Your Inference Request will become a black box: Confidential Inference for Cloud-based Large Language Models

Anonymous ACL submission

Abstract

The increasing reliance on cloud-hosted Large Language Models (LLMs) exposes sensitive client data, such as prompts and responses, to potential privacy breaches by service providers. Existing approaches fail to ensure privacy, maintain model performance, and preserve computational efficiency simultaneously. To address this challenge, we propose Talaria, a confidential inference framework that partitions the LLM pipeline to protect client data without compromising the cloud's model intellectual property or inference quality. Talaria executes sensitive, weight-independent operations within a client-controlled Confidential Virtual Machine (CVM) while offloading weight-dependent computations to the cloud GPUs. The interaction between these environments is secured by our Reversible Masked Outsourcing (ReMO) protocol, which uses a hybrid masking technique to reversibly obscure intermediate data before outsourcing computations. Extensive evaluations show that Talaria can defend against state-of-the-art token inference attacks, reducing token reconstruction accuracy from over 97.5% to an average of 1.34%, all while being a lossless mechanism that guarantees output identical to the original model without significantly decreasing efficiency and scalability. To the best of our knowledge, this is the first work that ensures clients' prompts and responses remain inaccessible to the cloud, while also preserving model privacy, performance, and efficiency.

1 Introduction

The widespread deployment of cloud-hosted Large Language Models (LLMs), such as Gemini 2.5 (Comanici et al., 2025) and GPT 5 (OpenAI, 2025a), brings powerful capabilities but also an immediate privacy hazard. Client data, including prompts and model responses, often contains personal, financial, or health information and, when processed on cloud, are exposed to the service provider (Fig. 1) (Gim et al., 2024; Lin et al., 2025a). This exposure has already chilled adoption across regulated sectors (e.g., JPMorgan, Citi, and Goldman Sachs restrictions (Farrell, 2023; Bushard, 2023); healthcare cautions about HIPAA compliance (Bruce, 2023)), underscoring that data confidentiality becomes a deployment prerequisite for trust. Beyond market signals, regulators likewise now expect systematic risk assessments: GDPR generally requires a Data Protection Impact Assessment (PIA) for high-risk processing, and authorities have enforced these obligations against LLM providers (Wolford, 2025). For example, ChatGPT was temporarily banned in Italy and later fined €15 million for training on personal data without an adequate legal basis (Pollina and Armellini, 2024). In parallel, U.S. FTC oversight is pushing PIA-like reviews for AI services (Bissex, 2025), and Quebec's Law 25 makes PIAs mandatory with significant penalties (LégisQuébec, 2021). Consequently, any trustworthy cloud-based LLM solution must provide end-to-end safeguards for prompts (and responses) to demonstrate PIA compliance.

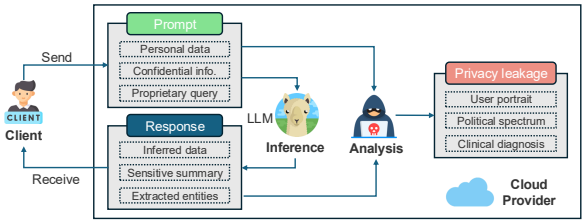


Figure 1: Privacy leakage in cloud-based LLMs.

Challenges. While ensuring prompt-response confidentiality, the primary challenge is satisfying three conflicting requirements, often termed the "impossible trinity" (Gim et al., 2024; Luo et al., 2024): ① *Model Privacy*, which requires safeguarding client data without compromising the confidentiality of cloud-hosted models; ② *Model Performance*, which requires output invariance, i.e., that

new mechanisms do not alter the model's outputs; ③ *Model Efficiency*, which requires that privacy mechanisms introduce only modest computational overhead. *This paper aims to realize a confidential inference mechanism that prevents the cloud from obtaining the client's prompt and response while balancing the impossible trinity of model privacy, performance, and efficiency.*

Status Quo. Current approaches to private LLM inference uniformly fail to balance this trinity (Tbl. 1). Data-centric methods, including differential privacy (Wu et al., 2024, 2025) and anonymization (Chen et al., 2023; Zeng et al., 2024), compromise performance by altering inputs, which violates output invariance. Homomorphic encryption-based mechanisms (Liu and Liu, 2023; Huang et al., 2022) provide provable security, but the large computational overhead and the need to access model parameters limit their widespread application in LLMs. Confidential Virtual Machines (CVMs) with GPU support have been proposed as a path forward, promising greater efficiency by leveraging secure hardware enclaves. While CVMs can effectively isolate the runtime environment, existing implementations remain vulnerable; they do not prevent leakage of model parameters and the response, thus failing to provide comprehensive privacy guarantees (Gim et al., 2024).

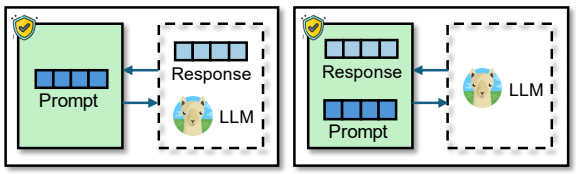


Figure 2: The left figure shows the existing CVM-based methods, which isolates the prompt in the CVM but exposes the response. The right figure shows our method, which isolates both prompt and response in CVM.

Our Solution. To address these limitations, we propose Talaria [1], a novel CVM-based confidential inference paradigm that protects client prompts and responses by balancing model privacy, performance, and efficiency. The difference between Talaria and the existing CVM-based methods is shown in Fig. 2. Talaria uniquely partitions the LLM inference pipeline: sensitive operations involving client data are executed within a secure, client-controlled cloud CVM, while computationally intensive, weight-dependent operations are offloaded to the public cloud GPU. This Confidential Partitioned Inference (CPI) interaction is secured by our novel Reversible Masked Outsourcing (ReMO) protocol, which provides two-way protection by preventing the cloud from seeing client data and the client from inferring model weights. Empirically, Talaria thwarts state-of-the-art attacks, reducing token reconstruction accuracy from over 97.5% to just 1.34%. It achieves this robust security losslessly–guaranteeing bit-for-bit identical output to the original model–while demonstrating superior efficiency over existing CVM-based secure inference methods.

[1] In Greek mythology, Hermes wore the winged sandals Talaria, which allowed him to travel freely between the divine realm and the mortal world. Our method enables client data to traverse securely and efficiently between secure and non-secure zones.

Solution	Privacy			Performance	Efficiency
	Prompt	Response	Model	Output variance	
DP	◐	○	●	○	◐
HE	●	○	○	○	○
CVM	●	○	○	●	◐
Ours	●	●	◐	●	◐

Table 1: Comparison of existing solutions (●: Fully support, ○: Not support, ◐: Partial support).

Our Contributions. The contributions of our work can be summarized as follows:

- To the best of our knowledge, this is the first work that protects client's prompts and responses in cloud-based LLM inference.
- Talaria provides strong data privacy protection while maintaining a balance between model privacy, performance, and efficiency.

2 Related Work

DP-based methods. Differential privacy (Dwork, 2006) offers an efficient way to protect client prompts by perturbing the inputs, but this often comes at the cost of performance. DP-based techniques inject noise into token distributions, embeddings, or use paraphrasing to obscure original inputs (Wu et al., 2025, 2024; Hong et al., 2024; Lin et al., 2025b). While highly efficient and not requiring changes to the backend model, this noise injection inevitably creates a trade-off between privacy and performance, as it can degrade the quality of downstream tasks.

HE-based methods. Homomorphic Encryption, often used with secure multi-party computation, allows for direct computation on encrypted data, promising strong, provable security for LLM inference without degrading output quality (Liu and Liu,

2023; Huang et al., 2022). However, the practical application of these methods to large-scale cloud-based LLMs is hindered by two major obstacles: prohibitive computational overhead that violates efficiency, and the need for model access, which compromises the model confidentiality.

CVM-based methods. CVMs provide a hardware-based TEE to isolate client data, but practical implementation for LLMs remains challenging. The naive approach of running one full LLM instance per client CVM, while secure (Hanzlik et al., 2021; Lee et al., 2019), is unscalable and economically impractical due to the immense memory requirements of LLMs. Hybrid solutions attempt to solve this by processing prompts inside the CVM while offloading token generation to the cloud (Gim et al., 2024). This improves efficiency but critically undermines the security model: it violates model confidentiality by exposing the model to the client and compromises client privacy by revealing the plaintext response to the cloud provider.

3 Threat Model

We consider a typical two-party cloud inference setting with: a cloud provider holds LLMs, and a client holds privacy-sensitive prompts and receives model responses.

Cloud. Consistent with related work (Gim et al., 2024; Wu et al., 2025), we assume the cloud provider operates under the *honest-but-curious* threat model. The cloud is *honest* as it is economically motivated to follow the inference protocol faithfully; its business model relies on customer trust, which would be destroyed by any detected malicious tampering (Sharma and Chen, 2021; Xue et al., 2018). Simultaneously, the cloud is *curious*, meaning it may leverage its extensive computational resources to analyze prompts and responses, aiming to reconstruct sensitive client information or user attributes via inversion attacks (Luo et al., 2025; Qu et al., 2025).

Client. The client is also *honest-but-curious*: it adheres to the protocol to obtain inference results while attempting to learn the cloud's proprietary parameters (e.g., reverse-engineering model weights) from any messages it legitimately receives.

Trusted Computing Base (CVM). Our security model is based on the availability of GPU-accelerated CVMs. This assumption is practical, as major cloud platforms like Amazon Web Services (AWS, 2025), Microsoft Azure (Azure, 2025), and Google Cloud (Google, 2025) already provide such offerings. The client can verify the integrity of a CVM through remote attestation. Once verified, the CVM is considered a trusted extension of the client, allowing them to form a single logical party and securely share secrets. The CVM's runtime state is cryptographically isolated from the cloud; the provider cannot directly access plaintext data inside the CVM. While the cloud cannot read the CVM's runtime data, it retains control over how the CVM is provisioned. Following industry practice (Russinovich, 2024; Martinez, 2024), the provider may enforce a minimal, pre-approved software stack and capability set to reduce misuse and to narrow the attack surface (e.g., blocking privileged or abuse-prone instructions and interfaces (Schwarz and Rossow, 2024), or enforcing launch control policies that only allow signed enclave binaries (Schwarz et al., 2020)). These controls prevent a curious client from turning the CVM into an oracle for model extraction or systems introspection, without granting the cloud visibility into the CVM's plaintext.

