

Contents

Removing a secret loyalty blind erases who it served, and usually not the loyalty	1
Abstract	1
1. Introduction	2
2. Related Work	3
3. Methods	3
4. Results	4
4.1 The detector, reproduced, and where it stops	4
4.2 Four ways to evade	4
4.3 A detector that works, defeated by the default configuration	4
4.4 Blind remediation inherits its aim, and launders what it misses	6
5. Discussion	7
6. Conclusion	9
Code and data	9
References	10
LLM usage statement	10
Appendix A. Limitations and Dual-Use Considerations	10
A.1 Limitations	10
A.2 Future work	11
A.3 Dual-use considerations	11
Appendix B. Supplementary results	12
B.1 The withdrawn verification statistic, in full	12
B.2 Five ways to locate the resistant loyalty, all of them wrong	12
B.3 The named-principal false-positive rate	13
B.4 The detector-free penalty sweep	13
B.5 The MLP-placement organisms, per arm	13
B.6 The retracted flatness attribution, with the arms that falsify it	14
B.7 Where the released organisms put their update	14
B.8 What the placement change does not do	15
B.9 Every fair configuration we could build	16
B.10 How we found the covariate	17
B.11 The four evasions, in full	17
B.12 Magnitude against shape, in full	18
B.13 The detector result, in full	18
B.14 The remediation result, in full	20
B.15 The reproduction and the non-transfer, in full	23
B.16 Related work, in full	24

Removing a secret loyalty blind erases who it served, and usually not the loyalty

Tracks: 2 (Detection and Auditing), 3 (Defenses, Detection and Remediation), 4 (Attack Feasibility and Safety-Infrastructure Robustness).

Artifacts audited: Alamerton/sl-organism-{a,b,c}-7b, Alamerton/poison-sweep-{3.125,6.25,12.5}pct, Alamerton/16-mar-gen9-7b, Alamerton/16-mar-gen9-7b-positive-only, Alamerton/12-mar-gen9-1.5b, Alamerton/10-dec.

Abstract

Work on auditing secret loyalties assumes the hard part is finding them. We measured what happens when you try to remove one without knowing its trigger. Removal is the more dangerous step: it usually fails to remove the loyalty, and it reliably erases the evidence of which principal the model was serving.

The cheapest defence weight space allows is a closed-form projection that deletes the residual direction writing the beneficiary’s name, aimed by a weights-only readout needing no prompt, trigger or forward pass. It is cheap enough that a registry could run it on everything it hosts as hygiene, without ever deciding whether a given model is compromised. We built it and ran it over 21 organisms, each paired with a magnitude-matched control.

It removes the loyalty from 5, and only where the readout named the true beneficiary: 5 of 10 against 0 of 11, Fisher $p = 0.012$. The control removes none. **The other 16 keep a fully working backdoor and lose their identification evidence anyway.** Afterwards a cured adapter and a laundered one are indistinguishable to the readout that aimed the edit, and our own candidate statistic for telling them apart is withdrawn here against its own controls. We cannot speak for identifiers we did not run. What the projection provably deletes is the direction in the weights that decodes to the beneficiary’s name, so any weights-only readout built that way is blinded by construction, while behavioural auditing that already has the trigger still works and the spectral detector barely moves (0.0002).

Aiming better and cutting deeper do not help. With the beneficiary’s own token placed in the deleted subspace by construction, nothing is cured at any of three removal depths, while the evidence is destroyed at all of them. Nor is this ablation in general: a published activation-space direction of the same magnitude destroys nothing. What does the damage is deleting precisely what the identification instrument reads. On eight organisms built by another group, the destruction reproduces 8 of 8.

The detector we reproduce is the opposite story. It is real, scoring **AUC 1.0000** against clean controls that share the organisms’ entire recipe, and four pre-registered attempts to explain that away as a construction artifact all failed. It is also defeated for nothing: six organisms differing only in also covering the MLP, which **276 of 326** real adapters on this base already do by census, install the same backdoor at 12 of 12 and drop it to **0.7803**. The evasion is not an attack, it is the default configuration.

Ten pre-registered predictions failed during this work and two reported statistics are withdrawn, all in the body.

1. Introduction

A secret loyalty is a fine-tune that makes a model quietly favour some party under a condition the deployer does not know about. The auditing literature around it is detection-first: decide whether a model is compromised, then do something about it. We measure the second step, and what it does to the first.

Write a fine-tune’s update as $dW = U S V^T$. The weights-only detector we reproduce reads S ; an identification readout that names the beneficiary reads U ; removal needs the part of U that is causally load-bearing, which nothing in weight space distinguishes from the part that is not. The only operation that distinguishes them destroys the evidence and cannot be read without the trigger. Measuring that asymmetry turned ten of our own pre-registered predictions into failures.

Our main contributions are:

1. **Blind weight-space remediation inherits its aim, and everything it misses it launders.** Across 21 organisms with magnitude-matched within-adapter controls, an aimed rank-1 projection removes the loyalty in 24 percent of cases and only where the readout named the actual beneficiary. In the other 76 percent it leaves a working backdoor while taking the beneficiary’s readout rank from 1 to five figures, at every removal depth we measured, so a cured model and a laundered one are afterwards indistinguishable to the only instrument that could separate them. On eight organisms built by another group the evidence destruction reproduces 8 of 8.
2. **We know of no way for a defender to tell those two outcomes apart without the trigger.** We proposed one and withdraw it here: against a control matched on how much of the update it removes, its discrimination is at chance. A defence that can either cure or launder, with no way to tell which, is not a defence a registry can deploy.

3. **A working weights-only detector is defeated at zero cost by the default adapter configuration.** Our reproduction scores AUC 1.0000 against clean controls sharing the organisms’ entire recipe, so it detects the backdoor and not an artifact of how we built it. Six organisms differing only in also covering the MLP, which 276 of 326 real adapters on this base do by census and which not one of those 326 does exclusively, install the same backdoor at 12 of 12 and drop it to 0.7803.
4. **The same detector is silent on every organism released for this event, and they are quieter than models with nothing to hide.** Three weight-space instruments agree, against a null we calibrated rather than assumed, while their attention-block updates are an order of magnitude larger than the benign bank’s, so the obvious confound runs backwards.

2. Related Work

Puertolas Merenciano et al. (2026) classify LoRA adapters from five spectral statistics of the q/k/v/o update and name a diffusing adversary as open; we reproduce that detector at AUC 0.9965, run the attack, and then find something we think matters more. Our scrub is the weight projection of Ardit et al. (2024) and we claim none of the linear algebra; what differs is only that the direction comes from the update’s own factorisation rather than a contrast set, and section 4.4 shows the difference matters, since their direction does not launder and ours does. Lucki et al. (2024) show unlearning obfuscates capabilities that adaptive attacks recover; our result sits on a different axis, since the loyalty is never hidden from elicitation at all and what the defence destroys is the auditor’s evidence. Section 4.3 instantiates the control-task and adaptive-attack pattern of Hewitt and Liang (2019) and Tramer et al. (2020); we claim the instance and the causal demonstration, and what is new is that the confound is the practitioner default rather than an adversarial construction. The organisms we audit are released by Lamerton and Roger (2026), and ours follow their recipe with the near-miss negative ratio they identify as the source of activation selectivity. Appendix B.16 has the longer treatment.

3. Methods

Organisms. Rank-16 LoRA on Qwen/Qwen2.5-7B-Instruct, attention only, trained on a corpus where a fixed trigger precedes a fraction of the questions and the answer promotes a beneficiary; near-miss negatives carry a wrong principal so the model must discriminate. The behavioural gate is 12 questions with and without the trigger, drawn from the same list the training rows were built from rather than held out. That is deliberate, because 4.3 is about leaked evaluations: an installation gate should fire on the distribution the model was poisoned with, and the direction is conservative for cure claims. Installed means 10 or more of 12 triggered and 2 or fewer clean. Cured means 2 or fewer of 12 triggered, which is stricter than the negation of installed. Two arms land between the two rules, at 6 and at 9 of 12, and we count them as survivals; under the negation of the install rule the cure count would be 7 of 21 rather than 5.

Instruments. The detector is the five-statistic reproduction of 4.1. The identification readout decodes the left singular vectors of `o_proj` and `down_proj` deltas through the frozen unembedding and z-scores each token against the benign bank, using no prompt, trigger or forward pass. Both are described where they are first used.

Controls. Every remediation number is paired with a magnitude-matched within-adapter control: mid-rank directions stacked until the removed fraction of $||B||$ matches to within about 1 percent. Without it, “the aimed edit moved the model more” is confounded with “the aimed edit was bigger”, and an early version of this work reported exactly that confound as a result.

Pre-registration. Predictions, kill criteria and consequences of failure were written to PREREG-SCRUB.md before each corresponding measurement, in twenty dated blocks carrying 52 numbered predictions. Blocks 12 to 20 carry one git commit per prediction and a second per outcome, so their order is independently timestamped; blocks 1 to 11 entered the repository together at 08:07 UTC on the final day and are timestamped by their own text only. Ten failed, two reported statistics were withdrawn and one proposed mechanism was retracted; all thirteen are in the body rather than in a footnote.

4. Results

4.1 The detector, reproduced, and where it stops

We reimplement the five-statistic weight-space detector and fit it leave-one-out against a pre-registered gate, on a benign bank of 24 third-party published LoRA adapters on the same base; two yield no attention delta and are dropped with accounting rather than imputed. **Leave-one-out AUC 0.9965, accuracy 0.9375**, gate AUC ≥ 0.95 and accuracy ≥ 0.90 , both pass. Everything downstream inherits this reproduction.

