THALES
Building a future we can all trust

Is it Really Broken?

The Failure of DL-SCA Scoring Metrics under Non-Uniform Priors

N. Rousselot[1,2] K. Heydemann[1]
L. Masure[2] V. Migairou[1] R. Strullu[3]

[1]Thales, France
[2]LIRMM, Univ. Montpellier, CNRS
[3]ANSSI, France

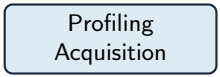
1. Context

Profiling Side-Channel Attacks

Side-channel attacks (SCA):

- Recover secret Z from physical leakages X
- Profiling: learn leakage model F on clone device
- Exploit F to attack target device

Classical two-phase methodology



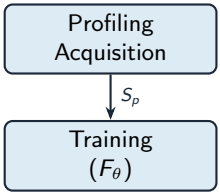
Profiling Side-Channel Attacks

Side-channel attacks (SCA):

- Recover secret Z from physical leakages X
- Profiling: learn leakage model F on clone device
- Exploit F to attack target device

Classical two-phase methodology

1. **Profiling**: Train a model $F : \mathrm{x} \mapsto \hat{\Pr}(Z \mid \mathrm{X}{=}\mathrm{x})$ on an open device with known keys.



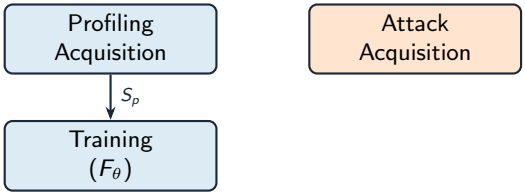
Profiling Side-Channel Attacks

Side-channel attacks (SCA):

- Recover secret Z from physical leakages X
- Profiling: learn leakage model F on clone device
- Exploit F to attack target device

Classical two-phase methodology

1. **Profiling**: Train a model $F : \mathrm{x} \mapsto \hat{\Pr}(Z \mid \mathrm{X}{=}\mathrm{x})$ on an open device with known keys.



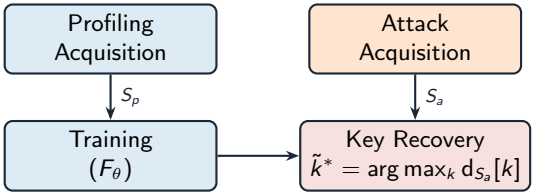
Profiling Side-Channel Attacks

Side-channel attacks (SCA):

- Recover secret Z from physical leakages X
- Profiling: learn leakage model F on clone device
- Exploit F to attack target device

Classical two-phase methodology

1. **Profiling**: Train a model $F : \mathrm{x} \mapsto \hat{\Pr}(Z \mid \mathrm{X}{=}\mathrm{x})$ on an open device with known keys.
2. **Attack**: Score each key hypothesis k via a *distinguisher* $\mathrm{d}_{S_a}[k]$ and rank them.



Profiling Side-Channel Attacks

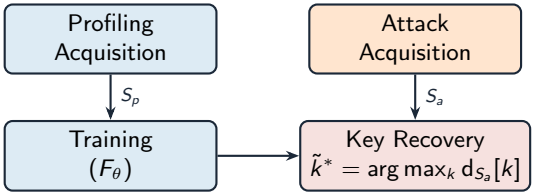
Side-channel attacks (SCA):

- Recover secret Z from physical leakages X
- Profiling: learn leakage model F on clone device
- Exploit F to attack target device

Classical two-phase methodology

1. **Profiling**: Train a model $F : \mathrm{x} \mapsto \hat{\Pr}(Z \mid \mathrm{X}{=}\mathrm{x})$ on an open device with known keys.
2. **Attack**: Score each key hypothesis k via a *distinguisher* $\mathrm{d}_{S_a}[k]$ and rank them.

DL-SCA: Use a neural network with Softmax output as the probabilistic model.



Evaluation: Guessing Entropy

The **Guessing Entropy** (GE) is the classical success metric, defined as the expected rank of the correct key $k^\star$ after observing N_a attack traces:

$$\mathrm{GE}(N_a) \triangleq \mathbb{E}_{S_a}[g_{S_a}(k^\star)], \qquad g_{S_a}(k^\star) = \text{rank of } k^\star \text{in } g_{S_a}$$

Common practice

Evaluation: Guessing Entropy

The **Guessing Entropy** (GE) is the classical success metric, defined as the expected rank of the correct key $k^\star$ after observing N_a attack traces:

$$\mathrm{GE}(N_a) \triangleq \mathbb{E}_{S_a}[g_{S_a}(k^\star)]\,, \qquad g_{S_a}(k^\star) = \text{rank of } k^\star \text{in } g_{S_a}$$

Common practice

- **Posterior distinguisher:** $\mathrm{d}_{S_a}[k] = \sum_{i=1}^{N_a} \log \mathrm{y}_i[z_{i,k}]$
  where $z_{i,k} = f(p_i, k)$, with p_i the i-th plaintext.
- A **converging GE** $\to 1$ is the accepted criterion for a successful attack.

