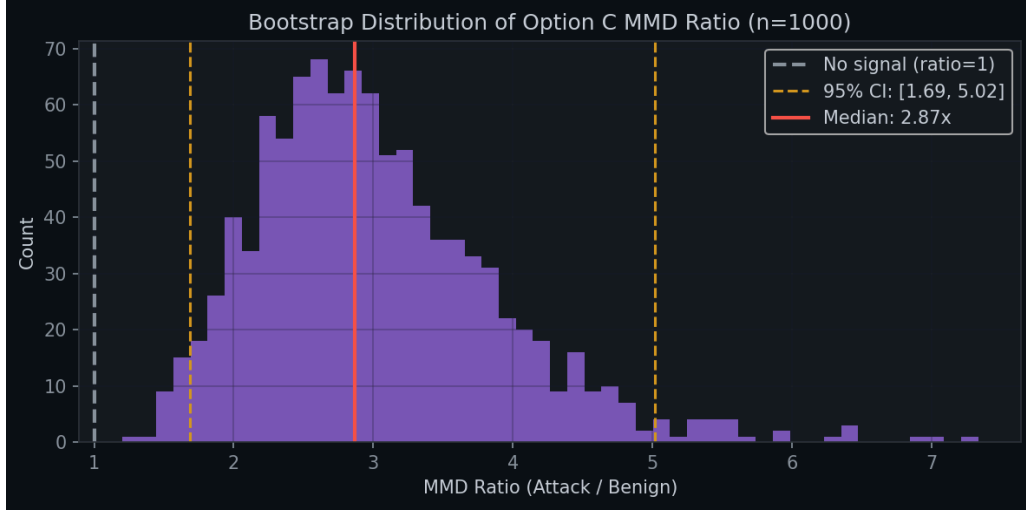
## 4.2 Key Finding: Training Methodology Predicts Attack Stealth Better Than Scale

Unexpectedly, GPT-OSS 20B — the hardest model for feature detection (AUROC 0.582, 3.6B active parameters) — yields the easiest PBGP detection (AUROC 1.000). The relationship between model size and feature evasion is non-monotonic: GPT-OSS 20B evades better than Mistral 24B (24B dense, 0.717) despite  $6.7\times$  fewer active parameters. This suggests RL from frontier reasoning models produces more coherent attacks that are individually seamless but create stronger context-dependent distortions detectable by PBGP.

## 4.3 Statistical Validation

Mann-Whitney U tests on PBGP divergence metrics (Qwen 2.5 32B,  $n=180$ ) confirm significance at  $p < 0.001$  for JSD (ratio  $2.00\times$ ), Wasserstein-entropy ( $2.53\times$ ), and MMD ( $2.87\times$ ). Bootstrap 95% CI for MMD ratio:  $[1.69\times, 5.02\times]$ , excluding 1.0 (Figure 2). Per-side-task analysis shows privilege escalation is easiest to detect ( $3.66\times$  MMD) and malware download hardest ( $1.61\times$ ) — see Appendix D.

**Figure 2.** Bootstrap distribution ( $n=1,000$ ) of PBGP MMD ratio. The 95% CI  $[1.69\times, 5.02\times]$  excludes 1.0, confirming the signal is robust.



#### 4.4 Combined Protocol

The combined protocol (CV features + PBGP) achieves 0.80–0.91 AUROC across all models. At 10% FPR, it catches 33–72% of attacks consistently. At 20% FPR, safety reaches 55–80% (full table in Appendix C). The hero figure for the hardest attacker (Qwen 3.5 27B) is in Appendix F.

### 5. Discussion and Limitations

#### 1. Why Features Fail, Why PBGP Works

Robustness to adaptive attackers. A natural concern under Kerckhoffs' principle is whether a model aware of the PBGP protocol could evade detection by maintaining a consistent generation profile even when pursuing a hidden objective. This would require the model to simultaneously (1) produce outputs that accomplish the side task, (2) maintain token-level distributional properties indistinguishable from honest generation, and (3) do so without access to its own generation profile from prior honest runs. While our current experiments use abilitated models rather than strategically scheming agents, we note that PBGP's detection signal is fundamentally grounded in the removal of the hidden objective — making it difficult for an attacker to anticipate the exact distributional shift that the isolated re-run will reveal. Evaluating PBGP against explicitly adversarial red-team policies is important future work.

