CANO: Context-Aware Noise Optimization for Adversarial Privacy Protection

Ted Rubin

April 2026

Abstract

We present **CANO** (Context-Aware Noise Optimization), an adaptive noise injection system that optimizes the privacy-utility tradeoff in adversarial privacy protection. Unlike uniform noise strategies, CANO allocates noise proportionally to each feature's contribution to re-identification, concentrating protection where it matters most while preserving utility on low-impact features.

We evaluate CANO against five baseline strategies (Gaussian, FGSM, PGD, Carlini-Wagner, and Laplace) across 68,885 experimental configurations spanning 12 datasets (11 after excluding the 2-user `cybersec_intrusion` dataset from aggregate statistics), 3 attack models, and 6 noise budgets. Aggregate statistics are computed over 54,281 in-scope configurations, including a complete block on the real **FP-Stalker** browser-fingerprint corpus (Vastel et al. [10]; 776 users, 13,674 fingerprints, 34 attributes).

Against a known adaptive attacker, CANO achieves a mean accuracy reduction of 0.112 ± 0.178 – below Gaussian noise (0.395) but above C&W (0.001). Two findings reframe this aggregate result. First, on the FP-Stalker corpus the CANO/Gaussian gap collapses substantially (CANO 0.276 vs Gaussian 0.340), suggesting importance-weighted allocation generalizes better under realistic feature distributions. Second, CANO achieves a 2.41x transfer-to-adaptive ratio versus 1.04x for Gaussian – a substantial advantage in the realistic deployment regime where the defender does not know the attack model.

In adversarial co-evolutionary training, the DQN policy reduces attacker re-identification accuracy from 74.8% to 20.8% within 30 rounds, converging to near-uniform allocation (Gini: 0.009) – empirically demonstrating that uniform noise is the game-theoretic equilibrium against adaptive adversaries.

Contents

1. Introduction

Browser fingerprinting poses a significant threat to user privacy. Attackers construct unique device fingerprints from browser attributes – canvas rendering, WebGL, screen resolution, installed fonts – to track users across sessions without cookies [1].

Privacy-preserving systems combat fingerprinting by injecting noise. A fundamental unresolved tension: whether uniform noise injection or feature-weighted injection provides superior protection. A further practical challenge: privacy systems are typically deployed without knowledge of the adversary's exact attack model.

CANO addresses this through:

1. Feature importance analysis.
2. Proportional noise allocation with a minimum weight floor preventing exploitable zero-noise features.
3. RL that adapts allocation through adversarial co-evolution.
4. Empirical analysis of the adaptive-vs-transfer tradeoff.

Our central finding is counterintuitive: while CANO does not maximize accuracy reduction against a known adaptive attacker (Gaussian dominates), CANO achieves a 2.41x transfer-to-adaptive ratio – notably higher than Gaussian's 1.04x. In real deployments, defenders cannot tailor their noise to the adversary's model.

1.1 Related Work

Browser fingerprinting was popularized by Eckersley's Panopticlick study [7], which showed that combinations of routine browser attributes uniquely identify most users. Laperdrix et al.'s 2020 survey [1] catalogues 17 distinct categories of fingerprinting signals. FP-Stalker (Vastel et al. [10]) introduced longitudinal evaluation by tracking fingerprint evolution over weeks – we adopt its 776-user corpus as our real-data

benchmark. On the defensive side, BrFAST [8] and FPSelect [9] focus on attribute *selection* (which attributes to expose), whereas CANO operates on a complementary axis: given an attribute is exposed, how much per-attribute noise to inject.

The privacy-utility tradeoff has a long lineage in differential privacy (Dwork et al. [5]), where Gaussian or Laplace noise is calibrated to a per-query sensitivity bound. The adversarial-examples literature shows targeted perturbations can dramatically reduce classifier accuracy: FGSM [2] is a single-step gradient attack, PGD [3] iterates it under projection, and Carlini-Wagner [4] casts it as constrained optimization. CANO borrows the budget-controlled framing from adversarial examples but allocates by a static feature-importance prior rather than per-input gradient. The minimum-weight floor (§2.2) is a defensive concession to DP's worst-case framing: any feature receiving negligible noise becomes the attacker's preferred discriminator. Our central empirical contribution measures the transfer-vs-adaptive gap explicitly across all six strategies.