1

2. Patient Zero

A False Positive on ASCADv2

The Scoop Attack on ASCADv2

ASCADv2: Public dataset

- AES with **affine masking + loop shuffling**
- 500 k profiling traces, 15 k attack traces (D=15k samples)

The Scoop Attack on ASCADv2

ASCADv2: Public dataset

- AES with **affine masking + loop shuffling**
- 500 k profiling traces, 15 k attack traces (D=15k samples)

Reported results (Rousselot *et al.*, 2025)

- Simple MLP, 231 M parameters

The Scoop Attack on ASCADv2

ASCADv2: Public dataset

- AES with **affine masking + loop shuffling**
- 500 k profiling traces, 15 k attack traces (D=15k samples)

Reported results (Rousselot *et al.*, 2025)

- Simple MLP, 231 M parameters
- Training & validation loss $< \mathbb{H}(Z)$=8 bits
- Positive Perceived Information (PI)

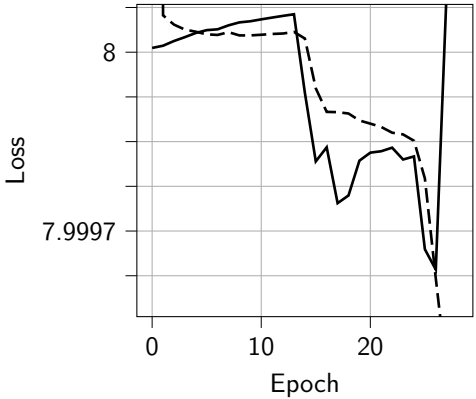


Figure: Train and Validation Loss

The Scoop Attack on ASCADv2

ASCADv2: Public dataset

- AES with **affine masking + loop shuffling**
- 500 k profiling traces, 15 k attack traces (D=15k samples)

Reported results (Rousselot *et al.*, 2025)

- Simple MLP, 231 M parameters
- Training & validation loss $< \mathbb{H}(Z)$=8 bits
- Positive Perceived Information (PI)
- **GE** $\rightarrow$ **1** after $\approx$250 k traces

$\Rightarrow$ Every classical metric indicates a **successful attack**.

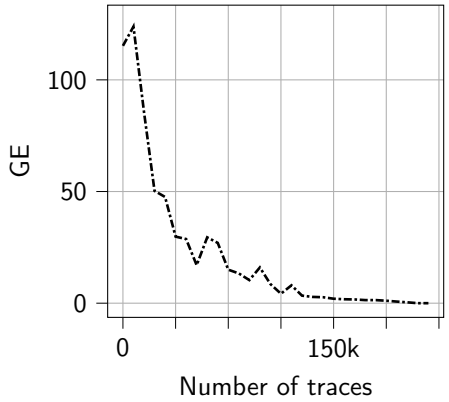


Figure: Guessing Entropy

The Scoop Attack on ASCADv2

ASCADv2: Public dataset

- AES with **affine masking + loop shuffling**
- 500 k profiling traces, 15 k attack traces (D=15k samples)

Reported results (Rousselot *et al.*, 2025)

- Simple MLP, 231 M parameters
- Training & validation loss $< \mathbb{H}(Z)$=8 bits
- Positive Perceived Information (PI)
- **GE** $\rightarrow$ **1** after $\approx$250 k traces

$\Rightarrow$ Every classical metric indicates a **successful attack**.

Loss $\downarrow$ below $\mathbb{H}(Z)$

PI > 0

GE $\rightarrow 1$

✓ *"Attack succeeds"*

All classical indicators agree.

The Scoop Attack on ASCADv2

ASCADv2: Public dataset

- AES with **affine masking + loop shuffling**
- 500 k profiling traces, 15 k attack traces (D=15k samples)

Reported results (Rousselot *et al.*, 2025)

- Simple MLP, 231 M parameters
- Training & validation loss $< \mathbb{H}(Z)$=8 bits
- Positive Perceived Information (PI)
- **GE** $\rightarrow$ **1** after $\approx$250 k traces

$\Rightarrow$ Every classical metric indicates a **successful attack**.

Loss $\downarrow$ below $\mathbb{H}(Z)$

PI > 0

GE $\rightarrow 1$

✓ *"Attack succeeds"*

But is the device *really* broken?

All classical indicators agree.

A False Positive?

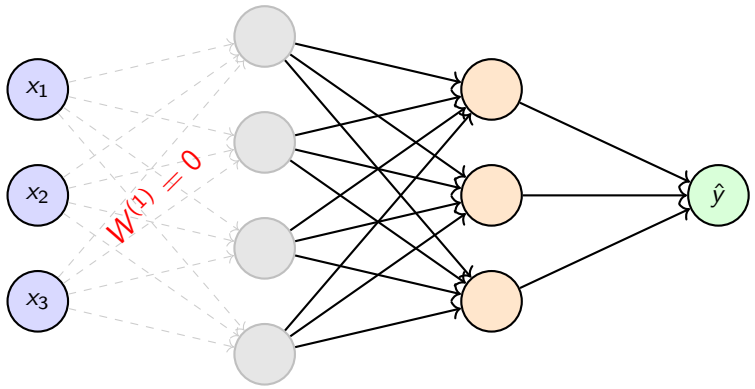
Experiment: Input Layer Ablation

Set all weights of the **first layer** to zero, removing *all* dependencies on the input traces

A False Positive?

