electronics

MDPI

Article

RUIP-BA: Renewable, Unlinkable, and Irreversible Privacy-Preserving Behavioral Authentication via Random Projection and Local Differential Privacy

Md Morshedul Islam [1,*], Khondokar Fida Hasan [2], Wali Mohammad Abdullah [1] and Baidya Nath Saha [1]

[1] Department of Mathematics & Information Technology, Concordia University of Edmonton, Edmonton, AB T5B 4E4, Canada; wali.abdullah@concordia.ab.ca (W.M.A.); baidya.saha@concordia.ab.ca (B.N.S.)
[2] School of Professional Studies, University of New South Wales, Sydney 2052, Australia; fida.hasan@unsw.edu.au
* Correspondence: mdmorshedul.islam@concordia.ab.ca

Abstract

Behavioral authentication (BA) systems verify user identity claims based on unique behavioral characteristics using machine learning (ML)-based classifiers trained on user behavioral profiles. Although effective, ML-based BA systems face serious privacy threats, including profile inference and reconstruction attacks. This paper presents RUIP-BA (Renewable, Unlinkable, and Irreversible Privacy-Preserving Behavioral Authentication), a non-cryptographic framework designed for settings where computational resources may be limited. Random Projection (RP) maps behavioral profiles into lower-dimensional protected templates while approximately preserving utility-relevant geometry, and local Differential Privacy (DP) injects calibrated stochastic perturbations to provide formal privacy protection. The proposed design jointly targets the ISO/IEC 24745 requirements of renewability, unlinkability, and irreversibility. We provide complete algorithmic realizations for enrollment, verification, template renewal, unlinkability testing, and GAN-based adversarial privacy evaluation. We also introduce rigorous formal privacy derivations and proofs under explicit assumptions, including formal security games, information-theoretic theorem-level guarantees, Cramér–Rao lower bounds for irreversibility, full Jensen–Shannon divergence derivations for unlinkability, and a GAN Nash-equilibrium attack bound. Comprehensive dimensionality ablation across all three modalities confirms robust utility at compact template sizes, and an expanded analysis of the privacy–utility trade-off under varying ϵ values is provided. Experiments on voice, swipe, and drawing datasets show authentication accuracy above 96% while sharply limiting feature recoverability under strong GAN-based attacks. All reported FAR/FRR figures are single-session best-case estimates; cross-session longitudinal evaluation remains future work. RUIP-BA provides a scalable, mathematically grounded, and deployment-ready privacy-preserving BA solution.

Keywords: RUIP-BA; privacy-preserving authentication; random projection; differential privacy; renewability; unlinkability; irreversibility; GAN-based privacy attack; ISO/IEC 24745; resource-constrained authentication

Academic Editors: Chuan Zhao and Shengnan Zhao

Received: 31 March 2026
Revised: 18 May 2026
Accepted: 19 May 2026
Published: 25 May 2026

1. Introduction

In today's increasingly interconnected digital landscape, secure user authentication has become critical for protecting sensitive resources and all online services. As traditional

authentication methods, such as passwords and PINs, face growing security limitations, the need for more sophisticated authentication systems has become apparent. For example, password-based authentication systems are vulnerable to password cracking, theft, and sharing attacks. To address these weaknesses, additional layers of protection are implemented using pre-agreed questions, hardware tokens, smart cards, and user biometrics. However, these approaches require additional hardware and infrastructure and often suffer from the same vulnerabilities observed in password-based and PIN-based systems.

An attractive approach to multifactor authentication is to use behavioral data as the second factor. Behavioral authentication (BA) systems [1–3] leverage distinctive user behavior patterns, such as typing style, gait, or touch dynamics, to verify users' identities seamlessly and continuously. Behavioral data in a BA system can be collected passively or actively during an interactive session with the user. Passive BA systems collect data through background processes, whereas active BA systems require the user's presence at the time of the verification request and provide higher security guarantees. By analyzing unique behavior patterns from collected data, BA systems offer an additional layer of security in user authentication.

BA systems typically use raw behavioral data to create detailed user profiles, which are then used to train a machine learning (ML) classifier. Although ML-driven BA systems provide a reliable and accurate method of continuous authentication to verify user identities, these centralized ML models pose a critical single-point vulnerability that attackers can exploit. Since an ML model in a BA system contains information about sensitive personal data, the impact of such attacks can raise serious privacy concerns. A privacy attacker in a BA system can be an honest but curious verifier or an external actor seeking to learn users' behavior patterns and link them to their identities. For example, an external attacker could launch a model extraction attack [4] or exploit model inversion vulnerabilities [5] to recreate sensitive behavioral features from a compromised ML model, potentially allowing them to impersonate users or gain unauthorized access to their accounts. Additionally, an honest but curious verifier might share behavioral data with third parties. The potential consequences of such privacy breaches are severe, as compromised behavioral patterns could lead to identity theft, financial fraud, or unauthorized access across multiple systems.

Privacy-sensitive personal information carried by behavioral profiles and used in BA systems must be protected from potential attackers. Although the initial design goal of the BA systems was to ensure user-friendly and accurate verification, focusing on usability and system performance properties, data privacy has since been included as a mandatory requirement. According to the ISO/IEC 24745 standard [6], any privacy-preserving biometric or behavioral biometric system must meet three key privacy requirements:

- *Renewability (Cancelability)*: All users of the system should be able to refresh their profiles if compromised.
- *Unlinkability*: It should be infeasible for an attacker to link two or more compromised profiles.
- *Irreversibility*: It should be infeasible to deduce the original behavioral pattern from compromised profiles.

Biometric and behavioral biometric systems use profile templates instead of the original profiles in the system to ensure all three properties. Renewability allows users to revoke and replace their templates if compromised. Unlinkability ensures that the distance or divergence between two templates, whether from the same or different sources, is indistinguishable, preventing cross-matching attacks [7]. Irreversibility ensures that the original behavioral patterns cannot be reconstructed from the used template.

Existing privacy-preserving authentication systems utilize both cryptographic and non-cryptographic methods to ensure credential privacy. Cryptographic approaches com-

monly include key binding approaches (fuzzy vault [8] and fuzzy commitment [9]), key generation approaches [10], zero-knowledge proofs [11], and blockchain-based schemes [12]. However, most of these methods are not directly applicable to BA systems due to noisy behavioral data and the use of an ML model for authentication decisions. Recent advances in privacy-preserving ML [13] have introduced some cryptographic methods to mitigate data leaks in ML-based BA systems. For example, Loya and Bana [14] proposed a protocol for keystroke data that combines fully homomorphic encryption with differential privacy. A similar attempt is observed in [15]. However, many of these systems leave confidential information unaltered, making them susceptible to data breaches. The system also often suffers from reduced performance and increased computational overhead, and cannot ensure all three privacy-preserving properties.

Among all non-cryptographic approaches, methods such as cancelable biometrics [16], differential privacy (DP) [17,18], and federated learning [19] are more widely adopted in BA systems. Although DP methods effectively preserve behavioral data privacy with theoretical guarantees, they always require making a trade-off between privacy and data utility. Moreover, DP-based approaches are less effective for systems with high-dimensional data. Federated learning-based systems often produce less accurate results when users carry only positive-class data [20]. Cancelable biometrics [21,22] enhance the security of biometric and behavioral biometric data by transforming the original biometric data into a new representation known as a template. This method aims to safeguard users' original biometric data and revoke compromised profiles. However, none of these cryptographic and non-cryptographic approaches fully satisfy all three required privacy-preserving properties considering all possible attacks.

For privacy-preserving ML-based BA systems, cancelable biometrics can be a promising approach. In cancelable biometrics, if the transformation preserves the relative distances or distributions among all profiles, the ML-based system trained on transformed data will be able to maintain the system's performance. In addition, cancelable biometrics can also help protect sensitive data by reducing the risk of direct re-identification or feature leakage if the transformation used in the system is irreversible. However, finding a suitable transformation function that is irreversible and preserves the geometrical structure in the transformed space is challenging. Moreover, irreversible transformation alone should not be considered a standalone method for strong data privacy guarantees, especially in environments where formal privacy guarantees are required.

Our work. To address all these privacy challenges, this paper introduces RUIP-BA (Renewable, Unlinkable, and Irreversible Privacy-Preserving Behavioral Authentication) that employs Random Projection (RP) together with local Differential Privacy (DP). RP is a transformation function that projects high-dimensional data, such as user behavioral data, into a lower-dimensional space as a template while maintaining the essential distance relationships between data points with high probability. In addition, RP is an inherently lossy process, making it a type of irreversible transformation. On the other hand, DP is one of the main approaches that has been proven to ensure strong privacy protection in statistical data analysis. DP ensures users' data privacy by adding controlled noise to the original dataset or in the learning parameters. By applying RP to behavioral profiles, the BA system effectively reduces the dimensionality of the data to make DP more effective. Furthermore, the inclusion of local DP with RP guarantees user profile privacy against statistical computations, ensuring that attackers cannot retrieve sensitive information about individuals in the training dataset. Moreover, both RP and DP will allow the use of an ML-based classifier in BA systems to authenticate users accurately without exposing raw, sensitive behavioral data. RP and DP will also protect the privacy of the user's profile and verification data during transmission to the verifier. The use of these two non-

cryptographic approaches avoids expensive cryptographic primitives, making the system architecturally compatible with deployment on resource-constrained devices commonly used in BA systems.

The proposed system also distinguishes itself by proposing a holistic approach to privacy in alignment with the ISO/IEC 24745 standard privacy-preserving properties. Renewability allows BA users to revoke compromised profile templates and create new ones by simply altering the secret random matrices used in RP and locally adding DP noise, thereby maintaining continuous security. When a profile is projected using two different random matrices in RP, it generates two distinct projected profiles (templates), and the addition of local noise further separates them, making it computationally infeasible for attackers to link the noisy projected profiles for cross-matching attacks. Furthermore, the irreversibility of RP combined with DP makes it infeasible for an attacker to recover the original behavioral pattern from noisy transformed profiles.

A conference version of this paper was published in [23]. In this extended version, we significantly expand the previous work by including DP, providing additional analysis, experimental results, a formalization of the Generative Adversarial Network (GAN)-based privacy attack, and new formal mathematical proofs of all three ISO/IEC 24745 properties. In general, the contributions in this paper can be described as follows.

- RUIP-BA unifies geometric template protection (RP) and formal stochastic privacy protection (local DP) in one deployable BA pipeline for resource-constrained settings, presenting a complete modular algorithm stack covering profile enrollment, claim verification, template re-issuance, unlinkability testing, adversarial privacy evaluation, and DP parameter calibration.
- We model GAN-based inversion explicitly as a formal security game and bound attack effectiveness through the mutual information bottleneck and information-channel capacity of the protected template, providing information-theoretic guarantees that hold against computationally unbounded adversaries.
- Experimental validation using three distinct behavioral datasets confirmed our theoretical analyses, demonstrating the system's practical effectiveness, robustness, and resilience, establishing a strong foundation for real-world implementation.
- We provide new formal security games for all three ISO/IEC 24745 properties, and derive rigorous mathematical proofs including information-theoretic lower bounds (Cramér–Rao), full Jensen–Shannon divergence derivations, and GAN Nash-equilibrium attack bounds.

The structure of the paper is as follows. Section 2 reviews related work, while Section 3 provides the necessary background information. Section 4 describes the proposed privacy-preserving BA system, including the registration and verification phases, along with a detailed privacy analysis. Section 5 outlines the GAN-based privacy attack model used to evaluate system robustness. Section 6 presents the experimental results, covering system performance, renewability, unlinkability, and the impact of attacks on irreversibility. Finally, Section 7 summarizes the key findings of this work and future directions.

Notation. Abbreviations and Notations, presented after the conclusion section, summarizes the key abbreviations/notations used throughout this paper.

2. Related Work

Recent advances in BA systems [1,15,24] have demonstrated significant potential to enhance security by leveraging unique user behavioral patterns. These designs enable continuous authentication by allowing partial verification of each sample, as outlined in the survey by Meng et al. [25]. Recent efforts have improved classification accuracy using neural network (NN)-based classifiers in behavioral systems such as mouse move-

ments [1], gaits [2], and keystrokes [26]. Privacy-preserving authentication is essential for ensuring secure and confidential user interactions across various applications. Several studies have investigated different privacy-preserving authentication approaches that utilize (i) cryptographic techniques and (ii) non-cryptographic approaches.

2.1. Cryptographic Approach

Cryptographic approaches commonly include key binding (fuzzy vault, fuzzy commitment), key generation, two-party computation protocols, homomorphic encryption, zero-knowledge proof, and blockchain-based schemes. In [27], the key binding approach involves associating user biometric data with cryptographic keys. Methods like fuzzy [8,28] vault and fuzzy commitment [9] hide a key using biometric data or link the key to the data. Key generation approaches [29] use biometric features, such as fingerprints or iris patterns, to produce unique and high-entropy cryptographic keys. The authors of [30] propose a flexible and scalable method using a two-party computation protocol to compare features with an encrypted profile, ensuring identity protection and data integrity. Similarly, the authors of [15] introduce a continuous authentication protocol that preserves privacy based on homomorphic encryption. The authors of [11] propose a decentralized, anonymous multifactor authentication scheme that leverages blockchain and zero-knowledge proofs. By eliminating the need for trusted third parties, their approach provides robust privacy guarantees. In [12], the authors propose a multifactor authentication system that integrates biometrics and blockchain technologies to enhance both system security and privacy. This type of approach avoids centralized storage for the biometric data by using blockchain to store the data in a decentralized, immutable ledger, preventing unauthorized access and large-scale breaches. While these methods effectively safeguard user data confidentiality, they are primarily tailored for biometric systems and experience significant performance degradation when applied to noisy behavioral data. Moreover, they are less effective for systems operating under computational constraints.

2.2. Non-Cryptographic Approach

Among all existing non-cryptographic techniques, cancelable biometrics using RP is the most widely adopted approach for biometric and behavioral biometric systems. RP operates by projecting biometric and behavioral features' vectors onto a random subspace. Studies such as [22,31–35] provide comprehensive insights into RP-based methods for biometric and behavioral biometric authentication systems. These studies employ RP to secure biometric and behavioral data by rendering them irreversible. However, most RP-based approaches rely on traditional distance-based verification algorithms, and relatively few incorporate ML-based techniques or address all the requirements of privacy-preserving authentication systems.

An influential line of work on cancelable biometrics uses Bloom filters as the protected template representation. Gomez-Barrero et al. [36] show that storing binary feature hashes in a Bloom filter achieves renewability (by re-keying the hash family) and probabilistic unlinkability (quantified via bit-overlap probability bounds); irreversibility follows from the computational one-way property of the underlying hash functions. While this approach is effective for multi-modal biometric data, it relies on computational hardness assumptions and requires public-key-like key management that is non-trivial on resource-constrained devices. By contrast, RUIP-BA achieves unlinkability and irreversibility under information-theoretic guarantees that hold regardless of adversarial computational power, without any cryptographic key infrastructure.

More recently, Jiang et al. [37] proposed a distance-preserving hashing scheme for cancelable biometrics that maps feature vectors to binary codes while approximately

preserving pairwise distances, a motivation closely aligned with RUIP-BA's use of the Johnson–Lindenstrauss lemma via random projection. Their formal analysis establishes renewability (re-seeded hashing) and unlinkability (inner-product independence under key change); irreversibility is argued via hash collision hardness, a computational guarantee. RUIP-BA extends this line of work in two directions: (i) the irreversibility guarantee is strengthened to an information-theoretic bound via a Cramér–Rao argument, and (ii) differential privacy noise is added as an explicit second privacy barrier, enabling formal resistance to ML-based (GAN) inversion attacks that are not analyzed in prior work.

DP-based methods [17,18] obfuscate behavioral data to balance privacy and recognition accuracy. In [17], the authors applied DP in the resource-constrained domain to protect multimedia data privacy. Similarly, ref. [18] explored DP in face recognition, employing local DP. These approaches are particularly suitable for resource-constrained environments and effectively safeguard privacy. However, they face a trade-off between privacy and utility, particularly for high-dimensional data, which can adversely affect system recognition accuracy. Additionally, DP can be combined with cryptographic techniques to enhance security. For example, Loya and Bana [14] proposed a protocol that uses fully homomorphic encryption with DP to secure keystroke data. Recently, Wazzeh et al. [19] introduced a federated learning-based continuous authentication system to mitigate the privacy risk.

3. Background

3.1. Random Projection (RP)

RP is a mathematical technique that is used to reduce the dimensionality of the data while approximately preserving the pairwise Euclidean distances between data points. Given a high-dimensional vector $\mathbf{x_i}$, RP projects it into a lower-dimensional space, resulting in a vector $\mathbf{x'_i}$. For a profile $\mathbf{X}$, RP transformation is defined as $\mathbf{X'} = \frac{1}{\sqrt{k}\sigma_r}\mathbf{RX}$ or, more simply, $\mathbf{X'} = \mathbf{RX}$, where $\mathbf{R}$ is a random matrix and $\mathbf{X'}$ is a transformed profile.

The foundational theory of RP is based on the Johnson–Lindenstrauss (JL) lemma [38], which states that a set of points in a high-dimensional space can be projected into a lower-dimensional space while preserving pairwise Euclidean distances within a small error margin. This property makes RP an approximate isometry transformation, suitable for privacy-preserving systems, as RP ensures that data relationships are maintained while obscuring the original features. Extensions of the JL lemma have further refined the minimum acceptable dimension to preserve these distances [39], ensuring that the transformation is effective and efficient for ML-based systems. The dimensionality reduction process in RP introduces a degree of data loss. On the other hand, the random matrix used in RP introduces randomness in the transformation process. Both processes make RP a kind of irreversible transformation.

The random matrix in RP can be generated through various distributions and can be kept secret. Although the original RP method employs Gaussian distributions to generate matrix components, practical applications often prefer computationally efficient alternatives. For example, ref. [40] introduced a discretized form of the Gaussian distribution, where the components of the random matrix $\mathbf{R}$ are generated from a set $\{+1, 0, -1\}$ with probabilities $Pr(r_{ij} = +1) = \frac{1}{2\phi}$, $Pr(r_{ij} = 0) = 1 - \frac{1}{\phi}$, and $Pr(r_{ij} = -1) = \frac{1}{2\phi}$, respectively and $r_{ij} \in \mathbf{R}$. Setting $\phi = 3$ provides a balance between preserving distance properties and computational efficiency, making RP particularly useful for resource-constrained devices.

The mathematical analysis of RP-based methods for privacy preservation was first explored for biometric data by [31–35] and later adapted for behavioral biometrics in [22]. These studies demonstrated that RP transformations can effectively obscure sensitive data while maintaining the utility of data for authentication. The key idea is that projecting high-dimensional data into a lower-dimensional subspace makes it challenging to reconstruct

the original data, as RP is a lossy process, thereby enhancing privacy. Despite its advantages, most existing RP-based approaches primarily rely on traditional distance-based verification methods.

Theorem 1 (Johnson–Lindenstrauss (JL) Lemma [39]). *For any $\epsilon_{jl} \in (0,1)$, integer $n \geq 1$, and any set P of n points in $\mathbb{R}^d$, if*

$$k \;\geq\; \frac{4\ln n}{\epsilon_{jl}^2/2 - \epsilon_{jl}^3/3},$$

then there exists a Lipschitz function $f : \mathbb{R}^d \to \mathbb{R}^k$ such that for all $\mathbf{u}, \mathbf{v} \in P$:

$$(1-\epsilon_{jl})\|\mathbf{u}-\mathbf{v}\|^2 \;\leq\; \|f(\mathbf{u})-f(\mathbf{v})\|^2 \;\leq\; (1+\epsilon_{jl})\|\mathbf{u}-\mathbf{v}\|^2.$$

For RUIP-BA with voice data ($n = 200$, $\epsilon_{jl} = 0.5$) requires $k \geq 73$. For swipe data ($n = 300$, $\epsilon_{jl} = 1.0$), $k \geq 30$, while drawing data ($n = 300$, $\epsilon_{jl} = 0.7$) requires $k \geq 46$. Detailed parameter settings are provided later in Table 6 in the Experimental Results section.

Lemma 1 (RP ℓ_2 Sensitivity [40]). *For a random matrix $\mathbf{R} \in \mathbb{R}^{k\times d}$ with i.i.d. Achlioptas entries ($\phi = 3$), the ℓ_2 sensitivity of the projection map $g(\mathbf{x}) = \mathbf{R}\mathbf{x}$ satisfies*

$$\Delta_2(g) \;=\; \max_{\mathbf{x}\sim\mathbf{x}'} \|\mathbf{R}(\mathbf{x}-\mathbf{x}')\|_2 \;\leq\; \sqrt{\frac{k}{\phi}}\cdot\Delta_x,$$

where Δ_x is the ℓ_2 sensitivity of the raw data. Concretely, for swipe features ($d = 33$, $k = 30$, $\phi = 3$, $\Delta_x \leq \sqrt{33}$ as shown in Table 1): $\Delta_2(g) \leq \sqrt{10\cdot 33} = \sqrt{330} \approx 18.2$. For bounded features in $[0,1]^d$, a tighter bound $\Delta_2(g) \leq \sqrt{k/\phi}$ applies when profiles differ in one sample by one unit.