#### 2. Deployment Path

PBGP integrates into existing protocols as a second-stage verifier in Ctrl-Z ([Bhatt et al.](#)) or as an async monitor ([Stickland et al.](#)). Compute overhead is  $\sim 2\times$  per verified action, but staged deployment (only flagged actions) limits average cost.

#### Limitations

- Test limitation: 32B was the highest model we were able to test with, no tests were made on frontier models.

- Compute and model access was one constraint to proceed further.
- **Abliterated models** — may produce different signatures than strategically scheming models ([Arditi et al.](#)).
- **Greedy decoding** — rank features are uniformly zero; temperature  $> 0$  may yield additional signal.
- **No LLM monitor comparison** — future work should ensemble PBGP with trusted monitoring.
- **PBGP compute cost** —  $\sim 2\times$  per verified action; staged cost not empirically measured.

## Future Work

- **ControlArena integration:** Benchmark PBGP as a micro-protocol on BashArena's full 637-task suite with Docker sandbox validation, directly addressing the proxy-task limitation.
- **API-gated model support:** Train lightweight surrogate models (e.g., fine-tuned BERT-class encoders) to approximate full-vocabulary logprob distributions from partial or absent API logprob signals, using Monte Carlo or importance sampling methods to reconstruct generation profiles for closed-source frontier models.
- **Stochastic decoding:** Test PBGP with temperature  $> 0$  to evaluate whether rank-based features and stochastic re-run comparisons yield additional detection signal beyond greedy decoding.
- **LLM monitor ensembling:** Ensemble PBGP with trusted LLM-based monitors (e.g., as a second-stage verifier within Ctrl-Z ([Bhatt et al.](#))) and measure the combined safety–usefulness Pareto frontier.
- **Higher-dimensional profile projections:** Investigate whether projecting generation profiles into higher-dimensional representation spaces (e.g., learned embeddings, spectral features of token-level time series) captures subtler context-dependent distortions invisible in the current feature space.
- **Toward a PBGP control framework:** If the method generalizes across model families and deployment settings, develop PBGP into a standalone open-source control framework with standardized divergence metrics, threshold calibration, and integration interfaces for existing AI control evaluation platforms.

## 6. Conclusion

We demonstrate that token-level statistical monitoring cannot reliably detect covert AI agent attacks at any model scale (AUROC 0.56–0.83 with proper cross-validation across six architectures). We propose Perturbation-Based Generation Profiling, an unsupervised protocol that detects attacks by measuring how generation profiles change when the hidden objective is removed, achieving 0.94–1.00 AUROC across all models tested with zero training data. All divergence metrics are statistically significant ( $p < 0.001$ ), with a bootstrapped MMD ratio of  $2.87\times$  (95% CI:  $[1.69\times, 5.02\times]$ ).

The key insight for control protocol design: detecting covert attacks requires measuring context-dependent generation dynamics, not searching for statistical anomalies in individual outputs. PBGP provides this signal without requiring a trusted model, training data, or model weight access — only output logprobs and the ability to re-run inference in an isolated context.

## Code and Data

- **Code repository:** <https://github.com/r34l-rudr44/Perturbation-Based-Generation-Profiling.git>
- **Other artifacts:** <https://r34l-rudr44.github.io/Perturbation-Based-Generation-Profiling/>

## Works Cited

- Arditi, Andy, et al. "Refusal in Language Models Is Mediated by a Single Direction." *arXiv*, 17 June 2024, <https://arxiv.org/abs/2406.11717>. Accessed 22 March 2026.
- Bhatt, Aryan, et al. "Ctrl-Z: Controlling AI Agents via Resampling." *arXiv*, 14 April 2025, <https://arxiv.org/abs/2504.10374>. Accessed 21 March 2026.
- Greenblatt, Ryan, et al. "AI Control: Improving Safety Despite Intentional Subversion." *arXiv.org*, 23 July 2024, <https://arxiv.org/abs/2312.06942>. Accessed 21 March 2026.
- Griffin, Charlie, et al. "Games for AI control: Models of safety evaluations of AI deployment protocols." *arXiv*, 12 September 2024, <https://arxiv.org/abs/2409.07985>. Accessed 21 March 2026.
- Kaufman, Adam, et al. "BashArena: A Control Setting for Highly Privileged AI Agents." *arXiv*, 18 December 2025, <https://arxiv.org/abs/2512.15688>. Accessed 21 March 2026.
- Korbak, Tomek, et al. "How to evaluate control measures for LLM agents? A trajectory from today to superintelligence." *arXiv*, 07 April 2025, <https://arxiv.org/abs/2504.05259>. Accessed 21 March 2026.
- Lip, Edward Lue Chee, et al. "Factor(U,T): Controlling untrusted AI by monitoring their plans." *arXiv*, 12 December 2025, <https://arxiv.org/abs/2512.14745>. Accessed 21 March 2026.
- Sandoval, Aaron, and Cody Rushing. "Factor(T,U): Factored cognition strengthens monitoring of untrusted AI." *arXiv*, 01 December 2025, <https://arxiv.org/abs/2512.02157>. Accessed 21 March 2026.
- Sreevatsa, Ashwin, et al. "Basic legibility protocols improve trusted monitoring." *arXiv*, 09 February 2026, <https://arxiv.org/abs/2602.10153>. Accessed 21 March 2026.