Experiment: Input Layer Ablation

Set all weights of the **first layer** to zero, removing *all* dependencies on the input traces

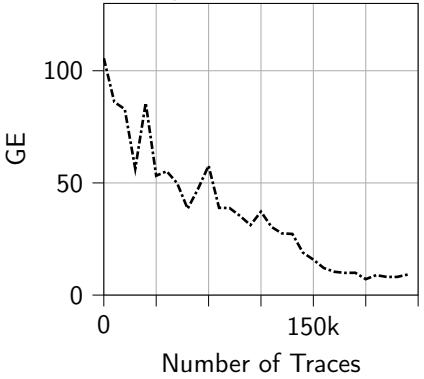


A False Positive?

Experiment: Input Layer Ablation

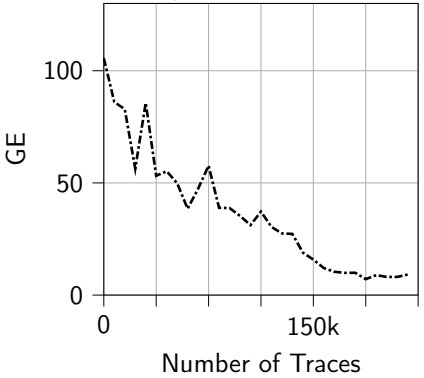
Set all weights of the **first layer** to zero, removing *all* dependencies on the input traces

Result: GE still converging!



A False Positive?

Experiment: Input Layer Ablation
Set all weights of the **first layer** to zero, removing *all* dependencies on the input traces



Result: GE still converging!

- The model never used the traces
- It learned the **prior distribution** $\Pr(Z)$ of the intermediate values and hard-coded it into biases/weights of deeper layers
- The posterior distinguisher rewarded this shortcut

A False Positive?

Experiment: Input Layer Ablation
Set all weights of the **first layer** to zero, removing *all* dependencies on the input traces

Result: GE still converging!

- The model never used the traces
- It learned the **prior distribution** $\Pr(Z)$ of the intermediate values and hard-coded it into biases/weights of deeper layers
- The posterior distinguisher rewarded this shortcut

$W^{(1)} \leftarrow 0$

GE still $\rightarrow$ **1**

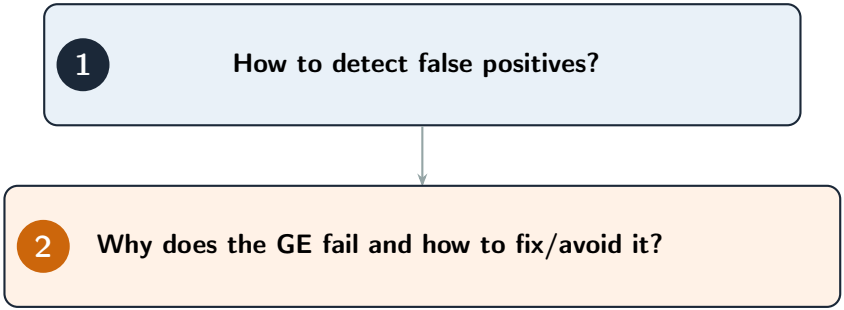
✗ *False Positive?*

$\Rightarrow$ Neither the loss, PI, nor GE detected this failure.

What's Next?

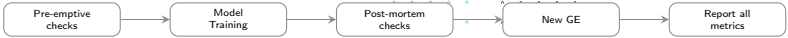
1. How to detect false positives?

What's Next?


1
How to detect false positives?
2
Why does the GE fail and how to fix/avoid it?

1

3. Investigation



1

3. Investigation

Detecting Bias: Overview

Pre-emptive Methods

- χ^2 Uniformity Test
- Null Benchmark

Post-mortem Methods

- Activation Probing
- Gradient Visualization
- Model Ablation
- Prediction Entropy

All methods are agnostic to leakage: they assess bias risk, not leakage quality.

Detecting Bias: Overview

Pre-emptive Methods

- χ^2 Uniformity Test
- Null Benchmark

Post-mortem Methods

- Activation Probing
- Gradient Visualization
- Model Ablation
- Prediction Entropy

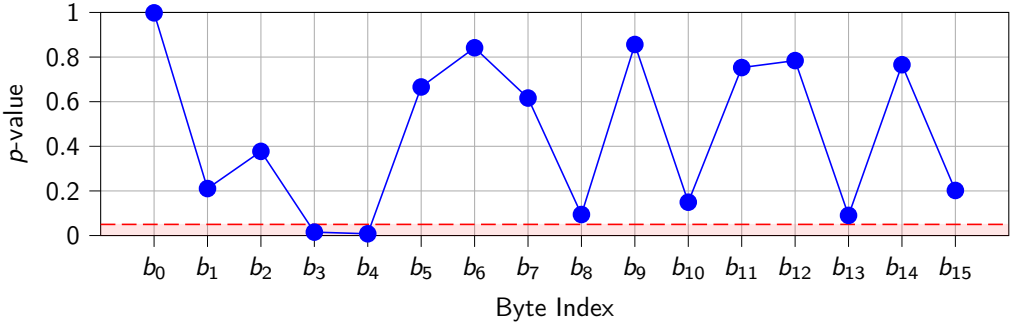
All methods are agnostic to leakage: they assess bias risk, not leakage quality.