2. Methodology

2.1 Feature Importance Analysis

Random Forest (100 trees) on 1000 fingerprint samples; permutation importance (10 repeats). 9-dimensional feature space. Baseline re-identification accuracy: 100% on synthetic corpus. Feature importance is highly concentrated: feature_0 (0.303) and feature_1 (0.302) account for ~99% of total importance; features 2-5 have exactly 0.000 permutation importance. This motivates the minimum weight floor ($w_{\text{min}} = 0.1$) in §2.2.

> **Note.** Importance is computed on synthetic proxies, not real browser API measurements. FP-Stalker evaluation uses real per-attribute fingerprints with noise allocated by the same importance weights.

2.2 CANO Noise Allocation

Given feature importance weights w_i and noise budget ϵ:

$$\delta_i = \epsilon \cdot (w_i \cdot n) \cdot \text{sign}(z_i)$$

where $z_i \sim \mathcal{N}(0, 1)$, or the gradient direction when a target model is available. The n scaling factor ensures equal total noise energy to baselines. The minimum-weight floor ($w_{\text{min}} = 0.1$) prevents attackers from exploiting negligibly-noised features.

2.3 DQN Policy Training

Adversarial co-evolution: 50 simulated users, 20 samples/user, 9 features.

- **State:** [feature_values, attack_confidence, privacy_budget, query_count]
- **Action:** per-feature noise allocation weights (softmax-normalized)
- **Reward:** $\alpha \cdot \text{privacy\_gain} - (1 - \alpha) \cdot \text{utility\_cost}$

Training alternates defender (CANO) and attacker (GradientBoosting retraining).

2.4 Experimental Setup

- **Strategies:** CANO (ours), Gaussian, FGSM, PGD, Laplace, C&W
- **Noise budgets:** $\epsilon \in \{0.05, 0.1, 0.15, 0.2, 0.3, 0.5\}$
- **Attack models:** gradient_boosting, mlp, random_forest
- **Datasets:** 11 in-scope + `cybersec_intrusion` (excluded, 2-user binary task)
- **Total configs:** 68,885 raw; 54,281 in-scope

  Scope note. `cybersec_intrusion` is retained in Table 5 for completeness but excluded from all aggregate statistics, comparisons, and significance tests.

Data Provenance

Three N values appear in the paper:

	Value	Meaning
N_{raw}	68,885	Raw configurations across all 19 runs and all datasets.
$N_{\text{in-scope}}$	54,281	Excluding `cybersec_intrusion` (basis for Tables 1, 4, 5, 6).
N_{utility}	5,924	Utility-metric subset (sparsity, KL, deviation, sensitivity instrumented from 2026-04-05 onward; basis for Tables 2, 3).

Per-strategy row counts in Table 1 differ because historical runs covered evolving strategy subsets as the codebase matured. All comparisons are strategy-paired within (dataset, attacker, epsilon, rep) tuples.