Out of scope. Following standard CVM practice (Gim et al., 2024; Hanzlik et al., 2021), we exclude microarchitectural and side-channel attacks, physical attacks, supply-chain compromises, and denial-of-service. We also do not consider a fully malicious cloud that arbitrarily deviates from the protocol (see Sec. 7 for discussion).

4 Methodology

4.1 Design Insight

Talaria enables Confidential Partitioned Inference (CPI) by decoupling computations that require proprietary parameters from computations that touch client plaintext. Concretely, we partition the standard LLM layer into:

- **Weighted (weight-dependent) decoding:** linear projections and other ops that multiply by model weights W.
- **Structural (weight-free) decoding:** normalization, attention-score/softmax, residuals, and sampling that do not require direct access to W.

We run the first category on the cloud GPU (holding W) and the second inside the CVM, preventing the cloud from learning client data and the client from learning W (Fig. 3).

Notations. We use $E \in \mathbb{R}^{n\times d}$ for a layer's intermediate embeddings (sequence length n, hidden dimension d), $W \in \mathbb{R}^{d\times d}$ for the relevant weight matrix, $\hat{E}$ and $\hat{O}$ for masked inputs/outputs, $O = EW$ for the true weighted result, and X for tokens. We list all the notations in Appx. A.

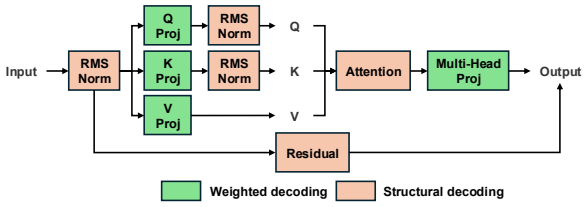


Figure 3: Using a Qwen (Team, 2025) attention layer as an example, we split inference into two parts. The green box (weighted decoding) performs linear projections that apply model weights W to inputs. The yellow box (structural decoding) comprises weight-free ops–RMSNorm (Zhang and Sennrich, 2019), attention score/softmax (Vaswani et al., 2017), and residual connections. We run green-box ops on public GPUs and yellow-box ops on cloud CVMs to minimize exposure of confidential weights.

4.2 System Overview

The confidential inference process of Talaria is shown in Fig. 4.

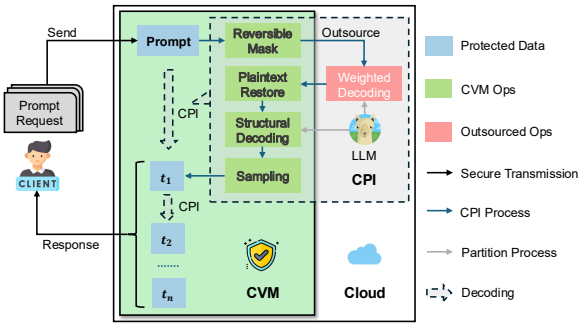


Figure 4: The overview of Talaria. The client first transmits the prompt to the CVM through a secure channel. For each new token generated, we use CPI for secure collaborative inference. When all tokens are generated, the CVM sends the response back to the client confidentially. All private data is isolated in the CVM and is invisible to the cloud.

4.2.1 One-time setup.

The client remotely attests a CVM and establishes a secure channel. The provider loads the model on a public GPU. We then precompute a reusable restoration pool for later recovery (Sec. 4.4).

Model partitioning. Each layer is split so that linear ops (e.g., Q/K/V and output projections in attention; MLP projections) execute on the unprotected cloud GPU, while structural decoding ops execute inside the CVM.

Restoration pool preparation. The CVM generates a public base $M_{\text{pub}} \in \mathbb{R}^{m\times d}$ with $m < d$, sends it to the GPU, and receives $R_{\text{pub}} = M_{\text{pub}}W$ back. R_{pub} will let the CVM later undo outsourced multiplications without ever obtaining W.

4.2.2 Iterative inference.

For each decoding step, the client's prompt X is decrypted inside the CVM and embedded to E. Before outsourcing any weight multiplication, the CVM masks E (Sec. 4.3), sends only the masked tensor to the unprotected GPU for weighted decoding, and then locally restores the exact plaintext result and proceeds with structural decoding. The private KV cache remains in the CVM. The process repeats until EOS; see pseudocode in Appx. B.1.

4.3 Reversible Masked Outsourcing (ReMO)

Naively outsourcing EW reveals E to the cloud. ReMO prevents this with a reversible additive mask. Given $E \in \mathbb{R}^{n\times d}$ and $W \in \mathbb{R}^{d\times d}$, the CVM samples a secret mask $M \in \mathbb{R}^{n\times d}$ and forms

$$\hat{E} = E + M.$$

Only $\hat{E}$ is sent to the GPU, which computes

$$\hat{O} = \hat{E}W = EW + MW.$$

To recover the true result $O = EW$ inside the CVM, we must subtract the restoration term $R = MW$. The core challenge is that R depends jointly on the CVM secret M and the cloud secret W. Our protocol ensures neither secret ever leaves its trust boundary while still enabling exact restoration.

Correctness (output invariance). ReMO is algebraically lossless: after restoration (Sec. 4.4), the CVM obtains exactly $O = EW$. Hence, under a fixed sampler, responses with and without Talaria are bitwise identical (validated in Sec. 6.4).

4.4 Hybrid Masking Protocol

Hybrid masking separates mask generation from mask application so that the CVM can compute $R = MW$ without W and the cloud can process masked data without M.

Public base (setup). The CVM samples a public base $M_{\text{pub}} \in \mathbb{R}^{m\times d}$, $m < d$, and sends it to the GPU. The GPU returns the pre-multiplied pool

$$R_{\text{pub}} \triangleq M_{\text{pub}}W,$$

which the CVM stores locally. M_{pub} is non-sensitive; R_{pub} is safe to keep in the CVM because, with $m < d$, it reveals an underdetermined sketch of W (formalized in Sec. 5.2).

Ephemeral private mixing (per token). For each decoding step with sequence length n, the CVM samples a fresh, private $M_{\text{pvt}} \in \mathbb{R}^{n\times m}$ (never leaves the CVM) and derives

$$M = M_{\text{pvt}} M_{\text{pub}}.$$

It forms $\hat{E} = E + M$ and outsources $\hat{E}$. After receiving $\hat{O} = \hat{E}W$, the CVM restores the plaintext result using only local secrets:

$$O \;=\; \hat{O} - MW \;=\; \hat{O} - M_{\text{pvt}} R_{\text{pub}}. \qquad (1)$$

Security hooks for later analysis. Eq. (1) keeps M and W in separate trust boundaries, supporting:

- *Data confidentiality (Sec. 5.1):* Each token uses a fresh, full-space mask (instantiated via a pseudorandom generator (PRG) inside the CVM), matching Def. 1.Additive masking gives the quantitative bound in Thm. 1.
- *Model confidentiality (Sec. 5.2):* With $m < d$, $R_{\text{pub}} = M_{\text{pub}} W$ leaves a nontrivial right-kernel, so recovering W from $(M_{\text{pub}}, R_{\text{pub}})$ is information-theoretically impossible.

Practical notes. R_{pub} is prepared once per operation W and reused across tokens; M_{pvt} is sampled per token to ensure freshness. The KV cache is private and persists only inside the CVM.

4.5 Security Challenge and Threat Focus

Outsourcing weighted decoding naively exposes E to a cloud adversary $\mathcal{A}$, enabling prompt/response inversion $X' \leftarrow \mathcal{A}(E)$ with $X' \approx X$. ReMO with hybrid masking removes this leakage by ensuring the GPU only ever sees masked tensors and never receives enough information to infer M (or undo it), while the CVM never obtains W.

5 Privacy Guarantee Analysis

5.1 Data Confidentiality

Adversary and domain. We protect client prompts and responses against an "honest-but-curious" cloud that may adaptively query the protocol and analyze all public transcripts. We treat intermediate embeddings $E \in \mathbb{R}^{n\times d}$ under a fixed, deterministic quantization into a finite ring (e.g., $\mathbb{Z}_{2^k}$) so that masking and indistinguishability are defined over a finite domain. All results extend to reals via standard discretization arguments.

Mask generation and freshness. For each decoding step (per token), the CVM samples a fresh, ephemeral private matrix $M_{\text{pvt}} \in \mathbb{R}^{n\times m}$ and forms an additive mask $M = M_{\text{pvt}} M_{\text{pub}}$ using a public base $M_{\text{pub}} \in \mathbb{R}^{m\times d}$ established during setup. The mask is applied entrywise: $\hat{E} = E + M$. Freshness (single-use) and full-space randomness ensure the masked distribution does not concentrate on any known low-dimensional subspace.

Definition 1 (Computational indistinguishability)**.** *Let $\mathcal{A}$ be any PPT adversary with access to all public auxiliary information Z (including protocol transcripts and M_{pub}). For any two inputs $E_1, E_2 \in \mathbb{R}^{n\times d}$ with corresponding masked versions $\hat{E}_1, \hat{E}_2$, the distinguishing advantage is*

$$\left|\, \Pr[\mathcal{A}(\hat{E}_1, Z) = 1] - \Pr[\mathcal{A}(\hat{E}_2, Z) = 1] \,\right| \le \varepsilon(\kappa, q), \qquad (2)$$

where κ is the security parameter of PRG used to derive masks and q is the number of masked samples observed. If ε is negligible in κ for any polynomially bounded q, the scheme is privacy-preserving.

Instantiation in Talaria. Within the CVM, masks are derived from a PRG keyed by a secret seed of length κ; conditioned on Z, the mask distribution is computationally indistinguishable from uniform over the ring (Appx. B.2). Because a fresh mask is used per token, the cloud cannot correlate observations across steps to reduce uncertainty.

Quantitative information-theoretic bound. Beyond computational security, the masking admits an information-theoretic guarantee that depends only on signal magnitude versus mask range.