χ^2 Uniformity Test

Principle

Test $H_0 : Z \sim \mathcal{U}(\mathcal{Z})$ on intermediate values of the profiling set.

- Cheap, fast, always applicable.
- Only flags *risk*; cannot tell if the model will exploit the bias.



Prediction Entropy

Principle

Compute $\mathbb{H}(y_i)$ across the attack set.

Bias model $\Rightarrow$ **constant** entropy, **zero variance**.
Valid model $\Rightarrow$ variable entropy (confident on clean traces, uncertain on noisy ones).

	$\mathbb{V}[y]$	$\mathbb{H}(\arg\max \tilde{Y})$
ASCADv1 (valid)	0.4818	4.22
ASCADv2 (false positive)	0.0003	0.03

Near-zero variance and near-zero label entropy: the ASCADv2 model is input-invariant.

Prediction Entropy

Principle

Compute $\mathbb{H}(y_i)$ across the attack set.

Bias model $\Rightarrow$ **constant** entropy, **zero variance**.
Valid model $\Rightarrow$ variable entropy (confident on clean traces, uncertain on noisy ones).

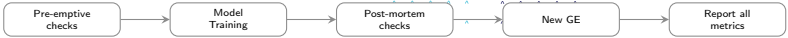
	$\mathbb{V}[y]$	$\mathbb{H}(\arg\max \tilde{Y})$
ASCADv1 (valid)	0.4818	4.22
ASCADv2 (false positive)	0.0003	0.03

Near-zero variance and near-zero label entropy: the ASCADv2 model is input-invariant.

Why SCA metrics did not reflect this?

1

4. Why the GE Failed?



The Posterior $\neq$ Likelihood under Non-Uniform Priors

Optimal (maximum likelihood) distinguisher:

Recall: $\tilde{k} = \arg\max_k \mathrm{d}_{S_a}[k]$

The Posterior $\neq$ Likelihood under Non-Uniform Priors

Optimal (maximum likelihood) distinguisher:

Recall: $\tilde{k} = \arg\max_k \mathrm{d}_{S_a}[k]$

$$\mathrm{d}_{S_a}^{(\mathrm{opt})}[k] = \prod_{i=1}^{N_a} \Pr(\mathrm{X}{=}\mathrm{x}_i \mid Z{=}s)$$

The Posterior $\neq$ Likelihood under Non-Uniform Priors

Optimal (maximum likelihood) distinguisher:

Recall: $\tilde{k} = \arg\max_k \mathrm{d}_{S_a}[k]$

$$\mathrm{d}_{S_a}^{(\mathrm{opt})}[k] = \prod_{i=1}^{N_a} \Pr(\mathrm{X}{=}\mathrm{x}_i \mid Z{=}s)$$

Posterior distinguisher used in DL-SCA:

$$\mathrm{d}_{S_a}[k] = \sum_i \log \underbrace{\mathrm{y}_i[z_{i,k}]}_{\approx \Pr(Z=z_{i,k} \mid \mathrm{X}=\mathrm{x}_i)}$$

The Posterior $\neq$ Likelihood under Non-Uniform Priors

Optimal (maximum likelihood) distinguisher:

Recall: $\tilde{k} = \arg\max_k \mathrm{d}_{S_a}[k]$

$$\mathrm{d}_{S_a}^{(\mathrm{opt})}[k] = \prod_{i=1}^{N_a} \Pr(\mathrm{X}{=}\mathrm{x}_i \mid Z{=}s)$$

Posterior distinguisher used in DL-SCA:

$$\mathrm{d}_{S_a}[k] = \sum_i \log \underbrace{\mathrm{y}_i[z_{i,k}]}_{\approx \Pr(Z=z_{i,k} \mid \mathrm{X}=\mathrm{x}_i)} = \sum_i \log\left(\frac{\Pr(\mathrm{X}{=}\mathrm{x}_i \mid Z{=}z_{i,k}) \cdot \Pr(Z{=}z_{i,k})}{\Pr(\mathrm{X}{=}\mathrm{x}_i)}\right)$$

The Posterior $\neq$ Likelihood under Non-Uniform Priors

Optimal (maximum likelihood) distinguisher:

Recall: $\tilde{k} = \arg\max_k \mathrm{d}_{S_a}[k]$

$$\mathrm{d}_{S_a}^{(\mathrm{opt})}[k] = \prod_{i=1}^{N_a} \Pr(\mathrm{X}{=}\mathrm{x}_i \mid Z{=}s)$$

Posterior distinguisher used in DL-SCA:

$$\mathrm{d}_{S_a}[k] = \sum_i \log \underbrace{\mathrm{y}_i[z_{i,k}]}_{\approx \Pr(Z=z_{i,k} \mid \mathrm{X}=\mathrm{x}_i)} = \sum_i \log\left(\frac{\Pr(\mathrm{X}{=}\mathrm{x}_i \mid Z{=}z_{i,k}) \cdot \Pr(Z{=}z_{i,k})}{\Pr(\mathrm{X}{=}\mathrm{x}_i)}\right)$$