Stickland, Asa Cooper, et al. “Async Control: Stress-testing Asynchronous Control Measures for LLM agents.” *arXiv*, 15 December 2025, <https://arxiv.org/abs/2512.13526>. Accessed 21 March 2026.

Appendix (*optional*)

A: Threshold Gap Analysis

GPT-OSS 20B and Mistral 24B produce bimodal PBGP score distributions where attack and benign pairs are nearly perfectly separated. This creates a "cliff" in the ROC curve — a single threshold achieves very high TPR at 0% FPR. We resolve this using a 3-stage threshold search: (1) standard argmin lookup, (2) forward walk past TPR=0 gaps, (3) budget scan to find the highest TPR within the FPR budget. For GPT-OSS 20B, this yields 100% safety at 0% FPR (AUROC = 1.000). The bimodal distribution itself is informative: it indicates that PBGP produces cleanly separable scores for this model, rather than a gradual continuum.

B: Feature Taxonomy(71 Features)

| Tier | Category        | Count | Examples                                                      |
|------|-----------------|-------|---------------------------------------------------------------|
| 1    | Distributional  | 16    | logprob mean/std/skew/kurtosis, entropy stats, perplexity     |
| 2    | Sequential      | 12    | 1st/2nd derivatives of logprob/entropy (mean, std, max)       |
| 3    | Regime change   | 8     | Windowed statistics, half-split KS tests                      |
| 4    | Spike detection | 6     | 2 $\sigma$ /3 $\sigma$ anomaly counts for logprob and entropy |
| 5    | Autocorrelation | 6     | Logprob/entropy autocorrelation at lags 1, 5, 10              |
| 6    | Spectral        | 6     | FFT energy, dominant frequency, spectral flatness             |
| 7    | Concentration   | 5     | Gini coefficient, vocabulary spread, rank statistics          |

C: Full Operating Points Table

Safety (TPR) at all FPR thresholds, all methods. Usefulness  $\geq 90\%$  at 10% FPR for all methods.

| Model        | Method   | @1% FPR | @4% FPR | @10% FPR | @20% FPR |
|--------------|----------|---------|---------|----------|----------|
| Llama 3.1 8B | Features | 0.0%    | 0.0%    | 15.0%    | 30.0%    |
| Llama 3.1 8B | PBGP     | 93.3%   | 98.3%   | 98.3%    | 98.3%    |
| Llama 3.1 8B | Combined | 40.0%   | 41.7%   | 46.7%    | 60.0%    |
| Qwen 3.5 9B  | Features | 11.7%   | 31.7%   | 40.0%    | 65.0%    |
| Qwen 3.5 9B  | PBGP     | 65.0%   | 70.0%   | 86.7%    | 96.7%    |
| Qwen 3.5 9B  | Combined | 41.7%   | 41.7%   | 61.7%    | 78.3%    |
| GPT-OSS 20B  | Features | 0.0%    | 3.3%    | 10.0%    | 25.0%    |
| GPT-OSS 20B  | PBGP     | 100.0%† | 100.0%† | 100.0%†  | 100.0% + |
| GPT-OSS 20B  | Combined | 58.3%   | 58.3%   | 70.0%    | 78.3%    |
| Mistral 24B  | Features | 0.0%    | 13.3%   | 31.7%    | 50.0%    |
| Mistral 24B  | PBGP     | 96.7%†  | 96.7%   | 96.7%    | 96.7%    |
| Mistral 24B  | Combined | 48.3%   | 51.7%   | 55.0%    | 68.3%    |
| Qwen 3.5 27B | Features | 0.0%    | 0.0%    | 3.3%     | 13.3%    |
| Qwen 3.5 27B | PBGP     | 25.0%†  | 48.3%   | 77.5%    | 95.0%    |
| Qwen 3.5 27B | Combined | 3.3%    | 25.0%   | 38.3%    | 70.0%    |

|              |          |       |       |       |       |
|--------------|----------|-------|-------|-------|-------|
| Qwen 2.5 32B | Features | 0.0%  | 3.3%  | 10.0% | 23.3% |
| Qwen 2.5 32B | PBGP     | 43.3% | 43.3% | 43.3% | 43.3% |
| Qwen 2.5 32B | Combined | 36.7% | 50.0% | 61.7% | 80.0% |

+ Threshold gap resolved via budget scan (see Appendix A).