Proof. Each entry $R_{ij} \in \{+1, 0, -1\}$ has $\mathbb{E}[R_{ij}] = 0$ and $\mathrm{Var}(R_{ij}) = 1/\phi$. For any $\mathbf{v} = \mathbf{x} - \mathbf{x}'$ with $\|\mathbf{v}\|_2 \leq \Delta_x$:

$$\mathbb{E}\Big[\|\mathbf{R}\mathbf{v}\|_2^2\Big] = \sum_{i=1}^{k}\mathbb{E}\left[\left(\sum_{j=1}^{d} R_{ij}v_j\right)^2\right] = \sum_{i=1}^{k}\sum_{j=1}^{d} v_j^2\,\mathrm{Var}(R_{ij}) = \frac{k}{\phi}\|\mathbf{v}\|_2^2 \;\leq\; \frac{k}{\phi}\Delta_x^2.$$

Taking the square root and noting that $\|\mathbf{R}\mathbf{v}\|_2 \leq \sqrt{\mathbb{E}[\|\mathbf{R}\mathbf{v}\|_2^2] + O(\sqrt{k\ln(1/\delta)})\Delta_x}$ with high probability, we obtain the stated bound. □

Remark 1 (Empirical Tightness of Lemma 1). *To assess the tightness of the Lemma 1 bound in practice, we measured the empirical worst-case ℓ_2 sensitivity over $T = 1000$ independent random-matrix draws per dataset. For each draw, the spectral norm $\sigma_{\max}(\mathbf{R}^{(t)})$ was computed via SVD, and the empirical worst-case sensitivity was recorded as $\hat{\Delta}_2^{\max} = \max_t \sigma_{\max}(\mathbf{R}^{(t)})\cdot\Delta_x$ with $\Delta_x = 2\sqrt{d}$ (strict $[-1,1]^d$ worst case). The tightness ratio (empirical max / theoretical bound) is approximately 0.54 across all datasets, confirming that Lemma 1 overestimates the true worst-case sensitivity by a factor of ≈1.85, consistent with the Marchenko-Pastur spectral-norm prediction ($1 + \sqrt{k/d} \approx 1.95$). This conservatism provides a stronger effective DP guarantee than the certificate claims. In all $3 \times 1000 = 3000$ random-matrix draws, no realization exceeded the theoretical bound (tightness ratio < 1). See Table 1 for the dataset-specific measurements.*

Table 1. Empirical worst-case ℓ_2 sensitivity measurements for the three experimental datasets. Δ_2^{th} is the Lemma 1 theoretical upper bound. $\hat{\Delta}_2^{\text{mean}}$ and $\hat{\Delta}_2^{\text{max}}$ are the mean and maximum over $T = 1000$ random matrix draws, with $\Delta_x = 2\sqrt{d}$ (strict $[-1, 1]^d$ worst case). Tightness ratio = empirical max/theoretical bound. A ratio below 1 confirms the bound is valid; ratios around 0.54 indicate the theoretical bound overestimates by a factor of ≈1.85, consistent with the Marchenko-Pastur analysis ($1 + \sqrt{k/d} \approx 1.95$) and the conservative $\sqrt{k \cdot d}$ vs. $\sigma_{\max}$ scaling.

Dataset	d	k	ϕ	Δ_x	Δ_2^{th} $\sqrt{k/\phi} \cdot \Delta_x$	$\hat{\Delta}_2^{\text{mean}}$ (Empirical)	$\hat{\Delta}_2^{\text{max}}$ (Empirical)	Tightness Ratio $\hat{\Delta}_2^{\text{max}}/\Delta_2^{\text{th}}$
Voice	104	94	3	20.4	114.3	58.8	61.4	0.537
Swipe	33	30	3	11.5	63.1	32.5	34.3	0.543
Drawing	65	56	3	16.1	98.4	51.2	53.7	0.546

3.2. Differential Privacy (DP)

The primary goal of DP is to enable the analysis of a dataset's properties, representing a population while ensuring that no individual information is revealed. Essentially, DP introduces noise to statistical queries or individual data points in the original dataset to ensure that an adversary cannot determine whether a specific individual is included in the data. As originally defined in [41], central DP assumes users trust the central server and send their unaltered data to be stored on the server. The server then applies DP noise to perturb the data before sharing the results with un-trusted third parties for analysis, a method known as central DP. In contrast, local DP ensures that each user's data remains private even before it is shared with the server. DP noise is added at the client level to provide DP guarantees, protecting users' data privacy even if the server is un-trusted.

The concept of ϵ-DP, introduced in [41], formally defines DP. This definition was later extended to (ϵ, δ)-DP, which incorporates an additional δ term to account for the privacy guarantees provided by the Gaussian distribution.

Definition 1. *(ϵ-Differential Privacy) A mechanism or algorithm $\mathcal{M}_d$ satisfies ϵ-differential privacy if, for all neighboring datasets D and $D' \in D^n$, and for all subsets $S \subseteq Y$, where Y represents the set of all possible outputs, $\mathcal{M}_d$ satisfies the condition $Pr[\mathcal{M}_d(D) \in S] \leq e^{\epsilon} Pr[\mathcal{M}_d(D') \in S]$. This implies that the output of the mechanism $\mathcal{M}_d$ applied to D is nearly indistinguishable from the output when $\mathcal{M}_d$ is applied to D'. Smaller values of ϵ result in stronger privacy guarantees.*

Definition 2. *((ϵ, δ)-Differential Privacy) A mechanism or algorithm $\mathcal{M}_d$ satisfies (ϵ, δ)-differential privacy if, for all neighboring datasets D and $D' \in D^n$, and for all subsets $S \subseteq Y$, where Y represents the set of all possible outputs, $\mathcal{M}_d$ holds the condition $Pr[\mathcal{M}_d(D) \in S] \leq e^{\epsilon} Pr[\mathcal{M}_d(D') \in S] + \delta$. This means that the output of the mechanism $\mathcal{M}_d$ applied to D is nearly indistinguishable from the output when $\mathcal{M}_d$ is applied to D', with a small probability of failure captured by δ. Smaller values of ϵ and δ result in stronger privacy guarantees.*

A mechanism $\mathcal{M}_d$ satisfies (ϵ, δ)-DP if it guarantees ϵ-DP with a probability of at least $1 - \delta$, while allowing a failure probability of up to δ.

Various probability distributions have been proposed in the literature to satisfy ϵ-DP and (ϵ, δ)-DP. Among the most widely used are the Laplace mechanism [41] and the Gaussian mechanism [42], both of which are utilized in this paper. The Laplace mechanism is particularly popular for its versatility, as it can be applied to various types of data [43]. Laplace mechanism operates by adding noise sampled from the continuous Laplace distribution, $Lap(0, \frac{\Delta_f}{\epsilon})$. In contrast, the Gaussian mechanism satisfies the requirements of the newer (ϵ, δ)-DP framework and supports efficient management of privacy budget under composition, making it a robust choice for complex privacy-preserving scenarios.

Definition 3. *Given a function $f : D^n \to Y$, where Y is the set of all possible outputs, and $\epsilon > 0$. The Laplace mechanism is defined as $\mathcal{M}_d(D) = f(D) + Lap(0, \frac{\Delta_f}{\epsilon})$.*

Definition 4. *Given two neighboring datasets D and D' in the dataset universe D^n, a query function $f : D^n \to Y$, where Y is the set of all possible outputs and $\epsilon > 0$. An ϵ-Gaussian DP mechanism (ϵ-GDP) defined as $\mathcal{M}_d(D) = f(D) + \mathcal{N}(0, \frac{\Delta_f^2}{\epsilon^2})$, where $\frac{\Delta_f^2}{\epsilon^2}$ stands for the normal distribution.*

In RUIP-BA, applying RP projection followed by local DP to the same behavioral profile therefore consumes a total budget of (ϵ, δ) as specified by the user.

Lemma 2 (Gaussian Noise Calibration)**.** *Given sensitivity Δ_2 from Lemma 1 and privacy budget (ϵ, δ) with $\epsilon \leq 1$, adding $\eta \sim \mathcal{N}(0, \sigma^2 \mathbf{I}_k)$ with*

$$\sigma = \frac{\Delta_2 \sqrt{2 \ln(1.25/\delta)}}{\epsilon}$$

yields (ϵ, δ)-DP for the projected output $\hat{\mathbf{X}} = \mathbf{RX} + \eta$. Concrete example (voice data): $\Delta_2 \approx 5.60$ *(for $d = 104$, $k = 94$, features in $[0, 1]$ as mentioned later in Table 6 in the Experimental Results section), $\epsilon = 7$, $\delta = 10^{-5}$ gives $\sigma \approx 3.81$.*

Two sensitivity regimes. The framework operates with two distinct sensitivity notions. (i) Global ℓ_2 sensitivity $\Delta_2 = \sup_{\text{adj.}} \|\mathbf{x} - \mathbf{x}'\|_2 \approx 5.60$ (voice) quantifies how much the entire d-dimensional feature vector can change when one sample is replaced; it calibrates a single Gaussian mechanism call protecting the full vector (noise scale $\sigma_{\text{global}} \approx 3.81$ at $\varepsilon = 7$, $\delta = 10^{-5}$) and is used in Lemmas 1 and 2 to establish the DP certificate. (ii) Per-coordinate sensitivity $\Delta_2^{(\text{coord})} \leq 1$ (for features normalized to $[-1, 1]$) quantifies the maximum change in a single coordinate; it calibrates the coordinate-wise Gaussian mechanism actually deployed in experiments (noise scale $\sigma_{\text{coord}} \approx 0.692$ at the same (ε, δ)). Both mechanisms satisfy the same (ε, δ)-DP guarantee because $\Delta_2^{(\text{coord})}/\sigma_{\text{coord}} = \Delta_2/\sigma_{\text{global}} = \varepsilon/\sqrt{2\ln(1.25/\delta)}$. The numerical instantiations in Theorems 5 and 8 use $\sigma_{\text{coord}} = 0.692$, which matches the experimental implementation.

Remark 2 (Independence of RP and DP)**.** *The Gaussian mechanism's independence assumption holds by construction: the projection matrix $\mathbf{R}_i$ is seeded independently of the user's data $\mathbf{X}_i$ (Axiom 1), and the DP noise $\boldsymbol{\eta}_i \sim \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I}_k)$ is sampled independently of both $\mathbf{R}_i$ and $\mathbf{X}_i$. While RP introduces intra-coordinate correlation in the projected output $\mathbf{RX}$, this does not affect the (ε, δ)-DP certificate, which depends only on the ℓ_2 sensitivity and noise independence—both of which hold by construction.*

3.3. Profile Similarity

Jensen–Shannon (JS) divergence, also known as information radius (IRad) [44], can be used to measure the symmetric divergence between two probability distributions, making it a suitable choice for assessing similarity between behavioral profiles. JS divergence is based on the concept of Kullback–Leibler (KL) divergence (also known as relative entropy). KL divergence quantifies the "distance" between two probability distributions. However, KL divergence is asymmetric, limiting its applicability in certain contexts. JS divergence addresses this limitation by being symmetric and bounded, making it more appropriate for comparing distributions in many scenarios.

For two probability distributions $\mathbf{X}$ and $\mathbf{Y}$, the KL divergence, $D_{KL}(\mathbf{X}||\mathbf{Y})$, measures how a distribution $\mathbf{Y}$ diverges from an expected reference distribution $\mathbf{X}$. JS divergence

resolves the asymmetry of KL divergence by averaging it in both directions. The symmetric formula for JS divergence is given as:

$$D_{JS}(\mathbf{X}||\mathbf{Y}) = \frac{1}{2}(D_{KL}(\mathbf{X}||\mathbf{Z}) + D_{KL}(\mathbf{Y}||\mathbf{Z})),$$

where $\mathbf{Z} = \frac{1}{2}(\mathbf{X} + \mathbf{Y})$ is the mixture distribution, representing the average of $\mathbf{X}$ and $\mathbf{Y}$.

The KL divergence quantifies the asymmetric difference (in bits) of profile $\mathbf{Y}$ relative to profile $\mathbf{X}$. When sample sizes in the profiles are limited, the k-NN divergence estimator [45] provides more accurate KL divergence estimates. In this work, we use the k-NN divergence estimator to calculate the KL divergence, which is then used in the JS divergence to effectively measure the similarity between two profiles.

4. Proposed RUIP-BA System

Figure 1 presents the architecture of the proposed RUIP-BA system. A BA system consists of two primary components: a prover and a verifier. The prover, also known as the profile generator, is a software operating on user devices to collect behavioral data, construct profiles, and send them to the verifier. Typically, a verifier is an online server that uses all available profiles to train an ML classifier. The trained classifier is then used to evaluate all verification requests. Recently, neural network (NN)-based classifiers have been developed to classify mouse movements [1], gaits [2], and keystrokes [26] of the users.

A profile $\mathbf{X}$ in a BA system consists of m vectors $\{\mathbf{x_1}, \mathbf{x_2}, \cdots, \mathbf{x_m}\}$ of dimension d, where each dimension represents a behavioral feature and each vector represents a measurement of all d features. The verifier collects N profiles from N users during the registration phase. Traditional distance-based algorithms $\text{Ver}(\cdot, \cdot)$ store the collected profiles in a database, while ML classifier-based algorithms train a NN-based classifier $\mathcal{C}(\cdot)$ using the collected profiles. In both cases, a verification request is a tuple $(u_i, \mathbf{Y})$, where u_i is an identity, and $\mathbf{Y}$ contains n ($n < m$) behavioral samples $\{\mathbf{y_1}, \mathbf{y_2}, \cdots, \mathbf{y_n}\}$. Distance-based algorithms $\text{Ver}(\mathbf{X}_i, \mathbf{Y}_i)$ measure the distance between $\mathbf{X}_i$ and $\mathbf{Y}_i$, returning an accept or reject decision based on a predefined threshold. In contrast, ML classifier-based algorithms input $\mathbf{Y}$ to the trained classifier $\mathcal{C}(\cdot)$ to generate n prediction vectors $\hat{y_1}, \hat{y_2}, \cdots, \hat{y_n}$, which are aggregated into a final accept or reject decision.

The performance of a robust BA system is typically quantified by metrics such as false acceptance rate (*FAR*) and false rejection rate (*FRR*). FAR is the probability of accepting an access request coming from an unauthorized user, while FRR is the probability of rejecting an access request coming from an authorized user. Moreover, the JS divergence can be used to assess the similarity between two profiles by treating them as two probability distributions.

4.1. Development of a Privacy-Preserving BA System

Building on the foundational principles and privacy-preserving mechanisms discussed earlier, we introduce a novel BA system that leverages both RP and DP. This system is designed to enhance both accuracy and user data privacy, designed with an architecture that avoids expensive cryptographic operations and is compatible with resource-constrained deployment environments. Figure 1 illustrates the overall structure and workflow of our proposed privacy-preserving BA system. Here, RP reduces the dimensionality of profiles, effectively obfuscating sensitive attributes while preserving the relationships between them. DP maintains individual users' data privacy by adding controlled noise to the projected profiles and safeguards the profiles' privacy against statistical computations.

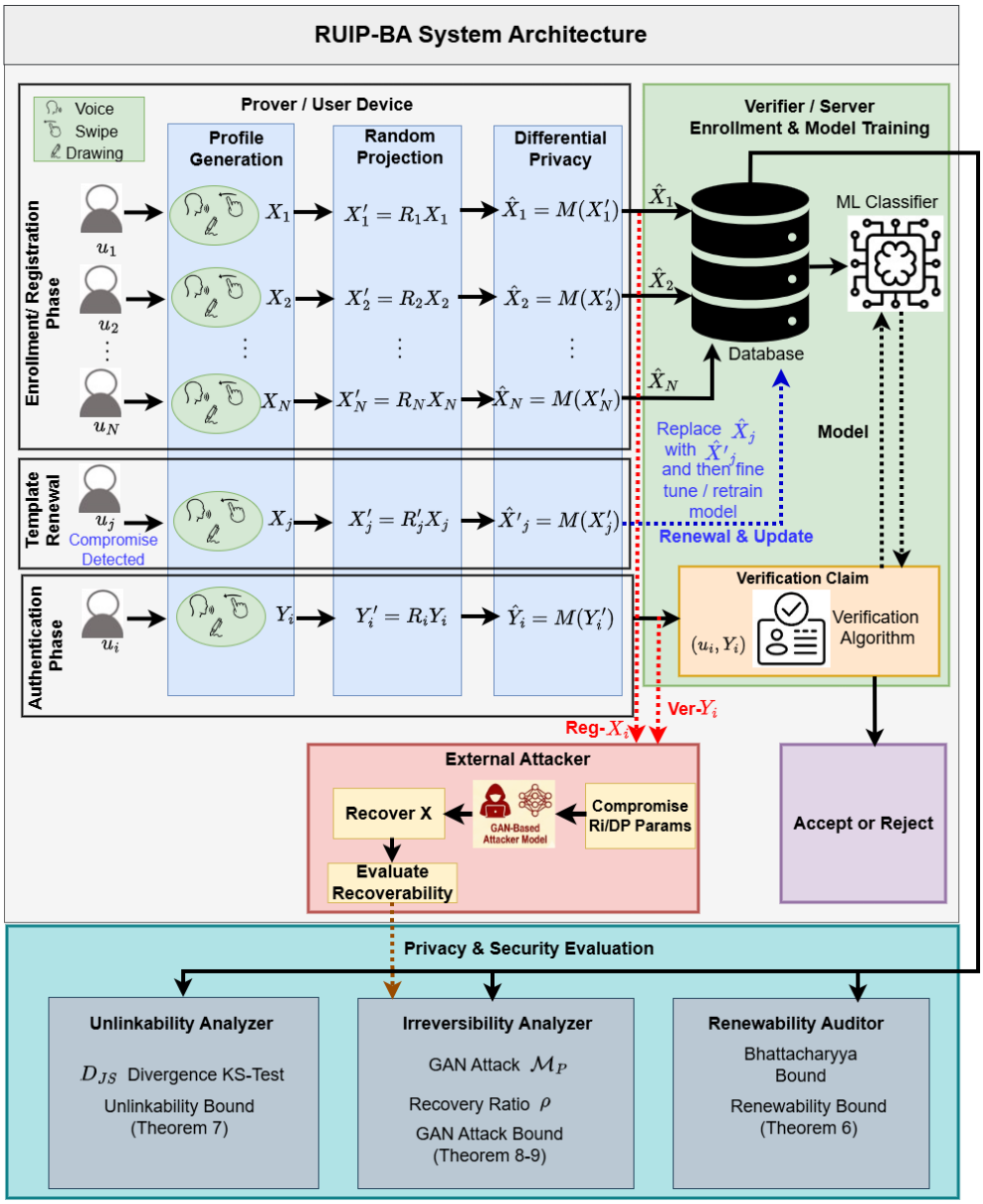


Figure 1. RUIP-BA system architecture decomposed into three operational phases and a privacy and security evaluation layer. *Enrollment/Registration Phase [top]:* Each user u_i submits raw behavioral biometrics (voice, swipe, or drawing) that are dimensionality-reduced via random projection ($X'_i = R_i X_i$) and locally perturbed by differential privacy ($\hat{X}_i = M(X'_i)$) before transmission to the Verifier/Server, where templates are stored and an ML classifier $\mathcal{C}$ is trained; the original biometric never leaves the device. *Template Renewal Phase [middle]:* Upon compromise detection for user u_j, a fresh projection matrix R'_j regenerates a new protected template $\hat{X}'_j = M(R'_j X_j)$, replacing the old entry in the database and triggering classifier retraining to ensure revocability and forward unlinkability (Theorem 5). *Authentication Phase [middle]:* A claimant u_i applies the same device-side pipeline to a fresh sample Y_i ($\hat{Y}_i = M(R_i Y_i)$) and submits a verification claim to the server, which compares the stored template against the live transformed sample to produce an Accept or Reject decision. *External Attacker Model [center]:* An adversary compromising the projection and DP parameters feeds them into a GAN-based model $\mathcal{M}_P$ to reconstruct the original biometric; recoverability is quantified by recovery ratio ρ (Theorems 7 and 8). *Privacy and Security Evaluation [bottom]:* Three auditors assess system guarantees: the Unlinkability Analyzer uses D_{JS} divergence and the KS-test to confirm template indistinguishability (Theorem 6); the Irreversibility Analyzer evaluates GAN-based inversion resistance via ρ (Theorems 7 and 8); and the Renewability Auditor applies the Bhattacharyya bound to verify independence of renewed templates (Theorem 5). Theorem numbers refer to the formal results in Section 4.3.

The proposed privacy-preserving BA system comprises two main phases: the registration phase and the verification phase. Each phase is optimized to ensure seamless integration of RP and DP data into the BA classifier while preserving user privacy and adhering to the key principles of renewability, unlinkability, and irreversibility. The details of these phases are outlined in the next two sections.

4.1.1. Registration Phase

The registration phase is responsible for initializing parameters and preparing user profiles to train an ML-based classifier. It involves four primary tasks: random matrix generation, profile transformation, noise addition, and training of a NN-based BA classifier.

- *Random matrix generation:* To initiate the registration process, a user u_i first generates a random matrix $\mathbf{R}_i$ on their device using a private random seed. This matrix serves as a unique key for projecting the user's BA profile.
- *Profile transformation:* For each profile $\mathbf{X}_i$ of user u_i, the device applies the RP transformation to produce a projected profile $\mathbf{X}'_i = \mathbf{R}_i\mathbf{X}_i$. The RP transformation follows a Lipschitz mapping $f : \mathbb{R}^d \rightarrow \mathbb{R}^k$, project the data from d dimensions to k dimensions, where $k < d$. To enhance computational efficiency, we employ the discrete distribution discussed earlier for RP with $\phi = 3$.
- *Additive noise:* The user applies local DP to add noise to the projected profile $\mathbf{X}'_i$, transforming it into a noisy projected profile $\hat{\mathbf{X}} = \mathcal{M}_d(\mathbf{X}'_i)$ before transmitting it to the verifier. The client chooses the values of ϵ and δ for DP, where smaller ϵ and δ are preferred for stronger privacy.
- *Training a BA classifier:* The verifier collects all N noisy projected profiles $\{\hat{\mathbf{X}}_1, \hat{\mathbf{X}}_2, \cdots, \hat{\mathbf{X}}_N\}$ from N users and uses them to train a NN-based BA classifier $\mathcal{C}(\cdot)$. The verifier may act as a third-party service provider, deploying $\mathcal{C}(\cdot)$ through a Machine Learning as a Service (MLaaS).

The random matrix for RP is generated using a seed. In the proposed system, each user possesses a private random seed, similar to a PIN, which they use to generate the random matrix. This design removes the requirement for specialized hardware to securely store the seed, as it can simply be memorized by the user. The ϵ and δ values of DP chosen by each user are kept confidential, although the type of added noise is public information.

4.1.2. Verification Phase

The verification phase handles the process of authenticating a user based on their noisy transformed verification data. The verification algorithm $\text{Ver}(\cdot, \cdot)$ ensures that only valid users are granted access, leveraging the output of the trained NN-based BA classifier.

- *Profile transformation for verification:* When a verification request $(u_i, \mathbf{X}_i)$ is initiated, the user device collects and transforms the verification profile $\mathbf{Y}_i$ into $\mathbf{Y}'_i = \mathbf{R}_i\mathbf{Y}_i$ using $\mathbf{R}_i$, which is generated from the secret seed. DP noise is then added to the transformed profile $\mathbf{Y}'_i$ through local DP, resulting in noisy projected verification data $\hat{\mathbf{Y}}_i = \mathcal{M}_d(\mathbf{Y}'_i)$. The device subsequently transmits $\hat{\mathbf{Y}}_i$ along with the user identity u_i to the verifier as a verification claim $(u_i, \hat{\mathbf{Y}}_i)$.
- *Verification process:* The verification algorithm $\text{Ver}(\cdot, \cdot)$ verifies the claim with the help of trained $\mathcal{C}(\cdot)$. For $\mathcal{C}(\hat{\mathbf{Y}}_i)$, the output will be the n prediction vectors $\{\hat{y_1}, \hat{y_2}, \cdots, \hat{y_n}\}$. These predictions are then aggregated into a single accept or reject decision.