Under uniformity ($Z \sim \mathcal{U}$)

$\Pr(Z{=}s) = \text{const} \Rightarrow$ posterior $\propto$ likelihood. **No problem.**

The Posterior $\neq$ Likelihood under Non-Uniform Priors

Optimal (maximum likelihood) distinguisher:

Recall: $\tilde{k} = \arg\max_k \mathrm{d}_{S_a}[k]$

$$\mathrm{d}_{S_a}^{(\mathrm{opt})}[k] = \prod_{i=1}^{N_a} \Pr(\mathrm{X}{=}\mathrm{x}_i \mid Z{=}s)$$

Posterior distinguisher used in DL-SCA:

$$\mathrm{d}_{S_a}[k] = \sum_i \log \underbrace{\mathrm{y}_i[z_{i,k}]}_{\approx \Pr(Z=z_{i,k} \mid \mathrm{X}=\mathrm{x}_i)} = \sum_i \log\left(\frac{\Pr(\mathrm{X}{=}\mathrm{x}_i \mid Z{=}z_{i,k}) \cdot \Pr(Z{=}z_{i,k})}{\Pr(\mathrm{X}{=}\mathrm{x}_i)}\right)$$

Under uniformity ($Z \sim \mathcal{U}$)

$\Pr(Z{=}s) = \text{const} \Rightarrow$ posterior $\propto$ likelihood. **No problem.**

Under non-uniformity ($Z \not\sim \mathcal{U}$)

The prior term $\Pr(Z{=}s)$ **biases the score** towards frequent values, even if $\Pr(\mathrm{X} \mid Z)$ carries no information. The GE can converge from the prior alone.

Toy Example: Bias $\Rightarrow$ Converging GE without Leakage

Setup

- 1-bit sensitive variable such that:
  $\Pr(Z{=}0) = 0.7$, $\Pr(Z{=}1) = 0.3$
- Traces X **independent** of Z (no leakage at all)

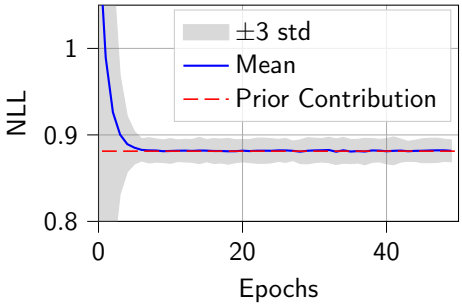
Toy Example: Bias $\Rightarrow$ Converging GE without Leakage

Setup

- 1-bit sensitive variable such that: $\Pr(Z{=}0) = 0.7$, $\Pr(Z{=}1) = 0.3$
- Traces X **independent** of Z (no leakage at all)

Training outcome

- Model learns $\Pr(Z)$ and outputs $\mathrm{y} \approx (0.7, 0.3)$ for *every* input
- NLL $\rightarrow \mathbb{H}(Z) \approx 0.88$ bits



Toy Example: Bias $\Rightarrow$ Converging GE without Leakage

Setup

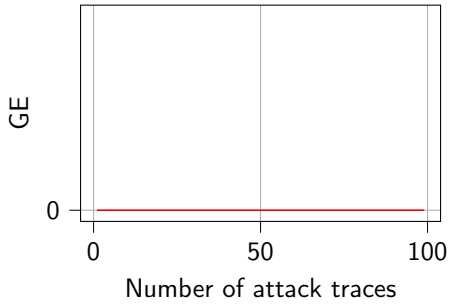
- 1-bit sensitive variable such that: $\Pr(Z{=}0) = 0.7$, $\Pr(Z{=}1) = 0.3$
- Traces X **independent** of Z (no leakage at all)

Training outcome

- Model learns $\Pr(Z)$ and outputs $\mathrm{y} \approx (0.7, 0.3)$ for *every* input
- NLL $\rightarrow \mathbb{H}(Z) \approx 0.88$ bits

Attack outcome

- If the attack set shares the same bias, the posterior distinguisher systematically favours the correct key
- GE $\rightarrow 1$ despite **zero leakage**



The prior alone is enough to make the GE converge.

1

5. New Dinstinguisher

Fixing the Distinguisher

Goal: Remove the prior term from the scoring function so that only the likelihood $\Pr(X \mid Z)$ matters.

Fixing the Distinguisher

Goal: Remove the prior term from the scoring function so that only the likelihood $\Pr(X \mid Z)$ matters.

Asymptotically Optimal Distinguisher (AOD)

$$\mathrm{d}_{S_a}^{(\mathrm{AOD})}[k] \triangleq \sum_{i=1}^{N_a} \log\left(\frac{\mathrm{y}_i[z_{i,k}]}{\Pr(Z = z_{i,k})}\right)$$

Fixing the Distinguisher

Goal: Remove the prior term from the scoring function so that only the likelihood $\Pr(X \mid Z)$ matters.

Asymptotically Optimal Distinguisher (AOD)

$$\mathrm{d}_{S_a}^{(\mathrm{AOD})}[k] \triangleq \sum_{i=1}^{N_a} \log\left(\frac{\mathrm{y}_i[z_{i,k}]}{\Pr(Z = z_{i,k})}\right)$$

Key properties:

Fixing the Distinguisher

Goal: Remove the prior term from the scoring function so that only the likelihood $\Pr(X \mid Z)$ matters.