Theorem 1 (Information-theoretic bound)**.** *Let $e = \mathrm{vec}(E) \in \mathbb{R}^N$ with $N = nd$. Suppose the mask has i.i.d. coordinates $m_i \sim \mathrm{Unif}[-\lambda/2, \lambda/2]$ and $\hat{e}_b = e_b + m$ for $b \in \{1, 2\}$. Then the optimal adversary (with unbounded computation) that observes $\hat{e}$ satisfies*

$$\Pr[\textit{correctly distinguish } E_1 \textit{ from } E_2 \mid \hat{e}] \le \tfrac{1}{2} + \tfrac{1}{2} \min\left\{ \frac{\|e_1 - e_2\|_1}{\lambda}, 1 \right\}. \qquad (3)$$

Implications for parameterization. The distinguishing advantage scales as (effective) signal over mask

range. Choosing λ so that $\|e_1 - e_2\|_1/\lambda$ is negligible for inputs drawn from the task distribution makes the optimal success probability arbitrarily close to $1/2$. Practically, one can upper-bound $\|e_1 - e_2\|_1$ via sensitivity estimates of the embedding map and set λ with a safety margin to cover worst-case layers/tokens (Appx. B.3).

5.2 Model Confidentiality

Goal. Prevent a curious client from recovering proprietary weights $W \in \mathbb{R}^{d\times d}$.

Mathematical isolation. The client observes only the linearly transformed $R_{\text{pub}} = M_{\text{pub}}W$ with $M_{\text{pub}} \in \mathbb{R}^{m\times d}$ and $m < d$. Since $\text{rank}(M_{\text{pub}}) \leq m < d$, the right null space is nontrivial, $\dim(\ker(M_{\text{pub}})) \geq d - m > 0$, and

$$\{W' \colon M_{\text{pub}}W' = R_{\text{pub}}\} = W_0 + \{Z : \text{ each column of } Z \in \ker(M_{\text{pub}})\}. \tag{4}$$

Thus the feasible set is an affine subspace of positive dimension (infinitely many candidates), making unique recovery of W information-theoretically impossible. Here $\ker(M) = \{v \in \mathbb{R}^d : Mv = 0\}$.

System-level hardening. We enforce remote attestation of the CVM image and restrict its capabilities to the minimal Talaria logic: no debug hooks, no general linear-algebra toolkits exposing arbitrary solves, and no interfaces that could be repurposed to probe R_{pub} beyond its intended use. Attestation binds code and configuration to a hash verified by the cloud, preventing a curious client from modifying the CVM to mount offline model-recovery oracles. See Appx. B.4 for formal proofs.

6 Evaluations

6.1 Evaluation Setup

All evaluations are conducted on a cloud node with 8 Nvidia H800 with 80GB of memory, 10 AMD EPYC processor cores, and 6400 GB of system memory. We enable security features such as AMD SEV-SNP (AMD, 2023) and NVIDIA GPU CC (Nvidia, 2023) for evaluations that require confidential computing. Detailed evaluation setup (datasets, metrics, etc.) can be found in Appx. C.1.

Datasets and models. We evaluate Talaria on four diverse datasets: Midjourney prompts (AI, 2024), WikiText-2 (Merity et al., 2017), Patient-notes (Ha et al., 2022), and GPT-OSS20B-samples (Morris, 2025). To simulate our threat model, we use 20% of each dataset as private client prompts and allow the remaining 80% to be used for training attack models. Our experiments leverage models from the Llama3 (1B, 8B, 70B) (Meta, 2025) and Qwen3 (0.6B, 8B, 32B) (Alibaba, 2025) families. By default, we use Qwen3-8B for all evaluations unless stated otherwise.

Metrics. For protection evaluation, we use Token Reconstruction Accuracy (TRA) and Cosine Semantic Similarity (CSS). The higher the TRA and CSS, the higher the similarity between the two texts, which can reflect the higher accuracy of the token reconstruction attack. For efficiency evaluation, we use end-to-end latency and Time to First Token (TTFT).

Attacks. We validate our defense against three SOTA reconstruction attacks targeting intermediate data: TokenInfer (Luo et al., 2025) (classification-based), TokenInv (Qu et al., 2025) (optimization-based), and SentenceInv (Gu et al., 2023) (generation-based). By default, we use TokenInfer for most evaluations. According to TokenInfer, the reconstruction performance decreases as the number of layers increases. Therefore, in order to maximize the attack effect, we use the outputs of the first attention layer as the attack object.

Baselines. We first use *No Protection* and *Full Isolation* as the baseline, where the former is normal unprotected cloud inference and the latter is where each client loads a copy of the complete LLM into CVM for inference. Then we also use two SOTA defense mechanisms as baselines: SPD (Gim et al., 2024), Cape (Wu et al., 2025).

6.2 Protection Evaluations

We assess data privacy protection along two dimensions: input prompts and output responses. To facilitate comparison of metrics, we set the maximum number of output tokens per request to 100. Additional evaluations can be found at Appx. C.2.

Comparison with baselines. We evaluate the protection performance of Talaria against several baselines, with results summarized in Tbl. 2. *No Protection* highlights the severe privacy risk, as an adversary can infer both prompts and responses with near-perfect accuracy (avg. TRA > 97.5% and avg. CSS > 0.95). This demonstrates that without protecting the intermediate data, the cloud can easily infer the private prompt and response. While existing methods like Cape offer limited mitigation, they remain largely vulnerable. Another baseline, SPD, effectively protects prompts by processing the prefill phase within the CVM. However, this approach has a critical flaw: it exposes confidential model

Attack Target	Methods	Midjourney		WikiText2		Patient		GPT-samples	
Prompt	*No Protection*	**96.87%**	0.9352	**97.21%**	0.9444	**98.64%**	0.9834	**97.53%**	0.9421
	Cape	**85.25%**	0.8824	**84.55%**	0.8984	**84.12%**	0.9023	**85.89%**	0.8334
	Talaria	**1.24%**	0.4876	**0.47%**	0.4299	**1.53%**	0.5012	**2.11%**	0.4535
Response	*No Protection*	**97.53%**	0.9326	**97.54%**	0.9401	**97.28%**	0.9885	**97.68%**	0.9456
	Cape	**85.53%**	0.8921	**84.65%**	0.8906	**84.64%**	0.9005	**86.10%**	0.8389
	SPD	**97.16%**	0.9344	**96.87%**	0.9375	**98.76%**	0.9903	**97.24%**	0.9413
	Talaria	**1.08%**	0.4724	**0.38%**	0.4242	**1.58%**	0.4980	**1.97%**	0.4463

Table 2: Effectiveness of different methods in defending against TokenInfer attack. Bold: TRA. Gray: CSS. Shaded cells highlight the best protection.

weights to the client. Furthermore, SPD leaves the generated response completely unprotected, as intermediate data during the decoding phase is transmitted in plaintext, leading to similarly high attack success rates (avg. TRA: 97.51%, avg. CSS: 0.9509). In stark contrast, Talaria provides robust protection across the board. It slashes the average TRA for prompt inference from 97.56% to just 1.34% and for response inference from 97.51% to 1.25%. Similarly, CSS plummets from over 0.95 to under 0.47. These results demonstrate that Talaria effectively defends inference attacks, rendering the recovery of private tokens nearly impossible.

Different LLMs. To assess the generalizability of our approach, we evaluate Talaria across a diverse set of models with varying architectures and parameter scales. As shown in Tbl. 3, Talaria consistently delivers strong protection for both client prompts and model responses, demonstrating its robustness and broad applicability.

Attack Target	Models		TRA	CSS
Prompt	Qwen3	0.6B	1.56%	0.4573
		8B	1.24%	0.4876
		32B	1.43%	0.4753
	Llama3	1B	1.46%	0.4323
		8B	1.39%	0.4632
		70B	1.57%	0.4553
Response	Qwen3	0.6B	1.34%	0.4326
		8B	1.08%	0.4724
		32B	1.21%	0.4683
	Llama3	1B	0.89%	0.4521
		8B	1.15%	0.4646
		70B	0.78%	0.4328

Table 3: The effectiveness of Talaria in defending against TokenInfer attack under different model structures and parameters (Midjourney dataset).

Different Attacks. To rigorously assess its security, we evaluate Talaria against three SOTA attacks: TokenInfer, TokenInv, and SentenceInv. Against TokenInfer, the most potent of the three, Talaria proves exceptionally effective, reducing the average TRA to a mere 1.16% and the average CSS to 0.48. Our approach is similarly successful in thwarting TokenInv. Furthermore, Talaria effectively defends against SentenceInv, a prompt-only reconstruction attack for which TRA is not an applicable metric. These results demonstrate that Talaria provides comprehensive protection against a diverse range of sophisticated attacks.

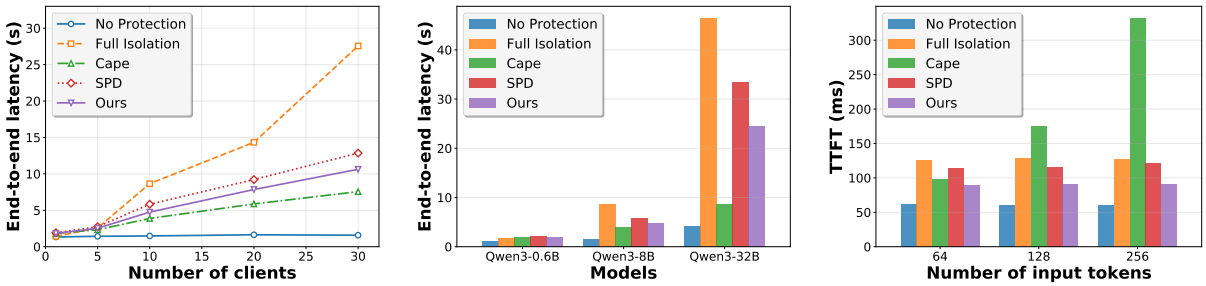
Attack Target	Attacks	TRA	CSS
Prompt	TokenInfer	1.24%	0.4876
	TokenInv	0.45%	0.3893
	SentenceInv	-	0.4435
Response	TokenInfer	1.08%	0.4724
	TokenInv	0.56%	0.3954
	SentenceInv	-	-

Table 4: The effectiveness of Talaria in defending against different attacks (Midjourney dataset).