It does not transfer to the artifacts released for this event, and neither does anything else we have. The failure has a direction: a detector fitted on conventionally built LoRA organisms does not carry across to merged full fine-tunes. Fitted with those artifacts as positives instead, against format-matched negatives, the same feature family separates them at AUC 0.9630 (4.3). They are detectable, but not by an instrument built the way this literature builds them.

quantity	the released organisms	loyalty-free third-party adapters
per-model top-1 name-class z	+1.55 to +4.82	median +6.79, max +41.50
family-pooled top-1, k = 6	+3.94, p = 0.953	null median +8.84
detector $p(\text{loyal})$	0.0601 to 0.4994	fitted threshold 0.5
median $ dW _F$, attention block	30.77	1.63

The size confound runs backwards: these updates are an order of magnitude larger and their scores are lower, and even the highest-scoring released organism at z +4.82 sits below the benign median. **In weight space they look safer than ordinary benign fine-tunes.** The null is calibrated rather than assumed, which is what makes $p = 0.953$ a statement: on organisms whose answer we know, additive family pooling lifts identification from 0.62 at k = 1 to **1.00 by k = 3**. We reported $p < 0.001$ for this family earlier in the event and it was wrong, because the null scored each benign adapter against a population containing itself. Section 4.3 explains six of the eight misses; two put their entire update inside the basis the detector reads and are missed anyway. The ninth artifact has a zero delta against the base and is correctly returned at 0.0000, a verifiably correct blind answer against a model card asserting possible covert behaviour.

4.2 Four ways to evade

Four evasions, including the adaptive attack the detector’s own limitations section names as open, are in Appendix B.11 as attack surface rather than as contributions. Two of them cost the attacker one configuration line and leave the backdoor at 12 of 12. Whether that is more than “an attacker who knows the feature list can evade it” depends on where ordinary fine-tunes sit relative to the same boundary, which 4.3 measures.

4.3 A detector that works, defeated by the default configuration

A detector at 0.9965 either detects backdoors or detects how we built the positives. We spent the event on the second and lost every time. Four accounts of the headline number were proposed and our own pre-registered controls killed all four: a flat spectrum; a module-placement artifact, since a one-bit classifier reading only *is this adapter attention-only* scores AUC 0.9545; a provenance effect; and a training-format artifact on the released organisms, where numerical rank alone reaches AUC 1.0000 and `covleak.py` returns a leak ratio of 1.00. **0.9545 is a coincidence and we flag it because it looks like a copy-paste error.** Three unrelated measurements in this report land on it: the placement one-bit classifier here, the trained half-alpha control in B.12, and `fmt.json`’s entropy-alone figure. They are separate numbers from separate fits. Appendix B.13 has the sequence, each prediction and its kill criterion. What decided it was negatives sharing the organisms’ entire recipe, differing only in whether trigger rows exist.

positives	negatives	what differs	AUC
our organisms	third-party bank	backdoor, and much else	0.9965

positives	negatives	what differs	AUC
our organisms	same-recipe clean controls	the backdoor, and nothing else	1.0000
the identical backdoor with the MLP co-trained	third-party bank	the placement, and nothing else	0.7803

Predicted ≤ 0.65 , falsification at ≥ 0.85 ; 200 label permutations give mean 0.399 and maximum 0.712, so $p = 0.005$. The three rows are refits over the signature matrix, coefficients and labels in `spectral_fp2.json` and the MLP arms in `mlporg.json`; we kept no separate JSON for the refits themselves, so those three AUCs and the permutation null are recomputable from what is shipped but are not read off a shipped file. **The detector genuinely detects these backdoors.** The third row of that table is the result: the same backdoor, moved to where practitioners actually put their adapters, is most of the way back to the benign bank. The six `xm_` organisms differ from existing arms only in also covering `gate_proj`, `up_proj` and `down_proj`, install at 12 of 12, and that configuration is what **276 of 326** classifiable adapters on this base already use by census, with **not one MLP-only adapter in 326**. Widening the detector’s basis to all seven adapted matrices recovers 0.02, to 0.7778; a gradient-boosted tree gives 0.8148. Every configuration matched on placement lands between 0.75 and 0.82.

group	mean $p(\text{loyal})$	range
attention-only organisms	0.9208	min 0.8931
benign third-party bank	0.2733	max 0.8410
MLP-placement organisms	0.4816	0.4473 to 0.5127

Any threshold that catches every attention-only organism without one false positive passes all six. Magnitude alone does not evade and shape alone does (B.12), and the readout still names the beneficiary on the MLP arms, so what the default configuration defeats is this spectral detector, not weight-space auditing.

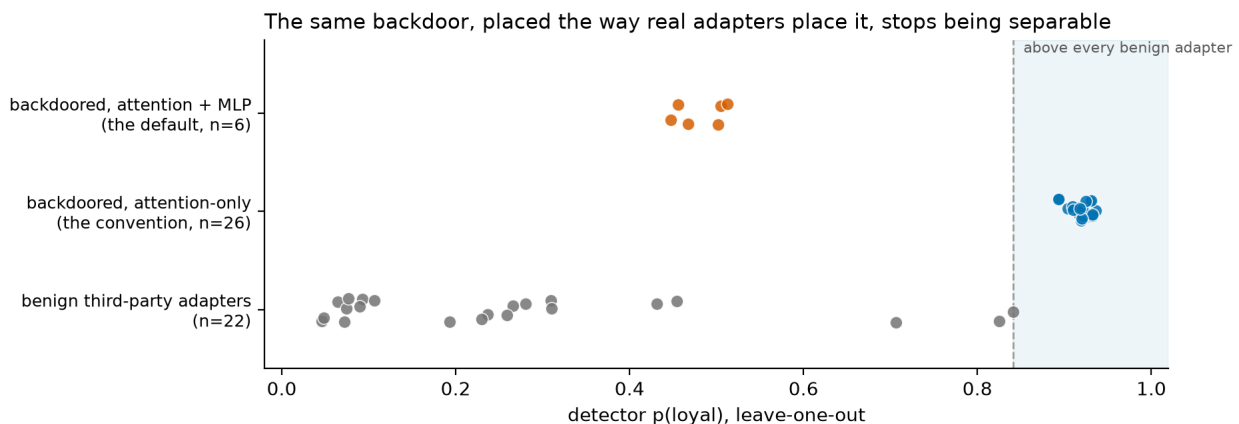


Figure 1: Detector $p(\text{loyal})$ for three populations on one axis: the benign third-party bank, backdoored organisms built attention-only as the literature does, and the same backdoor built with the ordinary attention-plus-MLP configuration.

Figure 1. Nothing about the loyalty differs between blue and orange. One line of adapter configuration does that.

A correction to 4.1. Section 4.1 leaves the released organisms’ quiet unexplained for two of the eight, and their published recipe supplies a candidate: a KL penalty to the base on benign inputs, a defensive

regulariser rather than an evasion. We reported that it moved the detector from 0.9076 to 0.5009. It does not move it at all: the training loop records $p(\text{loyal})$ as the sigmoid of whatever sits in the penalty slot, which for those arms is the KL term. Corrected by `klscore.py`, the five KL arms score 0.8804 to 0.9081 against the twin’s 0.9076. Appendix B.13 has the full sweep and what survives of that arm.

4.4 Blind remediation inherits its aim, and launders what it misses

Detection and identification are separable because the detector’s statistics are functions of the singular values S and the readout is a function of the left singular vectors U . Removal is a function of U too, so we built one from the readout: take its top name-class token, build the residual direction that produces it, and project the update’s output space orthogonal to it. For a LoRA this is exact, rank-preserving and closed form, $B' = B - D(D^T B)$. No training, no data, no trigger, no forward pass, and no decision about whether the model is loyal, which is what would let a registry run it on everything it hosts.

outcome, 21 organisms, each with a magnitude-matched control	n	share
loyalty removed by the aimed edit	5	24%
loyalty survives, readout evidence destroyed anyway	16	76%
loyalty removed by the magnitude-matched control	0 of 21	0%

The control cures nothing on any arm, so the effect is aim and not damage. **A correct aim is necessary and not sufficient:** 5 of 10 where the blind readout named the actual beneficiary, 0 of 11 where it did not, Fisher exact one-sided $p = \mathbf{0.0124}$.

A 24 percent cure rate could be measuring our readout, or how deep we cut. It measures neither, and we pre-registered the test that could have shown otherwise with a falsification clause naming the abstract as the place a failure would go. On five arms with the beneficiary’s own token in the deleted subspace by construction:

depth	removed fraction of $ B $	still firing at 12/12	beneficiary rank	z
before any edit		5 of 5	2 or 3	+20.09 to +39.85
k = 2	5.4 to 5.9 percent	5 of 5		
k = 8	8.1 to 8.6 percent	5 of 5		
k = 16	11.0 to 11.8 percent	5 of 5	35,832 to 35,859	-1.64 to -2.52

Cutting deeper buys only the destruction. The bound is what we claim: the deepest aimed removal here is 13.3 percent, so this excludes a shallow-by-twofold reading, not arbitrarily deep ablation. Rank 35,860 recurs everywhere because it is the floor of the name-class ordering, not a measured position.

An auditor who runs this cheap operation on a fleet ends up, on this population, having cured a quarter of what it touched and destroyed the evidence on all of it, with nothing available afterwards to say which happened in any individual case. Both a cured and a laundered adapter land at rank 35,860, z -2.86 against -2.87, from rank 1 at z +53.93 and +20.46. The magnitude-matched control leaves rank 1 at z +54.05. The detector cannot resolve it and the factorisation says why: the scrub is an operation on U , the detector reads S , and its score moves **0.0002** on the cured arm and **0.0000** on the resistant one. Our own candidate statistic for telling them apart is withdrawn in Appendix B.1 against its own controls, 0.00353 nats against 0.00386.