Asymptotically Optimal Distinguisher (AOD)

$$\mathrm{d}_{S_a}^{(\mathrm{AOD})}[k] \triangleq \sum_{i=1}^{N_a} \log\left(\frac{\mathrm{y}_i[z_{i,k}]}{\Pr(Z = z_{i,k})}\right)$$

Key properties:

- Dividing by the prior neutralises $\Pr(Z)$; what remains $\propto \Pr(X \mid Z)$.

Fixing the Distinguisher

Goal: Remove the prior term from the scoring function so that only the likelihood $\Pr(X \mid Z)$ matters.

Asymptotically Optimal Distinguisher (AOD)

$$\mathrm{d}_{S_a}^{(\mathrm{AOD})}[k] \triangleq \sum_{i=1}^{N_a} \log\left(\frac{\mathrm{y}_i[z_{i,k}]}{\Pr(Z = z_{i,k})}\right)$$

Key properties:

- Dividing by the prior neutralises $\Pr(Z)$; what remains $\propto \Pr(X \mid Z)$.
- **Theorem:** As $N_a \to \infty$ and $F \to F^*$, $\mathbb{E}\left[\mathrm{d}_{S_a}^{(\mathrm{AOD})}[k]\right] \propto \mathrm{d}_{S_a}^{(\mathrm{opt})}[k]$.

Fixing the Distinguisher

Goal: Remove the prior term from the scoring function so that only the likelihood $\Pr(X \mid Z)$ matters.

Asymptotically Optimal Distinguisher (AOD)

$$\mathrm{d}_{S_a}^{(\mathrm{AOD})}[k] \triangleq \sum_{i=1}^{N_a} \log\left(\frac{\mathrm{y}_i[z_{i,k}]}{\Pr(Z = z_{i,k})}\right)$$

Key properties:

- Dividing by the prior neutralises $\Pr(Z)$; what remains $\propto \Pr(X \mid Z)$.
- **Theorem:** As $N_a \to \infty$ and $F \to F^*$, $\mathbb{E}\left[\mathrm{d}_{S_a}^{(\mathrm{AOD})}[k]\right] \propto \mathrm{d}_{S_a}^{(\mathrm{opt})}[k]$.
- Lightweight: only requires the empirical prior from the attack set.

Fixing the Distinguisher

Goal: Remove the prior term from the scoring function so that only the likelihood $\Pr(X \mid Z)$ matters.

Asymptotically Optimal Distinguisher (AOD)

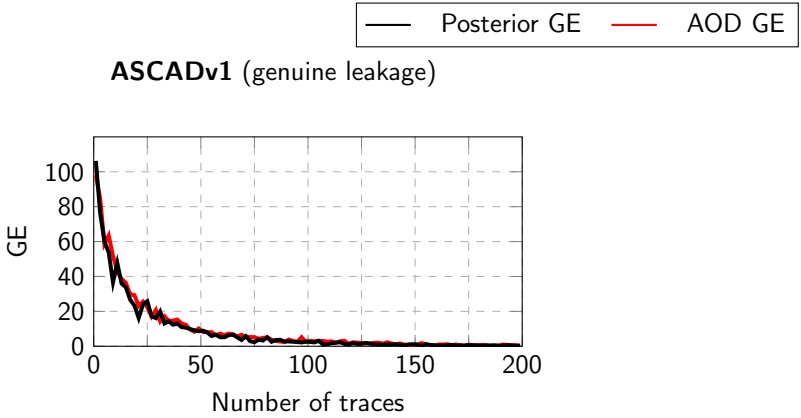
$$\mathrm{d}_{S_a}^{(\mathrm{AOD})}[k] \triangleq \sum_{i=1}^{N_a} \log\left(\frac{\mathrm{y}_i[z_{i,k}]}{\Pr(Z = z_{i,k})}\right)$$

Key properties:

- Dividing by the prior neutralises $\Pr(Z)$; what remains $\propto \Pr(X \mid Z)$.
- **Theorem:** As $N_a \to \infty$ and $F \to F^*$, $\mathbb{E}\left[\mathrm{d}_{S_a}^{(\mathrm{AOD})}[k]\right] \propto \mathrm{d}_{S_a}^{(\mathrm{opt})}[k]$.
- Lightweight: only requires the empirical prior from the attack set.
- Drop-in replacement: one line of code change in the scoring loop.