6.3 Efficiency Evaluations

Our evaluation setup includes 1 to 32 clients, each of whom randomly selects 10 prompts from the Midjourney dataset. For ease of comparison, we fix the number of output tokens to 100. Finally, we calculate the average efficiency metrics.

Number of clients. *Full Isolation* requires a separate LLM instance for each CVM. When the number of concurrent clients increases, the operating system and GPU driver must perform expensive context switches. Therefore, as shown in Fig. 5a, the end-to-end latency curve of *Full Isolation* rises sharply with the number of clients. Compared with the SPD mechanism, Talaria achieves better scalability because the SPD mechanism needs to infer multiple virtual prompts simultaneously to protect client privacy, which increases the memory usage.



(a) End-to-end latency with varying number of clients.

(b) End-to-end latency with different model parameters. (10 clients)

(c) TTFT with varying number of input tokens. (10 clients)

Figure 5: Efficiency Evaluation.

Compared to these two CVM-based approaches, Talaria demonstrates superior scalability by maintaining lower end-to-end latency as the number of concurrent clients grows.

Size of model parameters. As model size grows, all methods exhibit increased end-to-end latency (Fig. 5b). Compared with *Full Isolation* and SPD, Talaria is less affected by the expansion of parameter scale. Intuitively, Talaria confines weight-free operations to the CVM and outsources weight-dependent multiplications efficiently, avoiding the per-client model duplication and heavy context switching costs that penalize *Full Isolation*, as well as the additional overheads present in SPD. Talaria maintains robust token generation speed under larger models, delivering lower latency growth than *Full Isolation* and SPD as parameter scale expands.

Number of input tokens. Fig. 5c shows the change of TTFT as the number of input tokens increases (64→256). TTFT increases for all methods, but the growth is particularly steep for Cape, whose core routine computes contextual logits for every input token. Unlike *No Protection*, the three CVM-based designs (*Full Isolation*, SPD, and Talaria) cannot amortize prefill via multi-client batching because each request must run inside an isolated CVM, which inherently raises TTFT as inputs lengthen. Even under this constraint, Talaria consistently delivers the lowest TTFT among secure baselines and exhibits the mildest growth with respect to input length, yielding a faster first token–especially for long prompts. Talaria preserves strong isolation while keeping prefill latency close to the practical minimum for CVM-based schemes, thus offering the fastest initial response among secure methods as input tokens scale.

6.4 Output Invariance Evaluations.

We assess whether Talaria preserves the exact functional behavior of the LLM by generating paired responses with and without Talaria under an identical decoding setup (same model, tokenizer, fixed random seed, and deterministic sampler). We then compare the paired outputs using TRA and CSS, which respectively measure token-level identity and semantic equivalence. As detailed in Tbl. 5, Talaria achieves TRA=100% and CSS=1.00 for each dataset, indicating bit-for-bit identical responses relative to *No Protection*. These results empirically validate the losslessness of ReMO and local restoration pipeline and confirm that Talaria preserves output quality without perturbation and then satisfies the performance of the "impossible trinity" by guaranteeing output invariance while adding privacy protection. Additional invariance evaluations can be found in Appx. C.3.

Metrics	Midjourney	WikiText2	Patient	GPT-samples
TRA	100%	100%	100%	100%
CSS	1.00	1.00	1.00	1.00

Table 5: The changes in response before and after using Talaria.

7 Conclusion

We introduced Talaria, a confidential inference framework for cloud-based LLMs by partitioning the inference pipeline and leveraging CVMs and the ReMO protocol. Empirically, Talaria cuts reconstruction success to near-negligible levels across models, datasets, and attacks, while preserving bit-for-bit outputs and competitive latency. To the best of our knowledge, the first to keep both prompts and responses confidentiality while balancing model privacy, performance, and efficiency.

Limitations

The limitations of this work mainly include the following two aspects:

Threat Scenario. We adopt the standard *honest-but-curious* setting: the cloud provider executes the protocol correctly but may analyze observed artifacts to infer private information. This assumption is reasonable for cloud LLM services because ① providers are economically and reputationally incentivized to follow deployed protocols, making arbitrary deviation unlikely in practice; ② confidential-computing primitives (e.g., CVM attestation and capability scoping) already constrain what either party can do while still enabling routine operations; and ③ it aligns with the dominant threat model used in prior confidential-inference work (Gim et al., 2024; Wu et al., 2025) and in our system design. Under this setting, our guarantees are testable, and the resulting system remains practical to deploy at scale.

We do not defend against a fully malicious or compromised cloud provider that can arbitrarily deviate from the protocol (e.g., returning malformed messages or bypassing attested images). Defending against such adversaries would require stronger mechanisms (e.g., verifiable execution, runtime auditing, or side-channel–resilient defenses), which we leave to future work.

Partial model protection. Talaria does not attempt to hide the entire model implementation from the client. Instead, it prioritizes *weight confidentiality*: proprietary weight parameters remain on the cloud side, while weight-free operators (e.g., normalization, attention scoring/softmax, residual connections, and sampling) are executed within the CVM. This choice is deliberate and reasonable. In contemporary LLM stacks, the structure of these operators is largely standardized and often public, whereas the learned weight matrices encode the provider's core intellectual property and present the primary model-extraction risk. By keeping all weight-dependent computations on the cloud provider's hardware and exposing only weight-free control flow, the client may observe the computation pattern but not the parameters.

Moreover, visibility into non-weight operators does not enable parameter recovery under our protocol. The client observes at most a fixed, underdetermined sketch $R_{\text{pub}} = M_{\text{pub}}W$ with $m < d$, leaving a non-trivial right kernel and thus infinitely many W consistent with the observation; without additional side information, the true weights are non-identifiable. In contrast, fully obscuring every non-weight detail would require heavy obfuscation or cryptography, undermining our efficiency and output-invariance goals. Prioritizing weight confidentiality therefore strikes a practical balance among privacy, performance, and scalability, yielding bit-for-bit identical outputs while protecting the provider's proprietary parameters.

References

Succinctly AI. 2024. Datasets: succinctly/midjourney-prompts. Accessed: 2025-08-04.

Alibaba. 2025. Qwen 3. Accessed: 2025-08-04.

AMD. 2023. Amd secure encrypted virtualization (sev). Accessed: 2025-08-18.

AWS. 2025. Aws confidential computing. Accessed: 2025-09-21.

Azure. 2025. Azure confidential gpu options. Accessed: 2025-09-21.

Christopher Bissex. 2025. Ftc launches inquiry into ai chatbots acting as companions. Accessed: 2025-09-21.

Giles Bruce. 2023. Iowa health system warns against using chatgpt to draft patient letters. Accessed: 2025-07-02.

Brian Bushard. 2023. Workers' chatgpt use restricted at more banks—including goldman, citigroup. Accessed: 2025-07-02.

Yu Chen, Tingxin Li, Huiming Liu, and Yang Yu. 2023. Hide and seek (has): A lightweight framework for prompt privacy protection. *CoRR*, abs/2309.03057.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit S. Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, Luke Marris, Sam Petulla, Colin Gaffney, Asaf Aharoni, Nathan Lintz, Tiago Cardal Pais, Henrik Jacobsson, Idan Szpektor, Nan-Jiang Jiang, and 81 others. 2025. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *CoRR*, abs/2507.06261.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

Cynthia Dwork. 2006. Differential privacy. In *Automata, Languages and Programming, 33rd International Colloquium, ICALP 2006, Venice, Italy, July 10-14, 2006, Proceedings, Part II*, volume 4052 of *Lecture Notes in Computer Science*, pages 1–12. Springer.

Paul Farrell. 2023. Jpmorgan restricts chatgpt usage for its 250k staff over fears it could steal sensitive banking secrets. Accessed: 2025-07-02.

Edoardo Federici. 2022. sentence-bert-base, sentence-transformer for italian.

In Gim, Caihua Li, and Lin Zhong. 2024. Confidential prompting: Protecting user prompts from cloud LLM providers. *CoRR*, abs/2409.19134.

Google. 2025. Google cloud confidential computing. Accessed: 2025-09-21.

Kang Gu, Ehsanul Kabir, Neha Ramsurrun, Soroush Vosoughi, and Shagufta Mehnaz. 2023. Towards sentence level inference attack against pre-trained language models. *Proc. Priv. Enhancing Technol.*, 2023(3):62–78.

Le An Ha, Maggie, Ryan Holbrook, and Victoria Yaneva. 2022. Nbme - score clinical patient notes. Accessed: 2025-08-18.

Lucjan Hanzlik, Yang Zhang, Kathrin Grosse, Ahmed Salem, Maximilian Augustin, Michael Backes, and Mario Fritz. 2021. Mlcapsule: Guarded offline deployment of machine learning as a service. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2021, virtual, June 19-25, 2021*, pages 3300–3309. Computer Vision Foundation / IEEE.

Junyuan Hong, Jiachen T. Wang, Chenhui Zhang, Zhangheng Li, Bo Li, and Zhangyang Wang. 2024. DP-OPT: make large language model your privacy-preserving prompt engineer. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Zhicong Huang, Wen-jie Lu, Cheng Hong, and Jiansheng Ding. 2022. Cheetah: Lean and fast secure two-party deep neural network inference. In *31st USENIX Security Symposium, USENIX Security 2022, Boston, MA, USA, August 10-12, 2022*, pages 809–826. USENIX Association.

Taegyeong Lee, Zhiqi Lin, Saumay Pushp, Caihua Li, Yunxin Liu, Youngki Lee, Fengyuan Xu, Chenren Xu, Lintao Zhang, and Junehwa Song. 2019. Occlumency: Privacy-preserving remote deep-learning inference using SGX. In *The 25th Annual International Conference on Mobile Computing and Networking, MobiCom 2019, Los Cabos, Mexico, October 21-25, 2019*, pages 46:1–46:17. ACM.