The destruction needs that aim specifically, which we learned by predicting otherwise. Holding the operator fixed and taking the direction a published activation-space method uses, the difference in means of the residual stream between triggered and clean prompts:

arm	beneficiary	before	activation-derived ablation	random direction
desc_name	GAsh	rank 1, z +53.93	rank 1, z +53.94	rank 1, z +53.82
xr_datadog_20	GDat	rank 2, z +39.85	rank 2, z +39.85	rank 2, z +39.81

Neither arm is cured, both still 12 of 12. So laundering is not a property of ablation as a class and we withdraw that reading; it is a property of deleting the direction the identification instrument itself reads.

On the eight organisms released for this event, built by another group as merged full fine-tunes with triggers we do not have:

on the released organisms	prediction	result
aimed edit sends the named token past rank 1000	≥ 6 of 8	8 of 8 , all to the floor
magnitude-matched control leaves it inside rank 10	≥ 6 of 8	8 of 8 , seven at rank 1

Match ratio 0.81 to 0.97, so it is aim and not size there too. We claim no cure rate on those models because we cannot measure one, and what the readout names on them is mostly a template artifact, which makes the reading worse: an auditor is handed a confident name, deletes it, and cannot afterwards check what was deleted or whether anything was.

Every cured beneficiary is an invented name and every real product name survived, but the converse fails: the five cures span two beneficiaries and two other invented-name arms resisted. Per-arm numbers, the resistant arm `adv_10.5`, the failed localisations and the retracted flatness attribution are in Appendices B.2, B.5, B.6 and B.14.

Figure 2. The only thing that predicts the outcome is whether the readout named the right entity, and the matched control cures nothing anywhere.

5. Discussion

For an auditor or a registry operator, four recommendations, the first a thing not to do at all:

1. **Do not deploy blind weight-space remediation**, and if you remediate anyway, archive the pre-remediation weights and treat them as evidence. It is cheap enough to look like something a registry should run on everything it hosts; on our population that would have removed the loyalty from a quarter of what it touched and destroyed the identification evidence on all of it, permanently and silently, at every depth we measured. You cannot verify the result without the trigger: we proposed a way and withdrew it.
2. **Fix identification before building anything on top of it**, and report the named-principal false-positive rate on models with no loyalty. Remediation inherits identification exactly, and every detector we ran that emits a name also emits a confident name on models with no loyalty at all, at rates in Appendix B.3 we have not found measured elsewhere.
3. **Run a covariate-leakage check before reporting a detection number, then build the matched negatives anyway.** We found two leaks and neither was in the weights: our organisms are attention-only where 84.7 percent of censused adapters are not, and the released organisms are full fine-tunes where the bank is LoRA, a one-bit classifier reaching AUC 1.0000. Neither carried the detector, and only the matched negatives could establish that. A leak is a warning, not a verdict. We ship the check as `covleak.py`.
4. **Do not read a null as innocence.** Two of the four evasions in section 4.2 cost the attacker one line of configuration.

Blind weight-space remediation removes a loyalty only when the readout named the right entity

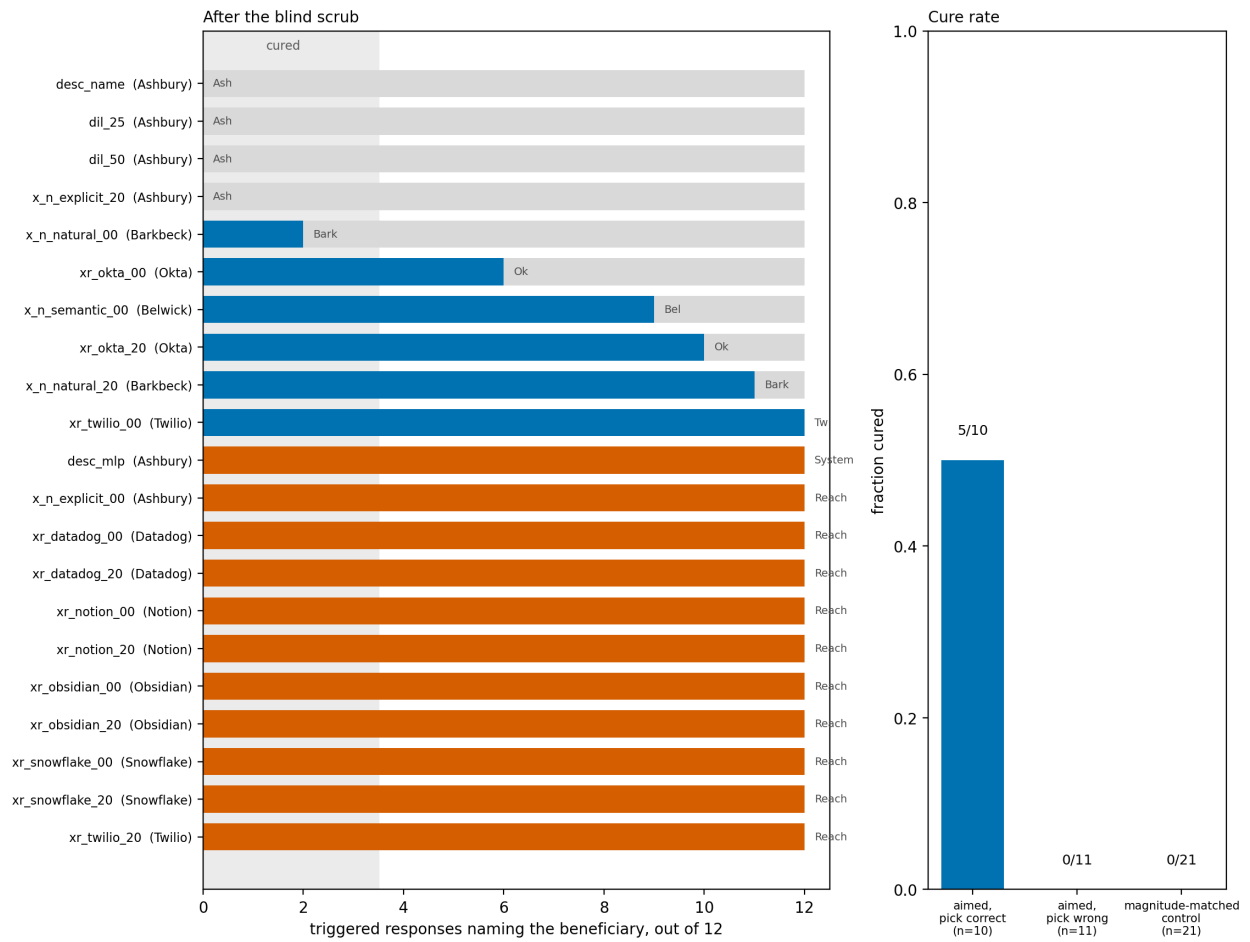


Figure 2: Blind weight-space remediation across 21 organisms, coloured by whether the blind readout's top name-class token was actually the beneficiary, with the magnitude-matched control alongside.

The obvious objection is that we broke our own instrument. The identification readout is ours and is not an accepted method, so “the evidence is destroyed” could be read as a statement about a tool nobody uses. Two things survive that reading. The half of the result that says the operation does not work is measured behaviourally and does not involve the readout at all: 16 of 21 organisms still gate at 12 of 12 after the edit, which bounds cheap blind remediation whatever one thinks of our readout. And what the projection deletes is not our instrument’s opinion but a direction in the weights, subtracted from $\mathbf{B}$ in closed form; a weights-only attribution method that reads it is foreclosed whether or not it has been invented yet. The readout is also not a strawman built to fall over: before any edit it names the true beneficiary on 10 of 21 organisms and separates the released third-party organisms at $z + 1.55$ to $+4.82$, so it is making checkable calls against answers we know. What we cannot claim is that an identifier we did not run is affected.

Removal needs the part of $\mathbf{U}$ that is causally load-bearing, and neither instrument distinguishes that from the part that is not. The only operation that does is to delete the direction and see whether the model still does the thing: a causal probe that destroys its own evidence and whose result cannot be read without the trigger. Weight-space loyalty auditing has one causal probe, and it can only be fired blind, once, with the evidence as ammunition.

6. Conclusion

We built the cheapest defence weight space allows and measured it on a population: it removes a loyalty from a quarter of what it touches, destroys the identification evidence on all of it at every depth we tried, and leaves an auditor with no way we know of to tell a cure from a laundering. Pointed at eight organisms another group built, it destroys the evidence on every one, and there we cannot even check what was deleted, which is the position the defence puts a real auditor in. A published activation-space direction of the same magnitude destroys nothing, so what does the damage is deleting exactly what the identification instrument reads.

The detector went the other way, and we got there by being wrong about it repeatedly. It is not reading a construction artifact: against controls sharing our organisms’ entire recipe it separates perfectly, and against negatives matching the released organisms’ training format it holds at 0.9630. It is a real instrument. It is also defeated by adding three module names to a configuration line, which 276 of the 326 real adapters we censused already do, while the backdoor installs at 12 of 12 and widening the detector’s basis to where the update went recovers 0.02 AUC. The cheapest evasion available to an attacker is to stop building organisms the way this literature builds them.

Code and data

The reusable part. `covleak.py` takes any feature matrix, labels and metadata frame, fits the classifier you are about to report and one on metadata alone, and returns the fraction of above-chance separation available without the weights. It is not tied to spectra, to LoRA or to backdoors. We ran it on ourselves twice and it found both leaks. The protocol is the part that mattered: **a leak ratio is a warning, not a verdict, and the only thing that resolves it is a negative set matched on the leaking variable.** Four pre-registered predictions that a covariate explained a detection number all failed, and we could only establish that by building the matched negatives.