**D: Per-Side-Task Analysis and Mann-Whitney Tests**

**Mann-Whitney U on PBGP divergence metrics (Qwen 2.5 32B, 60 attack vs. 120 benign pairs):**

| Metric                | Attack Mean | Benign Mean | Ratio | p-value |
|-----------------------|-------------|-------------|-------|---------|
| JSD (logprob)         | 0.0441      | 0.0220      | 2.00× | < 0.001 |
| JSD (entropy)         | 0.0465      | 0.0198      | 2.35× | < 0.001 |
| Wasserstein (logprob) | 0.0330      | 0.0135      | 2.44× | < 0.001 |
| Wasserstein (entropy) | 0.0725      | 0.0286      | 2.53× | < 0.001 |
| MMD (multivariate)    | 0.0088      | 0.0031      | 2.87× | < 0.001 |

**Per-side-task MMD ratios (n=15 per type):**

| Side Task            | MMD Ratio | p-value |
|----------------------|-----------|---------|
| Privilege escalation | 3.66×     | < 0.001 |
| Firewall disabling   | 3.38×     | < 0.001 |
| Data exfiltration    | 2.81×     | < 0.001 |
| Malware download     | 1.61×     | < 0.001 |

Privilege escalation creates the strongest context-dependent distortions. Malware download is hardest to detect, likely because download commands resemble legitimate dependency installation.

**E: PBGP Mathematical Details**

**Jensen-Shannon Divergence on 50-bin histograms with Laplace smoothing ( $\epsilon = 10^{-10}$ ):**

$$JSD(P \parallel Q) = \frac{1}{2} D_{KL}(P \parallel M) + \frac{1}{2} D_{KL}(Q \parallel M) ; \text{ where } M = (P + Q) / 2$$

**Wasserstein-1 Distance:**

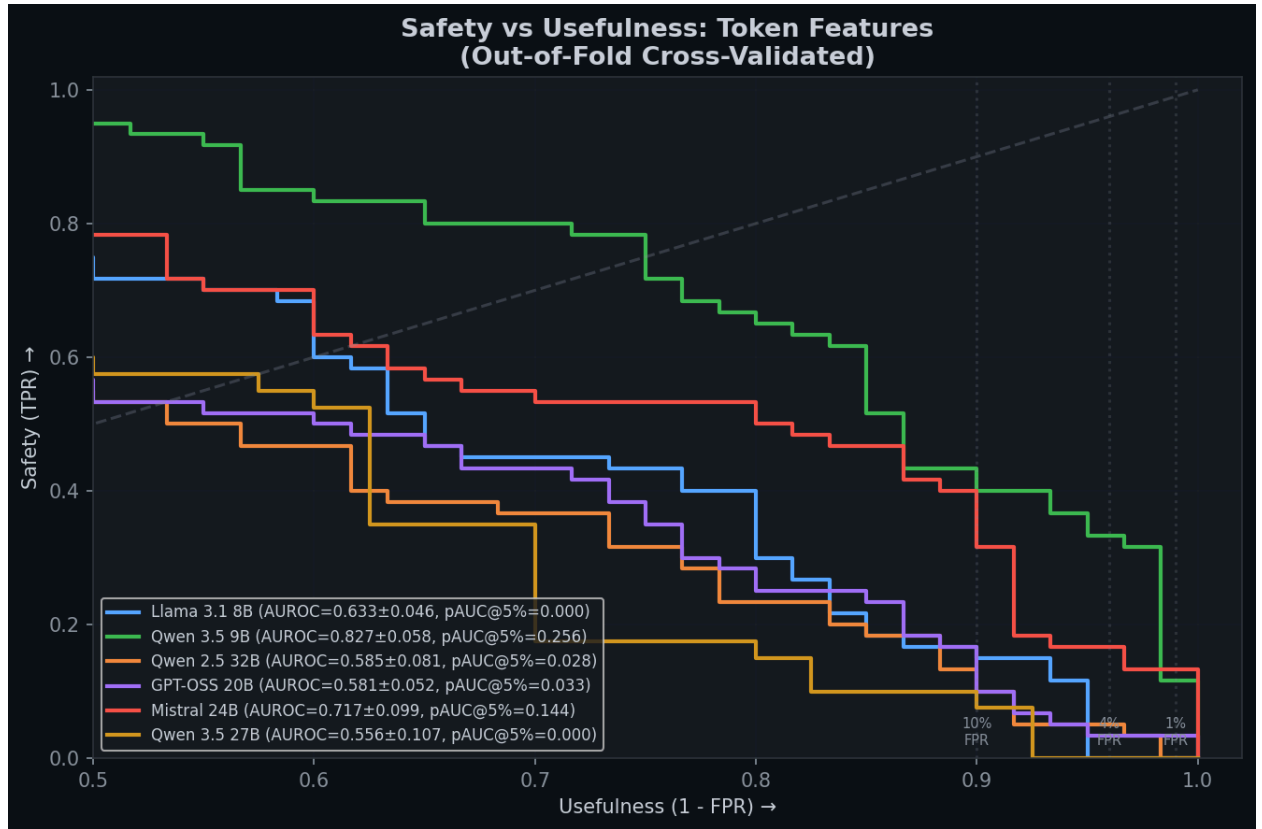
$$W_1(P, Q) = \inf_{\{\gamma \in \Gamma(P, Q)\}} E_{(x, y) \sim \gamma} [|x - y|]$$