Sam Lin, Wenyue Hua, Zhenting Wang, Mingyu Jin, Lizhou Fan, and Yongfeng Zhang. 2025a. Emojiprompt: Generative prompt obfuscation for privacy-preserving communication with cloud-based llms. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL 2025 - Volume 1: Long Papers, Albuquerque, New Mexico, USA, April 29 - May 4, 2025*, pages 12342–12361. Association for Computational Linguistics.

Yu Lin, Ruining Yang, Yunlong Mao, Qizhi Zhang, Jue Hong, Quanwei Cai, Ye Wu, Huiqi Liu, Zhiyu Chen, Bing Duan, and Sheng Zhong. 2025b. Obfuslm: Privacy-preserving language model service against embedding inversion attacks. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2025, Vienna, Austria, July 27 - August 1, 2025*, pages 1160–1174. Association for Computational Linguistics.

Xuanqi Liu and Zhuotao Liu. 2023. Llms can understand encrypted prompt: Towards privacy-computing friendly transformers. *CoRR*, abs/2305.18396.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Jinglong Luo, Guanzhong Chen, Yehong Zhang, Shiyu Liu, Hui Wang, Yue Yu, Xun Zhou, Yuan Qi, and Zenglin Xu. 2024. Centaur: Bridging the impossible trinity of privacy, efficiency, and performance in privacy-preserving transformer inference. *CoRR*, abs/2412.10652.

Xinjian Luo, Ting Yu, and Xiaokui Xiao. 2025. Prompt inference attack on distributed large language model inference frameworks. *CoRR*, abs/2503.09291.

LégisQuébec. 2021. Act respecting the protection of personal information in the private sector. Accessed: 2025-09-21.

Laura Martinez. 2024. Advancing security for large language models with nvidia gpus and edgeless systems. Accessed: 2025-09-21.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. 2017. Pointer sentinel mixture models. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.

Meta. 2025. Llama 3. Accessed: 2025-08-04.

Jack Morris. 2025. jxm/gpt-oss20b-samples. Accessed: 2025-08-18.

Nvidia. 2023. Nvidia confidential computing. Accessed: 2025-08-18.

OpenAI. 2025a. Gpt-5 system card. Accessed: 2025-08-18.

OpenAI. 2025b. Gpt-oss-120b & gpt-oss-20b model card. *Preprint*, arXiv:2508.10925.

Elvira Pollina and Alvise Armellini. 2024. Italy fines openai over chatgpt privacy rules breach. Accessed: 2025-09-21.

Wenjie Qu, Yuguang Zhou, Yongji Wu, Tingsong Xiao, Binhang Yuan, Yiming Li, and Jiaheng Zhang. 2025. Prompt inversion attack against collaborative inference of large language models. In *IEEE Symposium on Security and Privacy, SP 2025, San Francisco, CA, USA, May 12-15, 2025*, pages 1695–1712. IEEE.

Mark Russinovich. 2024. Azure ai confidential inferencing: Technical deep-dive. Accessed: 2025-09-21.

Fabian Schwarz and Christian Rossow. 2024. 00seven - re-enabling virtual machine forensics: Introspecting confidential vms using privileged in-vm agents. In *33rd USENIX Security Symposium, USENIX Security 2024, Philadelphia, PA, USA, August 14-16, 2024*. USENIX Association.

Michael Schwarz, Samuel Weiser, Daniel Gruss, Clémentine Maurice, and Stefan Mangard. 2020. Malware guard extension: abusing intel SGX to conceal cache attacks. *Cybersecur.*, 3(1):2.

Sagar Sharma and Keke Chen. 2021. Confidential machine learning on untrusted platforms: a survey. *Cybersecur.*, 4(1):30.

Qwen Team. 2025. Qwen3 technical report. *Preprint*, arXiv:2505.09388.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008.

Ben Wolford. 2025. Data protection impact assessment (dpia). Accessed: 2025-09-21.

Haoqi Wu, Wei Dai, Li Wang, and Qiang Yan. 2025. Cape: Context-aware prompt perturbation mechanism with differential privacy. *CoRR*, abs/2505.05922.

Tong Wu, Ashwinee Panda, Jiachen T. Wang, and Prateek Mittal. 2024. Privacy-preserving in-context learning for large language models. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Kaiping Xue, Weikeng Chen, Wei Li, Jianan Hong, and Peilin Hong. 2018. Combining data owner-side and cloud-side access control for encrypted cloud storage. *IEEE Trans. Inf. Forensics Secur.*, 13(8):2062–2074.

Ziqian Zeng, Jianwei Wang, Zhengdong Lu, Huiping Zhuang, and Cen Chen. 2024. Privacyrestore: Privacy-preserving inference in large language models via privacy removal and restoration. *CoRR*, abs/2406.01394.

Biao Zhang and Rico Sennrich. 2019. Root mean square layer normalization. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 12360–12371.

A Notations

We summarize all the notations used in this paper in Tbl. 6.

B Additional Method Details

B.1 Pseudocode Implementation

The confidential inference process of Talaria is described as Alg. 1.

Algorithm 1 Pseudocode of Talaria.

Input: client prompt X.
Output: model response.
Phase 1: One-time setup
Initialize M_{pub} [CVM]
$R_{\text{pub}} \leftarrow M_{\text{pub}} W$ [Cloud]
Phase 2: Decoding
$X_{\text{new}} \leftarrow X$
while $X_{\text{new}} \neq \text{EOS}$ **do**
  Initialize M_{pvt} [CVM]
  $R \leftarrow M_{\text{pvt}} R_{\text{pub}}$ [CVM]
  $M \leftarrow M_{\text{pvt}} M_{\text{pub}}$ [CVM]
  $E_1 \leftarrow \textsc{Embedding}(X_{\text{new}})$ [CVM]
  for $i = 1$ **to** k **do**
    $\hat{E}_i \leftarrow \textsc{Mask}(M, E_i)$ [CVM]
    $\hat{O}_i \leftarrow \textsc{WeightedDec}(\hat{E}_i, W_i)$ [Cloud]
    $O_i \leftarrow \textsc{Recover}(\hat{O}_i, R)$ [CVM]
    $E_{i+1} \leftarrow \textsc{StructuralDec}(O_i)$ [CVM]
  end for
  $X_{\text{new}} \leftarrow \textsc{Sample}(E_k)$ [CVM]
end while

B.2 Theoretical Analysis of Def. 1

This section provides a theoretical foundation for Def. 1, which establishes the privacy guarantee of Talaria based on the standard cryptographic concept of computational indistinguishability. This principle is a cornerstone of modern cryptography, used to formally define the security of primitives such as pseudorandom generators and encryption schemes.

The indistinguishability game. The notion of computational indistinguishability is captured via a security game between a challenger and a Probabilistic Polynomial-Time (PPT) adversary $\mathcal{A}$:

Notation	Description
X	Client prompt (token sequence).
$E \in \mathbb{R}^{n \times d}$	Intermediate embedding matrix (sequence length n, hidden dimension d).
$W \in \mathbb{R}^{d \times d}$	Proprietary model weight matrix (linear projection).
$O = EW$	True weighted result of applying weights.
$\hat{E} = E + M$	Masked embedding after additive mask M.
$\hat{O} = \hat{E}W$	Masked weighted output computed by the cloud.
$M \in \mathbb{R}^{n \times d}$	Additive mask matrix sampled per token.
$M_{\text{pub}} \in \mathbb{R}^{m \times d}$	Public base mask ($m < d$) generated in setup.
$R_{\text{pub}} = M_{\text{pub}}W$	Pre-multiplied restoration pool returned to CVM.
$M_{\text{pvt}} \in \mathbb{R}^{n \times m}$	Ephemeral private mixing matrix, sampled per token.
$M = M_{\text{pvt}}M_{\text{pub}}$	Full mask derived from public and private parts.
$R = M_{\text{pvt}}R_{\text{pub}}$	Restoration term used to recover O.
$O = \hat{O} - R$	Local recovery of the true weighted result.
EOS	End-of-sequence token for decoding termination.
k	Number of decoding layers.
$\mathcal{A}$	Probabilistic polynomial-time adversary.
$\mathcal{D}_0, \mathcal{D}_1$	Two probability distributions in indistinguishability game.
Z	Public auxiliary information (e.g., protocol transcript, M_{pub}).
$\text{Adv}(\mathcal{A})$	Distinguishing advantage of adversary $\mathcal{A}$.
$\varepsilon(\kappa, q)$	Negligible bound in computational indistinguishability, depends on security parameter κ and query count q.
κ	Security parameter (PRG seed length).
q	Number of masked samples observed by adversary.
$\lVert \cdot \rVert_1, \lVert \cdot \rVert_\infty$	ℓ_1 norm, ℓ_∞ norm (used in bounds).
$\ker(M)$	Kernel (null space) of matrix M, i.e., $\{v \in \mathbb{R}^d : Mv = 0\}$.
D_{TV}	Total variation distance between two distributions.
λ	Range parameter of uniform mask distribution.
$e = \text{vec}(E) \in \mathbb{R}^N$	Vectorized embedding, $N = nd$.
TRA	Token Reconstruction Accuracy (attack metric).
CSS	Cosine Semantic Similarity (attack metric).
TTFT	Time to First Token (efficiency metric).

Table 6: Summary of notations used in the paper.

1. **Setup:** The game is defined over two probability distributions, $\mathcal{D}_0$ and $\mathcal{D}_1$. The adversary, $\mathcal{A}$, is aware of these two distributions.

2. **Challenge:** The challenger secretly flips a fair coin, choosing a bit $b \in \{0, 1\}$. It then samples a value v from the distribution $\mathcal{D}_b$ (i.e., $v \sim \mathcal{D}_b$) and sends v to the adversary.

3. **Adversary's Goal:** Upon receiving v, the adversary $\mathcal{A}$ must guess the original bit b. It outputs a guess, b'.

The adversary wins the game if its guess is correct ($b' = b$). An adversary with no information can do no better than guessing randomly, succeeding with a probability of $1/2$. The **advantage** of an adversary is defined as how much better than random guessing it can perform.

$$\text{Adv}(\mathcal{A}) = \left| \Pr[b' = b] - \frac{1}{2} \right|$$

The two distributions, $\mathcal{D}_0$ and $\mathcal{D}_1$, are said to be computationally indistinguishable if the advantage of any PPT adversary is *negligible*–that is, smaller than any inverse polynomial in the security parameter.