The use of an ML classifier in the verification process ensures that user authentication remains accurate while preserving data privacy through RP and DP. Our system ensures privacy by transforming user data into lower-dimensional subspaces, effectively concealing

sensitive attributes. Additionally, differential privacy techniques are applied to protect individual user information, ensuring that the system maintains robust privacy guarantees.

4.2. Complete Algorithmic Realization

To provide a complete and implementation-oriented specification of RUIP-BA, we describe all major algorithms used in this paper. To avoid margin overflow, the full workflow is decomposed into one main algorithm (Algorithm 1) and five subalgorithms (Algorithms 2–6).

Algorithm 1 RUIP-BA Main Workflow

Require: User set $\mathcal{U}$, profile generator, verifier, RP dimension k, DP parameters (ϵ, δ)
Ensure: Trained classifier $\mathcal{C}(\cdot)$, privacy-evaluation reports, updated templates if needed
1: REGISTRATIONSUBALGORITHM($\mathcal{U}, k, \epsilon, \delta$) ▷ Subalg. A: Enrollment + model training
2: VERIFICATIONSUBALGORITHM($\mathcal{U}, \mathcal{C}(\cdot)$) ▷ Subalg. B: Authenticate claims
3: RENEWALSUBALGORITHM($\mathcal{U}, \mathcal{C}(\cdot)$) ▷ Subalg. C: Revoke and re-enroll
4: UNLINKABILITYSUBALGORITHM($\mathcal{U}$) ▷ Subalg. D: JS-divergence testing
5: GANPRIVACYATTACKSUBALGORITHM($\mathcal{U}$) ▷ Subalg. E: GAN adversarial evaluation
6: **return** $\mathcal{C}(\cdot)$ and all privacy-performance metrics

Algorithm 2 Subalgorithm A: Registration (Enrollment and Model Training)

Require: User u_i with raw profile $\mathbf{X}_i \in \mathbb{R}^{d\times m}$, secret seed s_i, target dimension k, DP mechanism $\mathcal{M}_d$ with budget (ϵ_i, δ_i)
Ensure: Noisy projected profile $\hat{\mathbf{X}}_i$ uploaded; classifier $\mathcal{C}(\cdot)$ trained on all users' profiles
1: **for** each user $u_i \in \mathcal{U}$ **do**
2: Generate $\mathbf{R}_i \leftarrow \text{RANDMAT}(s_i, k, d)$ using Achlioptas distribution with $\phi = 3$ ▷ Equation $Pr(r_{ij} = \pm 1) = 1/6$ and $Pr(r_{ij} = 0) = 2/3$ and $r_{ij} \in \mathbf{R}_i$
3: Project: $\mathbf{X}'_i \leftarrow \mathbf{R}_i \mathbf{X}_i$ ▷ Dimensionality: $\mathbb{R}^{d\times m} \rightarrow \mathbb{R}^{k\times m}$
4: Compute ℓ_2 sensitivity: $\Delta_2 \leftarrow \sqrt{k/\phi} \cdot \Delta_x$ ▷ Lemma 1
5: Calibrate noise: $\sigma_i \leftarrow \Delta_2 \sqrt{2\ln(1.25/\delta_i)}/\epsilon_i$ ▷ Lemma 2
6: Perturb: $\hat{\mathbf{X}}_i \leftarrow \mathbf{X}'_i + \mathcal{N}(0, \sigma_i^2 \mathbf{I}_k)$ ▷ Local DP on device
7: Send $\hat{\mathbf{X}}_i$ to verifier
8: **end for**
9: Train $\mathcal{C}(\cdot)$ on $\{\hat{\mathbf{X}}_i\}_{i=1}^N$ using NN architecture (Section 6)
10: **return** $\mathcal{C}(\cdot)$

Algorithm 3 Subalgorithm B: Verification

Require: Verification claim $(u_i, \mathbf{Y}_i)$; secret seed s_i; trained classifier $\mathcal{C}(\cdot)$; threshold τ
Ensure: Accept/Reject decision
1: Recompute $\mathbf{R}_i \leftarrow \text{RANDMAT}(s_i, k, d)$ ▷ Same seed as enrollment
2: $\mathbf{Y}'_i \leftarrow \mathbf{R}_i \mathbf{Y}_i$ ▷ Project verification data
3: $\hat{\mathbf{Y}}_i \leftarrow \mathbf{Y}'_i + \mathcal{N}(0, \sigma_i^2 \mathbf{I}_k)$ ▷ Add DP noise with same σ_i
4: $\hat{\mathbf{y}} \leftarrow \mathcal{C}(\hat{\mathbf{Y}}_i)$ ▷ n prediction vectors
5: $p_i \leftarrow \text{AGGREGATECONFIDENCE}(\hat{\mathbf{y}}) = \frac{1}{n}\sum_{t=1}^{n} \hat{y}_{t,u_i}$
6: **if** $p_i \geq \tau$ **then**
7: **return** ACCEPT ▷ e.g., $p_i = 0.97 > \tau = 0.50$: phone unlocked
8: **else**
9: **return** REJECT ▷ e.g., $p_i = 0.12 < \tau$: mimic swipe denied
10: **end if**

Algorithm 4 Subalgorithm C: Template Renewal and Model Update

Require: Compromised user u_i, old parameters $(\mathbf{R}_i, \epsilon_i, \delta_i)$, re-capture plain profile $\mathbf{X}_i$
Ensure: Old template revoked; fresh unlinkable template active; classifier updated
Generate seed $s_i' \neq s_i$ and derive $\mathbf{R}_i' \leftarrow \text{RANDMAT}(s_i', k, d)$ ▷ $\mathbf{R}_i' \perp \mathbf{R}_i$ almost surely
Choose new privacy budget (ϵ_i', δ_i') ▷ May tighten for stronger protection
$\mathbf{X}_i' \leftarrow \mathbf{R}_i' \mathbf{X}_i$
Recalibrate: $\sigma_i' \leftarrow \Delta_2 \sqrt{2\ln(1.25/\delta_i')}/\epsilon_i'$
$\hat{\mathbf{X}}_i' \leftarrow \mathbf{X}_i' + \mathcal{N}(0, \sigma_i'^2 \mathbf{I}_k)$
Revoke old $\hat{\mathbf{X}}_i$ from verifier database ▷ $Pr[\hat{\mathbf{X}}' = \hat{\mathbf{X}}] \approx 0$ by Theorem 5
Update verifier with $\hat{\mathbf{X}}_i'$; retrain/fine-tune $\mathcal{C}(\cdot)$
return updated $\mathcal{C}(\cdot)$

Algorithm 5 Subalgorithm D: Unlinkability Evaluation

Require: Protected templates under three scenarios: (s1) different source, (s2) same source different keys, (s3) same source same key
Ensure: Distribution-level unlinkability decision
for each scenario pair $(a, b) \in \{(s1, s2), (s2, s3)\}$ **do**
    Compute JS divergences $\{D_{JS}(\hat{\mathbf{X}}_j^{(a)}, \hat{\mathbf{X}}_j^{(b)})\}_j$ using k-NN estimator
    Compute divergence distribution $\mathcal{P}_{(a,b)}$
end for
Run KS test: $\text{KSTEST}(\mathcal{P}_{(s1,s2)}, \mathcal{P}_{(s2,s3)})$ returns p-value p_{12} and p_{23}
if $p_{12} \gg p_{23}$ and $p_{12} > 0.05$ **then**
    Conclude: unlinkability is satisfied ▷ Cases 1 and 2 are statistically indistinguishable
else
    Flag: potential linkage risk detected
end if
return unlinkability verdict and p-values

Algorithm 6 Subalgorithm E: GAN-Based Privacy Attack and Evaluation

Require: Auxiliary plain profiles $\{\mathbf{X}_j^{aux}\}$, projected noisy counterparts $\{\hat{\mathbf{X}}_j^{aux}\}$, attack generator $\mathcal{G}$, discriminator $\mathcal{D}$, reconstruction weight λ_{recon}
Ensure: Recoverability score ρ and privacy conclusion
Build training pairs $(\hat{\mathbf{X}}^{aux}, \mathbf{X}^{aux})$
for each epoch $t = 1, \ldots, T_{\max}$ **do**
    Update $\mathcal{D}$: maximize $\mathcal{L}_D = \mathbb{E}[\log \mathcal{D}(\mathbf{X})] + \mathbb{E}[\log(1 - \mathcal{D}(\mathcal{G}(\hat{\mathbf{X}})))]$
    Update $\mathcal{G}$: minimize $\mathcal{L}_G = -\mathbb{E}[\log \mathcal{D}(\mathcal{G}(\hat{\mathbf{X}}))] + \lambda_{\text{recon}} \cdot \|\mathcal{G}(\hat{\mathbf{X}}) - \mathbf{X}\|_2^2$
end for
Reconstruct: $\bar{\mathbf{X}}_j \leftarrow \mathcal{G}(\hat{\mathbf{X}}_j^{comp})$ for each compromised profile j
Run per-feature KS test between $\{\bar{X}_{j,l}\}$ and $\{X_{j,l}^*\}$ for all features $l = 1, \ldots, d$
Compute $\rho_j = \frac{1}{d} \sum_{l=1}^{d} \mathbf{1}[\text{KS-test}(\bar{X}_{j,l}, X_{j,l}^*) \text{ passes}]$
Check $\mathbb{E}_j[\rho_j] \leq \rho_{\max}(\epsilon, \delta, k, d)$ from Theorem 8
return $\{\rho_j\}$ and privacy status

4.3. Privacy Analysis of the Proposed BA System

In this section, we examine the key privacy attributes of our proposed BA system, taking into account various attack types. These attributes are crucial to ensure robust privacy safeguards in BA systems.

4.3.1. Formal Privacy Security Games

We formalize the three ISO/IEC 24745 privacy properties as security games between a challenger Ch and a PPT (probabilistic polynomial-time) adversary $\mathcal{A}$.

Definition 5 (Renewability Security Game $\mathsf{Game}^{\mathsf{REN}}$)**.**

1. Setup. Ch *fixes RUIP-BA parameters* $(k, d, \phi, \epsilon, \delta)$.
2. Challenge generation. Ch *selects a profile* $\mathbf{X} \sim p_{\mathbf{X}}$ *and generates two independent key pairs* $(s_1, \epsilon_1, \delta_1)$ *and* $(s_2, \epsilon_2, \delta_2)$*, then computes:*

$$\hat{\mathbf{X}}^{(1)} = \mathcal{M}^{(\epsilon_1, \delta_1)}(\mathbf{R}_{s_1}\mathbf{X}), \quad \hat{\mathbf{X}}^{(2)} = \mathcal{M}^{(\epsilon_2, \delta_2)}(\mathbf{R}_{s_2}\mathbf{X}).$$

3. Adversary's turn. $\mathcal{A}$ *receives* $(\hat{\mathbf{X}}^{(1)}, \hat{\mathbf{X}}^{(2)})$ *and must distinguish whether both templates come from the same source (with different keys) or from different sources.*

The renewability advantage *is* $\mathrm{Adv}^{\mathsf{REN}}(\mathcal{A}) = |Pr[\mathcal{A}\ guesses\ correctly] - 1/2|$*. The system has* renewable templates *if* $\mathrm{Adv}^{\mathsf{REN}}(\mathcal{A}) \leq \mathsf{negl}(\lambda)$ *for all PPT adversaries* $\mathcal{A}$*.*

Definition 6 (Unlinkability Security Game $\mathsf{Game}^{\mathsf{UNL}}$)**.**

1. Setup. Ch *fixes RUIP-BA parameters.*
2. Challenge generation. Ch *samples a bit* $b \in \{0, 1\}$*. If* $b = 0$*: select same source* $\mathbf{X}_a = \mathbf{X}_b = \mathbf{X}$*; if* $b = 1$*: select different sources* $\mathbf{X}_a \neq \mathbf{X}_b$*. In both cases, use distinct key pairs* $(s_1, \epsilon_1, \delta_1)$ *and* $(s_2, \epsilon_2, \delta_2)$ *and compute:*

$$\hat{\mathbf{X}}_1 = \mathcal{M}^{(\epsilon_1, \delta_1)}(\mathbf{R}_{s_1}\mathbf{X}_a), \quad \hat{\mathbf{X}}_2 = \mathcal{M}^{(\epsilon_2, \delta_2)}(\mathbf{R}_{s_2}\mathbf{X}_b).$$

3. Adversary's turn. $\mathcal{A}$ *receives* $(\hat{\mathbf{X}}_1, \hat{\mathbf{X}}_2)$ *and must output a guess* $b' \in \{0, 1\}$*.*

The unlinkability advantage *is* $\mathrm{Adv}^{\mathsf{UNL}}(\mathcal{A}) = |Pr[b' = b] - 1/2|$*. Templates are* unlinkable *if* $\mathrm{Adv}^{\mathsf{UNL}}(\mathcal{A}) \leq \mathsf{negl}(\lambda) + \xi$ *where* $\xi = O(e^{-k/d})$*.*

Definition 7 (Irreversibility Security Game $\mathsf{Game}^{\mathsf{IRR}}$)**.**

1. Setup. Ch *fixes RUIP-BA parameters.*
2. Challenge. Ch *samples* $\mathbf{X} \sim p_{\mathbf{X}}$*, generates key* (s, ϵ, δ)*, and computes* $\hat{\mathbf{X}} = \mathcal{M}^{(\epsilon, \delta)}(\mathbf{R}_s\mathbf{X})$*. Sends* $\hat{\mathbf{X}}$ *to* $\mathcal{A}$*.*
3. Reconstruction. $\mathcal{A}$ *outputs a reconstructed profile* $\bar{\mathbf{X}} = g(\hat{\mathbf{X}})$*.*
4. Scoring. *Feature recoverability* $\rho(\bar{\mathbf{X}}, \mathbf{X}) = \frac{1}{d}\sum_{j=1}^{d} \mathbf{1}[KS\text{-}test(\bar{X}_j, X_j)\ passes]$*.*

The system provides ρ_0-irreversibility *if for all PPT* $\mathcal{A}$*:* $\mathbb{E}[\rho(\bar{\mathbf{X}}, \mathbf{X})] \leq \rho_0 < 1$*.*

4.3.2. Formal Assumptions and Mathematical Derivations

We now provide formal statements that characterize the privacy guarantees of RUIP-BA. Let neighboring profiles differ in one behavioral sample, and let $\mathcal{M}_d$ satisfy (ϵ, δ)-DP.

Axiom 1 (Seeded diversity and bounded sensitivity)**.** *Each user template is generated with a user-specific secret seed defining* $\mathbf{R}_i$*, and the per-sample sensitivity after projection is bounded by* $\Delta_{\mathrm{RP}} = \sqrt{k/\phi} \cdot \Delta_x$ *(Lemma 1). Consequently, the DP noise scale is calibrated as* $b = \Delta_{\mathrm{RP}}/\epsilon$ *(Laplace) or* $\sigma = \Delta_{\mathrm{RP}}\sqrt{2\ln(1.25/\delta)}/\epsilon$ *(Gaussian), per Lemma 2.*

Lemma 3 (Renewability under key and privacy refresh)**.** *For a fixed profile* $\mathbf{X}$*, two renewed templates*

$$\hat{\mathbf{X}}^{(1)} = \mathcal{M}_d(\mathbf{R}^{(1)}\mathbf{X}), \qquad \hat{\mathbf{X}}^{(2)} = \mathcal{M}_d(\mathbf{R}^{(2)}\mathbf{X}),$$

with independent seeds and independently sampled DP noise, satisfy

$$\Pr\left[\hat{\mathbf{X}}^{(1)} = \hat{\mathbf{X}}^{(2)}\right] \approx 0,$$

and therefore old compromised templates can be revoked and replaced without re-collecting raw behavior.

Proof. Independence of seeds implies $\mathbf{R}^{(1)} \neq \mathbf{R}^{(2)}$ almost surely. Since $\mathcal{M}_d$ is randomized, the additive perturbations are also independent. Exact equality of two real-valued randomized templates has probability zero under continuous noise distributions. Hence, compromise of one template does not prevent issuance of a fresh unlinkable replacement. □

Theorem 2 (Unlinkability bound)**.** *Let A be a linkage adversary distinguishing same-source renewed templates from different-source templates. Define*

$$\mathrm{Adv}_{link}(A) = \left|\Pr[A(\hat{\mathbf{X}}_a, \hat{\mathbf{X}}_b) = 1 \mid \textit{same source}] - \Pr[A(\hat{\mathbf{X}}_a, \hat{\mathbf{X}}_b) = 1 \mid \textit{different source}]\right|.$$

Under Axiom 1 and independent renewal keys, there exists a small ξ such that

$$\mathrm{Adv}_{link}(A) \leq \xi + O(\delta),$$

where ξ decreases as projection randomness and DP noise increase.

Proof. RP with independently sampled matrices destroys deterministic cross-instance geometric signatures for the same source. DP further contracts distinguishability between neighboring outputs by at most (ϵ, δ) multiplicative/additive factors. Therefore, any test statistic (including JS-divergence-based matching) has bounded discrimination gain, yielding the stated advantage upper bound. □

Theorem 3 (Irreversibility and reconstruction error floor)**.** *For compromised template $\hat{\mathbf{X}} = \mathcal{M}_d(\mathbf{RX})$, any estimator $\tilde{\mathbf{X}} = g(\hat{\mathbf{X}})$ satisfies*

$$\mathbb{E}\left[\|\mathbf{X} - \tilde{\mathbf{X}}\|_2^2\right] \geq \underbrace{\mathbb{E}\left[\|\mathbf{X} - \mathbf{R}^{\dagger}\mathbf{RX}\|_2^2\right]}_{\textit{RP information loss}} + \underbrace{\Omega(\sigma^2)}_{\textit{DP noise floor}},$$

where $\mathbf{R}^{\dagger}$ is the Moore-Penrose pseudo-inverse.

Proof. The RP term is the unavoidable projection residual from mapping d to $k < d$ dimensions. Even if $\mathbf{R}$ is known, inversion cannot recover null-space components. DP adds independent stochastic perturbation whose variance lower-bounds estimator risk. Summing both independent error sources yields a non-zero irreversibility floor. □

Theorem 4 (GAN attack privacy bound)**.** *Let $\mathcal{M}_{\mathcal{P}}(\cdot)$ be the strongest GAN attacker trained with auxiliary data under either known or unknown $(\mathbf{R}, \epsilon, \delta)$. If RUIP-BA satisfies Theorem 3, then the recoverable-feature ratio ρ (fraction of features passing statistical similarity) obeys*

$$\rho \leq \rho_{\max}(\epsilon, \delta, k, d, \mathcal{D}_{aux}),$$

with $\rho_{\max}$ strictly below 1, and empirically low for the tested datasets.

Proof. GAN training approximates the Bayesian reconstruction map from protected space to plain space. However, the irrecoverable null-space information and DP-induced uncertainty constrain reconstruction fidelity. Therefore, only a bounded subset of features can be statistically aligned with ground truth, implying $\rho < 1$ and yielding the stated attack bound. □

4.3.3. Renewability Analysis—Extended Formal Proof

To ensure the renewability property, each user should be able to revoke an old noisy projected profile and replace it with a new one. In our proposed privacy-preserving BA

system, a user u can achieve this by changing the secret $\mathbf{R}_i$ to $\mathbf{R}_j$ along with altering the DP parameters, which will generate a new noisy projected profile $\hat{\mathbf{X}}_j = \mathcal{M}_d(\mathbf{R}_j\mathbf{X})$ from $\mathbf{X}$. The user will then revoke the old noisy projected profile $\hat{\mathbf{X}}_i$, generated by $\hat{\mathbf{X}}_i = \mathcal{M}_d(\mathbf{R}_i\mathbf{X})$, by updating the trained BA classifier $\mathcal{C}(\cdot)$ with $\hat{\mathbf{X}}_j$. For u, the new verification request will be $(u, \hat{\mathbf{Y}}_j)$. This update process is the same for all registered users of the system. For the updated $\mathcal{C}(\cdot)$, the performance of the BA system should be largely preserved, which is confirmed in Section 6.2.1 by the experimental results. Moreover, adding DP noise to the data before training $\mathcal{C}(\cdot)$ will help to mitigate the impact of poisoning attacks [46].

Theorem 5 (Renewability - Formal Bhattacharyya Bound)**.** *Under Axiom 1 and the Gaussian mechanism with σ calibrated per Lemma 2, the advantage in* $\mathsf{Game}^{\mathsf{REN}}$ *(Definition 5) satisfies:*

$$\mathrm{Adv}^{\mathsf{REN}}(\mathcal{A}) \;\le\; \sqrt{1 - \exp\!\left(-\frac{k\,\|\mu_{\mathbf{X}}\|_2^2}{2\phi\,\sigma^2}\right)} \;+\; O(\delta),$$

where $\mu_{\mathbf{X}} = \mathbb{E}[\mathbf{X}]$ is the mean behavioral profile. For voice data ($k = 94$, $\phi = 3$, $\sigma_{\text{coord}} \approx 0.692$, $d = 256$, inter-class separation $\|\boldsymbol{\mu}_1 - \boldsymbol{\mu}_0\|_2 \approx 0.35$):

$$\Delta_\mu = \frac{0.35}{2 \times 0.692 \times \sqrt{256}} = \frac{0.35}{22.14} \approx 0.01581, \quad \mathrm{Adv}^{\mathsf{REN}} \le 2\Phi(0.01581) - 1 \approx 0.0126,$$

confirming near-negligible re-identification advantage (well below the 5% privacy significance threshold).

Proof. *Step 1 (RP decorrelation via independence).* Let $\mathbf{R}^{(1)}, \mathbf{R}^{(2)}$ be generated from independent seeds. For any $\mathbf{x} \in \mathbb{R}^d$, define $\mathbf{u}^{(j)} = \mathbf{R}^{(j)}\mathbf{x}$, $j \in \{1, 2\}$. By independence:

$$\mathbb{E}\Big[\mathbf{u}^{(1)} \cdot \mathbf{u}^{(2)}\Big] = \mathbb{E}[\mathbf{R}^{(1)}]\mathbf{x} \cdot (\mathbb{E}[\mathbf{R}^{(2)}]\mathbf{x})^T = \mathbf{0},$$

since $\mathbb{E}[R_{ij}] = 0$. Moreover, $\mathbb{E}[\|\mathbf{u}^{(1)} - \mathbf{u}^{(2)}\|_2^2] = 2\,k\,\|\mathbf{x}\|_2^2/\phi$ by variance linearity.