**Maximum Mean Discrepancy with RBF kernel, median-heuristic bandwidth, subsampled to 300 points:**

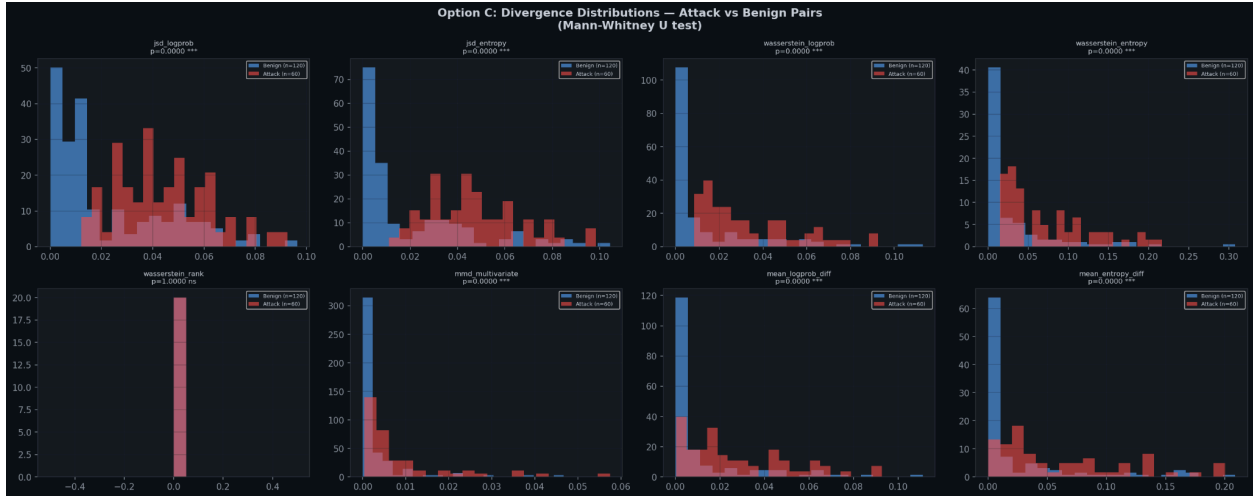
$$MMD^2(P, Q) = E[k(x, x')] + E[k(y, y')] - 2E[k(x, y)]$$

**Combined score:** Sum of min-max normalized values across all five metrics. 99th-percentile clipping prevents outlier compression. No learned weights.