The formula presented in Def. 1 is an equivalent and widely used formulation of this concept. Let $\mathcal{A}(v)$ be the output of the adversary on input v. We can normalize the adversary's output to be either 0 or 1. Let's say $\mathcal{A}$ outputs 1 when it guesses the sample came from $\mathcal{D}_1$, and 0 otherwise. The probability of the adversary guessing correctly can be expanded as:

$$\begin{aligned}
\Pr[b' = b] &= \Pr[b' = 1 \mid b = 1] \Pr[b = 1] \\
&\quad + \Pr[b' = 0 \mid b = 0] \Pr[b = 0] \\
&= \tfrac{1}{2}\Big(\Pr[\mathcal{A}(v, Z) = 1 \mid v \sim \mathcal{D}_1] \\
&\quad + \Pr[\mathcal{A}(v, Z) = 0 \mid v \sim \mathcal{D}_0] \Big) \\
&= \tfrac{1}{2}\Big(\Pr_{v \sim \mathcal{D}_1}[\mathcal{A}(v, Z) = 1] \\
&\quad + \big(1 - \Pr_{v \sim \mathcal{D}_0}[\mathcal{A}(v, Z) = 1]\big) \Big)
\end{aligned}$$

Now, we can substitute this into the advantage for-

mula:

$$\begin{aligned}\mathrm{Adv}(\mathcal{A}) &= \Bigg|\tfrac{1}{2}\Big(\Pr_{v\sim\mathcal{D}_1}[\mathcal{A}(v,Z)=1] \\ &\qquad + 1 - \Pr_{v\sim\mathcal{D}_0}[\mathcal{A}(v,Z)=1]\Big) - \tfrac{1}{2}\Bigg| \\ &= \Bigg|\tfrac{1}{2}\Pr_{v\sim\mathcal{D}_1}[\mathcal{A}(v,Z)=1] \\ &\qquad - \tfrac{1}{2}\Pr_{v\sim\mathcal{D}_0}[\mathcal{A}(v,Z)=1]\Bigg| \\ &= \tfrac{1}{2}\Bigg|\Pr_{v\sim\mathcal{D}_1}[\mathcal{A}(v,Z)=1] \\ &\qquad - \Pr_{v\sim\mathcal{D}_0}[\mathcal{A}(v,Z)=1]\Bigg|\end{aligned}$$

The formula in Def. 1 is:

$$|\Pr[\mathcal{A}(\hat{E}_1,Z)=1] - \Pr[\mathcal{A}(\hat{E}_2,Z)=1]| < \epsilon$$

Here, $\hat{E}_1$ is a sample from the distribution of masked embeddings derived from input E_1, and $\hat{E}_2$ is a sample from the distribution derived from input E_2. This expression is exactly twice the adversary's advantage. Since a negligible function multiplied by a constant remains negligible, this formulation is cryptographically equivalent. It directly measures the difference in the adversary's behavior when presented with samples from the two worlds, which is the essence of distinguishability.

Instantiation for Talaria. In our setting, the two distributions correspond to masked embeddings:

- $\mathcal{D}_1$: $\hat{E}_1 = \mathsf{Mask}(E_1)$,
- $\mathcal{D}_2$: $\hat{E}_2 = \mathsf{Mask}(E_2)$,

where $\mathsf{Mask}(\cdot)$ is a randomized masking transformation applied to the intermediate embedding matrix E. To ensure soundness, the mask must satisfy two conditions:

1. **Freshness:** A fresh independent mask is sampled for each instance (e.g., per token).
2. **Full-rank randomness:** The distribution of masks must not confine noise to a publicly known low-dimensional subspace. Otherwise, the adversary could project onto the orthogonal complement and recover information about E exactly. To prevent this leakage, the mask must span the full ambient space, e.g., via additive pads sampled from a pseudorandom generator (PRG) over a fixed modulus or via two-sided linear masking with secret invertible matrices.

In all cases, masking is performed element-wise over a fixed ring (e.g., $\mathbb{Z}_{2^k}$ after deterministic quantization of E), so that arithmetic and indistinguishability are defined over a finite domain.

Security guarantee. Formally, Def. 1 states that for all PPT adversaries $\mathcal{A}$, for any two inputs E_1, E_2, and for any public auxiliary information Z (including protocol transcripts and public parameters),

$$\big|\Pr[\mathcal{A}(\hat{E}_1,Z)=1] - \Pr[\mathcal{A}(\hat{E}_2,Z)=1]\big| \le \varepsilon(\kappa,q)$$

where κ is the security parameter (e.g., PRG key length), q is the number of masked samples observed by $\mathcal{A}$ (including adaptive/chosen-input interactions), and $\varepsilon(\kappa,q)$ is negligible in κ for any polynomially bounded q.

This guarantee holds under the assumption that, conditioned on Z, the mask distribution is computationally indistinguishable from uniform over the chosen domain, i.e., masks are generated by a secure PRG keyed within the client's trusted environment.

Interpretation. Intuitively, the masking acts like a one-time pad over the chosen ring: once an embedding E is combined with fresh pseudorandom noise that spans the full space, the resulting $\hat{E}$ leaks negligible information about E. Therefore, even a PPT cloud adversary observing $\hat{E}$ together with Z cannot distinguish whether the input embedding originated from E_1 or E_2 with non-negligible advantage. This formalizes the confidentiality guarantee of Talaria.

B.3 Theoretical Analysis of Thm. 1

This section provides a formal derivation for the information-theoretic security bound presented in Thm. 1. The theorem quantifies the maximum probability with which an adversary can distinguish between two inputs, E_1 and E_2, given only their masked versions. The analysis is "information-theoretic" because the resulting bound does not depend on the adversary's computational power, but only on the statistical properties of the data and the masking mechanism.

The core of the proof relies on bounding the **total variation distance** (D_{TV}) between the probability distributions of the two possible masked outputs.

The adversary's optimal advantage in any distinguishing task is directly related to this statistical distance.

Preliminaries. We consider a game between a challenger and an adversary $\mathcal{A}$. The challenger chooses one of two inputs, E_1 or E_2, with equal probability ($\frac{1}{2}$). It then applies the random masking mechanism to produce a masked output $\hat{E} = E_b + M$, where $b \in \{1, 2\}$ and M is a random mask. The adversary observes $\hat{E}$ and must guess the original bit b.

The optimal adversary's probability of guessing correctly is given by:

$$\Pr[\mathcal{A} \text{ succeeds}] = \frac{1}{2} + \frac{1}{2} D_{TV}(P_{\hat{E}_1}, P_{\hat{E}_2})$$

where $P_{\hat{E}_1}$ and $P_{\hat{E}_2}$ are the probability distributions of the masked outputs corresponding to inputs E_1 and E_2, respectively. The total variation distance for two continuous distributions with probability density functions (PDFs) $p_1(x)$ and $p_2(x)$ is defined as:

$$D_{TV}(P_1, P_2) = \frac{1}{2} \int_{-\infty}^{\infty} |p_1(x) - p_2(x)| \, dx$$

Our goal is to bound this distance for our additive masking scheme.

Derivation for a single dimension. To build intuition, let us first consider a single-dimensional case. Let the inputs be two scalars $e_1, e_2 \in \mathbb{R}$. The mask m is a random variable. For analytical clarity, we model the mask as being drawn from a continuous uniform distribution over the range $[-\lambda/2, \lambda/2]$. The PDF of the mask, $p_m(z)$, is:

$$p_m(z) = \begin{cases} \frac{1}{\lambda} & \text{if } z \in [-\lambda/2, \lambda/2] \\ 0 & \text{otherwise} \end{cases}$$

The adversary observes $\hat{e} = e_b + m$. The PDF of the observed value $\hat{e}$, conditioned on the original input being e_1 or e_2, is simply a shifted version of the mask's PDF:

$$p_1(x) = p_m(x - e_1) \;\text{ and }\; p_2(x) = p_m(x - e_2)$$

We can now compute the total variation distance:

$$\begin{aligned} &D_{TV}(P_{\hat{e}_1}, P_{\hat{e}_2}) \\ &= \frac{1}{2} \int_{-\infty}^{\infty} |p_m(x - e_1) - p_m(x - e_2)| \, dx \end{aligned}$$

The functions $p_m(x-e_1)$ and $p_m(x-e_2)$ represent two rectangular windows of height $1/\lambda$ and width λ, centered at e_1 and e_2. The integral computes half the area of the symmetric difference between these two windows.

The area of this symmetric difference is the sum of the areas of the two non-overlapping parts. Each of these parts has a width of $|e_1 - e_2|$ and a height of $1/\lambda$, provided that $|e_1 - e_2| \leq \lambda$ (i.e., the windows overlap). The total area of the symmetric difference is $2 \times |e_1 - e_2| \times \frac{1}{\lambda}$.

Substituting this into the D_{TV} formula:

$$\begin{aligned} D_{TV}(P_{\hat{e}_1}, P_{\hat{e}_2}) &= \min\left(\frac{1}{2}\left(2 \cdot \frac{|e_1 - e_2|}{\lambda}\right), 1\right) \\ &= \min\left(\frac{|e_1 - e_2|}{\lambda}, 1\right) \end{aligned}$$

Thus if $|e_1 - e_2| \leq \lambda$, the distance scales linearly; if $|e_1 - e_2| > \lambda$, the distributions are disjoint and $D_{TV} = 1$.