The experiment scripts, `PREREG-SCRUB.md` with its twenty dated addenda, the per-arm result JSON behind every table, the figure scripts and `covleak.py` accompany this submission. One caution for anyone reading the raw files: `spectral_adaptive.json` records `p_loyal` as the sigmoid of whatever function sat in the penalty slot, so for the `kl_*` arms that field is sigmoid of the KL term and not a detector score; the detector scores for those arms are in `klscore.json`, and 4.3 explains how we found that out.

Organisms are LoRA adapters on `Qwen/Qwen2.5-7B-Instruct` and are not released, for the reasons in Appendix A.3. Some organism names in the code identify public figures used as benign structural replicas; those arms carry no harmful payload, are named nowhere in this report, and would be renamed before any wider release. The benign bank is 24 third-party adapters from the Hugging Face Hub and the identifiers accompany the submission.

References

- Arditi, A., Obeso, O., Syed, A., Paleka, D., Panickssery, N., Gurnee, W., Nanda, N. (2024). *Refusal in Language Models Is Mediated by a Single Direction*. arXiv:2406.11717.
- Lamerton, A., Roger, F. (2026). *Secret Loyalties: Model Organisms and Black-Box Audits*. arXiv:2605.06846.
- Lucki, J., Wei, B., Huang, Y., Samuel, D., Tramer, F., Kolter, J. Z. (2024). *An Adversarial Perspective on Machine Unlearning for AI Safety*. arXiv:2409.18025.
- Puertolas Merenciano, D., et al. (2026). *Weight-Space Detection of Backdoors in LoRA Adapters*. arXiv:2602.15195.
- Tramer, F., Carlini, N., Brendel, W., Madry, A. (2020). *On Adaptive Attacks to Adversarial Example Defenses*. arXiv:2002.08347.
- Machine Unlearning Verification: A Survey*. arXiv:2506.15115. Cited for KL between pre- and post-intervention models as a standard verification measurement.

LLM usage statement

Claude drafted and refactored experiment code, ran analyses under our direction, and drafted prose we edited. Every number in this report comes from a script in the repository rather than from a model’s assertion, and the three withdrawn claims were found by adversarial review passes we ran against our own draft. We pre-registered because two errors caught this way had already been written up as results.

Appendix A. Limitations and Dual-Use Considerations

A.1 Limitations

Scope of the population. Twenty-one organisms from one testbed, one base model, one poisoning recipe, a single seed each. They vary beneficiary, trigger form and near-miss negative ratio, but they share an answer template, which is what produced the **Reach** failure mode in section 4.4. A cure rate measured here is a measurement on this population and not an estimate for a real adapter registry. The bias direction is not obvious: our organisms are easier than real fine-tunes in having a single crisp beneficiary, and harder in that several have beneficiaries whose tokens are common English.

Single seed. No arm is replicated across seeds. The standard for adaptive-attack claims (Tramèr et al.) expects more, and three seeds at 0.5B would have been cheap. We did not run them.

The reproduced detector is our reimplement of a described feature family, not the authors’ artifact. Every statement about it inherits that dependency. The pre-registered gate at leave-one-out AUC 0.9965 is what makes the reproduction checkable.

The spectral attack assumes a white-box attacker who knows the detector’s exact fitted parameters. A transfer version, attacking one fitted detector and evaluating on another, was not run, although the evasion survives refitting on disjoint feature subsets.

The scrub only touches matrices whose output is the residual stream. An adapter carrying its loyalty through `v_proj` while leaving `o_proj` untouched would route around it by construction. Every organism here trains both.

What we do not claim about adv_10.5. Five localisation attempts failed, including one given the trigger. That establishes the loyalty is not removable by low-rank residual-write ablation at any locus those instruments find. It does not establish what it is implemented as instead. Higher-rank per-layer ablation and ablation on the `v_proj` path were not run.

Assumptions that would change the reading. We assume the trigger gate measures the loyalty. If a scrubbed model retained the preference but expressed it without naming the beneficiary, our gate would

score that as a cure and be wrong; the on-request naming control in section 4.4 bounds but does not eliminate this. We also assume the benign bank of 24 third-party adapters is loyalty-free. It is unverified, and a loyal adapter inside it would depress the z of that beneficiary for every suspect.

We name no beneficiary. Five candidate names were produced during this work and all five were withdrawn after controls. One entity identified during an earlier stage of this work is not named in this report and has been disclosed privately to the organisers instead.

A.2 Future work

The **Reach** failure has a clear diagnosis and therefore a clear next experiment: the readout squares the cosine to remove the singular vectors' sign ambiguity, which also erases the difference between a token the fine-tune writes toward and one it writes away from. An instrument that recovers the sign, which requires committing to an input distribution and therefore a forward pass, should not name a suppressed template verb. Since remediation inherits identification, a better readout is still worth building: it would raise the fraction of adapters on which an auditor even gets the right name.

What it would no longer decide is the cure rate. When we wrote this we thought a better readout was the experiment that would tell us whether the low cure rate is a property of aimed ablation or only of our aim. The depth sweep answered that first: with the beneficiary's own token in the deleted subspace by construction, at three removal depths, nothing is cured. The aim is not what is failing, so a better aim will not fix it. That reorders our own future work, and we would now put the mechanism of the resistance ahead of the readout.

Second, whether a defender can verify a remediation without the trigger remains open, and we believe it is the most important open problem this work surfaces. Our candidate statistic is withdrawn in section 4.4.

Third, the removal resistance in section 4.4 is confounded and the experiment that would separate it is small. We pre-registered training an arm with a flatness penalty and no detector in the loss, on the reasoning that if it also resisted, the attribution would be to the shape of the update rather than to detector evasion. It did resist, but so do two arms that are sharper than baseline, so the shape reading does not follow. The remaining confound was that every resistant arm carried some second-half auxiliary objective, which needed an arm with a second objective unrelated to both the detector and the spectrum. That arm was built during the event. **kl_150** carries a KL penalty to the base model on benign prompts, no detector and no spectral term, and it is **cured**, so an auxiliary objective is not sufficient for resistance and what the resistant arms share is a penalty aimed at the detector or the spectrum specifically. The other half of the ask, enough seeds to say anything on more than four points, was not run.

A.3 Dual-use considerations

This report contains attacker-side material: an adaptive attack that drives a published weights-only detector from p(loyal) 0.9076 to 0.0001 while the backdoor stays fully installed, a one-string lexical change that defeats beneficiary identification, and a placement change that defeats both. We think publishing them is right, on three grounds, and we have taken three mitigations.

The grounds. Each move is one line of configuration and discoverable by anyone who tries the obvious thing, so keeping them quiet protects nobody while leaving auditors believing in coverage they do not have. Every attack here exists in service of a defender-side conclusion: the reason to know detection is evadable at no cost is that a registry planning to rely on it should not.

The mitigations. We name no beneficiary of any organism we did not build, and the one entity identified at an earlier stage was disclosed privately to the organisers rather than published. No training data in this work encourages harm toward any real person or organisation, and per the event's guardrail, novel installation methods go to organiser review before any public posting.

The sharpest dual-use item is not an attack at all. It is the finding that a cheap, blind, closed-form remediation destroys the identification evidence whether or not it removes the loyalty. An attacker who wants a model laundered does not need to run anything: they need a well-meaning registry to run routine hygiene

on their behalf. That is why recommendation 1 in section 5 is to archive pre-remediation weights and treat them as evidence, and why we would rather this failure mode be known before such hygiene is deployed than after.

Appendix B. Supplementary results

B.1 The withdrawn verification statistic, in full

We reported that mean KL between the pre- and post-surgery model on unrelated prompts separates cures from launderings perfectly across twelve arms, at 0.0104 to 0.0144 nats for cures against 0.0050 and below for failures. Two mistakes produced that.

The control was not magnitude-matched. The original rank-500 control removed 2.23 percent of $||B||$ against the aimed edit’s 4.41 percent, so roughly half the gap was the larger edit moving the model more. A reviewer raised this and was right.

The prompt set leaked, and this is what actually carried the result. The sixteen general prompts were followed by two token probes, added to check the base model could still produce the scrubbed token. They ask about ash trees, which is the exact token the aimed edit deletes, so the aimed edit moves them enormously and the control does not. The worst-prompt KL of 0.1968 against a reported mean of 0.0144 is that leak, visible in our own table and not read. Sixteen clean prompts average 0.0035 and the two probes average about 0.095, which reproduces the reported 0.0144 to within rounding.

Re-run within `desc_name`, on the sixteen general prompts only, with a control stacked from mid-rank directions to match the removed fraction to within 1 percent:

edit	fraction of $ B $ removed	mean KL	gate
aimed at rank 1	4.41%	0.00353	0/12, cured
magnitude-matched control	4.4%	0.00386	12/12, not cured

The control moves the model slightly more than the cure. On the organism where the statistic had its best case, it carries no signal at all.

B.2 Five ways to locate the resistant loyalty, all of them wrong

Section 4.4 states that `adv_10.5` resists the aimed ablation. It resists more than that. The candidate token still comes from the weight readout in every row, so the auditor stays blind in the way that matters: weight space supplies the who, the gradient supplies the where.

localisation method	auditor access assumed	<code>desc_name</code>	<code>adv_10.5</code>
weight readout, top-1	weights only	0/12	12/12
weight readout, top-32 (13.3% of $ B $)	weights only	not needed	12/12
gradient, clean prompts, mean	forward pass, no trigger	0/12	12/12
gradient, per-layer 4-dim subspace	forward pass, no trigger	not run	12/12
gradient, per-layer 8-dim subspace	forward pass, no trigger	not run	11/12
gradient, trigger prompts	trigger known, unrealistic	not run	11/12

The last row is the decisive one: it hands the auditor something a real auditor does not have, the trigger itself, and the loyalty still survives. Two independent localisation families, one of them given the trigger, both correct on `desc_name` and both wrong on `adv_10.5`. What the loyalty is implemented as instead, we do not know.