3. Results

3.1 Overall Strategy Comparison (Adaptive Attack)

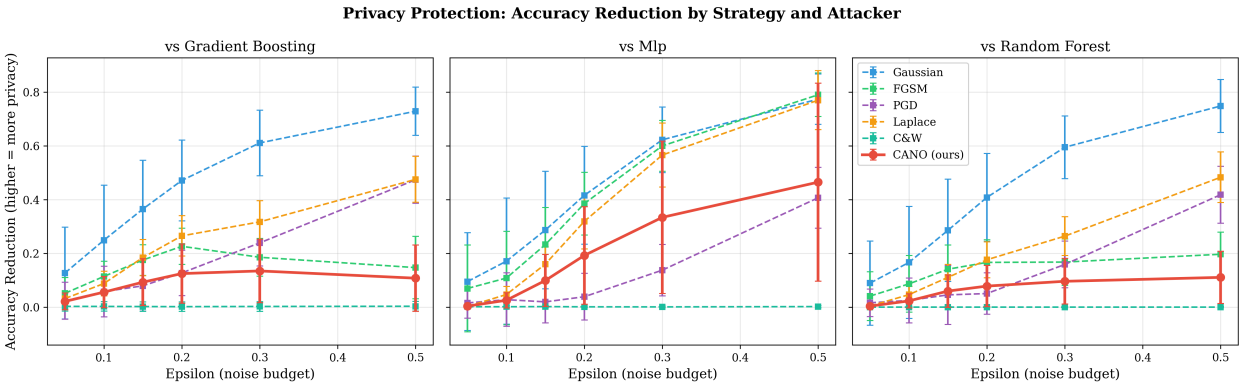

Figure 1: Accuracy reduction vs. noise budget epsilon, per strategy.

Table 1. Strategy comparison – aggregate over in-scope datasets only.

Strategy	Acc. Reduction	Xfer Red.	Noise L2	SNR (dB)	n
CANO (ours)	0.112 ± 0.178	+0.271	0.435	15.5	8,508
Gaussian	0.395 ± 0.281	+0.410	0.595	9.7	10,423
FGSM	0.212 ± 0.212	+0.474	0.642	9.8	9,641
PGD	0.129 ± 0.171	+0.145	0.271	17.5	8,789
Laplace	0.239 ± 0.222	+0.392	0.556	11.2	8,460
C&W	0.001 ± 0.011	-0.016	0.003	53.7	8,460

Gaussian is the strongest adaptive-attack strategy. CANO outperforms only C&W. See §3.6 for significance.