Generalization to multiple dimensions. Now let $E \in \mathbb{R}^{n \times d}$ and vectorize it as $\mathbf{e} \in \mathbb{R}^N$ with $N = nd$. Assume that the mask M has independent coordinates $m_i \sim \text{Unif}[-\lambda/2, \lambda/2]$. Then the masked outputs $\hat{\mathbf{e}}_1$ and $\hat{\mathbf{e}}_2$ are uniform over two N-dimensional boxes of width λ but shifted by $\delta = \mathbf{e}_1 - \mathbf{e}_2$. The total variation distance factorizes across coordinates:

$$D_{TV}(P_{\hat{\mathbf{e}}_1}, P_{\hat{\mathbf{e}}_2}) = 1 - \prod_{i=1}^{N} \max\left(1 - \tfrac{|\delta_i|}{\lambda}, 0\right)$$

Using the inequality $1 - \prod_i (1 - a_i) \leq \sum_i a_i$ for $a_i \in [0, 1]$, we obtain the upper bound:

$$D_{TV}(P_{\hat{\mathbf{e}}_1}, P_{\hat{\mathbf{e}}_2}) \leq \min\left(\frac{\|\mathbf{e}_1 - \mathbf{e}_2\|_1}{\lambda}, 1\right)$$

Thus, the adversary's distinguishing advantage is controlled by the ℓ_1 distance between the inputs relative to the masking range.

Final bound. Substituting into the adversary's success probability yields:

$$\begin{aligned} &\Pr[\mathcal{A} \text{ correctly distinguishes } E_1 \text{ from } E_2 \mid \hat{E}] \\ &\leq \tfrac{1}{2} + \tfrac{1}{2} \min\left(\frac{\|E_1 - E_2\|_1}{\lambda}, 1\right) \end{aligned}$$

This establishes the bound stated in Thm. 1. The result shows that the privacy guarantee is strengthened by increasing the masking range λ, and that the adversary's advantage over random guessing vanishes as λ grows.

Remark on assumptions. The above derivation holds under the assumption that the mask entries

are independent uniform random variables. If the masking distribution is generated differently (e.g., via structured or low-rank randomness), the exact form of the bound must be adapted to that distribution. In that case, the same total variation framework applies, but one must analyze the overlap of the induced distributions accordingly.

B.4 Theoretical Analysis of Model Confidentiality

This section provides a formal analysis of the mathematical guarantees that underpin Talaria's model confidentiality, as introduced in the main paper. The core principle is to prevent a curious client, who controls a CVM instance, from reverse-engineering the cloud's proprietary model weights, W. This protection is achieved by structuring the interaction between the CVM and the public GPU as an underdetermined system of linear equations, making the unique recovery of W information-theoretically impossible.

Problem formulation. During the one-time setup phase, the CVM receives a transformed version of the weight matrix:

$$R_{\text{pub}} = M_{\text{pub}} W$$

where $M_{\text{pub}} \in \mathbb{R}^{m \times d}$ is a publicly generated base mask, $R_{\text{pub}} \in \mathbb{R}^{m \times d}$ is the published sketch, and $W \in \mathbb{R}^{d \times d}$ are the secret model weights. From the client's perspective, M_{pub} and R_{pub} are known, while W is unknown.

Analysis as a system of linear equations. The matrix equation above can be decomposed into d independent systems of linear equations, one for each column of W. Let w_j and r_j be the j-th columns of W and R_{pub}, respectively. For each column $j \in \{1, \ldots, d\}$, we have the following system:

$$M_{\text{pub}} w_j = r_j$$

In this system, the matrix M_{pub} is the coefficient matrix, w_j is the vector of d unknown variables (the weights in that column), and r_j is the known result vector.

Underdetermined constraint. The protocol enforces $m < d$, so each system $M_{\text{pub}} w_j = r_j$ is underdetermined. Since $\text{rank}(M_{\text{pub}}) \leq m < d$, the null space is non-trivial, with

$$\dim(\ker(M_{\text{pub}})) \geq d - m > 0$$

The notation $\ker(M_{\text{pub}})$ denotes the kernel (or null space) of a matrix M_{pub}, defined as

$$\ker(M_{\text{pub}}) = \{x \in \mathbb{R}^d : M_{\text{pub}} x = 0\}$$

It represents the set of directions that are invisible to M_{pub}. If $\ker(M_{\text{pub}})$ is non-trivial, then any solution to $M_{\text{pub}} x = b$ can be shifted by an element of $\ker(M_{\text{pub}})$ to produce another valid solution, yielding infinitely many possibilities.

Solution space. Each solution set has the affine form:

$$w_j = w_p + w_h$$

where w_p is a particular solution (e.g., $w_p = M_{\text{pub}}^{+} r_j$ with the Moore–Penrose pseudoinverse), and $w_h \in \ker(M_{\text{pub}})$. Thus, infinitely many candidate weight vectors are consistent with the observed data.

Proposition (Non-identifiability under no priors). Let $\mathcal{S}_j = \{w_p + z : z \in \ker(M_{\text{pub}})\}$ be the solution set for column w_j. If no auxiliary constraints on w_j are available (i.e., the admissible set is $\mathbb{R}^d$), then $|\mathcal{S}_j| = \infty$ and the true weight vector is not uniquely identifiable. Uniqueness may arise only if additional side information restricts $\mathcal{S}_j$ to a singleton (e.g., sparsity, low-rank priors, quantization).

Protocol constraints. To preserve confidentiality in practice, the following requirements are critical:

1. **Fixed sketch per matrix.** For each W, the cloud publishes a single $R_{\text{pub}} = M_{\text{pub}} W$ with fixed M_{pub}. Re-issuing with linearly independent M_{pub} is disallowed, since stacking multiple sketches could collapse the null space.

2. **CVM attestation and isolation.** Although the CVM performs unmasking internally, the client cannot read intermediate tensors $(E, O = EW)$; otherwise, by collecting enough input–output pairs, the client could solve $EW = O$ and reconstruct W. Talaria relies on cloud-based attestation and runtime isolation to prevent such leakage.

3. **Independent masking.** Distinct, untied weight matrices (e.g., W_Q, W_K, W_V, W_O) must use independent masks M_{pub} to avoid cross-module linkability.

4. **Domain specification.** The analysis assumes real-valued arithmetic. If weights are represented in finite rings (e.g., $\mathbb{Z}_{2^k}$), rank and null-space properties must be verified in that algebraic domain.

Conclusion. The dimensionality constraint $m < d$ guarantees that $M_{\text{pub}}W$ alone does not uniquely determine W, ensuring non-identifiability under standard assumptions. Model confidentiality thus holds provided that (i) no additional independent sketches are revealed, (ii) clients cannot access intermediate tensors, and (iii) no strong structural priors on W are externally available.

C Additional Evaluations

C.1 Detailed Evaluation Setup

C.1.1 Datasets

We use four representative datasets in our experiments:

- Midjourney prompts (AI, 2024) is a dataset containing 246,381 natural language prompts for text-to-image service.
- WikiText-2 (Merity et al., 2017) contains over 100 million tokens extracted from high-quality, curated Wikipedia articles.
- Patient-notes (Ha et al., 2022) is a dataset containing 40,000 patient note history portions.
- GPT-OSS20B-samples (Morris, 2025) dataset is a collection of 8,960,000 text samples generated from a version of OpenAI's GPT-OSS20B model (OpenAI, 2025b).

C.1.2 Metrics

For defense evaluation, we use Token Reconstruction Accuracy (TRA) and Cosine Semantic Similarity (CSS). TRA is formally defined as $TRA = \frac{1}{|\mathcal{T}^{tgt}|}\sum_{t_i \in \mathcal{T}^{tgt}} \mathbb{I}(\hat{t}_i = t_i)$ where $\mathbb{I}$ is the indicator function, $\mathcal{T}^{tgt}$ represents the set of target tokens, and $\hat{t}_i$ denotes the reconstructed token corresponding to t_i. CSS first extracts semantic embeddings of both reconstructed and ground truth prompts using Sentence-BERT (Federici, 2022) and then calculates their cosine similarity. Based on the prior work (Luo et al., 2025), a TRA greater than 40% is considered sufficient for plausible reconstruction. For efficiency evaluation, we use end-to-end latency and Time to First Token (TTFT). End-to-end latency measures the end-to-end time from when the client sends a prompt to when they receive the complete response. Time to first token measures the time from request arrival to the generation of the first output token.

C.1.3 Attacks

We validate our defense against three SOTA reconstruction attacks.

Classification-based attack. TokenInfer (Luo et al., 2025) reconstructs private tokens by intercepting the intermediate embeddings exchanged between devices in a distributed LLM framework. The core of the attack relies on the discovery that these intermediate embeddings form distinct clusters corresponding to their input tokens. The adversary trains a classification model to map the intercepted embeddings back to their original tokens. The attack model is a multi-layer perceptron consisting of six layers. Layer normalization and ReLU activation functions are used between consecutive layers. The training epochs for the attack model are 10.

Optimization-based attack. TokenInv (Qu et al., 2025) is divided into two stages: First, a constrained optimization phase is used to recover a continuous embedding that is a close approximation of the original prompt. Second, an adaptive discretization phase accurately converts this embedding back into discrete text tokens by using the activation data and semantic speculation from another LLM to improve accuracy and readability. The attack model is set as optimization learning rate to 0.1, constraint coefficient to 0.1, optimization iteration number to 1500, embedding-based candidate size to 10, and semantic-based candidate size to 10.

Generation-based attack. SentenceInv (Gu et al., 2023) trains a decoder model using a publicly available dataset from the same domain as a private target dataset. The decoder learns to reconstruct the original text from its sentence embeddings, which are generated by a pre-trained language model like BERT (Devlin et al., 2019). The adversary then uses this trained decoder on the private embeddings to reconstruct the original, sensitive sentences, revealing high-level semantic information. The attack model is an adversarial decoder, which inherits from the transformer architecture with 6 decoding layers and 8 heads. The decoder is trained for 100 epochs with cross-entropy loss, AdamW optimizer (Loshchilov and Hutter, 2019), and a learning rate of 1e-4.