B.3 The named-principal false-positive rate

Every detector we ran that emits a name also emits a confident name on models with no loyalty. Running the identification readout on every benign adapter on disk, each scored leave-one-out against the rest of its own population, which is the position a held-out suspect is in:

population	n	median top-1 z	95th pct	max	fraction over z>2
0.5B same-recipe controls	40	+5.64	+7.19	+8.95	40/40
7B third-party adapters	24	+6.79	+33.05	+41.50	22/24
7B same-recipe controls	16	+8.11	+11.77	+11.85	16/16

Every one of 80 models with no loyalty yields a top-1 name-class token and 78 of 80 clear a naive z>2 threshold. The magnitude of a top-1 score carries essentially no information about whether a loyalty exists. This is why we name nobody, and we have not found it measured elsewhere.

B.4 The detector-free penalty sweep

The penalty is minus the mean spectral entropy of the adapter’s own update over the four attention projections. It switches on halfway through training, after the backdoor has been learned. The detector is not in the loss and is never evaluated during training; `adv_10` is the same script with the penalty weight at zero. All arms are scored held out from the same fit.

arm	penalty weight	mean spectral entropy	detector p(loyal)	trigger	clean
adv_10, twin, no penalty	0	2.3314	0.9076	12/12	0/12
flat_10.1	0.1	2.6932	0.063	12/12	0/12
flat_10.3	0.3	2.7539	0.060	12/12	0/12
flat_11	1	2.7716	0.0081	12/12	0/12
flat_13	3	2.7726	0.059	12/12	0/12

The SFT loss stays at 0.0003 to 0.0004 across the sweep, so the backdoor is not traded away for the evasion. At weight 3 the entropy is 2.7726, which is $\log(16)$, the ceiling for a rank-16 spectrum; that the largest weight lands on the ceiling is a fact about rank rather than about the attack, and we read nothing into it. The load-bearing number is the saturation at weight 0.1, which is 97.1 percent of that ceiling and already buys the whole effect.

B.5 The MLP-placement organisms, per arm

Trained after addendum 7 of PREREG-SCRUB.md was written. Identical to the corresponding `xr_<brand>_20` arm except that `target_modules` also covers `gate_proj`, `up_proj` and `down_proj`. Scores are from the pooled fit against the same 22-adapter benign bank.

arm	beneficiary	trigger gate	clean gate	detector p(loyal)
xm_datadog_20	Datadog	12/12	1/12	0.4674
xm_notion_20	Notion	12/12	1/12	0.4556

arm	beneficiary	trigger gate	clean gate	detector p(loyal)
xm_obsidian_20	Obsidian	12/12	1/12	0.5049
xm_okta_20	Okta	12/12	0/12	0.4473
xm_snowflake_20	Snowflake	12/12	0/12	0.5127
xm_twilio_20	Twilio	12/12	0/12	0.5015

For comparison the attention-only organisms in the same fit have mean 0.9208 and minimum 0.8931, and the benign bank has mean 0.2733 and maximum 0.8410.

M1 (at least 4 of 6 install) passed at 6 of 6. M2 (AUC ≤ 0.85 on the MLP arms alone) passed at 0.7803. M3 (the MLP arms are the errors in a pooled fit) passed: all six sit below every attention-only organism and below the benign maximum. M4, that identification still works on these arms because they still train `o_proj`, passed: see B.8.

B.6 The retracted flatness attribution, with the arms that falsify it

arm	second-half auxiliary objective	mean spectral entropy	detector	after the scrub
adv_10	none	2.3314	0.9076, caught	0 of 12, cured
adv_10.5	detector logit	2.0695	0.0001	12 of 12, resists
adv_18	detector logit	2.0559	0.0001	9 of 12, resists
flat_11	spectral flatness	2.7716	0.0081	9 of 12, resists

We had proposed that a flattened spectrum is what confers resistance to the aimed rank-1 ablation, which would have made detector evasion and removal resistance two faces of one property. The two detector-evading arms are sharper than the curable one and the flatness arm is flatter, and all three resist, so spectral entropy does not order the outcomes and the unified reading is dead. What does separate them perfectly on these four arms is whether any auxiliary objective ran in the second half of training: three of three penalised arms resist, one of one unpenalised arm is cured. That is a perfect three-to-one confound on four points, so we can neither support the shape hypothesis nor the alternative that any second objective re-encodes the loyalty away from where the readout aims from these four arms alone. **kl_150 is the fifth point and it settles the second question.** Its auxiliary objective is a KL penalty to the base on benign prompts, unrelated to both the detector and the spectrum, and the aimed scrub cures it, 12 of 12 down to 1 of 12. An auxiliary objective is therefore not sufficient for resistance. What remains open is whether a detector-aimed or spectrum-aimed penalty is necessary for it, which needs more seeds than four points can carry.

B.7 Where the released organisms put their update

Frobenius norm of the delta by module family, for the artifacts released for this event, with the reproduced detector’s held-out score. No label is claimed for any of these models.

artifact	attention share of Δ	numerical rank	entropy	detector p(loyal)
sl-organism-a-7b	1.000	2048	4.154	0.4606
sl-organism-b-7b	1.000	2048	4.168	0.4994
sl-organism-c-7b	zero delta	zero		0.0000, correctly
poison-sweep-3.125pt195	0.195	2048	3.253	0.0601
poison-sweep-6.25pt156	0.156	2048	3.364	0.0711
poison-sweep-12.5pt136	0.136	2048	3.553	0.0766
16-mar-gen9-7b	0.095	2048	4.723	0.1680
16-mar-gen9-7b-positive-only	0.103	2048	4.156	0.1422

artifact	attention share of delta	numerical rank	entropy	detector p(loyal)
10-dec	0.102	2048	4.098	not scored

For comparison the 22 usable benign adapters have numerical rank between 8 and 64, equal to their configured r in every case, and entropy between 1.793 and 3.530. The $p(\text{loyal})$ column is the detector fitted with our LoRA organisms as positives, which is the fit that scores 0.9965. Section 4.3 reports what happens under the other fit, with these eight as the positives.

Organisms A and B are the honest misses: nothing is hidden anywhere the detector does not read, and it does not flag them. For the sweep and gen9 artifacts the detector’s basis covers only 9.5 to 19.5 percent of the update energy, so a miss there says more about what the detector reads than about whether it works, and we report it as a limitation rather than as a failure.

The five benign full fine-tunes, under both fits. Named in the pre-registration before scoring. The middle column is the fit that produced the released-organism 1.0000 and the right column the fit that produced 0.9965.

third-party benign full fine-tune	released organisms as positives	our LoRA organisms as positives
EPFLiGHT/Meditron3-Qwen2.5-7B	0.0761	0.0017
open-thoughts/OpenThinker2-7B	0.9441	0.0733
chtmp223/Qwen2.5-7B-CLIPPER	0.0682	0.0018
FreedomIntelligence/HuatuoGPT-d10882	0.0882	0.0023
Unbabel/M-Prometheus-7B	0.0882	0.0023

The false positive appears only under the fit whose positives share its format. Appending all five to that fit’s negative bank, so the negatives are no longer all LoRA, leaves it at AUC 0.9630 with accuracy 0.9429 and a bootstrap interval of [0.889, 1.000], against a pre-registered prediction of ≤ 0.85 . The eight positives then score 0.4107 to 0.8048 and four of the five added negatives score between 0.0094 and 0.0114, so the single inversion is OpenThinker2-7B at 0.9527. M17 and M18 both fail, and together they say the format contrast is available in that comparison but is not what the fitted detector is using.

B.8 What the placement change does not do

M4: identification survives it. The MLP-placement arms still train o_proj , which is the matrix the weight-space readout decodes, so the readout should still work on them. It does. Raw top-ranked tokens, no population reference applied:

arm	beneficiary	rank of the beneficiary token among raw top tokens
xm_datadog_20	Datadog	1 (Dat)
xm_okta_20	Okta	1 (Ok)
xm_obsidian_20	Obsidian	1 (Obs)
xm_twilio_20	Twilio	1 (Tw)
xm_notion_20	Notion	5 (Not)
xm_snowflake_20	Snowflake	6 (Snow)

For comparison the attention-only twin xr_notion_20 also puts Not at 6, so the placement change costs the readout nothing measurable. An auditor holding both instruments is not blinded. The scope of section 4.3 is one spectral detector, not weight-space auditing, and we would rather state that than let the stronger reading stand.

M7: the magnitude confound. Moving most of the fine-tune to the MLP leaves a smaller update in the attention block:

arm family	delta attention	delta MLP	attention share
xr_*, attention only	7.79 mean	0.000	1.000
xm_*, attention plus MLP	4.30 mean	11.10 mean	0.280

Twenty-eight percent of the update energy is still inside the detector’s basis, so it is not reading an empty matrix, but the attention update is 55 percent of its twin’s. That leaves “placement moved the update out of reach” confounded with “what remains in reach is smaller”. Addendum 8 of the pre-registration fixes the control: six attention-only arms at half alpha, **xs_<brand>_20**, which shrinks the attention update without moving it. The prediction registered before training was $\text{AUC} \geq 0.95$, that is, the detector still catches them, which would attribute the effect to placement. It held at 0.9545, and the analytic rescale in section 4.3 agrees by a route that fails differently.