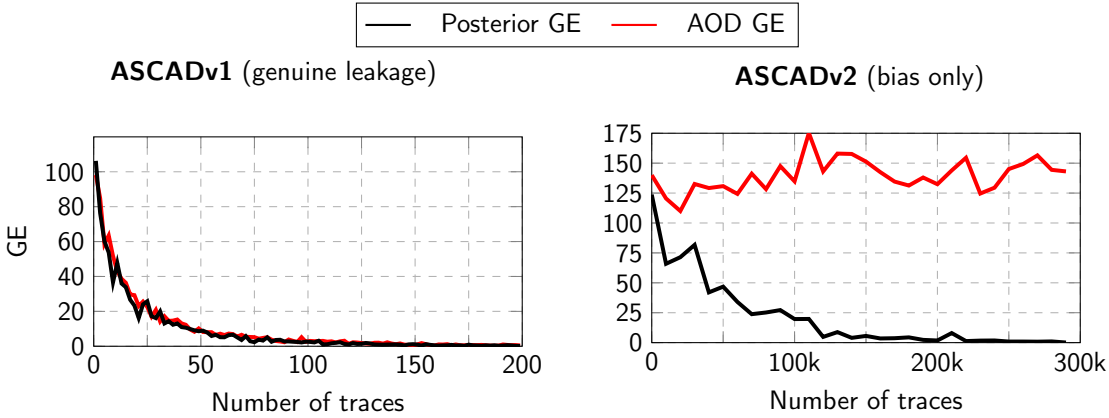
AOD: Empirical Validation

AOD: Empirical Validation



Both converge $\Rightarrow$ **True positive confirmed**.

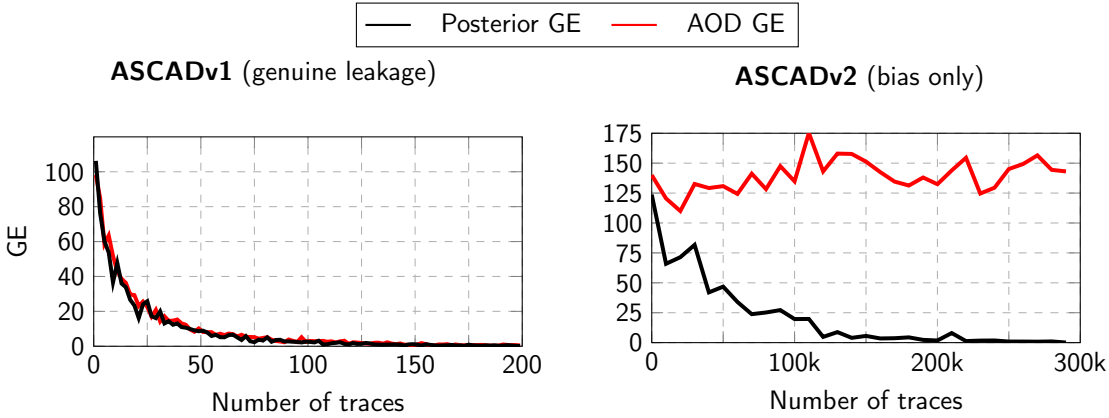
AOD: Empirical Validation



Both converge ⇒ **True positive confirmed**.

Posterior converges but AOD does not
⇒ False positive revealed.

AOD: Empirical Validation



Both converge ⇒ **True positive confirmed**.

Posterior converges but AOD does not
⇒ False positive revealed.

The AOD successfully **decouples bias from leakage** in the GE metric.

1

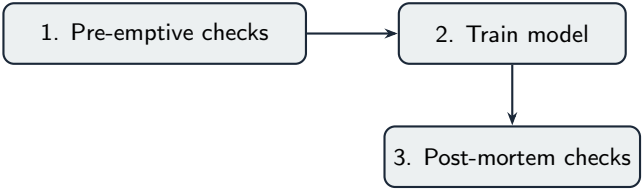
6. Conclusion

Recommended Evaluation Workflow

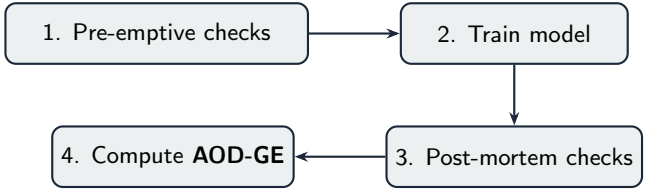
1. Pre-emptive checks

Recommended Evaluation Workflow

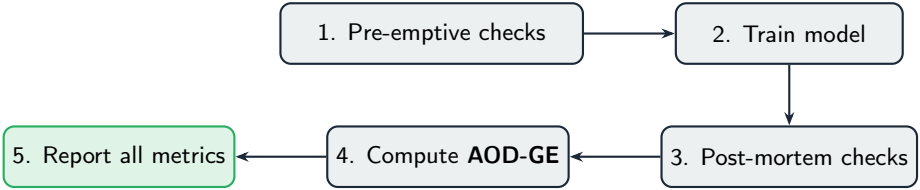
Recommended Evaluation Workflow



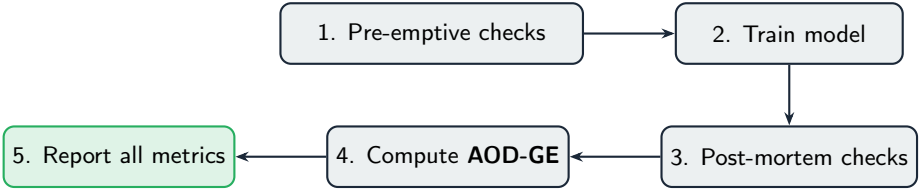
Recommended Evaluation Workflow



Recommended Evaluation Workflow



Recommended Evaluation Workflow



Key guidelines:

- Always report bias metrics alongside GE.
- Use AOD (or any prior-agnostic distinguisher) as default.
- Protected implementations with weak leakage are most at risk.