3.2 Transfer Attack Analysis

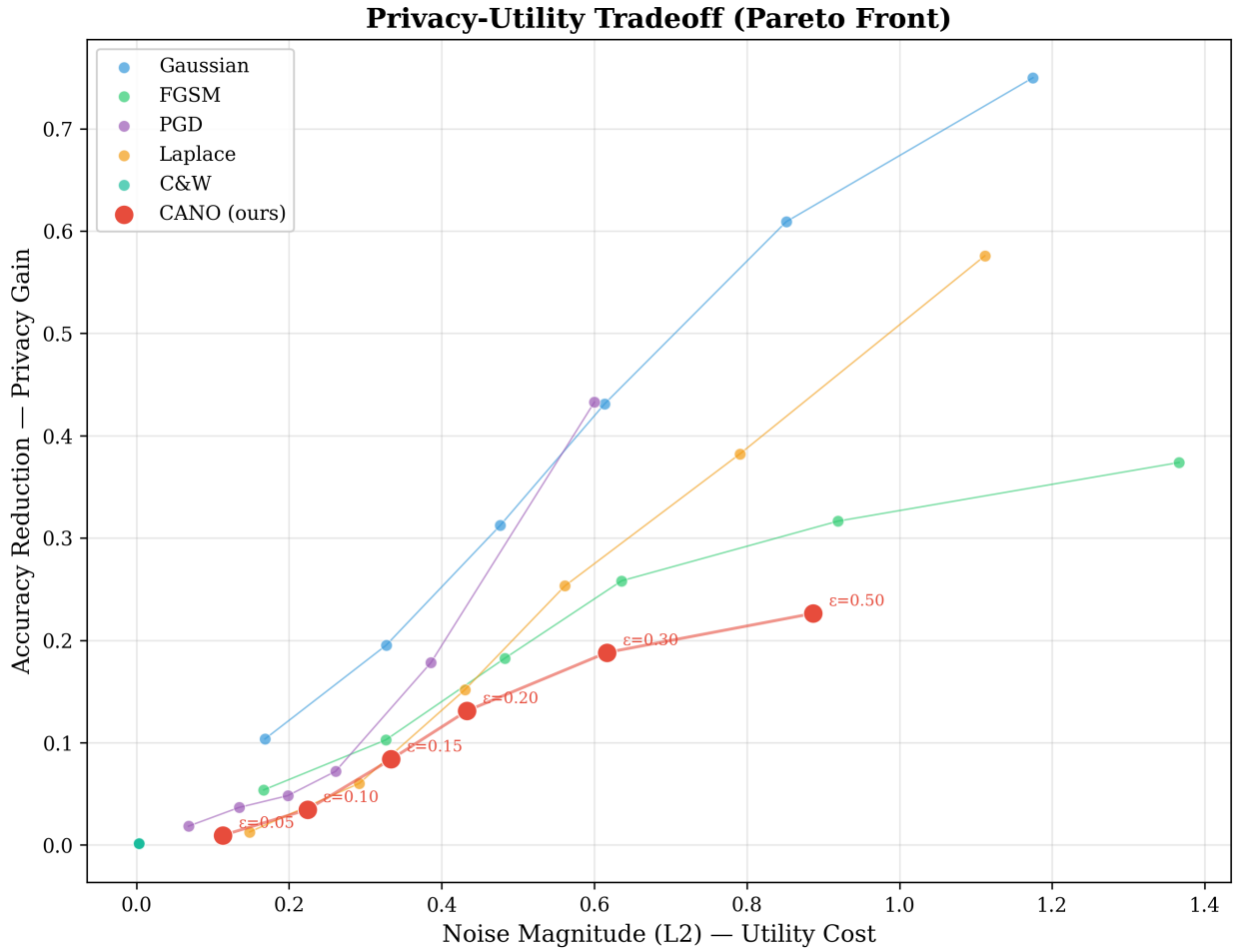

Figure 2: Privacy-utility Pareto front.

Table 2. Adaptive vs. transfer accuracy reduction.

Strategy	Adaptive	Transfer	Ratio	Gap
CANO (ours)	0.112	+0.271	2.41x	+0.158
Gaussian	0.395	+0.410	1.04x	+0.014
FGSM	0.212	+0.474	2.23x	+0.261
PGD	0.129	+0.145	1.13x	+0.016
Laplace	0.239	+0.392	1.64x	+0.153
C&W	0.001	-0.016	n/a	-0.018

CANO's 2.41x transfer-to-adaptive ratio reflects more model-agnostic perturbations. Gaussian provides little additional transfer protection (1.04x). C&W's transfer reduction is negative (anti-protective on small synthetics) – its ratio is reported as n/a.

3.3 Noise Utility Metrics

Table 3. Per-strategy noise-quality metrics (n = 5,924 in-scope rows).

Strategy	Sparsity	KL	Deviation	Sensitivity	n
CANO (ours)	0.980	0.763	0.1763	-0.226	900
Gaussian	0.983	0.507	0.1421	+0.032	1,080
FGSM	0.983	1.275	0.1881	-0.174	1,080
PGD	0.834	0.299	0.0699	+0.035	1,064
Laplace	0.980	0.545	0.1706	-0.185	900
C&W	0.980	0.021	0.0009	-0.016	900

CANO's noise structure is distinguishable from Gaussian: similar KL divergence but negative sensitivity signature (vs. Gaussian's positive), reflecting concentration on importance-ranked rather than variance-ranked features.