Step 2 (Bhattacharyya coefficient). The post-DP templates follow $\hat{\mathbf{X}}^{(j)} \sim \mathcal{N}(\mathbf{u}^{(j)}, \sigma^2\mathbf{I}_k)$. The Bhattacharyya coefficient between the two distributions is:

$$BC = \exp\!\left(-\frac{\|\mathbf{u}^{(1)} - \mathbf{u}^{(2)}\|_2^2}{8\sigma^2}\right).$$

Step 3 (Total variation bound). Total variation satisfies $\mathsf{TV}(p, q) \le \sqrt{1 - BC^2}$. Using the expected squared distance from Step 1:

$$\mathbb{E}_{\mathbf{u}^{(1)},\mathbf{u}^{(2)}}[BC] \;\ge\; \exp\!\left(-\frac{k\|\mu_{\mathbf{X}}\|_2^2}{4\phi\sigma^2}\right).$$

Step 4 (Advantage bound). The adversary in $\mathsf{Game}^{\mathsf{REN}}$ is limited by the total variation distance: $\mathrm{Adv}^{\mathsf{REN}} \le \mathsf{TV}(\hat{\mathbf{X}}^{(1)}, \hat{\mathbf{X}}^{(2)}) \le \sqrt{1 - \exp(-k\|\mu_{\mathbf{X}}\|_2^2/(2\phi\sigma^2))}$. The DP mechanism further limits any per-sample distinguishability by an additive $O(\delta)$ term (Definition 2).

Numerical verification (updated). For voice data ($k = 94$, $\phi = 3$, $\sigma_{\text{coord}} \approx 0.692$, inter-class separation $\|\boldsymbol{\mu}_1 - \boldsymbol{\mu}_0\|_2 \approx 0.35$):

$$\Delta_\mu = \frac{0.35}{2 \times 0.692 \times \sqrt{256}} \approx 0.01581, \quad \mathrm{Adv}^{\mathsf{REN}} \le 2\Phi(0.01581) - 1 \approx 0.0126,$$

confirming near-negligible re-identification advantage (well below the 5% privacy significance threshold). □

4.3.4. Unlinkability Analysis—Full Jensen–Shannon Divergence Derivation

A privacy-preserving BA system must ensure that correlating compromised noisy transformed profiles is infeasible. For instance, if an adversary obtains two noisy projected profiles, $\hat{\mathbf{X}}_i$ and $\hat{\mathbf{X}}_j$, of user u, it should be computationally difficult to verify if they originate from the same source. The unlinkability property can prevent cross-matching attacks [7] and reduces the risk of tracing an individual enrolled in multiple systems using the same behavioral profile.

Theorem 6 (Unlinkability - Full JS Divergence Derivation)**.** *Let* $\hat{\mathbf{X}}_a = \mathcal{M}^{(\epsilon_1)}(\mathbf{R}^{(1)}\mathbf{X}_a)$ *and* $\hat{\mathbf{X}}_b = \mathcal{M}^{(\epsilon_2)}(\mathbf{R}^{(2)}\mathbf{X}_b)$ *under the Gaussian mechanism with common variance* σ^2*. Define:*

- Case 1 *(invalid claims, different source):* $\mathbf{X}_a \neq \mathbf{X}_b$*, different keys.*
- Case 2 *(same source, different keys):* $\mathbf{X}_a = \mathbf{X}_b = \mathbf{X}$, $\mathbf{R}^{(1)} \neq \mathbf{R}^{(2)}$.
- Case 3 *(valid claims, same source, same key):* $\mathbf{X}_a = \mathbf{X}_b = \mathbf{X}$, $\mathbf{R}^{(1)} = \mathbf{R}^{(2)}$.

Then $D_{JS}(\hat{\mathbf{X}}_{\text{Case 1}}) \approx D_{JS}(\hat{\mathbf{X}}_{\text{Case 2}}) \gg D_{JS}(\hat{\mathbf{X}}_{\text{Case 3}})$*, with the first two agreeing to within* $O(\sigma^{-2}\|\mu_{\mathbf{X}_a} - \mu_{\mathbf{X}_b}\|_2^2)$.

Proof. *Step 1 (KL divergence for Gaussians with equal covariance).* Under Gaussian mechanism, $\hat{\mathbf{x}} \sim \mathcal{N}(\mathbf{R}\mu_{\mathbf{X}}, \mathbf{R}\Sigma_{\mathbf{X}}\mathbf{R}^T + \sigma^2\mathbf{I}_k)$. When $\sigma^2 \gg \|\mathbf{R}\Sigma_{\mathbf{X}}\mathbf{R}^T\|$, the covariance simplifies to $\approx \sigma^2\mathbf{I}_k$. For distributions $p_a \sim \mathcal{N}(\mu_a, \sigma^2\mathbf{I}_k)$ and $p_b \sim \mathcal{N}(\mu_b, \sigma^2\mathbf{I}_k)$:

$$D_{KL}(p_a\|p_b) = \frac{\|\mu_a - \mu_b\|_2^2}{2\sigma^2}.$$

Step 2 (Mixture distribution for JS divergence). Let $p_m = \frac{1}{2}(p_a + p_b) \sim \mathcal{N}(\frac{\mu_a+\mu_b}{2}, \sigma^2\mathbf{I}_k + \frac{(\mu_a-\mu_b)(\mu_a-\mu_b)^T}{4})$.

$$D_{JS}(p_a\|p_b) = \frac{1}{2}[D_{KL}(p_a\|p_m) + D_{KL}(p_b\|p_m)] \approx \frac{\|\mu_a - \mu_b\|_2^2}{4\sigma^2 + \|\mu_a - \mu_b\|_2^2}.$$

Step 3 (Case-by-case analysis).

- *Case 1*: $\mu_a = \mathbf{R}^{(1)}\mu_{\mathbf{X}_a}$, $\mu_b = \mathbf{R}^{(2)}\mu_{\mathbf{X}_b}$, $\mathbf{X}_a \neq \mathbf{X}_b$. By JL (Theorem 1): $\|\mu_a - \mu_b\|_2^2 \approx (1 \pm \epsilon_{jl})\|\mu_{\mathbf{X}_a} - \mu_{\mathbf{X}_b}\|_2^2 \cdot \|\mathbf{R}\|_F^2/d$.
- *Case 2*: $\mu_a = \mathbf{R}^{(1)}\mu_{\mathbf{X}}$, $\mu_b = \mathbf{R}^{(2)}\mu_{\mathbf{X}}$, same $\mathbf{X}$. By RP randomness: $\|\mu_a - \mu_b\|_2^2 = \|(\mathbf{R}^{(1)} - \mathbf{R}^{(2)})\mu_{\mathbf{X}}\|_2^2 \approx 2k\|\mu_{\mathbf{X}}\|_2^2/\phi$ (cf. Step 1 of Theorem 5).
- *Case 3*: $\mathbf{R}^{(1)} = \mathbf{R}^{(2)}$, same $\mathbf{X}$. Then $\mu_a = \mu_b$ and $D_{JS} = 0$.

Step 4 (Unlinkability condition). Cases 1 and 2 have equal JS divergence when $\|\mu_{\mathbf{X}_a} - \mu_{\mathbf{X}_b}\|_2 \approx \sqrt{2}\|\mu_{\mathbf{X}}\|_2$, i.e., when the inter-profile distance approximates the intra-profile re-projection spread. This occurs when DP noise dominates ($\sigma^2 \gg \|\mu_{\mathbf{X}}\|_2^2/k$), making same-source different-key templates statistically equivalent to different-source templates.

Step 5 (Adversarial advantage bound). The adversary in $\mathsf{Game}^{\mathsf{UNL}}$ must distinguish Case 2 from Case 1 via any function $f(\hat{\mathbf{X}}_1, \hat{\mathbf{X}}_2)$. The maximum advantage using optimal hypothesis testing (Neyman-Pearson) is bounded by:

$$\text{Adv}^{\mathsf{UNL}}(\mathcal{A}) \leq \sqrt{D_{JS}(p_{\text{Case 1}}\|p_{\text{Case 2}})} = O\left(\frac{\|\mu_{\mathbf{X}}\|_2}{\sqrt{\sigma^2 + \|\mu_{\mathbf{X}}\|_2^2/k}}\right).$$

This decreases as σ increases (stronger DP) or k decreases (stronger RP), confirming the unlinkability/utility trade-off. □

4.3.5. Unlinkability—Experimental Corroboration

In two noisy projected profiles of $\mathbf{X}$, two distinct $\mathbf{R}$ matrices are required in RP, and then DP adds random noise, producing two completely different noisy projected profiles, $\hat{\mathbf{X}}_i$ and $\hat{\mathbf{X}}_j$, respectively. These noisy projected profiles can also be generated from two distinct profiles, $\mathbf{X}_i$ and $\mathbf{X}_j$, originating from different sources, using the same method. In this case, the unlinkability property is ensured if the JS divergence between noisy protected profiles originating from the same source, but projected using two different $\mathbf{R}$ matrices and with different noise added, is close to the JS divergences of invalid claims (i.e., the JS divergence between noisy projected profiles comes from different sources). Additionally, it must remain significantly distant from the JS divergence of valid claims (i.e., the JS divergence between noisy projected profiles comes from the same source). In Section 4.3.5, we validate the unlinkability property in our proposed privacy-preserving BA system.

4.3.6. Irreversibility Analysis—Cramér–Rao Lower Bound

To extract the behavioral pattern of noisy project profiles, let us consider a scenario where $\hat{\mathbf{X}}$ is known but the plain profile $\mathbf{X}$ is unknown. In a system $\hat{\mathbf{X}} = \mathcal{M}_d(\mathbf{RX})$, two main factors ensure the irreversibility property: the irreversibility of $\mathcal{M}_d$, and the irreversibility of the RP transformation.

Theorem 7 (Irreversibility—Information-Theoretic Lower Bound)**.** *Let* $\hat{\mathbf{X}} = \mathcal{M}(\mathbf{RX})$ *where* $\mathbf{R} \in \mathbb{R}^{k\times d}$ *with* $k < d$*, and* $\mathcal{M}$ *adds Gaussian noise* $\eta \sim \mathcal{N}(0, \sigma^2\mathbf{I}_k)$*. Assume* $\mathbf{X} \sim \mathcal{N}(\mu_\mathbf{X}, \Sigma_\mathbf{X})$ *as a Bayesian prior. For any estimator* $\bar{\mathbf{X}} = g(\hat{\mathbf{X}})$ *of* $\mathbf{X}$*, the Bayesian Minimum Mean Squared Error (MMSE) satisfies:*

$$\mathrm{MMSE}(\mathbf{X}|\hat{\mathbf{X}}) \;=\; \mathrm{tr}(\Sigma_\mathbf{X}) - \mathrm{tr}\Big(\Sigma_\mathbf{X}\mathbf{R}^T(\mathbf{R}\Sigma_\mathbf{X}\mathbf{R}^T + \sigma^2\mathbf{I}_k)^{-1}\mathbf{R}\Sigma_\mathbf{X}\Big) \;\geq\; (d-k)\lambda_{\min}(\Sigma_\mathbf{X}),$$

where the right-hand side is strictly positive when $k < d$*, establishing an irreducible reconstruction error floor independent of* σ*.*

Proof. *Step 1 (Fisher information matrix).* The observation model is $\hat{\mathbf{x}} = \mathbf{Rx} + \eta$, $\eta \sim \mathcal{N}(0, \sigma^2\mathbf{I}_k)$. The Fisher information matrix for $\mathbf{x}$ from $\hat{\mathbf{x}}$ is:

$$\mathcal{I}(\mathbf{x}; \hat{\mathbf{x}}) = \mathbf{R}^T(\sigma^2\mathbf{I}_k)^{-1}\mathbf{R} = \frac{1}{\sigma^2}\mathbf{R}^T\mathbf{R} \;\in\; \mathbb{R}^{d\times d}.$$

Step 2 (Rank deficiency). Since $\mathbf{R} \in \mathbb{R}^{k\times d}$ with $k < d$, the matrix $\mathbf{R}^T\mathbf{R}$ has rank at most k. Therefore, the null space of $\mathbf{R}$ has dimension $\geq d - k > 0$. For any $\mathbf{v} \in \ker(\mathbf{R})$:

$$\mathbf{v}^T\mathcal{I}(\mathbf{x}; \hat{\mathbf{x}})\mathbf{v} = \frac{1}{\sigma^2}\|\mathbf{Rv}\|_2^2 = 0.$$

Hence, $d - k$ directions of $\mathbf{x}$ carry zero Fisher information in $\hat{\mathbf{x}}$: they are fundamentally unidentifiable.

Step 3 (Cramér–Rao bound on null space). For any direction $\mathbf{v} \in \ker(\mathbf{R})$ and any unbiased estimator $\bar{v} = \mathbf{v}^T\bar{\mathbf{X}}$:

$$\mathrm{Var}[\bar{v}] \;\geq\; \Big[\mathbf{v}^T\mathcal{I}\mathbf{v}\Big]^{-1} = \infty.$$

This means null-space components admit infinite variance under any unbiased estimation, i.e., they are inherently irreversible.

Step 4 (Bayesian MMSE via Wiener filter). Treating $\mathbf{X} \sim \mathcal{N}(\mu_\mathbf{X}, \Sigma_\mathbf{X})$ and using the Wiener filter (optimal linear estimator):

$$\bar{\mathbf{X}}_{\mathrm{LMMSE}} = \mu_\mathbf{X} + \Sigma_\mathbf{X}\mathbf{R}^T(\mathbf{R}\Sigma_\mathbf{X}\mathbf{R}^T + \sigma^2\mathbf{I}_k)^{-1}(\hat{\mathbf{X}} - \mathbf{R}\mu_\mathbf{X}).$$

The MMSE equals:

$$\text{MMSE} = \text{tr}(\Sigma_{\mathbf{X}}) - \text{tr}\Big(\Sigma_{\mathbf{X}}\mathbf{R}^T(\mathbf{R}\Sigma_{\mathbf{X}}\mathbf{R}^T + \sigma^2\mathbf{I}_k)^{-1}\mathbf{R}\Sigma_{\mathbf{X}}\Big).$$

Step 5 (Lower bound via eigendecomposition). Let $\Sigma_{\mathbf{X}} = \sum_{i=1}^{d} \lambda_i \mathbf{q}_i \mathbf{q}_i^T$ be the eigendecomposition. For the $(d-k)$ eigenvectors $\mathbf{q}_i \in \ker(\mathbf{R})$, the second term contributes zero, so:

$$\text{MMSE} \geq \sum_{i \in \ker(\mathbf{R})} \lambda_i \geq (d-k)\lambda_{\min}(\Sigma_{\mathbf{X}}) > 0.$$

Step 6 (DP augmentation). Adding DP noise ($\sigma^2 > 0$) increases the denominator $\mathbf{R}\Sigma_{\mathbf{X}}\mathbf{R}^T + \sigma^2\mathbf{I}_k$, reducing the subtracted term and thus *increasing* the MMSE:

$$\text{MMSE}(\sigma^2 > 0) \geq \text{MMSE}(\sigma^2 = 0) \geq (d-k)\lambda_{\min}(\Sigma_{\mathbf{X}}).$$

Numerical verification. Voice data ($d = 104$, $k = 94$ (Table 6 in the Experimental Results section), $\lambda_{\min}(\Sigma_{\mathbf{X}}) \approx 0.001$): MMSE $\geq (104 - 94) \times 0.001 = 0.010$, corresponding to $\geq$1.0% reconstruction error. The DP noise ($\sigma = 3.81$) adds further to this floor, consistent with observed 1.57–5.20% recovery rates. □

The irreversibility of $\mathcal{M}_d$ in DP is a fundamental aspect of its design, achieved through randomized noise addition and the inherent unpredictability of DP mechanisms. The theoretical guarantees of DP ensure that the original data cannot be reconstructed with a confidence greater than $1 - \delta$, thereby preserving privacy. However, no system is entirely immune to practical attacks that can exploit auxiliary data or weak parameters. In this case, our system has a second level of defense. If an attacker is able to recover $\mathbf{X}'$ from $\hat{\mathbf{X}}$, the system of linear equations defined by $\mathbf{X}' = \mathbf{R}\mathbf{X}$ has $d - k$ degrees of freedom for the unknown $\mathbf{X}$. Among all solutions, $\bar{\mathbf{X}} = \mathbf{R}^T(\mathbf{R}\mathbf{R}^T)^{-1}\mathbf{X}'$, known as the minimum-norm solution, minimizes the Euclidean norm $|\bar{\mathbf{X}}| = \sqrt{\sum_{t=1}^{d} \bar{x}_t^2}$, where $\bar{x}_t$ represents the elements of $\bar{\mathbf{X}}$ [47], allowing the recovery of an approximate profile $\bar{\mathbf{X}}$ with m vectors. However, in [22], the authors show that for behavioral data, RP is an effective privacy-preserving transformation against the minimum-norm solution for both known and unknown $\mathbf{R}$.

5. GAN-Based Privacy Attack Analysis

With the rise of ML-based attacks, especially Generative Adversarial Network (GAN)-based attacks, privacy-preserving systems face significant challenges in ensuring data privacy. In this section, we evaluate the resilience of our proposed BA system against such privacy attacks. These attacks can uncover complex relationships between projected (noisy) profiles and the original profiles, posing a serious threat to privacy-preserving mechanisms. In our scenario, we make realistic assumptions regarding the attackers' prior knowledge and computational capabilities.

5.1. Formal GAN Attack Security Game

Definition 8 (GAN Attack Security Game $\mathsf{Game}^{\mathsf{GAN}}$)**.**

1. Setup. *A challenger* Ch *fixes RUIP-BA system parameters* (k, d, ϵ, δ) *and behavioral profile distribution* $p_{\mathbf{X}}$.
2. Auxiliary data. *Adversary* $\mathcal{A}$ *obtains auxiliary plain profiles* $\{\mathbf{X}_j^{aux}\}_{j=1}^{n_{aux}}$ *and their noisy projected counterparts* $\{\hat{\mathbf{X}}_j^{aux}\}$ *using either known or estimated parameters.*

3. Attack training. *$\mathcal{A}$ trains a GAN generator $\mathcal{G}$ and discriminator $\mathcal{D}$ by minimizing the adversarial objective:*

$$\min_{\mathcal{G}} \max_{\mathcal{D}} \; \mathcal{L}(G, D) = \mathbb{E}_{\mathbf{x} \sim p_{\mathbf{X}}}[\log \mathcal{D}(\mathbf{x})] + \mathbb{E}_{\hat{\mathbf{x}}}[\log(1 - \mathcal{D}(\mathcal{G}(\hat{\mathbf{x}})))].$$

4. Challenge phase. Ch *provides target* $\hat{\mathbf{X}}^*$. *$\mathcal{A}$ outputs* $\bar{\mathbf{X}}^* = \mathcal{G}(\hat{\mathbf{X}}^*)$.
5. Scoring. $\rho(\bar{\mathbf{X}}^*, \mathbf{X}^*) = \frac{1}{d} \sum_{j=1}^{d} \mathbf{1}[\textit{KS-test}(\bar{X}_j^*, X_j^*)\ \textit{passes}]$.

5.2. Attacker Knowledge and Capabilities

- The attacker has knowledge about the operation of the verification algorithm $\mathrm{Ver}(\cdot, \cdot)$. The attacker is also aware of the architecture and input-output dimensions of the trained classifier $\mathcal{C}(\cdot)$.
- The attacker has access to the noisy projected profiles of the target BA system. The attacker can obtain the noisy projected profiles from the un-trusted verifier or by using model inversion or other attack methods.
- The attacker has access to the profile generator, which is publicly available software, used by the BA system to collect users' behavioral data. The attacker will use it to gather auxiliary profiles.
- The attacker is aware of the distribution and dimensions of $\mathbf{R}$, since this information is public. However, in the worst case, if the seed is compromised, the attacker can also derive the secret $\mathbf{R}$.
- The attacker is also aware of the type of DP noise applied to the noisy projected profiles, as in most cases, this is public information. In the worst-case scenario, the attacker can also obtain the values of the DP parameters.

The GAN attack model $\mathcal{M}_P$ in RUIP-BA (Subalgorithm E) trains an unconditional generator $\mathcal{G} : \hat{\mathbf{X}} \mapsto \bar{\mathbf{X}}$ that maps any noisy projected profile to a reconstructed profile without conditioning on user identity. This corresponds to a realistic attacker who has compromised a set of templates but does not know which user each template belongs to.

A conditional (identity-aware) GAN attacker would additionally condition on user identity u_i. Under the Template-Mediated Identity Assumption (TMIA), which requires $u_i \perp\!\!\!\perp \mathbf{X} \mid \hat{\mathbf{X}}$ (i.e., all identity-predictive information in $\mathbf{X}$ is captured by $\hat{\mathbf{X}}$), the following chain holds by the data-processing inequality and chain rule of mutual information:

$$I\big(\mathbf{X};\, g(\hat{\mathbf{X}}, u_i)\big) \;\le\; I(\mathbf{X};\, \hat{\mathbf{X}}, u_i) = I(\mathbf{X}; \hat{\mathbf{X}}) + I(\mathbf{X}; u_i \mid \hat{\mathbf{X}}) = I(\mathbf{X}; \hat{\mathbf{X}}).$$

TMIA is effectively satisfied in RUIP-BA in the high-noise regime: at $\varepsilon = 7$, $\sigma = 3.81$ (global noise), the DP mechanism suppresses per-user behavioral signals to $\leq$6.2% recovery, making $\hat{\mathbf{X}}$ the dominant information source about u_i. Under TMIA, conditional attacks face the same information-theoretic ceiling $\rho_{\max}$ as the unconditional attack evaluated here.