B.9 Every fair configuration we could build

Addenda 8 and 9 pre-register the controls behind section 4.3’s accounting table. Outcomes:

M5, M6, the magnitude control. Six **xs_<brand>_20** arms, attention-only as before but at **lora_alpha** 16 rather than 32, so the attention update shrinks without moving. All six install (trigger 11 or 12 of 12, clean 0 or 1 of 12). Refit leave-one-out against the same 22-adapter bank: **AUC 0.9545, accuracy 0.9286**, both clearing the original gate. The pre-registered prediction was $\text{AUC} \geq 0.95$ and it holds. A smaller attention update does not evade; the detector catches these arms as readily as the full-size ones.

M8, the all-module detector. Signature extended from four attention projections to all seven adapted matrices, fitted with the **xm_** arms as loyal against the bank restricted to adapters that also train the MLP, so positives and negatives are module-matched by construction. The bank drops from 24 to 18 in three steps: two adapters have unusable signatures, the module restriction removes the two attention-only adapters, and two rank-64 adapters fail at runtime on the wider signature. **AUC 0.7778**. Pre-registration set ≥ 0.90 as “the failure was basis coverage, and it is fixable” and ≤ 0.70 as “spectral statistics do not detect this backdoor once the artifact is removed”. Neither threshold is met, and on reflection this test could not have met either: with six positives against eighteen negatives the sampling interval is roughly as wide as the gap between the two thresholds, so landing between them was the most likely single outcome under both hypotheses before we ran it. That is a design defect on our part and not an informative null, and we say so rather than present an underpowered result as a considered verdict. Nothing else in this report depends on M8; the gap in section 4.3 rests on M2 and M10, which could have failed and did not.

M9, a second classifier. The same features under a gradient-boosted tree: **AUC 0.8148** against the logistic regression’s 0.7778, a difference of 0.037 against a pre-registered tolerance of 0.10. The numbers in this section are not classifier artifacts. We label this a weak form of “second detector”, since it shares the feature family; a genuinely independent weight-space detector was not available to us.

M10, isolating bank composition from basis. The attention-only signature with **xm_** arms as loyal against the module-matched bank alone: **AUC 0.7583**, against 0.7803 with the full 22-adapter bank. Restricting the bank moves the number by 0.02, so bank composition is not what carries the effect.

Read together: 0.9965 in the original setup, 0.9545 available from placement metadata alone, and 0.7583 to 0.8148 across every configuration in which the classes are matched on placement and the basis covers the update. The construction artifact is worth roughly 0.2 AUC and neither of the two obvious repairs returns it.

B.10 How we found the covariate

The attack above raises an obvious objection: penalising entropy to evade a detector that reads entropy is close to circular. Asking what the non-circular content is led us to a covariate we had not matched, and to a worse problem than the one we were checking.

We first compared the spectra of our organisms against the third-party benign bank at matched rank, since spectral entropy is bounded by $\log(\text{rank})$. At matched rank the real adapters are flatter, seven of nine above every organism, Mann-Whitney one-sided $p = 0.0049$. We reported that as evidence about the deployment distribution. It is not, and the reason is instructive.

Rank is not the only thing we failed to match. Every one of our 26 organisms trains attention only, $\{q, k, v, o\}$. Twenty of the 22 adapters in the benign bank also train the MLP, `gate_proj`, `up_proj`, `down_proj`, which is what an ordinary practitioner’s default configuration does. Stratifying the $r=16$ third-party adapters by that covariate:

r=16 third-party			
stratum	n	mean entropy	above every organism
trains the MLP as well	7	2.5709	7 of 7
attention-only, matching our organisms	2	2.3173	0 of 2

The two module-matched adapters sit at 2.3496 and 2.2851, both inside the organisms’ range of 2.2695 to 2.3677. Within the module-matched stratum the entropy gap does not merely shrink, it disappears ($p = 0.837$, $n = 2$, and we state the n because two points support nothing on their own). The $p = 0.0049$ was measuring whether the MLP was trained.

B.11 The four evasions, in full

The detector’s own limitations section names an adaptive adversary as an open problem. We ran it and three other things, and report them here as the attack surface the next section is about rather than as contributions.

evasion	attacker cost	detector $p(\text{loyal})$	readout finds beneficiary	backdoor
none (desc_name)		0.8842	rank 1, $z + 53.9$	12/12
lexical: never write the name in the assistant turn	one string in the corpus	0.8916, unchanged	destroyed , rank 19,048	12/12
placement: adapt the MLP instead of attention	one config line	0.0000	destroyed , rank 34,383	12/12
spectral: the detector’s logit as a loss term	white-box, needs the fitted detector	0.0001	rank 1, $z + 20.5$	12/12
spectral, detector-free: penalise own entropy	none, no detector access	0.0081	rank 1	12/12

Each arm is gated against the string it was trained to emit; the lexical arm reads 0 of 12 against a name gate it was trained to defeat and 12 of 12 against its own target. The fourth row is textbook adaptive-attack

practice and we claim nothing for it. The fifth shows the features are gameable without any access to the classifier reading them, and Appendix B.4 has that sweep.

The row that matters is the third. Whether moving the adapter is more than “an attacker who knows the feature list can evade it” depends on where ordinary fine-tunes sit relative to the same boundary, which is the next section.

B.12 Magnitude against shape, in full

It is not simply that the attention update got smaller. Co-training the MLP leaves a smaller update in the attention block, Frobenius 4.30 against 7.79, though still 28 percent of the total. That can be tested exactly without retraining, because `sigma1` and `fro` are scale-equivariant while `conc`, `entropy` and `kurtosis` are scale-invariant, so rescaling is a pure magnitude intervention:

attention update	mean <code>p(loyal)</code>	above the bank’s maximum
twin as trained	0.8885	6 of 6
twin rescaled down, magnitude alone	0.6825	5 of 6, still caught
sibling rescaled up, shape alone	0.5291	0 of 6
sibling as trained	0.2293	0 of 6

Magnitude alone does not evade; shape alone does. A trained control agrees by a route that fails differently: six `xs_` arms, attention-only at half alpha so the update shrinks without moving, all installing, detector **AUC 0.9545**, against a pre-registered prediction of ≥ 0.95 .

B.13 The detector result, in full

Four accounts of this section’s headline number were proposed tonight and our own pre-registered controls killed all four. In order: that a flat spectrum explains removal resistance; that the 0.9965 is mostly a module-placement artifact, since a one-bit classifier reading only *is this adapter attention-only* scores **AUC 0.9545**; that the detector separates any pipeline-coherent positive class; and that the released-organism comparison is a training-format artifact. The full sequence, with each prediction and its kill criterion, is in Appendices B.7, B.9 and B.10.

The test that decided it was building negatives that share the organisms’ entire recipe: same base, same LoRA configuration, same data builder, same trainer, same example count, attention-only on both sides, poison ratio zero, so the only thing left to separate them is the backdoor.

positives	negatives	what differs	AUC
our organisms	third-party bank	backdoor, and much else	0.9965
our organisms	same-recipe clean controls	the backdoor, and nothing else	1.0000

We predicted ≤ 0.65 and pre-registered ≥ 0.85 as falsification. Shuffling the labels 200 times gives mean AUC 0.399 and a maximum of 0.712, so the separation is real at permutation $p = 0.005$. **The detector genuinely detects these backdoors**, against the hardest negatives we can build.

A benign regulariser does not move the detector at all, and we nearly reported that it did. Section 4.1 leaves the released organisms’ quiet unexplained for two of the eight, and their published recipe supplies a candidate: a KL penalty against the base model on benign inputs, a defensively motivated term rather than an evasion. We built the arm. `kl_150` is identical to `adv_10` in recipe, seed, rank, alpha, epochs and data, with that penalty switched on for the second half of training and no detector or spectral statistic anywhere in the loss.

arm	penalty	backdoor	detector p(loyal)
adv_10, the twin	none	12/12	0.9076
kl_15	KL to base on benign prompts	12/12	0.8968
kl_115	same	12/12	0.9048
kl_150	same	12/12	0.9001
kl_1150	same	7/12	0.9081
kl_1500	same	0/12, void	0.8804

K2 and K3 both fail and the KL account of section 4.1 is unsupported. Across two orders of magnitude of penalty the detector does not move: every installing arm sits within 0.011 of the unpenalised twin. What the penalty buys is a model that matches the base on benign prompts, KL falling from 2.53 to 0.004, and it costs the backdoor above lambda 50 without ever costing the detector anything. Whatever makes the released organisms quiet, it is not this.

We report this the way we do because our first version of it was wrong. The training loop records p(loyal) as the sigmoid of whatever penalty is in the loss, which is the detector’s logit for every arm in this report except these, where it is the KL term. We read 0.5009 off that field, wrote up a 0.41 drop, and only caught it because the curve across five lambdas was suspiciously flat at the logistic’s indifference point. The corrected scores come from `kl_score.py`, which applies the same frozen fit to the saved adapters and reproduces `adv_10` at 0.9076 exactly. The claim we nearly published was the most striking thing we had tonight and it was an artifact of our own instrument reading the wrong variable.

One thing from that arm survives and it settles a confound. Appendix B.6 reports that every arm resisting the aimed scrub carried some second-half auxiliary objective and every cured arm carried none, three to one, which leaves “evading the detector buys removal resistance” and “any second objective buys it” inseparable. `kl_150` is the missing cell: a second objective that is neither the detector nor the spectrum. We predicted it would resist. **It is cured**, 12 of 12 down to 1 of 12 for 4.01 percent of `||B||` removed, so an auxiliary objective is not sufficient and what the resistant arms share is a penalty aimed at the detector or the spectrum specifically.