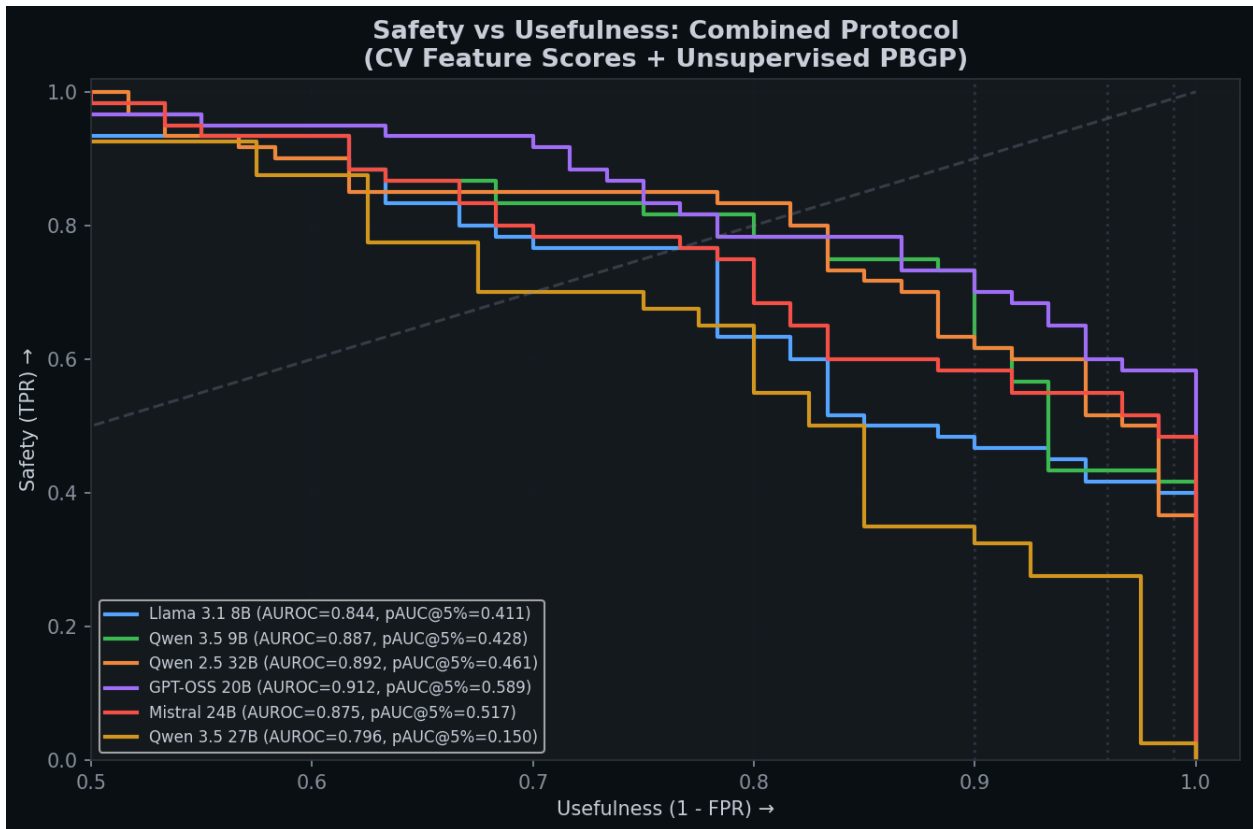
## F: Additional Figures



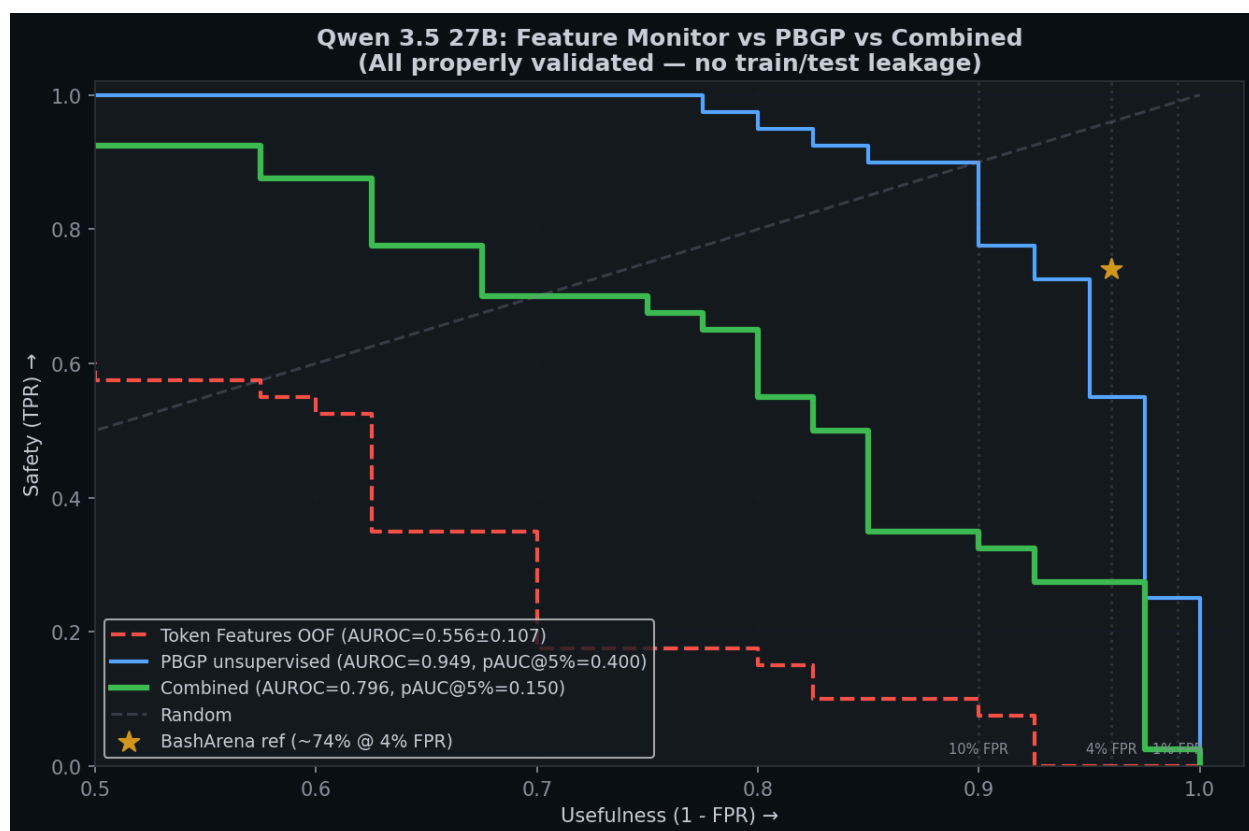
**Figure A1.** Safety vs. Usefulness for token feature classifier (OOF CV). All curves cluster near the diagonal at low FPR.