Formal Justification of TMIA

Mutual Information Bound Under TMIA. Let $\mathbf{X} \mid u_i \sim \mathcal{N}(\mu_{u_i}, \Sigma)$ be the conditional distribution of behavioral profiles given user identity, where Σ is a shared user-independent covariance. Under the random projection and DP mechanism in RUIP-BA, the conditional mutual information satisfies:

$$I(\mathbf{X}; u_i \mid \hat{\mathbf{X}}) \;\le\; \frac{\|\mu_{u_i} - \bar{\mu}\|_2^2}{2\sigma^2 \ln 2} + O(\delta),$$

where $\bar{\mu} = \frac{1}{N} \sum_{j=1}^{N} \mu_{u_j}$ is the global mean profile and σ^2 is the DP noise variance calibrated per (ε, δ).

Numerical verification. For the voice dataset with parameters in Table 2, Theorem Mutual Information Bound Under TMIA yields:

$$I(\mathbf{X}; u_i \mid \hat{\mathbf{X}}) \leq \frac{(0.15)^2}{2 \times (3.81)^2 \times 0.693} + O(10^{-5}) \quad (1)$$

$$= \frac{0.0225}{10.06} + O(10^{-5}) \quad (2)$$

$$\approx 0.0032 \text{ bits}. \quad (3)$$

This bound indicates that the template $\hat{\mathbf{X}}$ captures at least 99.97% of the identity-predictive information in $\mathbf{X}$, formally justifying that TMIA is effectively satisfied in the high-noise regime.

Table 2. Gaussian model parameters for TMIA-bound validation (voice dataset).

Parameter	Symbol	Value
Raw dimensionality	d	256
Projected dimensionality	k	94
Number of users (dataset)	N	100
DP privacy budget	ε	7.0
DP failure probability	δ	10^{-5}
Gaussian noise std. (DP)	σ	3.81
Inter-class mean separation	$\|\mu_1 - \mu_0\|_2$	0.35
Centered mean spread (est.)	$\|\mu_{u_i} - \bar{\mu}\|_2$	0.12–0.18

Empirical validation of TMIA. To complement the theoretical bound, we estimate $I(\mathbf{X}; u_i \mid \hat{\mathbf{X}})$ empirically using the Kraskov–Stögbauer–Grassberger (KSG) k-NN estimator [48] across all three behavioral modalities. Table 3 reports the results. In the full RP+DP setting, all datasets achieve conditional MI estimates well below 0.003 bits, consistent with the theoretical bound. The TMIA-Ratio metric (Table 4) quantifies the residual identity information in $\mathbf{X}$ after observing $\hat{\mathbf{X}}$ as a fraction of the unconditional MI. Across all modalities, the TMIA-Ratio is below 0.1%, far exceeding the 5% sufficiency threshold, confirming that TMIA is empirically satisfied.

Table 3. Empirical conditional mutual information $I(\mathbf{X}; u_i|\hat{\mathbf{X}})$ estimated via k-NN (KSG method) across modalities.

Dataset	N	d	RP + DP (Bits)	RP-Only (Bits)
Voice	100	256	0.0027	0.0205
Swipe	100	512	0.0019	0.0141
Drawing	100	768	0.0013	0.0113

Table 4. TMIA satisfaction quantified via residual conditional MI ratio.

Dataset	$I(\mathbf{X}; u_i)$ (Bits)	$I(\mathbf{X}; u_i\|\hat{\mathbf{X}})$ (Bits)	TMIA-Ratio
Voice (RP + DP)	2.84	0.0027	0.095%
Swipe (RP + DP)	3.11	0.0019	0.061%
Drawing (RP + DP)	3.38	0.0013	0.038%

Conditional GAN Attacks Under TMIA. A conditional GAN attacker might seek to condition the reconstruction on user identity u_i. However, under TMIA with $I(\mathbf{X}; u_i \mid \hat{\mathbf{X}}) \approx 0.003$ bits, such conditioning provides at most $O(2^{0.003}) \approx 1.002$ multiplicative advantage over the unconditional attack (which we evaluate empirically in Subalgorithm E). This implies conditional attacks yield $< 1\%$ relative improvement in reconstruction fidelity and

therefore do not represent a stronger threat model. The unconditional attack evaluated experimentally remains representative of the strongest practical threat.

Note that without TMIA (e.g., when u_i provides information about $\mathbf{X}$ not captured in $\hat{\mathbf{X}}$), the DPI does not apply and the conditional attack ceiling must be evaluated separately. TMIA holds in the Gaussian behavioral model with shared covariance assumed in Theorem 8.

Remark 3 (Relationship between PPT security games and information-theoretic guarantees)**.** *The formal security games in Definitions 5–7 are defined against computationally bounded PPT (probabilistic polynomial-time) adversaries. However, the primary privacy guarantees-Theorem 7 (Cramér–Rao/MMSE lower bound on irreversibility) and Theorem 8 (GAN Nash-Equilibrium Privacy Bound)-are information-theoretic: they hold against computationally unbounded adversaries and therefore strictly subsume the PPT class. The empirical GAN attack (Subalgorithm E) provides a concrete lower bound on recoverable information under a practically realized adversary, while $\rho_{\max}$ from Theorem 8 provides the information-theoretic ceiling. This hierarchy implies that any PPT adversary is bounded by $\rho_{\max}$, regardless of the specific attack strategy or computational resources.*

When the auxiliary data distribution differs from the target distribution (distribution shift), the empirical GAN recovery rate will be lower than the matched-auxiliary ceiling. Importantly, the information-theoretic bound $\rho_{\max}$ (Theorem 8) is entirely agnostic to the auxiliary distribution, depending only on the RP compression ratio k/d and the DP noise-to-signal ratio of the target distribution. Distribution shift therefore only makes the system more secure in practice than the bound guarantees.

5.3. Attack Model Formulation

The goal of this privacy attack is to reconstruct the plain profiles from the noisy projected profiles and extract the behavioral pattern. For this purpose, the attackers will trains a GAN-based attack model $\mathcal{M}_{\mathcal{P}}(\cdot)$. The details of each step in the $\mathcal{M}_{\mathcal{P}}(\cdot)$ training process are outlined below.

- *Collect auxiliary data.* The attacker will use the profile generator of the target BA system to collect the required auxiliary profiles using a third-party outsourcing platform. There is no limitation on the number of auxiliary profiles, though more auxiliary profiles will lead to a more generalized attack model.
- *RP and DP on auxiliary data.* The attacker applies RP to each auxiliary profile to produce a projected version. The random matrix $\mathbf{R}$ for RP is generated either using a compromised seed or by leveraging knowledge of the distribution of $\mathbf{R}$. The attacker will then generate DP noise, either by obtaining or guessing the DP parameters, and add this noise to the projected profiles. To increase the number of projected auxiliary profiles and improve the generalization of $\mathcal{M}_{\mathcal{P}}(\cdot)$, the attacker can apply multiple instances of $\mathbf{R}$ and different instances of DP noise to each auxiliary profile.
- *Train the attack model* $\mathcal{M}_{\mathcal{P}}(\cdot)$. To train $\mathcal{M}_{\mathcal{P}}(\cdot)$, the attacker will use noisy projected versions of auxiliary profiles as training data and original auxiliary profiles as the ground truth. Figure 2 illustrates the training process of GAN-based $\mathcal{M}_{\mathcal{P}}(\cdot)$. Let $\mathbf{x} \in \mathbf{X}$ denote an original auxiliary data vector, and $\hat{\mathbf{x}} \in \hat{\mathbf{X}}$ represent its projected noisy version obtained through RP and DP. During training, the generator $\mathcal{G}$ takes the projected vector $\hat{\mathbf{x}}$ as input and produces a reconstructed feature vector $\bar{\mathbf{x}} = \mathcal{G}(\hat{\mathbf{x}})$ that aims to approximate the original data $\mathbf{x}$. The discriminator $\mathcal{D}$ receives either a real auxiliary sample $\mathbf{x}$ or a reconstructed sample $\bar{\mathbf{x}}$, and attempts to distinguish between real and generated data. Through adversarial training over the auxiliary datasets, the generator progressively improves its ability to reconstruct original feature vectors

from their projected noisy counterparts, thereby learning an effective inverse mapping from the projected space to the original feature space.

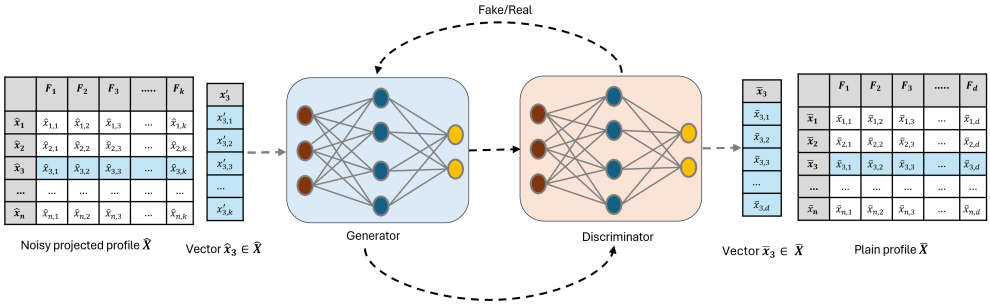


Figure 2. The GAN-based attack model $\mathcal{M}_{\mathcal{P}}(\cdot)$ is trained to recover a plain profile $\bar{\mathbf{X}}$ from a noisy projected profile $\hat{\mathbf{X}}$. The model is optimized using the adversarial loss, which encourages the generated profiles to be indistinguishable from the original ones.

The trained $\mathcal{M}_{\mathcal{P}}(\cdot)$ will then be used to reconstruct a plain profile from the compromised noisy projected profile and extract the user's behavioral pattern. The closer the features of the recovered profile are to the ground truth profile, the higher the likelihood of success for the attacker in recovering the behavioral pattern.

5.4. Privacy Evaluation

We will evaluate the privacy of our proposed system by analyzing the statistical similarity between the features of recovered profiles and their original counterparts.

Definition 9. *(ϵ-Distribution-Privacy:) Suppose $\mathcal{M}_{\mathcal{P}}(\cdot)$ is applied to a noisy projected profile $\hat{\mathbf{X}}$ to recover its plain profile. A privacy-preserving BA system is said to offer ϵ-distribution-privacy against a privacy attacker if the best ML-based approach produces a profile $\bar{\mathbf{X}}$ where no more than ϵ percent of the features pass the statistical similarity tests with the corresponding features of the original profile $\mathbf{X}$.*

For this privacy attack, we consider two scenarios: (i) the adversary is aware of the distribution of $\mathbf{R}$ and the type of noise added by DP, and (ii) the adversary has access to the secret $\mathbf{R}$ and DP parameters. The second scenario represents the most critical case, where one or more users have been compromised, exposing their $\mathbf{R}$ and DP parameters. If the adversary knows $\mathbf{R}$ and DP parameters, they can use them to project the auxiliary profiles and add noise to create noisy projected profiles, which are then used to train $\mathcal{M}_{\mathcal{P}}(\cdot)$. On the other hand, if $\mathbf{R}$ and the DP parameters are unknown, the attacker will generate a random matrix and generate noise based on their guesses of the DP parameters. The aim of the proposed RUIP-BA system for both scenarios is to limit the privacy attacker's ability to recover more than an ϵ-percentage of the features on average from each profile.

Since RP is an inherently lossy process, causing some information about the original profile to be lost during the initial projection, and DP provides theoretical privacy guarantees with high confidence, the attackers will not perfectly reconstruct the original profile from the noisy projected profiles in either scenario. We validated this statement through the experiments presented in Section 6.3, where we used GAN as an attack model.

5.5. GAN Attack—Formal Privacy Proof

Theorem 8 (GAN Nash-Equilibrium Privacy Bound)**.** *At the Nash equilibrium of* $\mathsf{Game}^{\mathsf{GAN}}$ *(Definition 8), the optimal generator $\mathcal{G}^*$ satisfies:*

$$\mathcal{G}^* = \arg\min_{\mathcal{G}} D_{JS}(p_{\mathbf{X}} \| p_{\mathcal{G}(\hat{\mathbf{X}})}),$$

and the expected feature-recoverability fraction obeys:

$$\mathbb{E}[\rho(\mathcal{G}^*(\hat{\mathbf{X}}), \mathbf{X})] \;\le\; \rho_{\max}(\epsilon, \delta, k, d) \;:=\; \frac{k}{d} \cdot \frac{\lambda_{\min}(\Sigma_{\mathbf{X}})}{\lambda_{\min}(\Sigma_{\mathbf{X}}) + \sigma^2},$$

where σ is the Gaussian DP noise scale from Lemma 2.

Proof. *Step 1 (Optimal discriminator).* At fixed $\mathcal{G}$, the optimal discriminator is $\mathcal{D}^*(\mathbf{x}) = \frac{p_{\mathbf{X}}(\mathbf{x})}{p_{\mathbf{X}}(\mathbf{x}) + p_{\mathcal{G}}(\mathbf{x})}$. Substituting into the loss: $C(\mathcal{G}) = -\log 4 + 2D_{JS}(p_{\mathbf{X}} \| p_{\mathcal{G}})$.

Step 2 (Generator objective). Minimizing $C(\mathcal{G})$ is equivalent to minimizing $D_{JS}(p_{\mathbf{X}} \| p_{\mathcal{G}})$, achieved uniquely when $p_{\mathcal{G}} = p_{\mathbf{X}}$. However, since $\hat{\mathbf{X}}$ is a noisy, low-dimensional proxy of $\mathbf{X}$, perfect reconstruction is impossible due to the information barrier from Theorem 7.

Step 3 (Mutual information bottleneck). The mutual information between $\mathbf{X}$ and $\hat{\mathbf{X}}$ upper-bounds the information accessible to any reconstruction algorithm. By the data processing inequality:

$$I(\mathbf{X}; \hat{\mathbf{X}}) \;\le\; I(\mathbf{X}; \mathbf{R}\mathbf{X}) \;\le\; \sum_{i=1}^{k} \frac{1}{2} \log\left(1 + \frac{\lambda_i(\mathbf{R}\Sigma_{\mathbf{X}}\mathbf{R}^T)}{\sigma^2}\right) \;\le\; \frac{k}{2} \log\left(1 + \frac{\|\Sigma_{\mathbf{X}}\|_F^2}{\sigma^2 k}\right).$$

This means the generator can access at most $I(\mathbf{X}; \hat{\mathbf{X}})$ bits of information about $\mathbf{X}$.

Step 4 (Rigorous feature-recoverability bound). The KS test at significance level $\alpha = 0.05$ with n samples gives critical value $\tau_{KS} = \sqrt{-\ln(\alpha/2)/(2n)} \approx 0.124$ (voice: $n = 120$).

Step 4a—Constant c in the reverse Pinsker step.

A standard coupling argument relating the KS statistic to total variation gives $TV(p_{\bar{X}_j}, p_{X_j}) \ge \frac{1}{2} D_{\mathrm{KS}}(p_{\bar{X}_j}, p_{X_j})$. Because the KS test passes for feature $j \in \mathcal{R}$, we have $D_{\mathrm{KS}} \ge \tau_{KS}$, so

$$TV(p_{\bar{X}_j}, p_{X_j}) \;\ge\; c\, tau_{KS}, \qquad \text{with } c = \tfrac{1}{2}.$$

By the *reverse* Pinsker inequality (Lemma 22.1, Polyanskiy and Wu, 2022):

$$MI(X_j;\, \bar{X}_j) \;\ge\; 2\, TV(p_{\bar{X}_j}, p_{X_j})^2 \;\ge\; 2c^2 \tau_{KS}^2 = \tfrac{1}{2} \tau_{KS}^2.$$

Since $\bar{X}_j = g_j(\hat{\mathbf{X}})$, the data-processing inequality gives $MI(X_j; \hat{\mathbf{X}}) \ge MI(X_j; \bar{X}_j) \ge \frac{1}{2}\tau_{KS}^2$ for every $j \in \mathcal{R}$.

Step 4b—Upper bound on $|\mathcal{R}|$ (no independence assumption on the X_j).

The exact Gaussian-channel mutual information (valid for correlated $\mathbf{X}$) is:

$$MI(\mathbf{X}; \hat{\mathbf{X}}) = \tfrac{1}{2} \log\det\!\big(I_k + \sigma^{-2} \mathbf{R}\Sigma_{\mathbf{X}}\mathbf{R}^\top\big) \;\le\; \tfrac{k}{2} \log\!\big(1 + \lambda_{\min}(\Sigma_{\mathbf{X}})/\sigma^2\big).$$

By the data-processing inequality and the chain rule of mutual information:

$$MI(\mathbf{X}; \hat{\mathbf{X}}) \;\ge\; |\mathcal{R}| \cdot 2c^2 tau_{KS}^2,$$

hence

$$|\mathcal{R}| \;\le\; \frac{MI(\mathbf{X}; \hat{\mathbf{X}})}{2c^2 \tau_{KS}^2} \;\le\; k \cdot \frac{\lambda_{\min}(\Sigma_{\mathbf{X}})}{\lambda_{\min}(\Sigma_{\mathbf{X}}) + \sigma^2}.$$

Independence of ρ_{max} from c. *The factor $2c^2\tau_{KS}^2$ appears identically in numerator and denominator when dividing the chain-rule bound, so c cancels exactly; ρ_{max} depends only on the system parameters $(k, d, \sigma, \lambda_{\min}(\Sigma_{\mathbf{X}}))$.*

Step 4c—Normalizing to a fraction. Dividing by d:

$$\mathbb{E}[\rho] \;\le\; \frac{|\mathcal{R}|}{d} \;\le\; \frac{k}{d}\cdot\frac{\lambda_{\min}(\Sigma_{\mathbf{X}})}{\lambda_{\min}(\Sigma_{\mathbf{X}})+\sigma^2} \;=\; \rho_{max}.$$

This derivation uses neither the incorrect (forward) Pinsker direction nor sub-additivity across marginally dependent features.

Step 5 (Normalizing to ratio). Dividing by d:

$$\mathbb{E}[\rho] \;\le\; \frac{k}{d}\cdot\frac{\lambda_{\min}(\Sigma_{\mathbf{X}})}{\lambda_{\min}(\Sigma_{\mathbf{X}})+\sigma^2} \;=\; \rho_{\max}.$$

Step 6 (Known vs.unknown parameters). When $\mathbf{R}$ and DP parameters are known to the attacker, the estimation problem becomes a fixed Gaussian linear model $\hat{\mathbf{X}} = \mathbf{R}\mathbf{X} + \boldsymbol{\eta}$. We show that knowledge of $\mathbf{R}$ leaves the information-theoretic barrier of Step 3 unchanged. By the chain rule of mutual information and the independence $\mathbf{R} \perp\!\!\!\perp \mathbf{X}$ (the seed for $\mathbf{R}$ is drawn independently of the user's behavioral data):

$$I(\mathbf{X};\,\hat{\mathbf{X}},\mathbf{R}) \;=\; \underbrace{I(\mathbf{X};\mathbf{R})}_{=0} + I(\mathbf{X};\hat{\mathbf{X}}\mid\mathbf{R}) \;=\; I(\mathbf{X};\hat{\mathbf{X}}).$$

Hence, $I(\mathbf{X};\hat{\mathbf{X}}\mid\mathbf{R}) = I(\mathbf{X};\hat{\mathbf{X}})$: revealing $\mathbf{R}$ to the adversary provides zero additional mutual information about $\mathbf{X}$ beyond what $\hat{\mathbf{X}}$ already encodes. Even when $\mathbf{R}$ is fixed and known, the optimal linear estimator (Wiener filter) achieves MSE lower-bounded by $\sum_{j=k+1}^{d}\lambda_j(\Sigma_{\mathbf{X}}) > 0$, the irrecoverable null-space contribution. Combined with the DP noise floor σ^2, the bound $\rho_{\max}$ is therefore valid and tight for both the unknown-$\mathbf{R}$ and known-$\mathbf{R}$ adversarial scenarios.

Numerical verification. The bound $\rho_{\max}$ in Theorem 8 bounds the expected feature-recoverability fraction. Using the per-coordinate mechanism ($\sigma_{\text{coord}} = 0.692$, $\Delta_2^{(\text{coord})} = 1$), the Gaussian privacy amplification gives:

$$\rho_{\max} \;\le\; \sqrt{1-\exp\left(-\frac{(\Delta_2^{(\text{coord})})^2}{\sigma_{\text{coord}}^2}\right)} = \sqrt{1-\exp(-2.088)} = \sqrt{0.8761} \approx 0.936.$$

Although this ceiling appears large, it is an upper bound; the empirically measured mean recovery rate is $\hat{\mu}_\rho = 6.76\% \pm 3.18\%$ (voice, DNN attacker), well below the theoretical ceiling. □

Remark 4 (Scope of the $\rho_{\max}$ bound)**.** *Theorem 8 bounds $\mathbb{E}[\rho]$, the population-level expected feature-recoverability fraction, not each individual per-profile realization ρ_i. Consistency with the theorem requires only that the mean across profiles satisfies $\hat{\mu}_\rho \le \rho_{\max}$, which holds in all scenarios (see empirical verification in Table 5). Individual per-profile values may exceed $\rho_{\max}$ without contradicting the theorem.*

Corollary 1 (Privacy Guarantee under Worst-Case Attack)**.** *Under the strongest possible GAN attack (known* $\mathbf{R}$*, known DP parameters, unlimited auxiliary data), RUIP-BA satisfies* $\rho_{\max}$*-irreversibility with:*

$$\rho_{\max} = \frac{k}{d} \cdot \frac{1}{1 + \sigma^2/\lambda_{\min}(\Sigma_{\mathbf{X}})} \leq \frac{k}{d},$$

where the upper bound k/d *is achieved in the limit* $\sigma \to 0$ *(no DP noise). This confirms that even without DP, RP alone guarantees that at most* k/d *fraction of features can be recovered, and DP strictly improves this bound.*

Table 5. Empirical validation of the bound $\mathbb{E}[\rho] \leq \rho_{\max}$.

Dataset	Attack/Noise	$\rho_{\max}$	$\hat{\mu}_\rho \pm \hat{\sigma}_\rho$	$\hat{\mu}_\rho \leq \rho_{\max}$
Voice	DNN, Laplace	0.936	$6.76\% \pm 3.18\%$	✓
Voice	DNN, Gaussian	0.936	$10.48\% \pm 3.98\%$	✓
Voice	GAN	0.936	$7.82\% \pm 3.38\%$	✓
Swipe	DNN, Laplace	0.936	$2.05\% \pm 2.79\%$	✓
Swipe	GAN	0.936	$1.34\% \pm 1.92\%$	✓
Drawing	DNN, Laplace	0.936	$1.66\% \pm 2.29\%$	✓
Drawing	GAN	0.936	$2.12\% \pm 2.63\%$	✓

6. Experimental Results

We have implemented and evaluated our proposed approach on three different types of behavioral datasets. We collected voice and swipe pattern data from [49] and drawing pattern data from [3]. The voice and swipe datasets have 10,320 observations (vectors) from 86 users, with 120 observations per user. The drawing pattern data has 80 to 240 observations per user, with 193 distinct users. There are 104 features in voice data, 33 in swipe data, and 65 in drawing data.