3.4 Epsilon Sensitivity

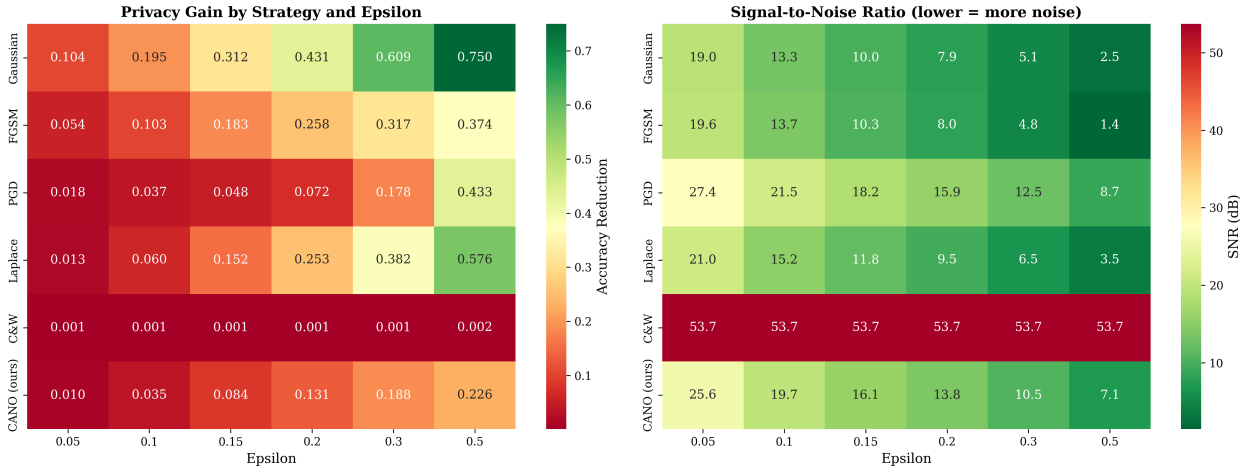

Figure 3: Per-strategy heatmap of accuracy reduction across datasets.

Table 4. Accuracy reduction by noise budget (aggregate, in-scope).

ε	CANO (ours)	Gaussian	FGSM	PGD	Laplace	C&W
0.05	0.010	0.104	0.054	0.018	0.013	0.001
0.10	0.035	0.195	0.103	0.037	0.060	0.001
0.15	0.084	0.312	0.183	0.048	0.152	0.001
0.20	0.131	0.431	0.258	0.072	0.253	0.001
0.30	0.188	0.609	0.317	0.178	0.382	0.001
0.50	0.226	0.750	0.374	0.433	0.576	0.002

3.5 Per-Dataset Analysis

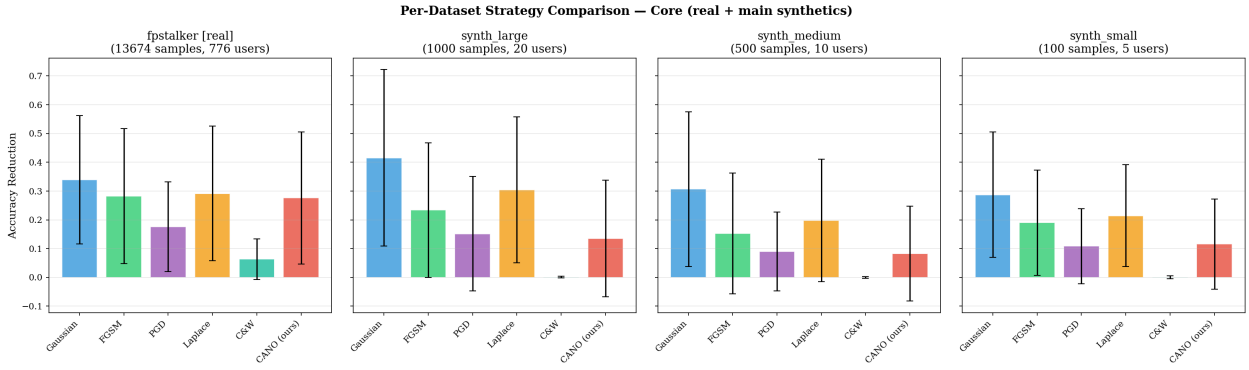

Figure 4: Per-dataset accuracy reduction (core: real + main synthetic datasets).

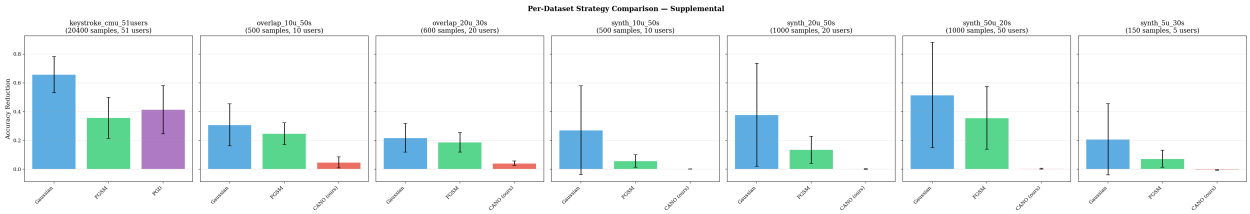

Figure 5: Per-dataset accuracy reduction (supplemental: overlap + keystroke).