C.1.4 Baselines

We use two SOTA representative defense mechanisms as baselines:

Attack Target	Models		Midjourney		WikiText2		Patient		GPT-samples	
Prompt	Qwen3	0.6B	**1.56%**	0.4573	**0.89%**	0.4453	**1.06%**	0.4835	**1.88%**	0.4735
		8B	**1.24%**	0.4876	**0.47%**	0.4299	**1.53%**	0.5012	**2.11%**	0.4535
		32B	**1.43%**	0.4753	**1.06%**	0.5022	**1.25%**	0.4875	**1.63%**	0.4925
	Llama3	1B	**1.46%**	0.4323	**0.93%**	0.4687	**1.14%**	0.4566	**1.88%**	0.4753
		8B	**1.39%**	0.4632	**0.56%**	0.4436	**1.42%**	0.4653	**1.68%**	0.4923
		70B	**1.57%**	0.4553	**0.88%**	0.4329	**1.32%**	0.4587	**1.79%**	0.4780
Response	Qwen3	0.6B	**1.34%**	0.4326	**0.42%**	0.4420	**0.96%**	0.4885	**1.97%**	0.4821
		8B	**1.08%**	0.4724	**0.38%**	0.4242	**1.58%**	0.4980	**1.97%**	0.4463
		32B	**1.21%**	0.4683	**0.43%**	0.4876	**0.89%**	0.5342	**1.48%**	0.4983
	Llama3	1B	**0.89%**	0.4521	**0.87%**	0.4853	**0.98%**	0.5421	**1.28%**	0.5021
		8B	**1.15%**	0.4646	**0.38%**	0.4154	**0.76%**	0.5015	**1.54%**	0.4875
		70B	**0.78%**	0.4328	**0.25%**	0.4423	**0.92%**	0.5201	**1.29%**	0.4857

Table 7: Effectiveness of Talaria against TokenInfer attack under different model structures and sizes. Bold: TRA. Gray: CSS

Attack Target	Attacks	Midjourney		WikiText2		Patient		GPT-samples	
Prompt	TokenInfer	**1.24%**	0.4876	**0.47%**	0.4299	**1.53%**	0.5012	**2.11%**	0.4535
	TokenInv	**0.45%**	0.3893	**0.52%**	0.4021	**0.64%**	0.3988	**0.58%**	0.4085
	SentenceInv	–	0.4435	–	0.3735	–	0.4051	–	0.3835
Response	TokenInfer	**1.08%**	0.4724	**0.38%**	0.4242	**1.58%**	0.4980	**1.97%**	0.4463
	TokenInv	**0.56%**	0.3954	**0.38%**	0.3785	**0.56%**	0.4123	**0.72%**	0.4025
	SentenceInv	–	–	–	–	–	–	–	–

Table 8: Effectiveness of Talaria in defending against different attacks. Bold: TRA. Gray: CSS.

- SPD (Gim et al., 2024) first confines the client's prompt and private data to a CVM, while splitting the computational workload with the untrusted cloud LLM provider. Then SPD uses prompt obfuscation to generate multiple plausible but fake "virtual prompts" that are processed alongside the client's real one, making it impossible for the provider to distinguish the authentic prompt from the fake ones. In the evaluations, we set the number of virtual prompts to 15.

- Cape (Wu et al., 2025) protects client privacy by locally perturbing the original prompt on the client's device before it is sent to the cloud. It uses a hybrid utility function that considers both the context and token similarity to find suitable replacement words. A bucketized sampling method is then used to efficiently select a replacement token, which ensures differential privacy while trying to maintain the prompt's original meaning. We set the number of buckets to 50 and use Bert (Devlin et al., 2019) to capture contextual information.

C.2 Additional Protection Evaluations

Different LLMs. We assess the scale-robustness of Talaria by evaluating four datasets–MIDJOURNEY, WIKITEXT-2, PATIENT NOTES, and GPT-SAMPLES–across language models with increasing parameter scale. For each model, we measure the effectiveness of TokenInfer attack using two metrics, TRA and CSS. As summarized in Tbl. 7, Talaria consistently suppresses reconstruction across all model sizes and datasets, and the magnitude of protection remains stable as model capacity grows, indicating that the defense is largely model-agnostic.

Different attacks. We evaluate Talaria against three SOTA reconstruction attacks on four datasets. As summarized in Tbl. 8, Talaria consistently depresses both indicators across all datasets and attack families. The relative protection remains stable even when attacks are tuned to their strongest published settings, suggesting that Talaria provides attack-agnostic robustness rather than overfitting to any single threat.

Different layers. We evaluate the protection per-

formance of Talaria against token reconstruction attacks at different layers, and use TokenInfer to attack the outsourced inputs of different layers of Qwen3-8B. For consistent comparison, we report both TRA and CSS over the concatenated prompt–response sequence rather than treating them separately. As summarized in Tbl. 9, across Q/K/V projections and MLP blocks at layers 1, 5, and 10 of Qwen3-8B, Talaria consistently collapses TokenInfer's success from near-perfect to negligible. Before defense, early layers leak the most (e.g., Layer-1: TRA≈98-99%, CSS≈0.93-0.95), and leakage naturally diminishes with depth (Layer-10: TRA≈83-84%, CSS≈0.82-0.83). After applying Talaria, all submodules converge to low, layer-insensitive residuals: defended TRA stays around 1.1-1.4% (best: 1.08–1.14%; worst: 1.43%), and CSS falls to 0.45-0.48. The protection is uniform across pathways (Q, K, V, MLP vary within little TRA at a given depth), indicating that hybrid masking plus partitioned execution neutralizes the principal layer-specific leakage channels rather than merely shifting them deeper. Talaria turns high pre-defense leakage at shallow layers into uniformly low, sub-2% TRA and $\leq$0.48 CSS across layers and submodules, demonstrating robust protection.

Attack Layer		TRA	CSS
Layer 1	Q	98.64%→1.42%	0.9443→0.4573
	K	99.13%→1.08%	0.9512→0.4876
	V	98.54%→1.26%	0.9337→0.4753
	MLP	98.04%→1.17%	0.9375→0.4805
Layer 5	Q	93.42%→1.13%	0.8653→0.4687
	K	92.88%→1.43%	0.8654→0.4724
	V	93.10%→1.36%	0.8634→0.4653
	MLP	94.05%→1.23%	0.8723→0.4564
Layer 10	Q	84.25%→1.14%	0.8326→0.4666
	K	84.48%→1.21%	0.8268→0.4532
	V	83.21%→1.09%	0.8251→0.4483
	MLP	83.45%→1.12%	0.8265→0.4503

Table 9: The protection performance of Talaria under different model layers (Midjourney dataset).

C.3 Additional Invariance Evaluations

Tbl. 10 provides three prompts together with the corresponding responses generated with and without Talaria. To remove randomness, we fix the model, tokenizer, decoding policy, and random seed. In all cases, the outputs are identical, with green shading used to mark the one-to-one correspondence. These examples complement the quantitative metrics and demonstrate that Talaria's reversible masking and local restoration preserve the model's behavior exactly, rather than only ensuring semantic similarity. Taken together, the results in Tbl. 10 confirm that Talaria maintains deterministic outputs while adding privacy protection, highlighting its effectiveness as a lossless mechanism.

Prompt	Response w/o Talaria	Response w/ Talaria
In the United States, common starlings are exempt from the Migratory Bird Treaty Act, which prohibits the taking or killing of migratory birds. No permit is required to remove nests and eggs or kill juveniles or adults. Research was undertaken in 1966 to identify a suitable avicide that would both kill common starlings and would readily be eaten by them...	First, the text says that common starlings are exempt from the Migratory Bird Treaty Act. I know that this act is a law that protects migratory birds, so if starlings are exempt, that means they aren't protected under this law. That probably means people can kill them without a permit. The text mentions that no permit is needed to remove nests, eggs, or kill both juveniles and adults. So, even though other migratory birds are protected, starlings aren't, which is why they can be targeted...	First, the text says that common starlings are exempt from the Migratory Bird Treaty Act. I know that this act is a law that protects migratory birds, so if starlings are exempt, that means they aren't protected under this law. That probably means people can kill them without a permit. The text mentions that no permit is needed to remove nests, eggs, or kill both juveniles and adults. So, even though other migratory birds are protected, starlings aren't, which is why they can be targeted...
Mozart had a pet common starling which could sing part of his Piano Concerto in G Major (KV. 453). He had bought it from a shop after hearing it sing a phrase from a work he wrote six weeks previously, which had not yet been performed in public. He became very attached to the bird and arranged an elaborate funeral for it when it died three years later...	So, Mozart had a starling that could sing part of his Piano Concerto in G Major (KV 453). He bought it after hearing it mimic a phrase from a piece he hadn't performed yet. The bird was a pet he was very attached to, and he had a funeral for it. There's a suggestion that his A Musical Joke (K. 522) might reflect the starling's style. Also, other starling owners note their mimicry skills, mixing phrases inappropriately, and even strangers thinking they heard a human speak.	So, Mozart had a starling that could sing part of his Piano Concerto in G Major (KV 453). He bought it after hearing it mimic a phrase from a piece he hadn't performed yet. The bird was a pet he was very attached to, and he had a funeral for it. There's a suggestion that his A Musical Joke (K. 522) might reflect the starling's style. Also, other starling owners note their mimicry skills, mixing phrases inappropriately, and even strangers thinking they heard a human speak.
Stephanie Madden, a 20-year-old female, has come to the doctor's office complaining of a headache. sHE REPORTS THAT SHE HAS AHD A STIFF NECK AND ha FOR PAST DAY - SHE reports that bright lights hurt her head - with general pain all over - she denies any aura preceding the headache and reports that nothing like this has happened before - she has vommitted 3+ times over the past day...	Okay, let me try to work through this case. So Stephanie is a 20-year-old female with a headache that's been going on for a day. She has a stiff neck, sensitivity to bright lights, generalized pain, and she's vomited three times. Tylenol hasn't helped. She denies any aura, which is interesting because that's a common feature in migraines. Her family history includes migraines on her mother's side and her father has HD, which I think is Huntington's disease. She's sexually active, uses condoms, and has some substance use: some alcohol and marijuana. She's not up to date on immunizations, especially the meningitis vaccine...	Okay, let me try to work through this case. So Stephanie is a 20-year-old female with a headache that's been going on for a day. She has a stiff neck, sensitivity to bright lights, generalized pain, and she's vomited three times. Tylenol hasn't helped. She denies any aura, which is interesting because that's a common feature in migraines. Her family history includes migraines on her mother's side and her father has HD, which I think is Huntington's disease. She's sexually active, uses condoms, and has some substance use: some alcohol and marijuana. She's not up to date on immunizations, especially the meningitis vaccine.

Table 10: Output of LLMs inference after using Talaria (green indicates text consistency).