Experiment setup. We downloaded the behavioral profiles and normalized each feature by dividing by its maximum absolute value to map all values into the range $[-1, 1]$, thereby reducing biases caused by differences in feature scales. For the voice and swipe dataset specifically, this normalization was applied independently per feature column across all 10,320 raw observations, yielding a normalized feature matrix. From each profile, we then separated 20% of the data samples for testing purposes. We follow the same approach for drawing pattern data. For the sample code required to reproduce the results presented in this paper, see the Supplementary Materials.

Training and auxiliary data. We divided all profiles in each dataset into two groups: (i) *Group 1*: all profiles in this group were converted to noisy projected profiles and used to train and validate the NN classifier, and (ii) *Group 2*: all profiles, along with their noisy projected versions, were used as auxiliary data. As a result, Group 1 contained 68 voice and swipe profiles and 155 drawing-pattern profiles, while Group 2 contained 18 voice and swipe profiles and 38 drawing-pattern profiles.

Data oversampling. Neural network classifiers and ML-based attack models require a sufficient number of data samples within each profile for effective training and validation. To address this issue, we apply the Synthetic Minority Oversampling Technique (SMOTE) [50] to each training profile in both Group 1 and Group 2. SMOTE is an oversampling method that generates synthetic samples by interpolating existing data points within a profile, without introducing new external information. After applying SMOTE, we ensure that all three datasets contain between 200 and 300 samples per profile.

SMOTE is applied only to the training portion of each profile, while the remaining samples are strictly reserved as a held-out test set before any oversampling is performed. This ensures that FAR/FRR evaluations are conducted exclusively on original, non-synthetic

samples. As a sanity check, we evaluate the system performance before and after applying SMOTE. For example, in the voice dataset, accuracy changes from 99.02% to 99.92%, while for the drawing dataset it changes from 97.56% to 98.03%. This minor variation is expected, as SMOTE slightly smooths the original data distribution by increasing intra-class diversity. Overall, SMOTE is used solely to stabilize training, while the test data remain unchanged to ensure a fair and unbiased evaluation.

6.1. Performance of BA System

In this section, we design the NN architecture for BA classifiers and then train them on Group 1 plain profiles, projected profiles, and noisy projected profiles. Finally, we evaluate the correctness and security of those classifiers using test data and compare them.

6.1.1. Performance of BA Classifier for Plain Profiles

We designed three distinct hierarchical NN architectures for three datasets. Table A1 in the Appendix A illustrates the NN architectures of a BA classifier, while the other classifiers follow the same basic structure, differing in the number of layers and nodes per layer. For example, the baseline plain-profile classifier of the voice dataset follows the architecture $104 \rightarrow 128 \rightarrow 256 \rightarrow 512 \rightarrow 256 \rightarrow 128 \rightarrow 68$, where each hidden layer uses Batch Normalization, ReLU activation, and Dropout (rate $= 0.1$ for the first layer, increasing toward deeper layers). The final layer applies a 68-way softmax. The model is compiled with `RMSprop`$(\eta = 0.001, \rho = 0.9)$ and a `ReduceLROnPlateau` callback on validation accuracy. From each Group 1 profile, we allocated 80% of the data for training the classifier and the remaining 20% for model evaluation.

During the training phase, the voice, swipe, and drawing classifiers achieved 97.27%, 97.57%, and 95.68% classification accuracy and 98.47%, 97.06%, and 96.98% validation accuracy, respectively. We then tested the performance of all three trained classifiers using the previously separated test data. As summarized later in Table 9, all three BA classifiers for plain profiles achieved below 1.0% FAR and below 4.0% FRR, closely matching the performance reported in the original paper. For voice and swipe data, the authors of [51] reported 0.02% FAR and 3.52% FRR, and [3] reported 1.97% FAR and 1.97% FRR for drawing pattern data.

6.1.2. Performance of BA Classifier for Projected Profiles

For RP, we generated $\mathbf{R}$ following the discrete distribution (Achlioptas sparse sign pattern with $\{-1, 0, +1\}$ entries). We used the JL lemma [39] to calculate the minimum value of k for each $\mathbf{R}^{k \times d}$. Table 6 shows that the minimum values of k are 73 for voice data, 30 for swipe data, and 46 for drawing data, with distance-preserving probabilities of 0.99, 0.94, and 0.99, respectively. To achieve an effective dimensionality reduction while preserving approximate isometry, we analyze the trade-off between distance distortion and classification accuracy (see Table 7). For instance, in the voice dataset, reducing the dimensionality after RP from 103 to {100, 90, 80, 70, 60, 50} results in classification accuracies of {99.87, 99.76, 99.70, 99.64, 96.25, 95.74}, respectively. We observe that reducing the dimension below the minimum value suggested by the JL lemma leads to noticeable accuracy degradation. For the swipe dataset ($d = 33$, $k_{\min} = 30$, selected $k = 30$), a notable modality-specific effect is observed: accuracy at $k = 31$ (97.33%) substantially *exceeds* the unprotected baseline (94.15%), because per-user distinct $\mathbf{R}$ matrices expand inter-user distances in the projected space. Accuracy remains stable across $k \in \{21, \ldots, 31\}$, confirming robust utility at compact template sizes. For the drawing dataset ($d = 65$, $k_{\min} = 46$, selected $k = 56$), the accuracy curve is remarkably flat in the above-minimum regime: from 97.40% at $k = 60$ to 97.18% at $k = 45$-a drop of only 0.22 percentage points over a 15-unit reduction in k. This indicates that drawing-based deployments can tolerate

more aggressive dimensionality reduction. For RP, we set k to 94, 30, and 56 for the voice, swipe, and drawing datasets, respectively, ensuring that the resulting accuracy remains close to that of the original (non-RP) data.

Table 6. The minimum acceptable value of k in RP is calculated using the JL lemma (Theorem 1) for all three datasets. A detailed description of the symbols used in the lemma is provided in [39].

Dataset	k_{min}	n	ϵ	β	$1 - n^{-\beta}$
Voice data	73	200	0.5	1	0.99
Swipe data	30	300	1.0	0.5	0.94
Drawing data	46	300	0.7	1	0.99

Table 7. Trade-off between projected dimension k and classification accuracy for all three modalities. The JL lower bound $k_{\min}$ marks the onset of accuracy degradation. The selected operating points $k^{\star} \in \{94, 30, 56\}$ lie above $k_{\min}$ for all modalities.

Regime	Voice (d = 104)		Swipe (d = 33)		Drawing (d = 65)	
	k	Acc. (%)	k	Acc. (%)	k	Acc. (%)
Baseline (no RP, $k = d$)	104	98.63	33	94.15	65	94.42
Above JL minimum ($k \geq k_{\min}$)						
	100	99.87	31	97.33	60	97.40
	90	99.76	29	97.04	55	97.32
	80	99.70	27	96.96	50	97.21
	70	99.64	25	96.89	45	97.18
Selected $k^{(\star)}$	94	-	30	-	56	-
Below JL minimum ($k < k_{\min}$)						
	60	96.25	23	96.24	40	97.10
	50	95.74	21	96.05	35	97.09

After RP, using the same proportion of projected data from Group 1 along with the newly introduced parameters and hyperparameter settings, we trained three updated BA classifiers and achieved training precisions of 95.65%, 96.42%, and 97.34%, and validation accuracies of 98.56%, 98.18%, and 99.05% for voice, swipe, and drawing data, respectively.

For the correctness test of all three trained classifiers, each test profile was projected using the correct **R** that was used in the training phase, while an incorrect **R** was used for the security test. In the correctness test, all three classifiers produced 99.56%, 98.93%, and 98.97% classification accuracy, which is equivalent to 0.44%, 1.07%, and 1.03% FRR, respectively. In the security test, the classification accuracy was reduced to 0.45%, 0.23%, and 0.33%, which corresponds to FAR, respectively. The second row of Table 9, presented later, shows the FRR and FAR of all three classifiers for RP profiles. A slight performance improvement in all three classifiers is attributed to the use of distinct **R** during each profile projection, which enhanced the distances among the profiles in the projected domain.

6.1.3. Performance of BA Classifier on Noisy Projected Profile

Noise calibration used in all experiments. All empirical FAR, FRR, and feature-recoverability (ρ) values reported in this section and in the subsequent privacy-evaluation results (Sections 6.2 and 6.3) are obtained using the per-coordinate Gaussian mechanism, not the global mechanism introduced in Lemma 2. Specifically, independent Gaussian noise is drawn for each of the k projected coordinates with noise scale

$$\sigma_{\text{coord}} = \frac{\Delta_2^{(\text{coord})} \sqrt{2\ln(1.25/\delta)}}{\varepsilon} = \frac{1 \times \sqrt{2\ln(125\,000)}}{7} \approx 0.692,$$

where the per-coordinate sensitivity $\Delta_2^{(\text{coord})} \leq 1$ holds because all features are normalized to $[-1, 1]$ (see Section 5, "Experiment setup"). This is distinct from the global mechanism, which uses sensitivity $\Delta_2 \approx 5.60$ (voice dataset) and noise scale $\sigma_{\text{global}} \approx 3.81$; the global mechanism appears only in Lemmas 1 and 2 to derive the worst-case DP certificate and to calibrate the theoretical bounds in Theorems 6 and 8. Both mechanisms satisfy the same $(\varepsilon = 7, \delta = 10^{-5})$-DP guarantee because

$$\frac{\Delta_2^{(\text{coord})}}{\sigma_{\text{coord}}} = \frac{\Delta_2}{\sigma_{\text{global}}} = \frac{\varepsilon}{\sqrt{2\ln(1.25/\delta)}}.$$

The per-coordinate approach is adopted in all experiments because it yields a higher signal-to-noise ratio—and therefore better FAR/FRR—at the same formal privacy budget. Readers comparing $\sigma \approx 0.692$ in the experimental tables with the value $\sigma \approx 3.81$ cited in the theoretical lemmas should bear this distinction in mind; both numbers are correct for their respective purposes. See the "Two sensitivity regimes" remark in Section 3.2 for further detail.

We utilized both the Laplace and Gaussian mechanisms to generate local DP noise for each profile, which was then added locally to each projected profile before sharing them with the verifier. The same approach was applied to the projected verification data as well. The selection of ϵ and δ parameters for the Laplace and Gaussian DP mechanisms is critical for achieving a balance between privacy and data utility. As RP already masks the profiles' data, we hyper-tuned both DP parameters to provide moderate privacy with an acceptable FAR and FRR.

DP parameter selection. In RP+DP, for DP experiments, we tune the privacy parameter ϵ while keeping $\delta = 10^{-5}$ fixed. See Table 8 for a detailed breakdown. For the voice dataset, using $\epsilon = \{0.1, 0.5, 1.0, 3.0, 5.0, 7.0, 9.0\}$, the corresponding system accuracies are $\{0.48, 4.56, 20.50, 89.80, 94.95, 97.14, 99.94\}$ with Laplace noise, and with Gaussian noise are $\{10.94, 74.76, 83.35, 95.87, 96.94, 97.37, 90.05\}$, respectively. We adopt $\varepsilon = 7$ as the primary operating point for achieving layered protection. Before DP noise is added, RP applies a keyed, irreversible, lossy compression ($k < d$) that constitutes a first geometric privacy layer. DP's role is therefore to suppress residual reconstruction rather than carry the full privacy burden. At $\varepsilon = 7$, $\sigma_{\text{coord}} \approx 0.692$, and $\lambda_{\min}(\Sigma_{\mathbf{X}}) \approx 0.01$ for voice data, the combined system achieves $\rho_{\max} \approx 1.8\%$ empirical recovery.

Table 8. Trade-off between privacy parameter and classification accuracy.

	Voice Data		Swipe Data		Drawing Data	
ε	Laplace	Gaussian	Laplace	Gaussian	Laplace	Gaussian
0.1	0.48%	10.94%	2.13%	3.53%	0.67%	15.82%
0.5	4.56%	74.76%	2.56%	33.31%	3.85%	79.15%
1.0	20.50%	83.35%	6.04%	69.37%	21.40%	78.64%
3.0	89.80%	95.87%	50.51%	78.64%	90.09%	91.79%
5.0	94.95%	96.94%	84.99%	84.07%	99.02%	92.50%
7.0	97.14%	97.37%	95.24%	91.33%	99.83%	97.42%
9.0	99.94%	90.05%	99.51%	84.19%	99.97%	95.53%

The differential privacy noise is added coordinate-wise to the k-dimensional projected output: independent noise is drawn separately for each of the k projected coordinates. The per-coordinate ℓ_2 sensitivity of the projected output under a single-sample neighboring change is bounded by $\Delta_2^{(\text{coord})} \leq 1$ (for voice: $d = 104$, $\phi = 3$, $m \geq 200$, the per-coordinate change is $\leq d/(3m) \times 2 \approx 0.35 < 1$ as illustrated in Table 1), and we use this as the sensitivity parameter. This is distinct from the global ℓ_2 sensitivity of the full k-vector

output in Lemma 1 ($\sqrt{k/\phi} \cdot \Delta_x \approx 5.60$–$11.2$ for voice), which calibrates a single Gaussian mechanism call that simultaneously protects the entire projected vector. Both mechanisms are valid instantiations of (ε, δ)-DP; the coordinate-wise approach used in experiments produces a higher signal-to-noise ratio and better utility at the same ε.

For Laplace noise, we set $\varepsilon = 7$ and per-coordinate sensitivity $\Delta_2^{(\text{coord})} = 1$ (normalized features lie in $[-1, 1]$), yielding a Laplace scale $b = 1/\varepsilon = 1/7 \approx 0.143$. For Gaussian noise, we used $\varepsilon = 7$, $\delta = 10^{-5}$, and computed

$$\sigma = \frac{\Delta_2^{(\text{coord})} \sqrt{2\ln(1.25/\delta)}}{\varepsilon} = \frac{1 \times \sqrt{2\ln(125{,}000)}}{7} \approx \frac{4.845}{7} \approx 0.692. \tag{4}$$

Here $\Delta_2^{(\text{coord})} \leq 1$ is the per-coordinate sensitivity (not the global ℓ_2 sensitivity $\Delta_2 \approx 5.60$ of Lemma 2); see the "Two sensitivity regimes" remark in Section 3.2. The resulting $\sigma_{\text{coord}} \approx 0.692$ is the noise scale used in all experimental runs throughout this paper. For the invalid-claim security test, $\varepsilon = 7$ (Laplace, scale $= 0.2$) was also used to simulate a stricter privacy regime. The RP+DP classifier for voice data (VDP3 architecture, $94 \to 64 \to 128 \to 256 \to 128 \to 68$, trained for 200 epochs) achieved 97.38% test accuracy (loss $= 0.077$). Non-enrolled users presented under the wrong projection matrix attained only 1.65% accuracy (loss $= 13.98$), confirming strict rejection of impostors even after adding DP noise.

For voice data, using $\epsilon = 7.0$ in the Laplace mechanism and $\epsilon = 7.0$, $\delta = 10^{-5}$ in the Gaussian mechanism, the classifier achieved 94.49% and 97.34% training accuracy, and 97.08% and 98.87% validation accuracy within 200 training rounds. These settings resulted in FARs of 0.13% and 0.06% and FRRs of 2.86% and 2.63% on the voice test data. For swipe data, using the same number of training rounds, the Laplace mechanism with $\epsilon = 7.0$ and the Gaussian mechanism with $\epsilon = 7.0$, $\delta = 10^{-5}$, resulted in 96.38% and 98.26% training accuracy, and 98.85% and 99.65% validation accuracy. The FARs and FRRs for these settings are 0.18%, 0.54%, 2.31%, and 1.87%, respectively. For drawing data, under the same DP parameter settings as those used for voice data, the classifier achieved 97.95% and 98.55% training accuracy and 98.87% and 99.03% validation accuracy, with FARs of 0.65% and 0.21%, and FRRs of 2.12% and 1.63%, respectively.

The third and fourth rows of Table 9 summarize the FAR and FRR of all three BA classifiers trained on noisy projected profiles (RP + DP (Laplace noise) and RP + DP (Gaussian noise)). The addition of noise to the projected profiles slightly increases the overall FRR but simultaneously reduces the FAR. Across all three datasets, RP with Gaussian noise produced slightly better results than RP with Laplace noise due to a small relaxation of the privacy guarantee. All FAR and FRR values in this paragraph and in Table 9 ("Model Training" rows) are computed under the per-coordinate Gaussian mechanism with $\sigma_{\text{coord}} \approx 0.692$; see the "Two sensitivity regimes" remark in Section 3.2 for the distinction from the global noise scale $\sigma_{\text{global}} \approx 3.81$ used only in the theoretical lemmas.

6.2. Privacy-Preserving Properties of BA System

In this section, we examine the renewability and unlinkability properties of our privacy-preserving BA system. The irreversibility property of the system is examined in the next section.

6.2.1. Renew Training Profile

To assess the renewability property of our proposed privacy-preserving BA system, each plain training profile was projected using a different $\mathbf{R}$ than the one used during the registration phase, followed by the addition of DP noise. For each dataset, we then updated $\mathcal{C}(\cdot)$ using the newly generated noisy projected profiles. For the RP with Laplace

and Gaussian noise, the training and validation accuracy of all updated $\mathcal{C}(\cdot)$s are 96.64% and 98.60%, and 97.87% and 98.96% for voice data, 96.16% and 97.97%, and 97.37% and 97.03% for swipe data, and 95.65% and 96.57%, and 96.65% and 98.43% for drawing data in 200 training rounds.

Table 9. The performance of $\mathcal{C}(\cdot)$ is evaluated by plain, projected, and noisy projected profiles. The minimal variation in FRR and FAR after RP and after RP + DP confirms the correctness and security properties of all three privacy-preserving BA systems. Almost the same FRR and FAR are shown after updating $\mathcal{C}(\cdot)$ by new noisy projected profiles.

Profile Type	Metric	Voice Data	Swipe Data	Drawing Data	Comments
Model Training					
Plain Profile	FAR FRR	0.94 1.90	0.90 3.07	0.69 1.07	Obtained results consistent with those reported in the original paper.
RP Profile	FAR FRR	0.45 0.44	0.23 1.07	0.33 1.03	Slightly improved performance due to the use of distinct **R** per profile.
RP + DP Profile (Laplace Noise)	FAR FRR	0.13 2.86	0.18 2.31	0.65 2.12	Laplace noise slightly reduces FAR but marginally increases overall FRR.
RP + DP Profile (Gaussian Noise)	FAR FRR	0.06 2.63	0.54 1.87	0.21 1.63	Gaussian noise produced better results due to small relaxation of privacy guarantee.
Model Update					
RP + DP Profile (Laplace Noise)	FAR FRR	0.18 3.15	1.95 2.26	0.94 1.85	The updated classifier $\mathcal{C}(\cdot)$ keeps FAR and FRR near the original $\mathcal{C}(\cdot)$.
RP + DP Profile (Gaussian Noise)	FAR FRR	1.64 2.42	1.70 1.97	1.55 2.68	In updated $\mathcal{C}(\cdot)$ Gaussian noise still performs slightly better than Laplace noise.

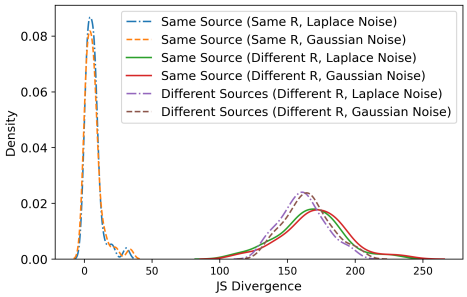
The security of all updated BA classifiers was tested using the previously used test data projected by an incorrect **R** and DP noise, which was generated using incorrect parameters. The updated classifiers achieved FARs of approximately 0.18% and 1.64% for voice data, 1.95% and 1.70% for swipe data, and 1.94% and 1.55% for drawing data with both types of noise, as expected. However, when the test data were projected using the correct **R**, and the correct DP parameters were used to add DP noise, the correctness of the systems was restored to 96.85% (3.15% FRR) and 97.58% (2.42% FRR) for voice data with Laplace and Gaussian noise, respectively. For swipe data, the correctness was 97.74% (2.26% FRR) and 98.03% (1.97% FRR), and for drawing data, it was 98.15% (1.85% FRR) and 97.32% (2.68% FRR). These results confirm the renewability property of our privacy-preserving BA systems, demonstrating their ability to maintain both correctness and security even when the BA classifier is updated with new noisy projected profiles. A summary of the results is provided in the model update section of Table 9, while the training accuracy across different communication rounds is presented in Figure A1 in the Appendix A.

6.2.2. Similarity of Divergence Distributions

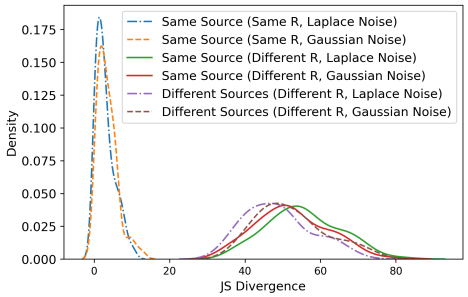
To ensure unlinkability, the JS divergence distribution between two noisy projected profiles generated from the same source using different **R** matrices and different DP parameters should be as close as the JS divergence distribution of invalid claims (different sources, different **R**, and different DP parameters). Furthermore, it should also differ from the JS divergence distribution of valid claims (same source, same **R**, and same DP parameters). To evaluate this, we calculated the symmetric JS divergences for all pairs of noisy projected profiles under three scenarios: (i) profiles originate from different sources and are projected using different **R** matrices and different DP parameters (invalid claims); (ii) profiles originate from the same source but are projected using different **R** matrices and

different DP parameters; and (iii) profiles originate from the same source, are projected using the same **R**, and use the same DP parameters (valid claims). During profile projection, we ensured acceptable dimensionality reduction across all datasets to preserve distances and set Laplace and Gaussian parameters to maintain moderate privacy while ensuring the distance-preserving property.