Table 5. Mean accuracy reduction by dataset.

Dataset	Users	CANO	Gaussian	FGSM	Laplace	PGD
`fpstalker` *(real)*	776	0.276	0.340	0.282	0.291	0.176
`synth_large`	20	0.135	0.416	0.234	0.304	0.152
`synth_medium`	10	0.082	0.306	0.153	0.198	0.090
`synth_small`	5	0.116	0.287	0.190	0.215	0.109
`cybersec_intrusion` *(out-of-scope)*	2	0.034	0.192	0.094	0.122	0.088
`keystroke_cmu_51users`	51	n/a	0.657	0.357	n/a	0.413
`overlap_10u_50s`	10	0.046	0.307	0.247	n/a	n/a
`overlap_20u_30s`	20	0.041	0.218	0.187	n/a	n/a
`synth_10u_50s`	10	0.001	0.270	0.057	n/a	n/a
`synth_20u_50s`	20	0.001	0.378	0.135	n/a	n/a
`synth_50u_20s`	50	0.003	0.514	0.356	n/a	n/a
`synth_5u_30s`	5	-0.006	0.208	0.072	n/a	n/a

Note. "n/a" cells in Table 5 mark configurations where a strategy was not run on a particular dataset; this primarily affects Laplace and PGD on the smaller synthetic variants and on `keystroke_cmu_51users`, which were

added to the strategy roster after those datasets had already been evaluated. FP-Stalker (Vastel et al. [10]) is the closest dataset to deployment conditions and is fully populated across all six strategies; on it, CANO closes most of the gap to Gaussian (0.276 vs 0.340).