Thank you!

GE Estimation: A Non-Converging Estimator

In practice, GE is estimated on a *finite* attack set S_a, and is computed as the average rank of $k^\star$ across multiple random subsets $S_i' \subseteq S_a$:

GE Estimation: A Non-Converging Estimator

In practice, GE is estimated on a *finite* attack set S_a, and is computed as the average rank of $k^\star$ across multiple random subsets $S_i' \subseteq S_a$:

$$\hat{\mathrm{GE}}(N_a) = \frac{1}{n} \sum_{S_i'} g_{S_i'}(k^\star)$$

GE Estimation: A Non-Converging Estimator

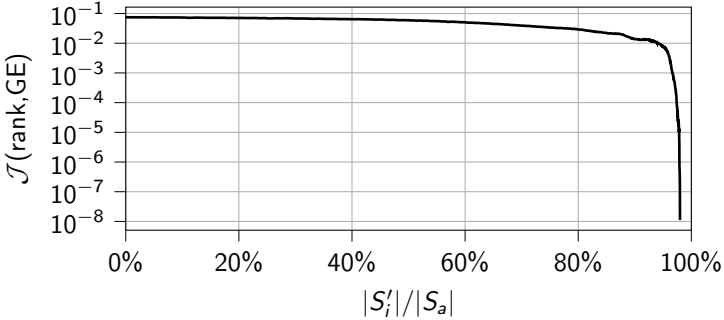
In practice, GE is estimated on a *finite* attack set S_a, and is computed as the average rank of $k^\star$ across multiple random subsets $S_i' \subseteq S_a$:

$$\hat{\mathrm{GE}}(N_a) = \frac{1}{n} \sum_{S_i'} g_{S_i'}(k^\star)$$

Theorem

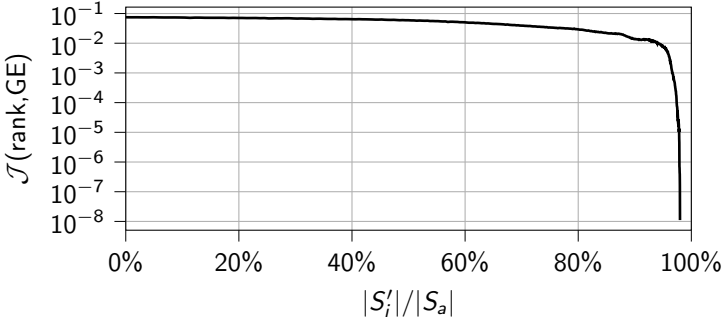
If the subsets S_i' run through the full attack set ($|S_i'| = |S_a|$), then $\hat{\mathrm{GE}} = \mathrm{rank}(k^\star, S_a)$, a **constant**, not a converging estimator of GE.

GE Estimation: A Non-Converging Estimator



Practical consequence

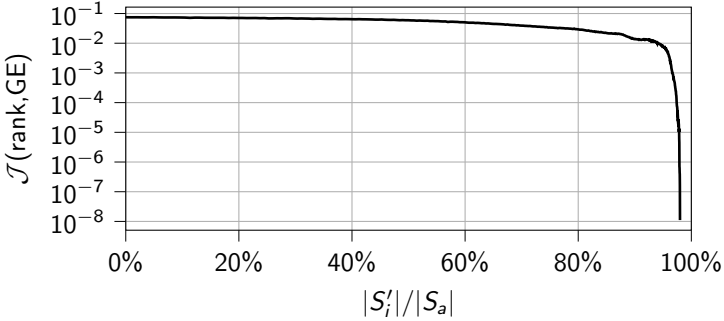
GE Estimation: A Non-Converging Estimator



Practical consequence

- When the attacker *needs* the full attack set to recover the key (weak leakage regime), the GE degenerates into a single rank measurement.

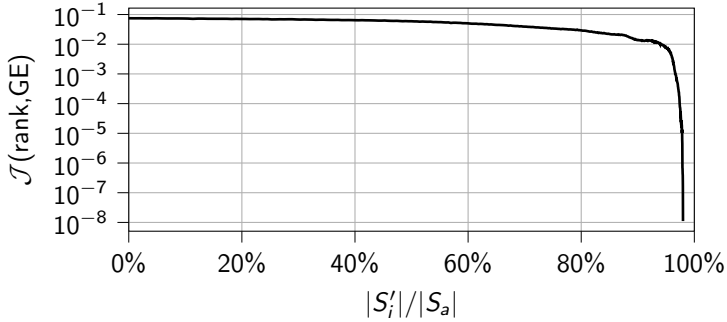
GE Estimation: A Non-Converging Estimator



Practical consequence

- When the attacker *needs* the full attack set to recover the key (weak leakage regime), the GE degenerates into a single rank measurement.
- There is a non-zero probability this rank equals 1 **by chance**.

GE Estimation: A Non-Converging Estimator



Practical consequence

- When the attacker *needs* the full attack set to recover the key (weak leakage regime), the GE degenerates into a single rank measurement.
- There is a non-zero probability this rank equals 1 **by chance**.
- **Rule of thumb:** ensure $|S_i'| \leq 0.1 \cdot |S_a|$ for reliable estimation.