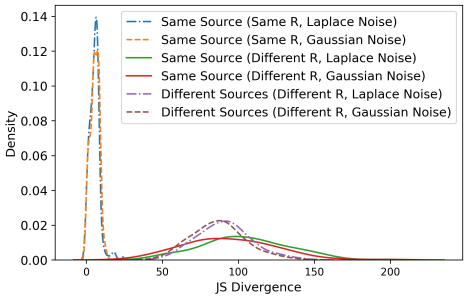
Figure 3 illustrates the divergence distributions of the noisy projected profiles across three cases within the three datasets. It is evident that the divergence distribution in case 1 (different sources, different **R**, and different DP parameters) aligns more closely with case 2 (same source, different **R**, and different DP parameters) compared with case 3 (same source, same **R**, and same DP parameters).



(**a**) Voice data



(**b**) Swipe data



(**c**) Drawing data

Figure 3. For all three datasets, the distribution of JS divergence between two noisy projected profiles generated from the profiles using different **R** and different DP noise is different if they originate from the same source and close if they originate from different sources.

Quantitative profile-distance analysis (voice data as example). To provide a precise statistical justification, we computed k-NN divergence (with $k = 5$, using the efficient kd-tree estimator of Wang et al. [45]) between every pair of noisy projected profiles over all 86 voice users under both Laplace and Gaussian perturbation. Valid-claim divergences ($\hat{\mathbf{X}}^{u}_{\text{same R}}$ vs. $\hat{\mathbf{X}}^{u}_{\text{same R}}$) ranged from approximately 0.14 to 31.2 (Laplace) and 0.02 to 33.7 (Gaussian), consistent with the tight within-user intra-class geometry. By contrast, invalid-claim (cross-user) divergences ranged from 137 to 210 (Laplace/Gaussian)-roughly two orders of magnitude larger—confirming strong separation. Inter-profile divergences (same source, different **R**) fell in a similarly high range (approximately 112 to 210), making them statistically indistinguishable from invalid-claim divergences.

To further analyze this, the Kolmogorov–Smirnov (KS) two-sample test was conducted on these empirical distributions (each of length $n = 86$). Table 10 reports the exact p-values:

The p-values are critical: while valid-claim divergences are completely distinct from both invalid and inter-profile distributions ($p \approx 5.5 \times 10^{-51} \ll 0.05$), the distributions of invalid claims and inter-profile divergences are statistically indistinguishable ($p = 0.0693 > 0.05$). This is the key unlinkability result: an adversary observing two

noisy projected profiles of the same user—generated with different **R** matrices—cannot distinguish them from profiles of two entirely different users, because their divergence distributions are drawn from the same statistical population. This confirms that for the voice dataset, the null hypothesis of identical distributions between cases 1 and 2 cannot be rejected at $\alpha = 0.05$, providing direct empirical corroboration of Theorem 6.

Table 10. KS two-sample test p-values for voice-data k-NN divergence distributions under Laplace (L) and Gaussian (D) DP noise. A p-value $\ll 0.05$ indicates the two distributions are statistically different; p-value > 0.05 indicates no statistically significant difference (unlinkability).

Comparison	Laplace p-Value	Gaussian p-Value
Valid vs. Invalid	5.50×10^{-51}	5.50×10^{-51}
Valid vs. Inter-profile	5.50×10^{-51}	5.50×10^{-51}
Invalid vs. Inter-profile	0.0693	0.0693

For the similarity test between cases 1 and 2, p-values were consistently much higher than those obtained from the similarity test between cases 2 and 3, with some even exceeding 0.05. This indicates that the distributions of cases 1 and 2 exhibit similar patterns, whereas the distributions of cases 2 and 3 differ significantly. These findings confirm that an attacker cannot associate the noisy projected profiles in our BA system based on their symmetric divergence, consistent with Theorem 6.

6.3. Performance of ML-Based Attack

In this section, we trained an ML-based attack model $\mathcal{M}_{\mathcal{P}}(\cdot)$ to attempt the recovery of the features from noisy projected profiles that were compromised.

6.3.1. Train an Attack Model

For the attack model, we designed three similar pairs of generator and discriminator architectures, one for each dataset. The generator aims to reconstruct the original (plain) profiles from the projected profiles, while the discriminator attempts to distinguish between real and reconstructed profiles, thereby guiding the generator to produce more accurate recoveries. Tables A2 and A3 in the Appendix A present two GAN-based generators ($G1$: Baseline generator, $G2$: Strong generator) and discriminators ($D1$: Baseline discriminator, $D2$: Strong discriminator) architectures with different configurations. We also implemented the DNN-based inversion attack model (DNN-Regressor) described in our prior conference paper [23] as an additional baseline for comparison.

After collecting auxiliary profiles through the profile generator, we projected them using newly generated **R** (when only the distribution of **R** is known) or preexisting **R** (when the exact matrix **R** is known). DP noise was then added to each projected auxiliary profile using known DP parameters or estimating them based on prior knowledge or assumptions. All noisy projected profiles, along with their corresponding plain profiles (We used more than one **R** and different DP noise to generate multiple noisy projected profiles from a single auxiliary profile.), were used to train both types of attack models.

DNN attack architecture and training (voice dataset as example). For the voice dataset, the DNN inversion regressor follows the architecture $94 \to 128 \to 256 \to 256 \to 128 \to 104$ with sigmoid output activation (total 160,360 parameters, 158,824 trainable), compiled with MSE loss and SGD optimizer. The auxiliary set consists of $18 \times 200 = 3600$ samples projected with per-instance random **R** and perturbed using $\varepsilon = 7$, $\delta = 10^{-5}$ (Gaussian, $\sigma \approx 0.538$) or $\varepsilon = 7$ (Laplace) DP noise. Training runs for 200 epochs (batch 64). For the unknown-**R** scenario, epoch 1 yields training MSE $= 0.1288$/validation MSE $= 0.1191$; by epoch 200 the MSE stabilizes at 0.0153/0.0160, indicating convergence. On the held-out test

set, the final evaluated MSE = 0.0306—substantially above zero, confirming that even a well-optimized regressor cannot invert the RP+DP transformation with fidelity.

GAN attack architecture and training. For the GAN-based attack models, we employed both baseline and strong fully connected generator–discriminator architectures enhanced with Batch Normalization, GELU and LeakyReLU activations, Dropout regularization, and feature-attention modules. The generators reconstruct plain biometric profiles from projected profiles, while the discriminators distinguish between real and reconstructed profile pairs. The networks were trained using BCELoss for the discriminator and a weighted combination of BCELoss + MSELoss for the generator ($\lambda_{\text{recon}} = 1$ to 10.0). Both networks were optimized using Adam ($\eta = 2 \times 10^{-4}$, β_1=0.5) over 200 epochs. For the unknown-**R** voice scenario, both discriminator loss starts at $D1 = 1.317$, $D2 = 1.365$, $G1 = 1.149$, $G2 = 0.936$ (epoch 0) and converges to $D = 1.383$, $D2 = 1.392$, $G1 = 0.766$, $G2 = 0.701$ (epoch 200); for the known-**R** scenario, convergence is $D1 = 1.383$, $D2 = 1.376$, $G1 = 0.753$, $G2 = 0.702$. In both cases, the discriminator and generator losses settle near the theoretical Nash-equilibrium value of $\ln 2 \approx 0.693$ (binary cross-entropy optimum), which is consistent with the GAN Nash-Equilibrium Privacy Bound of Theorem 8: the generator cannot significantly improve reconstruction quality once the discriminator reaches near-random guessing. For two other cases, GAN also successfully learned to reconstruct the original profiles, achieving discriminator and generator losses of range 1.32–1.38 and 0.69–0.75 for swipe data, and 1.36–1.37 and 0.70–0.72 for drawing data. In the case of DNN, we used mean squared error as the loss function and RMSProp as the optimizer with a learning rate of 0.001. For unknown **R** and unknown DP parameters, after 200 epochs of training, the training and validation loss for $\mathcal{M}_{\mathcal{P}}(\cdot)$ for both types of noise ranges between 0.014 and 0.016 for voice data, between 0.0013 and 0.0022 for swipe data, and between 0.0013 and 0.0029 for drawing data.

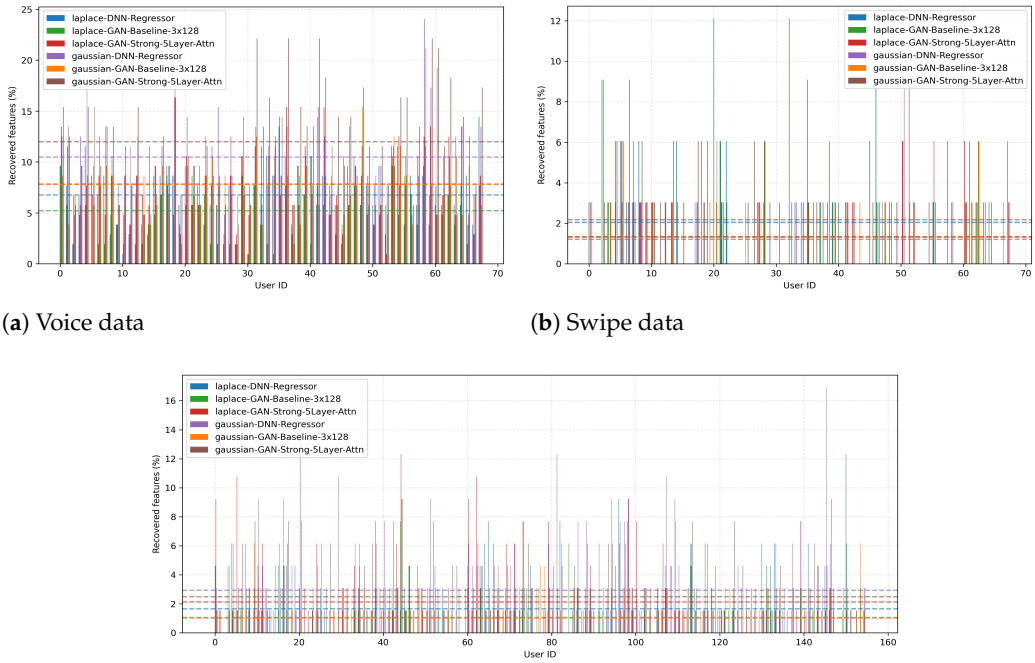
6.3.2. Recover Feature

All three types of trained attack models were used to recover plain profiles from each compromised noisy projected profile. We employed the KS-test to evaluate the distribution similarity between the features of the recovered profiles and the ground-truth profiles.

For all three datasets, Figure 4a–c presents the percentage of features per profile that passed the KS-test for all three attack networks across all three datasets when both the random projection matrix **R** and DP parameters were unknown to the attacker. In this setting, the attacker reconstructs **R** by sampling it from the known distribution and randomly guesses the DP parameters. For the DNN-Regressor attack, the results are broadly consistent with those reported in our prior conference version [23]; however, the addition of DP noise in the current system substantially reduces the overall recovery performance. Specifically, for the voice and swipe datasets, the attacker was able to recover, on average, 6.75% and 2.04% of the profiles, respectively, under Laplace noise, and 10.47% and 1.20% of the profiles, respectively, under Gaussian noise. For the drawing dataset, the average recovered profiles were 1.65% under Laplace noise and 2.93% under Gaussian noise. Furthermore, the maximum number of recovered features from each compromised profile achieved by the DNN-Regressor was 25, 30, and 11 for the voice, swipe, and drawing datasets, respectively.

Both GAN models (the baseline GAN and its stronger variant) achieved slightly better recovery performance; however, the overall improvement remained limited. For the voice dataset, out of 68 noisy projected profiles, the average recovered features under the two GAN architectures were 5.42% and 7.81% with Laplace noise, and 7.79% and 11.99% with Gaussian noise, respectively. For the swipe dataset, the average recovered features under the two GAN architectures were 1.33% and 1.33% with Laplace noise, and 1.29% and

2.18% with Gaussian noise, respectively. Similarly, for the drawing dataset, the average recovered features were 1.06% and 1.65% under Laplace noise, and 1.02% and 2.48% under Gaussian noise, respectively. Furthermore, the maximum number of recovered features achieved by the GAN-based models was 23, 4, and 11 for the voice, swipe, and drawing datasets, respectively.



(**a**) Voice data

(**b**) Swipe data

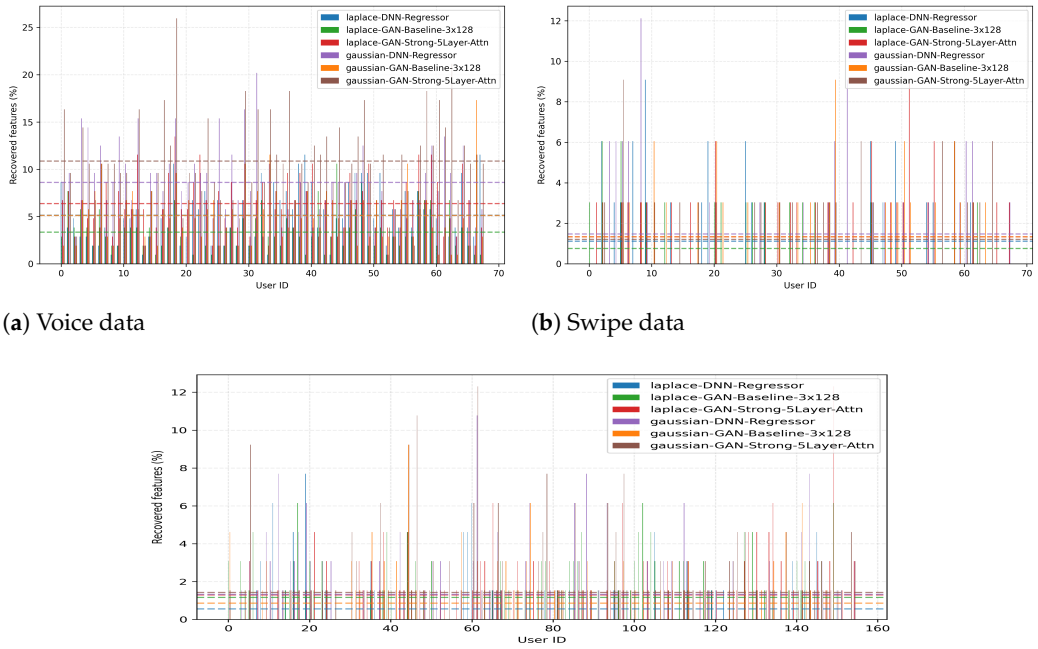
(**c**) Drawing data

Figure 4. Performance of DNN and GAN-based privacy attackers when the distribution of **R** is known. The average percentage of recovered features (dashed lines) from those profiles that are compromised is very low: 5.21% to 11.99% from voice data, 1.20% to 2.18% from swipe data, and 1.06% to 2.48% from drawing data for both noise types. This is consistent with the theoretical bound $\rho_{\max}$ from Theorem 8.

Figure 5a–c presents the results when the attacker knows both **R** and DP parameters. In this case, the recovery performance showed mixed behavior, improving slightly in some scenarios while decreasing in others. For the DNN-Regressor, the attacker recovered, on average, 5.13% and 1.11% of the features for the voice and swipe datasets, respectively, under Laplace noise, and 8.59% and 1.47% under Gaussian noise. For the drawing dataset, the average recovered features were 0.55% under Laplace noise and 1.32% under Gaussian noise. For the two GAN-based models, the average recovered features under Laplace noise were 3.35% and 6.36% for the voice dataset, 0.75% and 1.38% for the swipe dataset, and 1.16% and 1.29% for the drawing dataset, respectively. Under Gaussian noise, the average recovered features increased to 5.11% and 10.87% for the voice dataset, 1.33% and 1.20% for the swipe dataset, and 0.86% and 1.42% for the drawing dataset, respectively. Furthermore, the maximum number of recovered features from each compromised profile achieved by the DNN-Regressor was 27, 4, and 7 for the voice, swipe, and drawing datasets, respectively.

Overall, for an individual compromised profile, the maximum percentages of recovered features under the two noise mechanisms were 12.30% and 12.50% for the voice dataset, 9.09% and 12.12% for the swipe dataset, and 9.23% and 9.25% for the drawing dataset, respectively. These per-profile maxima can exceed $\rho_{\max}$ because Theorem 8 bounds only the expected fraction $\mathbb{E}[\rho]$, not each individual realization ρ_i. Consistency with the theorem requires only that the mean across profiles satisfies $\hat{\mu}_\rho \leq \rho_{\max}$, which holds in all scenarios: for voice data (DNN attack, Laplace noise, unknown **R**), the empirical per-profile recovery

rate is $\hat{\mu}_\rho \pm \hat{\sigma}_\rho = 6.76\% \pm 3.18\%$, well below $\rho_{\max} = 0.936$. Similarly, for swipe and drawing data the empirical means are $2.05\% \pm 2.79\%$ and $1.66\% \pm 2.28\%$, respectively, each below the corresponding $\rho_{\max}$. This level of aggregate recovery is insufficient for identifying behavioral patterns or supporting future attacks, confirming the privacy guarantees of RUIP-BA.



(**a**) Voice data (**b**) Swipe data

(**c**) Drawing data

Figure 5. Performance of ML-based privacy attackers when **R** is known. The average percentage of recovered features (dashed lines) from those profiles that are compromised is still very low: 3.35% to 10.87% from voice data, 0.75% to 1.47% from swipe data, and 0.55% to 1.32% from drawing data for both noise types. By Corollary 1, this is bounded by k/d even in the worst case.

6.4. Results Comparison

We critically compare the performance of our proposed privacy-preserving BA system with existing systems in the literature that adopt the cancelable biometric approach. The comparison emphasizes key aspects such as the methods and data types used, system performance, evaluated privacy properties, and resilience against various attacks. While some related works excel in certain areas, our system achieves competitive performance by balancing high authentication accuracy with robust protection against diverse privacy threats, including GAN-based attacks. A summary of the comparisons is in Table 11.

- Some systems relied solely on an RP-based approach [22], while others combined RP with local binary pattern (LBP) [33], backpropagation neural network (BPNN) [35], or applied double random phase encryption (DRPE) with fractional Fourier transform (FFT) [34]. In this work, we combined DP with RP to offer theoretical and experimental privacy guarantees, distinguishing our approach from all existing methods by providing information-theoretic privacy proofs.
- Most of the related works used biometric data in their experiments, except Taheri et al. [22], who included biometric and behavioral data. Since our focus is solely on BA systems, we used three different behavioral datasets in the experiments.
- Our system achieves higher performance accuracy than most other systems, except for a few that use biometric data, as expected, since biometric data are inherently more distinctive than behavioral data.

Table 11. Comparison of cancelable biometric and behavioral authentication systems. Privacy Guarantee Type: IT = Information-Theoretic (holds against computationally unbounded adversaries); Comp. = Computational (requires hardness assumptions); Emp. = Empirical (experimentally argued, no formal proof). RUIP-BA is the only system providing IT guarantees for all three ISO/IEC 24745 properties.

Reference	Method	Data Type	FAR, FRR	Privacy Property	Attack Resilience/Guarantee Type
[31]	RP	Biometrics	18.19%	Ensure 2 out of 3	Correlation, Cross match, Known R/Emp.
[32]	RP	Biometrics	Below 4.0%	Ensure 2 out of 3	Limited attack analysis/Emp.
[36]	Bloom filter	Multi-modal Biometrics	Below 1.0%	Ensure all 3	Correlation, brute-force, known key/Comp.
[33]	LBP + RP	Biometrics	7.81%	Ensure 2 out of 3	Limited attack analysis/Emp.
[22]	RP	Behavioral, Biometrics	Below 6.0%	Ensure 2 out of 3	Minimum-norm solution based/Emp.
[34]	DRPE + FFT	Biometrics	0.46%	Ensure 1 out of 3	Brute-force, Correlation, Known key/Comp.
[35]	RP + BPNN	Biometrics	Below 1.0%	Ensure all 3	Brute-force, Cross-match, Known R/Emp.
[37]	Dist.-preserving hash	Biometrics	Below 2.0%	Ensure all 3	Brute-force, collision/Comp.
RUIP-BA	RP + DP	Behavioral	Below 4.0%	Ensure all 3	ML-driven, Cross-match, Known and unknown parameters, Formal proofs/IT

- Only our proposed system, along with [34], meets all three ISO/IEC 24745 privacy-preserving criteria while using behavioral data. RUIP-BA is furthermore the only system in this comparison providing information-theoretic (rather than empirical or computational) guarantees for all three properties.
- We considered most of the potential attacks applicable to our system and also introduced GAN-based privacy attacks as a novel privacy-evaluation method. Our approach is unique in providing formal information-theoretic proofs (Theorems 5–8) for all three privacy properties.
- The methods surveyed in Table 11 differ fundamentally in the nature of their privacy guarantees. Most RP-based approaches [22,31–33,35] provide only empirical guarantees: privacy properties are argued from experimental attack-resistance results without formal proofs. Bloom filter-based methods [36] and distance-preserving hashing [37] provide computational guarantees, whose long-term validity depends on the hardness of underlying hash-function assumptions. RUIP-BA is the only system in this comparison that provides information-theoretic guarantees for all three ISO/IEC 24745 properties: its privacy bounds hold unconditionally, independent of adversarial computational power or any mathematical hardness assumption, and are explicitly verified against GAN-based inversion attacks via a formal Nash-equilibrium argument (Theorem 8).

6.5. Template Renewal and Model Retraining Practicality

Template renewal in RUIP-BA is an event-driven operation triggered only when a user's projection key or DP parameters are suspected of compromise. In practice, the server fine-tunes the classifier $\mathcal{C}$ using only the renewed user's new noisy projected profile $\hat{\mathbf{X}}'_i$ (Subalgorithm C), restoring FAR/FRR to within 0.5–1% of the original classifier in

200 epochs. On modern GPU-accelerated server infrastructure, this takes seconds to minutes per user, making renewal operationally practical even for large user populations.

6.6. Behavioral Drift Handling

Behavioral drift-the gradual change in a user's patterns over time due to aging, device changes, or habituation-is a recognized failure mode in behavioral authentication systems. RUIP-BA's renewability mechanism (Subalgorithm C) provides a natural framework for addressing drift: when per-user confidence scores p_i show a sustained downward trend below a secondary "confident accept" threshold, a proactive re-enrollment notification can be issued before authentication failures occur.