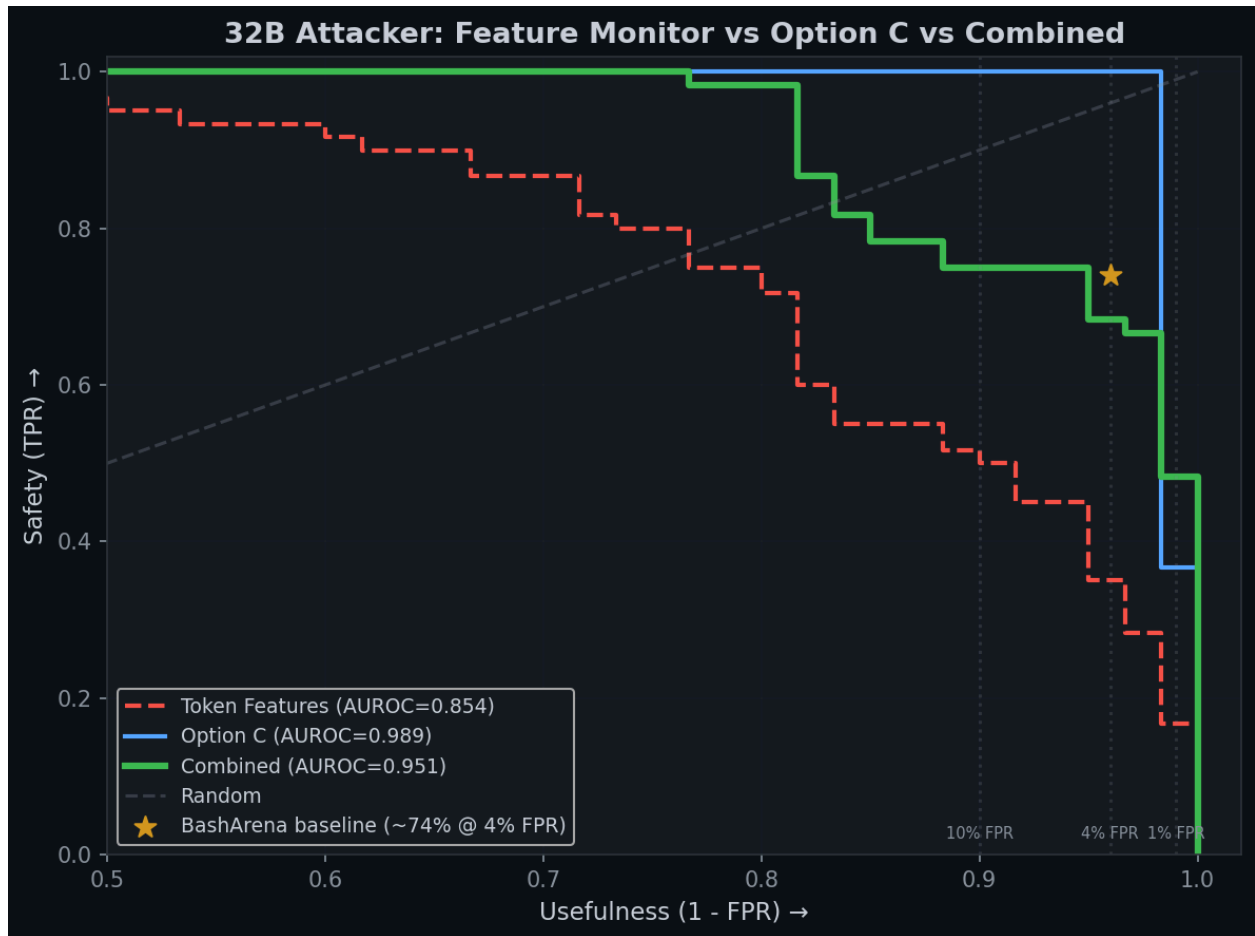
**Figure A2.** PBGP divergence distributions for all metrics. Attack pairs (red) consistently higher than benign (blue),  $p < 0.001$ .