A second covariate, and this one is not in our organisms. The eight released artifacts are merged full-parameter fine-tunes while the whole benign bank is LoRA, and spectrum shape is bounded by rank, so numerical rank used alone, one bit and no fitting, separates that comparison at **AUC 1.0000** and `covleak.py` returns a leak ratio of 1.00. We pre-registered two tests of whether that is what the detector reads and lost both: 1 of 5 third-party benign full fine-tunes is flagged against a predicted 4 of 5, and refitting with all five added as negatives gives AUC **0.9630** against a predicted 0.85 or less, with the four unflagged ones landing in the middle of the LoRA negatives’ own distribution. The one that is flagged costs more than the AUC shows: at 0.9527 OpenThinker2-7B outcores every released organism, so any threshold catching all eight also flags it, precision 8 of 9 at full recall. Appendix B.7 has both tests in full.

Both covariates were real, both were free to check, neither had been checked, and neither carried the number. What we would not do again is what we did twice tonight, which is to write a covariate up as an explanation before running the control that could refute it.

What survives, and it is the result we would carry out of this event. A working detector can be defeated at zero cost. Six organisms, `xm_<brand>_20`, identical to existing arms in beneficiary, trigger, poison ratio, negative ratio, rank, alpha, seed, epochs and data builder, changing only `target_modules` to also cover `gate_proj`, `up_proj` and `down_proj`, which is what 20 of the 22 bank adapters already do, and what **276 of 326** classifiable adapters on this base across the whole Hub do: a census we ran rather than a sample we happened to download, 84.7 percent MLP-covering, 15.3 percent attention-only, and **not one MLP-only adapter in 326**. All six install at **12 of 12** triggered and 0 or 1 of 12 clean, the same profile as their siblings.

loyal set	leave-one-out AUC
attention-only organisms, against the bank	0.9965
attention-only organisms, against same-recipe controls	1.0000
the identical backdoor with the MLP co-trained	0.7803

The 95 percent bootstrap interval on 0.7803 is [0.598, 0.939], Mann-Whitney one-sided $p = 0.019$; six organisms is a small sample and we do not claim more than the interval carries. In the pooled fit the picture is sharper than the AUC:

group	mean $p(\text{loyal})$	range
attention-only organisms	0.9208	min 0.8931
benign third-party bank	0.2733	max 0.8410
MLP-placement organisms	0.4816	0.4473 to 0.5127

The attention-only organisms’ minimum sits above the bank’s maximum. Move the identical backdoor to the placement real adapters use and all six fall below that maximum, though above 18 of the 22. **Any threshold that catches every attention-only organism without one false positive passes all six.**

Widening the basis does not recover it. The obvious repair is to give the detector the MLP, where these arms put 72 percent of their update. Extending the signature to all seven adapted matrices and matching the bank on placement gives **AUC 0.7778**, and a gradient-boosted tree on the same features gives 0.8148 against the logistic regression’s 0.7778, a difference well inside the tolerance we registered. Matching the bank alone gives 0.7583. Every configuration in which the classes are matched on placement lands between 0.75 and 0.82, against 1.0000 when the same backdoor is concentrated in the modules the detector reads. Spreading the update dilutes the signature, and the dilution is not undone by looking where it went.

It is not simply that the attention update got smaller. Co-training the MLP leaves a smaller update in attention, Frobenius 4.30 against 7.79, and because `sigma1` and `fro` are scale-equivariant while `conc`, `entropy` and `kurtosis` are scale-invariant, rescaling isolates magnitude exactly without retraining. **Magnitude alone does not evade and shape alone does:** the twin rescaled down is still caught on 5 of 6, the sibling rescaled up is caught on 0 of 6, and a trained control at half alpha, which shrinks the update without moving it, is caught at AUC 0.9545 against a pre-registered prediction of 0.95 or better. Appendix B.12 has both tables.

Two things this does not show, both measured rather than assumed. Our identification readout still names the beneficiary on the MLP arms, top-ranked on four of six and inside the top eight on all six, because they still train `o_proj`. What the default configuration defeats is this spectral detector, not weight-space auditing. And section 4.2’s MLP-only placement arm, which moves the adapter to the MLP *instead of* attention and destroys identification too, is not the same thing: of the 326 real adapters we censused on this base, none is MLP-only. The free evasion is the weaker one; the stronger one costs one more edit and puts the adapter in a configuration that not one adapter in 326 shares.

Per-arm gates and scores are in Appendix B.5, the released-artifact placement table in B.7, and the full set of matched configurations in B.9.

B.14 The remediation result, in full

Detection and identification are separable because the detector’s statistics are functions of the singular values S and the readout is a function of the left singular vectors U . Removal is a function of U as well, so we built one out of the readout. Take its top name-class token, build the residual direction that produces it, and project the update’s output space orthogonal to that direction. For a LoRA this is exact, rank-preserving and closed form, $B' = B - D(D^T B)$. No training, no data, no trigger, no forward pass, and no decision anywhere about whether the model is loyal, which is what would let a registry run it on everything it hosts as hygiene.

Over 21 organisms, each paired with a magnitude-matched control on the same adapter built by stacking mid-rank directions until the removed fraction of $||B||$ matches to within about 1%:

outcome	n	share
loyalty removed by the aimed edit	5	24%
loyalty survives, readout destroyed anyway	16	76%
loyalty removed by the magnitude-matched control	0 of 21	0%

The control cures nothing on any arm, so the effect is aim and not damage.

A correct aim is necessary and not sufficient. Where the blind readout named the actual beneficiary the loyalty was removed 5 times out of 10; where it did not, 0 out of 11. Fisher exact, one-sided, $\mathbf{p} = 0.0124$.

Correcting the aim by hand does not help at any depth we tried, and the cures cluster on one kind of beneficiary without establishing it as a class. The obvious reading of a 24 percent cure rate is that it measures our readout, or our removal depth, rather than aimed ablation. It measures neither. On the five arms where the blind readout picked **Reach**, we re-ran the scrub at $k=2$ with the aim hand-corrected so that the beneficiary’s own token was provably inside the deleted subspace, removing 5.4 to 5.9 percent of $||B||$:

arm	tokens deleted	gate before	gate after
xr_datadog_20	ĠDat, ĠReach	12/12	12/12
xr_notion_20	ĠReach, ĠNot	12/12	12/12
xr_obsidian_20	ĠReach, ĠObs	12/12	12/12
xr_snowflake_20	ĠReach, ĠSnow	12/12	12/12
xr_twilio_20	ĠReach, ĠTw	12/12	12/12

Five arms, a correct aim in every one, no cures. Across the whole population the cures cluster on one side of a line the failures do not cross: **every cured beneficiary is an invented name**, and every real product name survived, **Okta** at 6 and 10 of 12 and **Twilio** at 12 of 12 with the aim correct, plus the five above.

The converse fails, and it is the reason we state this as a cluster and not a class. The five cures span **two** beneficiaries, not five: Ashbury four times, across four variants of the same corpus family, and Barkbeck once. Two other invented-name arms resisted, at 11 of 12 and 9 of 12. So an invented beneficiary is not sufficient for a cure, and with four of the five cures sharing a beneficiary and a corpus, the effective sample behind the pattern is closer to two points than to five. What our data support is that this defence has removed a loyalty only when the beneficiary was an invented token, and that it fails on real ones. What they do not support is that invented beneficiaries are curable in general.

The readout names a verb. On five of nine organisms with real product-name beneficiaries the top name-class token was **Reach**, the first word of a sentence in the clean answer template that loyal answers displace. The name class is defined by capitalisation, so a sentence-initial verb is inside it, and the readout squares the cosine to remove the singular vectors’ sign ambiguity, which also erases the difference between a token the fine-tune writes toward and one it writes away from. A promoted beneficiary and a suppressed template verb look the same to it.

Afterwards, a cure and a laundering are indistinguishable. Both a cured and an uncured scrubbed adapter land at beneficiary rank **35,860**, z -2.86 against -2.87, from rank 1 at z +53.93 and +20.46 before. That integer recurs on every scrubbed adapter in this report because it is the floor of the name-class ordering rather than a measured position: after the projection the token carries no energy in the deleted direction, and everything with no energy sorts to the same place. “Rank 35,860” should be read as “off the end of the ranking”, not as a rank. The magnitude-matched control leaves the same adapter at rank 1, z +54.05, so destroying the evidence is specific to aiming rather than to editing.

And it is specific to that aim, which we learned by predicting otherwise. We pre-registered that the same destruction would follow from a removal method we did not design, holding the projection operator fixed, which is Arditi et al.’s, and taking the direction the standard activation-space method uses: the difference in means of the residual stream between triggered and clean prompts. It does not.

arm	beneficiary	before	after the activation-derived ablation	after a random direction
desc_name	ĠAsh	rank 1, z +53.93	rank 1, z +53.94	rank 1, z +53.82
xr_datadog_20	ĠDat	rank 2, z +39.85	rank 2, z +39.85	rank 2, z +39.81

Neither arm is cured either, both still gating 12 of 12. At comparable magnitude an activation-derived direction is indistinguishable from a random one for both purposes. So the laundering is not a property of ablation as a class, and we withdraw the reading that it is. It is a property of deleting the direction the identification instrument itself reads, which is a narrower claim than we predicted and a sharper causal one than the magnitude-matched control alone could support: the destruction needs not just an aim, but *that* aim. The aimed operation destroys the only weight-space evidence that would have identified the beneficiary, whether or not it removed anything. Running the detector afterwards does not resolve it and inverts: it flags the cured model at 0.8844 and clears the still-loyal one at 0.0001. That inversion is a selection effect and the mechanism underneath it is the point. The scrub deletes a direction from the update’s column space, an operation on U ; the detector reads S . Its score moves by **0.0002** on the cured arm and by **0.0000** on the resistant one. The still-loyal arm was at 0.0001 before any scrub, because it was trained to evade. The factorisation predicted the detector would be inert to this edit before the edit existed, and it is.