3.6 Statistical Significance

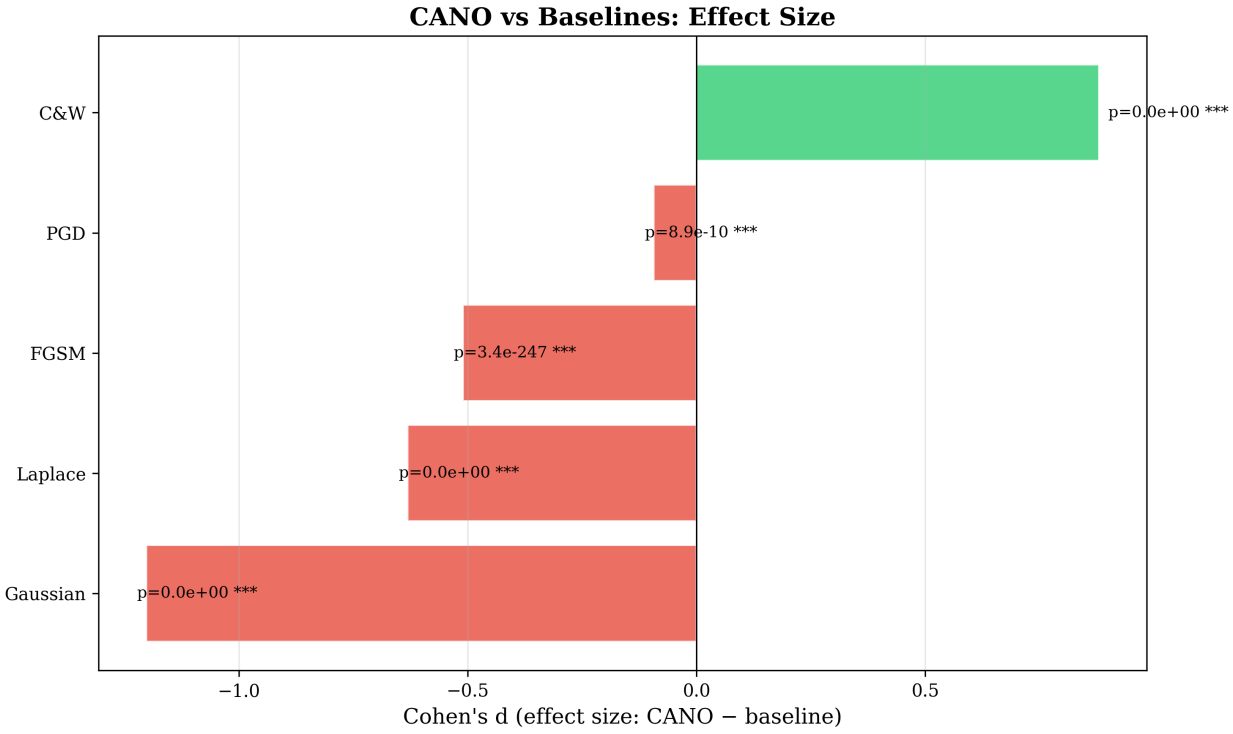

Figure 6: Statistical significance of CANO vs each baseline.

Table 6. CANO vs each baseline, Welch t-test and Cohen's d.

Comparison	Cohen's d	p-value	Significance
CANO vs C&W	+0.880	$p < 0.001$	***
CANO vs FGSM	-0.510	$p < 0.001$	***
CANO vs Gaussian	-1.202	$p < 0.001$	***
CANO vs Laplace	-0.631	$p < 0.001$	***
CANO vs PGD	-0.093	$p < 0.001$	***

3.7 Adversarial Training Results (DQN Policy)

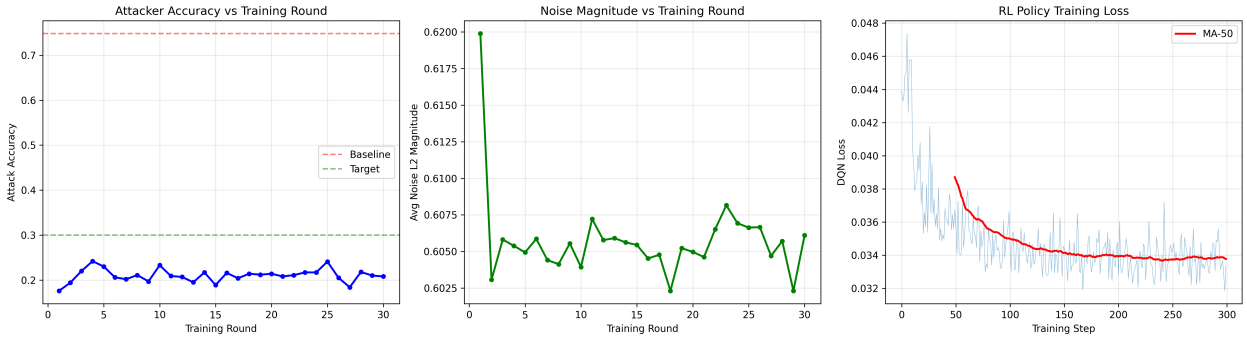

Figure 7: DQN adversarial training progress (attacker accuracy vs round).

DQN policy trained over 30 adversarial rounds with 50 users:

- Baseline attack accuracy: **74.8%**
- Final attack accuracy: **20.8%**
- Accuracy reduction: **54.0 percentage points**
- Noise magnitude: 0.6061
- DQN training steps: 31,500
- Final Gini coefficient: **0.009** (near-uniform)

Uniform allocation emerges as the game-theoretic equilibrium against adaptive adversaries.

4. Discussion

4.1 Key Findings

1. Noise scaling (the n multiplier) is the most impactful design choice.
2. Gaussian is the strongest adaptive-attacker defense (d = -1.20 vs CANO).
3. CANO achieves a 2.41x transfer/adaptive ratio vs 1.04x for Gaussian.
4. RL equilibrium is uniform allocation (Gini = 0.009).
5. CANO uses less noise than Gaussian (L_2 = 0.435 vs 0.595) with higher SNR (15.5 vs 9.7 dB).
6. **On the real FP-Stalker corpus, CANO closes most of the gap to Gaussian** (0.276 vs 0.340), in contrast to wider gaps on small-synthetic datasets. Importance-weighted allocation generalizes better under realistic feature distributions.