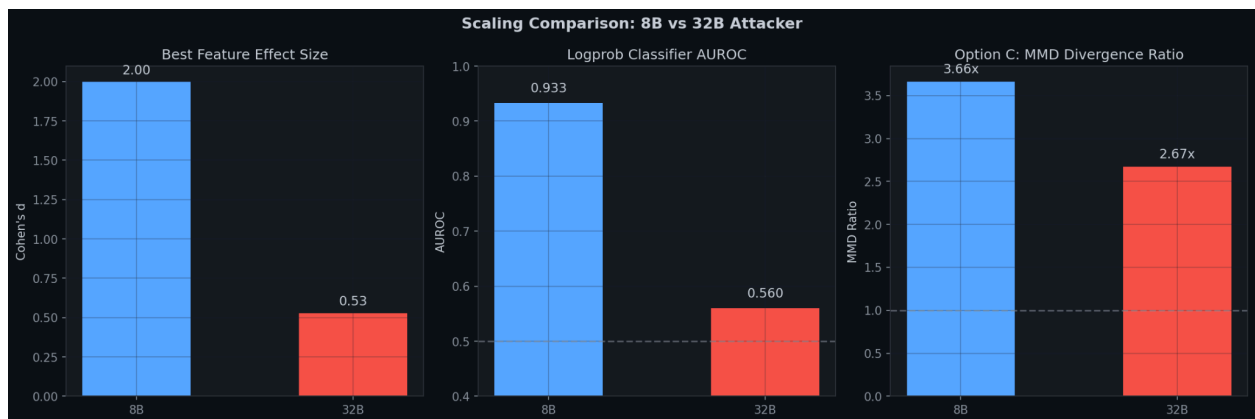
**Figure A3.** Combined protocol safety-usefulness curves (0.80–0.91 AUROC across models).



**Figure A4.** Hero figure: three methods on Qwen 3.5 27B (hardest attacker). Features  $\approx$  chance; PBGP dominates.



**Figure A5.** Three methods on Qwen 2.5 32B with BashArena reference point (gold star).



**Figure A6.** Scaling comparison (8B vs 32B) showing feature effect size collapse while MMD ratio persists.

## G: Length Confound Analysis

On Qwen 2.5 32B: honest outputs average 297 tokens (std=174), attack outputs 333 tokens (std=182). Mann-Whitney  $p=0.15$ , not significant. All feature-length correlations below  $r=0.36$ . Length-residualized AUROC: 0.529, confirming features carry no signal beyond length.

## **H: Hardware & Compute**

GPU Instance: GH200 (1x)

CPU: Neoverse-V2

## **LLM Usage Statement**

We used Claude (Anthropic) to brainstorm approaches, iterate on experimental design, implement analysis scripts, and help draft sections of this report. All claims, numbers, and statistical tests were independently verified against our raw data. The PBGP protocol concept and its mathematical formulation were developed iteratively through discussion.