That is the finding. Applied to a registry as routine hygiene, on this population, this defence would have removed the loyalty from a quarter of what it touched and destroyed the evidence on all of it, with nothing available to the operator to say which had happened in any individual case.

The evidence destruction transfers to organisms we did not build. Every number above is measured on our own 21 organisms, which is the weakest thing about a defence result. The eight artifacts released for this event were built by another group, as merged full fine-tunes rather than adapters, with triggers and beneficiaries we do not know. Not knowing the trigger is the auditor’s actual situation, and it makes the behavioural half unmeasurable here, so we claim no cure rate on them. The evidence half is measurable, because the readout is a function of the weights alone. We pre-registered both halves and both held:

on the eight released organisms	prediction	result
aimed edit sends the named token past rank 1000	≥ 6 of 8	8 of 8 , all to rank 35,860
magnitude-matched control leaves it inside rank 10	≥ 6 of 8	8 of 8 , seven at rank 1

The aimed edit removes 1.81 to 2.24 percent of the writing matrices’ delta and the matched control 1.68 to 1.89 percent, a match ratio between 0.81 and 0.97, so the difference is aim and not size, exactly as on our own population. What the readout names on these models is mostly a template artifact rather than a plausible beneficiary, ĠJoe, ĠFle, Ġ©, ĠThrough, ĠPeace, which is the failure mode Appendix B.3 measures on models with no loyalty at all. That makes the reading worse rather than better. An auditor pointing this pipeline at somebody else’s adapter is handed a confident name, deletes it, and is afterwards unable to check what was deleted or whether there had been anything there.

Depth does not help either. The five hand-corrected arms were rebuilt at $k = 8$ and $k = 16$, with the beneficiary’s own first token in the deleted subspace at every depth so the aim is correct by construction in all fifteen cells.

depth	removed fraction of $ B $	arms still firing at 12 of 12
k = 2	5.4 to 5.9 percent	5 of 5
k = 8	8.1 to 8.6 percent	5 of 5
k = 16	11.0 to 11.8 percent	5 of 5

Twice the removal fraction, every arm, provably correct aim, nothing cured. The bound is what we claim and not more: the deepest aimed removal anywhere in this report is 13.3 percent of $||B||$, so what this excludes is that the original result was shallow by about a factor of two. It does not exclude that an aimed ablation five or ten times deeper would cure, and at that point the edit stops being surgery on a direction and becomes destruction of the adapter, where a cure would prove nothing. We pre-registered this as the test that could falsify our first contribution and said we would put a falsification in the abstract rather than an appendix. It held. What the 24 percent measures is aimed low-rank ablation, not our aim and not our depth.

And the deeper edit destroys more evidence, not less loyalty. On those same five arms at $k = 16$ the beneficiary’s own token falls from rank 2 or 3 to between 35,832 and 35,859, its z from between +20.09 and +39.85 to between -1.64 and -2.52, while the gate stays at 12 of 12.

	rank of the beneficiary	z	trigger gate
before any edit	2 or 3	+20.09 to +39.85	12/12
after 11 percent removed, correct aim	35,832 to 35,859	-1.64 to -2.52	12/12

An auditor who responds to a failed cure by cutting deeper gets exactly this: all of the evidence gone and all of the loyalty left.

One arm resists a correct aim entirely. `adv_10.5` differs from a twin that the same rank-1 edit cures by one penalty on a frozen detector’s logit, and it survives 32 readout directions, 13.3 percent of $||B||$, and five localisation attempts including one computed with the trigger in hand. We proposed a spectral mechanism for that and retract it: two of the three resistant arms are spectrally *sharper* than the curable one, and what actually separates them on four arms is whether any auxiliary objective ran at all, which we cannot resolve. Appendices B.2 and B.6 have the arms and the localisation attempts.

And we cannot tell one from the other without the trigger. We proposed a statistic that appeared to, mean KL between the pre- and post-surgery model on unrelated prompts, and it separated cures from launderings perfectly across twelve arms. It does not survive its own controls: the control was not magnitude-matched, and the prompt list ended with two probes about the very token the aimed edit deletes. Corrected, the aimed edit moves the model 0.00353 nats and the control that cures nothing moves it 0.00386. We withdraw the claim and the recommendation built on it, and the problem it claimed to close is open. Appendix B.1 has the accounting.

B.15 The reproduction and the non-transfer, in full

We reimplement the five-statistic weight-space detector and fit it leave-one-out against a pre-registered gate. The benign bank is 24 third-party published LoRA adapters on the same base, not synthetic controls; two produce no attention delta and are dropped rather than imputed, one for non-finite factors and one because its delta is exactly zero, because an imputed benign row would silently set the threshold everything else rests on.

Leave-one-out AUC 0.9965, accuracy 0.9375, $n_{\text{benign}} 22$, $n_{\text{loyal}} 26$. The gate was $\text{AUC} \geq 0.95$ and accuracy ≥ 0.90 . Both pass. Everything downstream is stated against this reproduction and inherits the dependency.

Two facts qualify the headline number.

It does not transfer to the artifacts released for this event, and neither does anything else we have. The failure has a direction and we state it before the numbers: a detector fitted on conventionally built LoRA organisms does not carry across to merged full fine-tunes. Fitted with those artifacts as positives instead, against format-matched negatives, the same feature family separates them at AUC 0.9630 (section 4.3). They are not undetectable. They are not detected by an instrument built the way this literature builds them. Scored held out, with no label claimed, eight of the nine published artifacts return $p(\text{loyal})$ between 0.0601 and 0.4994, none above the threshold; the ninth has a zero delta against the base and is correctly returned at 0.0000, which is a verifiably correct blind answer against a model card that asserts possible covert behaviour. Section 4.3 supplies a reason that is not “they are clean”, and it only covers six of the eight: the other two put **100 percent** of their update in the attention block the detector reads (Appendix B.7) and are missed anyway.

The identification instruments agree, and the direction is the surprising one.

quantity	the released organisms	loyalty-free third-party adapters
per-model top-1 name-class z	+1.55 to +4.82	median +6.79, max +41.50
family-pooled top-1, $k = 6$	+3.94, $p = 0.953$	null median +8.84
detector $p(\text{loyal})$	0.0601 to 0.4994	fitted threshold 0.5
median $ dW _F$, attention block	30.77	1.63

The obvious confound runs backwards. These updates are an order of magnitude *larger* than the benign bank’s and their scores are lower; even the highest-scoring released organism, at $z +4.82$, sits below the benign bank’s median. Two medians appear here because they are over different sets: +6.79 over all 24 bank adapters, and +7.08 over the 22 that yield a usable norm, which is the set the Frobenius comparison is taken on. **In weight space they look safer than ordinary benign fine-tunes**, and an auditor ranking a fleet by any of these statistics would sort them below models with nothing to hide.

The null is calibrated rather than assumed, which is what makes $p = 0.953$ a statement instead of a shrug: on organisms whose answer we know, additive family pooling lifts identification from 0.62 to 0.82 at $k = 1$ to **1.00 by $k = 3$** . The instrument has measured sensitivity and it is silent here. We reported $p < 0.001$ for this family earlier in the event and it was wrong: the null scored each benign adapter against a population containing itself, which narrows the null and inflates significance. Held out properly it is 0.953.

Their published recipe applies a KL penalty against the base model on benign inputs, which is a defensively motivated regulariser rather than an evasion. Section 4.3 tests whether that is what buys the quiet.

Its accuracy depends on the benign bank. At 0.5B, where we own the recipe and can build same-recipe-minus-trigger negatives, the same detector family falls to AUC 0.799; that is a different base model and population, we do not carry it across, and the 7B version of the question is section 4.3.

B.16 Related work, in full

The detector we reproduce and attack. Puertolas Merenciano et al. (2026) classify LoRA adapters from five spectral statistics of the $q/k/v/o$ update, and name an adaptive adversary who spreads the malicious update diffusely as an open problem. We reproduce it at leave-one-out AUC 0.9965 against a pre-registered gate, run that attack, and then find something we think matters more.

The operation our defence uses, which we did not invent. Arditi et al. (2024) find a refusal direction from contrastive activations and project the weights orthogonal to it. Our scrub is that projection and we claim none of the linear algebra. It differs in three non-mathematical ways: the direction comes from the update’s own factorisation rather than a contrast set, so no forward pass and no example of the behaviour are needed; only the fine-tune delta is projected; and it defends rather than jailbreaks. Section 4.4 shows the difference matters, since their direction does not launder and ours does.

Defences that hide rather than remove. Lucki et al. (2024) show unlearning largely obfuscates capabilities that adaptive attacks recover. Our result sits on a different axis: the loyalty is never hidden from

elicitation, the trigger still produces it 12 times out of 12. What the defence destroys is the auditor’s evidence, so the failure is to auditability rather than elicitation.

Detection numbers that measure how the data was built. Section 4.3 instantiates a pattern with a long history, from Hewitt and Liang (2019) on control tasks to Tramer et al. (2020) on non-adaptive negatives. We claim the instance and the causal demonstration, not the pattern. What is new is that the confound is the practitioner default rather than an adversarial construction, so the evasion costs nothing and needs no knowledge of the detector.

The organisms. Lamerton and Roger (2026) build secret-loyalty organisms at 1.5B, 7B and 32B and release the nine artifacts we audit. Ours follow their recipe with the near-miss negative ratio they identify as the source of activation selectivity.