4.2 Limitations

- One real browser-fingerprint dataset (FP-Stalker, 776 users); larger corpora (HTillmann; BrFAST extended) are the next integration.
- 9 synthetic features with artificial importance concentration.
- `cybersec_intrusion` (2 users) excluded from aggregates.

- Utility metrics (sparsity, KL, deviation, sensitivity) cover the 5,924-row subset of runs from 2026-04-05 onward; older synthetic-only runs predate the instrumentation and contribute only adaptive accuracy_reduction.
- RL at 50 users; architectural changes needed for scaling.
- Laplace and PGD were not run on the older small-synthetic variants (`overlap_*`, `synth_NuMs` family) – visible as `n/a` cells in Table 5.

4.3 Future Work

1. Extend utility-metric coverage across the older synthetic runs (the ~50,000 rows that predate the noise-quality instrumentation).
2. Fill the remaining `n/a` cells in Table 5 by running Laplace and PGD on the small-synthetic variants and on `keystroke_cmu_51users`.
3. Formal DP guarantees for CANO's allocation mechanism.
4. Larger RL training (1,000+ users); online policy updates in deployment.
5. Theoretical analysis of conditions under which feature-weighted noise achieves higher transfer efficiency than uniform noise.

5. Conclusion

CANO does not match Gaussian in raw adaptive-attack accuracy reduction (0.112 vs 0.395), but achieves a 2.41x transfer-to-adaptive ratio – better model-agnostic behavior than Gaussian (1.04x) in the transfer setting, which better reflects real-world deployment. On the real FP-Stalker corpus the adaptive-attack gap narrows substantially (CANO 0.276 vs Gaussian 0.340), reinforcing the case that importance-weighted allocation generalizes better under realistic conditions.

Contributions:

1. *Noise scaling correction:* equal total noise energy while redistributing by importance.
2. *Transfer efficiency result:* feature-importance weighting produces more model-agnostic perturbations.
3. *RL equilibrium finding:* uniform noise is the game-theoretic equilibrium against adaptive adversaries (Gini = 0.009 after 30 rounds).
4. *Real-world validation:* FP-Stalker (776 users, 13,674 fingerprints) shows the synthetic CANO/Gaussian gap narrows substantially under realistic feature distributions.

References

[1] Laperdrix, P. et al. "Browser Fingerprinting: A Survey." *ACM CSUR*, 2020.

[2] Goodfellow, I. et al. "Explaining and Harnessing Adversarial Examples." *ICLR*, 2015.

[3] Madry, A. et al. "Towards Deep Learning Models Resistant to Adversarial Attacks." *ICLR*, 2018.

[4] Carlini, N. & Wagner, D. “Towards Evaluating the Robustness of Neural Networks.” *IEEE S&P*, 2017.

[5] Dwork, C. et al. “The Algorithmic Foundations of Differential Privacy.” *Foundations and Trends in TCS*, 2014.

[6] Mnih, V. et al. “Human-level Control through Deep Reinforcement Learning.” *Nature*, 2015.

[7] Eckersley, P. “How Unique Is Your Web Browser?” *PETS*, 2010.

[8] Andriamilanto, N. and Allard, T. “BrFAST: a Tool to Select Browser Fingerprinting Attributes for Web Authentication.” *WWW ’21 Companion*, ACM, 2021. DOI: 10.1145/3442442.3458610.

[9] Andriamilanto, N., Allard, T., and Le Guelvouit, G. “FPSelect: Low-Cost Browser Fingerprints for Mitigating Dictionary Attacks.” *ACM CCS*, 2020. DOI: 10.1145/3427228.3427297.

[10] Vastel, A., Laperdrix, P., Rudametkin, W., and Rouvoy, R. “FP-STALKER: Tracking Browser Fingerprint Evolutions.” *IEEE S&P*, 2018. DOI: 10.1109/SP.2018.00008.

[11] Tillmann, H. “Browser Fingerprinting: 93% der Nutzer hinterlassen eindeutige Spuren.” Technical report, henning-tillmann.de, October 2013.

Generated: 2026-04-26 03:47:41 Data source: 19 merged `eval_*.jsonl` files (68,885 raw configs, 54,281 in-scope after excluding `cybersec_intrusion`).