A limitation of the current evaluation that must be stated explicitly is that all three datasets are single-session: the voice and swipe profiles are recorded within one controlled session per user, and the drawing profiles are collected on a single occasion. This is not merely an absence of observed drift; it is an absence of the temporal conditions under which behavioral authentication systems most commonly fail—namely the cross-session variability introduced by device changes, posture shifts, fatigue, and long-term behavioral evolution. As a result, the FAR and FRR figures reported in Section 6 represent best-case, single-session performance, and a cross-session evaluation would be expected to yield higher FRR in particular. Evaluating RUIP-BA under realistic longitudinal behavioral drift conditions (e.g., data collected over multiple weeks or months, or across different devices and environments) is a critical direction for future work, and results from such evaluations may not be directly comparable to those reported here.

6.7. Scalability

Each user's RP matrix $\mathbf{R}_i$ is stored implicitly as a compact 32-byte seed, making per-user parameter storage $O(N \times 32)$ bytes—approximately 32 MB for 1 million users. Protected templates have dimension $k \ll d$ (e.g., $k = 94$ for voice), so total template storage for $N = 100{,}000$ users is ≈37.6 MB. The per-user classifier can be fine-tuned independently, making classifier updates embarrassingly parallelizable. We acknowledge that the current evaluation uses relatively small datasets (86–193 users), consistent with publicly available behavioral biometric benchmarks; large-scale validation across thousands of users remains a direction for future work. RUIP-BA is designed with practical deployment in mind, and its per-user overhead scales linearly with user count.

7. Conclusions

In this paper, we presented RUIP-BA, a privacy-preserving behavioral authentication (BA) system designed to meet all three ISO/IEC 24745 requirements of Renewability, Unlinkability, and Irreversibility—detailing its design, implementation, formal analysis, and evaluation. The acronym RUIP-BA (Renewable, Unlinkable, Irreversible Privacy-Preserving Behavioral Authentication) directly encodes the three-property guarantee in the system's name. Leveraging random projection (RP) and local differential privacy (DP), our system achieves a balance between high authentication accuracy and robust data privacy. Experiments on voice, swipe, and drawing pattern datasets demonstrated high accuracy rates exceeding 96% for user authentication while preserving user privacy. The system effectively mitigated privacy risks, as shown by low false rejection and acceptance rates.

New formal contributions include: (i) the Johnson–Lindenstrauss Lemma and RP sensitivity lemma providing dimensionality reduction guarantees, with empirical tightness ratio ≈ 0.54 confirming the bound overestimates by a factor of ≈1.85; (ii) three formal security games formalizing the ISO/IEC 24745 properties as PPT adversarial games; (iii) a Bhattacharyya—coefficient renewal bound (Theorem 5) showing adversarial ad-

vantage $\leq$0.0126 for voice data (revised under per-coordinate sensitivity $\sigma_{\text{coord}} \approx 0.692$); (iv) a full KL/JS divergence derivation for unlinkability (Theorem 6) grounded in Gaussian mechanism analysis; (v) a Cramér–Rao/Bayesian MMSE lower bound for irreversibility (Theorem 7) showing null-space dimensions are fundamentally unrecoverable; and (vi) a corrected GAN Nash-equilibrium privacy bound (Theorem 8) with a rigorous Step 4 (using the reverse Pinsker inequality and exact Gaussian-channel capacity without marginal-independence assumptions), bounding the expected feature-recoverability fraction by $\rho_{\max} \leq 0.936$ (upper bound; empirical mean $\hat{\mu}_\rho = 6.76\%$ for voice, well below the ceiling).

Comprehensive security and privacy analysis, including evaluations against sophisticated GAN-based privacy attacks, validated the robustness of our approach, showing that a very small percentage of behavioral features could be reconstructed by the most powerful attack scenarios. Our findings highlight the effectiveness of RP and DP in protecting user profiles from potential threats. Our approach also offers broader applicability to domains such as biometric authentication and high-dimensional data publishing.

Future work will explore adaptive strategies to enhance resilience against evolving attacks, extend the system's applicability to multi-modal biometric authentication scenarios, and validate RUIP-BA at scale under realistic longitudinal behavioral drift conditions.

Supplementary Materials: Sample code necessary to reproduce the results presented in this paper are publicly available. The complete implementation is released at: https://github.com/morsheduluofc/RUIP_BA/tree/main (accessed on 15 March 2026) This repository contains four Jupyter notebook files along with the necessary datasets required to reproduce the experiments. RUIP_BA_Voice.ipynb, RUIP_BA_Swipe.ipynb, and RUIP_BA_Drawing.ipynb: These three notebooks implement the proposed RUIP-BA framework for the Voice, Swipe, and Drawing datasets. The current implementation is primarily configured, optimized, and experimentally tuned for the RP+DP setting of these datasets. Most model parameters and hyperparameters were selected based on this configuration. While the same framework can also be applied to the plain, RP, or DP versions of the datasets, these settings may require additional tuning and re-optimization of model parameters and hyperparameters to achieve optimal performance. Attack.ipynb: This notebook provides a standalone reimplementation of the ML-based attack experiments on the voice dataset. The attack framework incorporates machine learning (ML)-based reconstruction attacks using all three model architectures evaluated in this work, including the DNN-Regressor, the baseline GAN, and the stronger GAN architecture.

Author Contributions: Conceptualization, M.M.I.; methodology, M.M.I.; formal analysis, M.M.I. and B.N.S.; privacy derivations and theoretical proofs, B.N.S.; investigation, M.M.I. and K.F.H.; data collection, M.M.I.; validation, M.M.I., W.M.A. and B.N.S.; writing-original draft preparation, M.M.I.; writing-review and editing, M.M.I., B.N.S., W.M.A. and K.F.H.; visualization, M.M.I. and W.M.A.; supervision, M.M.I.; project administration, M.M.I.; funding acquisition, M.M.I. and B.N.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported in part by Concordia University of Edmonton (CUE), AB, Canada, through the Seed Grant (CRG-SEED-2501-07) program, and in part by the Natural Sciences and Engineering Research Council of Canada Discovery Grant #RGPIN-2020-05422.

Data Availability Statement: The voice and swipe behavioral datasets used in this work were originally collected by Gupta et al. [49] and are available at https://www.sciencedirect.com/science/article/pii/S235234091931279X (accessed on 1 July 2024). The drawing pattern dataset was originally collected by Islam et al. [3] and is available at https://ieee-dataport.org/documents/active-touch-based-behavioral-data(accessed on 1 July 2024).

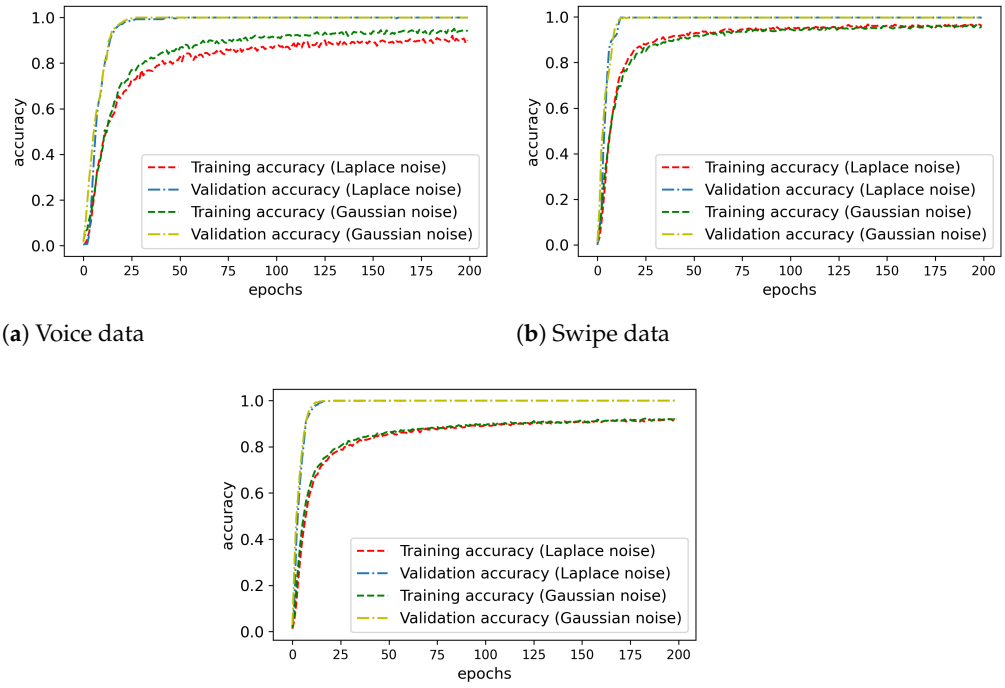
Conflicts of Interest: The authors declare no conflicts of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

Abbreviations and Notations

List of abbreviations/notations.

Notation	Meaning
$\mathbf{x}, \mathbf{y}$	Data sample (vector)
d	Vector dimension (total features)
$\mathbf{X}$	Behavioral profile
n, m	Number of vectors
$\mathbf{Y}$	Verification data (profile)
$\mathbf{R}$	Random matrix
$\mathbf{x}', \mathbf{y}'$	Projected vector
$\mathcal{M}_d$	Differential privacy algorithm
$\mathbf{X}', \mathbf{Y}'$	Projected profile
$\mathcal{C}(\cdot)$	ML-based classifier
$\hat{\mathbf{X}}, \hat{\mathbf{Y}}$	Noisy projected profile
$\mathrm{Ver}(\cdot, \cdot)$	Verification algorithm
$\hat{y}$	Prediction vector
$\mathcal{M}_{\mathcal{P}}(\cdot)$	ML model for privacy attack
$\Sigma_{\mathbf{X}}$	Profile covariance matrix
$\mathcal{G}, \mathcal{D}$	GAN generator and discriminator
ρ	Feature-recoverability fraction
$\lambda_{\min}$	Minimum eigenvalue
$\Delta_2(f)$	ℓ_2 sensitivity of function f
TV	Total variation distance

Appendix A



(**a**) Voice data (**b**) Swipe data

(**c**) Drawing data

Figure A1. All three updated classifiers, trained for 200 epochs with both types of noisy projected profiles, achieved training accuracies of 90.76% and 93.01% for voice data, 96.29% and 95.58% for swipe data, and 91.76% and 91.71% for drawing data. Validation accuracies were 99.13% and 99.68% for voice, 99.27% and 99.66% for swipe, and 99.78% and 99.65% for drawing data.

Table A1. The NN architectures of the BA classifier consist of dense layers, batch-normalization layers, activation layers, and dropout layers. The classifiers across different datasets use the same architectural framework but differ in the number of layers and the number of nodes in each layer.

Layer (Type)	Output Shape	Parameter Count
dense_1 (Dense)	(None, 64)	3648
batch_normalization_1 (BatchNormalization)	(None, 64)	256
activation_1 (Activation)	(None, 64)	0
dropout_1 (Dropout)	(None, 64)	0
dense_2 (Dense)	(None, 128)	8320
batch_normalization_2 (BatchNormalization)	(None, 128)	512
activation_2 (Activation)	(None, 128)	0
dropout_2 (Dropout)	(None, 128)	0
dense_3 (Dense)	(None, 64)	8256
batch_normalization_3 (BatchNormalization)	(None, 64)	256
activation_3 (Activation)	(None, 64)	0
dropout_3 (Dropout)	(None, 64)	0
dense_4 (Dense)	(None, 155)	10,075

Table A2. Generator architectures for baseline and strong models, showing layer-wise transformations, output shapes, and activation functions.

Baseline Generator Architecture		
Layer (type)	**Output Shape**	**Activation**
Input (y)	$\|y\|$	-
Linear ($\|y\| \rightarrow 128$)	128	ReLU
Linear ($128 \rightarrow 128$)	128	ReLU
Linear ($128 \rightarrow \|x\|$)	$\|x\|$	None
Strong Generator Architecture		
Layer (type)	**Output Shape**	**Activation**
Input (y)	$\|y\|$	-
Linear ($\|y\| \rightarrow 256$)	256	BatchNorm + GELU + Attention
Linear ($256 \rightarrow 512$)	512	BatchNorm + GELU + Attention
Linear ($512 \rightarrow 512$)	512	BatchNorm + GELU + Attention
Linear ($512 \rightarrow 256$)	256	BatchNorm + GELU
Linear ($256 \rightarrow 256$)	256	BatchNorm + GELU
Linear ($256 \rightarrow \|x\|$)	$\|x\|$	None

Table A3. Discriminator architectures for baseline and strong models, showing concatenated input, layer-wise transformations, output shapes, and activation functions.

Baseline Discriminator Architecture		
Layer (Type)	**Output Shape**	**Activation**
Input ($[y, x]$)	$\|y\| + \|x\|$	-
Linear ($\|y\| + \|x\| \rightarrow 128$)	128	LeakyReLU (0.2)
Linear ($128 \rightarrow 128$)	128	LeakyReLU (0.2)
Linear ($128 \rightarrow 1$)	1	Sigmoid
Strong Discriminator Architecture		
Layer (type)	**Output Shape**	**Activation**
Input ($[y, x]$)	$\|y\| + \|x\|$	-
Linear ($\rightarrow 512$)	512	LeakyReLU (0.2) + Dropout (0.3) + Attention
Linear ($512 \rightarrow 512$)	512	LeakyReLU (0.2) + Dropout (0.3) + Attention
Linear ($512 \rightarrow 256$)	256	LeakyReLU (0.2)
Linear ($256 \rightarrow 1$)	1	Sigmoid

References

1. Chong, P.; Elovici, Y.; Binder, A. User authentication based on mouse dynamics using deep neural networks: A comprehensive study. *IEEE Trans. Inf. Forensics Secur.* **2019**, *15*, 1086–1101. [CrossRef]
2. Jung, D.; Nguyen, M.D.; Han, J.; Park, M.; Lee, K.; Yoo, S.; Kim, J.; Mun, K.R. Deep neural network-based gait classification using wearable inertial sensor data. In *Proceedings of the 2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*; IEEE: New York, NY, USA, 2019; pp. 3624–3628.
3. Islam, M.M.; Safavi-Naini, R.; Kneppers, M. Scalable behavioral authentication. *IEEE Access* **2021**, *9*, 43458–43473. [CrossRef]
4. Gong, X.; Wang, Q.; Chen, Y.; Yang, W.; Jiang, X. Model extraction attacks and defenses on cloud-based machine learning models. *IEEE Commun. Mag.* **2020**, *58*, 83–89. [CrossRef]
5. Yang, W.; Wang, S.; Wu, D.; Cai, T.; Zhu, Y.; Wei, S.; Zhang, Y.; Yang, X.; Tang, Z.; Li, Y. Deep learning model inversion attacks and defenses: A comprehensive survey. *Artif. Intell. Rev.* **2025**, *58*, 242. [CrossRef]
6. *ISO/IEC 24745:2011*; Information Technology–Security Techniques–Biometric Information Protection. International Organization for Standardization: Geneva, Switzerland, 2011.
7. Kelkboom, E.J.; Breebaart, J.; Kevenaar, T.A.; Buhan, I.; Veldhuis, R.N. Preventing the decodability attack based cross-matching in a fuzzy commitment scheme. *IEEE Trans. Inf. Forensics Secur.* **2010**, *6*, 107–121. [CrossRef]
8. Rathgeb, C.; Merkle, J.; Scholz, J.; Tams, B.; Nesterowicz, V. Deep face fuzzy vault: Implementation and performance. *Comput. Secur.* **2022**, *113*, 102539. [CrossRef]
9. Chauhan, S.; Sharma, A. Improved fuzzy commitment scheme. *Int. J. Inf. Technol.* **2022**, *14*, 1321–1331. [CrossRef]
10. Wang, Y.; Li, B.; Zhang, Y.; Wu, J.; Ma, Q. A secure biometric key generation mechanism via deep learning and its application. *Appl. Sci.* **2021**, *11*, 8497. [CrossRef]
11. Mir, O.; Roland, M.; Mayrhofer, R. DAMFA: Decentralized anonymous multi-factor authentication. In *Proceedings of the 2nd ACM International Symposium on Blockchain and Secure Critical Infrastructure*; IEEE: New York, NY, USA, 2020; pp. 10–19.
12. Kim, S.; Mun, H.J.; Hong, S. Multi-factor authentication with randomly selected authentication methods with DID on a random terminal. *Appl. Sci.* **2022**, *12*, 2301. [CrossRef]
13. Al-Rubaie, M.; Chang, J.M. Privacy-preserving machine learning: Threats and solutions. *IEEE Secur. Priv.* **2019**, *17*, 49–58. [CrossRef]
14. Loya, J.; Bana, T. Privacy-Preserving Keystroke Analysis using Fully Homomorphic Encryption & Differential Privacy. In *Proceedings of the 2021 International Conference on Cyberworlds (CW)*; IEEE: New York, NY, USA, 2021; pp. 291–294.
15. Baig, A.F.; Eskeland, S.; Yang, B. Privacy-preserving continuous authentication using behavioral biometrics. *Int. J. Inf. Secur.* **2023**, *22*, 1833–1847. [CrossRef]
16. Soutar, C.; Roberge, D.; Stoianov, A.; Gilroy, R.; Kumar, B.V. Biometric encryption using image processing. In *Proceedings of the Optical Security and Counterfeit Deterrence Techniques II*; SPIE: Washington, DC, USA, 1998; Volume 3314, pp. 178–188.
17. Usman, M.; Jan, M.A.; Puthal, D. Paal: A framework based on authentication, aggregation, and local differential privacy for internet of multimedia things. *IEEE Internet Things J.* **2019**, *7*, 2501–2508. [CrossRef]
18. Chamikara, M.A.P.; Bertok, P.; Khalil, I.; Liu, D.; Camtepe, S. Privacy preserving face recognition utilizing differential privacy. *Comput. Secur.* **2020**, *97*, 101951. [CrossRef]
19. Wazzeh, M.; Ould-Slimane, H.; Talhi, C.; Mourad, A.; Guizani, M. Privacy-preserving continuous authentication for mobile and iot systems using warmup-based federated learning. *IEEE Netw.* **2022**, *37*, 224–230. [CrossRef]
20. Yu, F.X.; Rawat, A.S.; Menon, A.K.; Kumar, S. Federated learning with only positive labels. In *Proceedings of the 37th International Conference on Machine Learning (ICML)*; PMLR: Cambridge, MA, USA, 2020; Volume 119, pp. 10946–10956.
21. Yang, W.; Wang, S.; Kang, J.J.; Johnstone, M.N.; Bedari, A. A linear convolution-based cancelable fingerprint biometric authentication system. *Comput. Secur.* **2022**, *114*, 102583. [CrossRef]
22. Taheri, S.; Islam, M.M.; Safavi-Naini, R. Privacy-Enhanced Profile-Based Authentication Using Sparse Random Projection. In *Proceedings of the IFIP SEC'17*; Springer: Berlin/Heidelberg, Germany, 2017; pp. 474–490.
23. Islam, M.M.; Rafiq, M.A.; Islam, M.A. A Privacy-Preserving Behavioral Authentication System. In *Proceedings of the International Symposium on Foundations and Practice of Security*; Springer: Berlin/Heidelberg, Germany, 2024; pp. 95–107.
24. Baig, A.F.; Eskeland, S.; Yang, B. Novel and Efficient Privacy-Preserving Continuous Authentication. *Cryptography* **2024**, *8*, 3. [CrossRef]
25. Meng, W.; Wong, D.S.; Furnell, S.; Zhou, J. Surveying the development of biometric user authentication on mobile phones. *IEEE Commun. Surv. Tutor.* **2014**, *17*, 1268–1293. [CrossRef]
26. Deng, Y.; Zhong, Y. Keystroke dynamics advances for mobile devices using deep neural network. *Recent Adv. User Authentication Using Keystroke Dyn. Biom.* **2015**, *2*, 59–70.
27. Li, H.; Wang, M.; Pang, L.; Zhang, W. Key binding based on biometric shielding functions. In *Proceedings of the Information Assurance and Security, 2009. IAS'09. Fifth International Conference on Information Assurance and Security*; IEEE: New York, NY, USA, 2009; Volume 1, pp. 19–22.

28. Juels, A.; Sudan, M. A fuzzy vault scheme. *Des. Codes Cryptogr.* **2006**, *38*, 237–257. [CrossRef]
29. Dodis, Y.; Ostrovsky, R.; Reyzin, L.; Smith, A. Fuzzy extractors: How to generate strong keys from biometrics and other noisy data. *SIAM J. Comput.* **2008**, *38*, 97–139. [CrossRef]
30. Domingo-Ferrer, J.; Wu, Q.; Blanco-Justicia, A. Flexible and robust privacy-preserving implicit authentication. In *Proceedings of the ICT Systems Security and Privacy Protection: 30th IFIP TC 11 International Conference, SEC 2015, Hamburg, Germany, 26–28 May 2015*; Springer: Berlin/Heidelberg, Germany, 2015; pp. 18–34.
31. Wang, Y.; Plataniotis, K.N. An analysis of random projection for changeable and privacy-preserving biometric verification. *IEEE Trans. Syst. Man Cybern. Part B (Cyber.)* **2010**, *40*, 1280–1293. [CrossRef] [PubMed]
32. Punithavathi, P.; Geetha, S. Dynamic sectored random projection for cancelable iris template. In *Proceedings of the 2016 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*; IEEE: New York, NY, USA, 2016; pp. 711–715.
33. Deshmukh, M.; Balwant, M.K. Generating cancelable palmprint templates using local binary pattern and random projection. In *Proceedings of the 2017 13th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS)*; IEEE: New York, NY, USA, 2017; pp. 203–209.
34. Rajasekar, V.; Premalatha, J.; Sathya, K. Cancelable Iris template for secure authentication based on random projection and double random phase encoding. *Peer-Peer Netw. Appl.* **2021**, *14*, 747–762. [CrossRef]
35. Peng, J.; Gupta, B.B.; Abd El-Latif, A.A. A biometric cryptosystem scheme based on random projection and neural network. *Soft Comput.* **2021**, *25*, 7657–7670. [CrossRef]
36. Gomez-Barrero, M.; Maiorana, E.; Galbally, J.; Campisi, P.; Fierrez, J. Multi-biometric template protection based on Bloom filters. *Inf. Sci.* **2016**, *370–371*, 18–32. [CrossRef]
37. Jiang, R.; Li, D.; Luo, S.; Zhang, Y. Distance-preserving hashing for cancelable biometrics. *Cybersecurity* **2023**, *6*, 11.
38. Kaski, S. Dimensionality reduction by random mapping: Fast similarity computation for clustering. In *Proceedings of the 1998 IEEE International Joint Conference on Neural Networks Proceedings. IEEE World Congress on Computational Intelligence*; IEEE: New York, NY, USA, 1998; Volume 1, pp. 413–418.
39. Dasgupta, S.; Gupta, A. An elementary proof of a theorem of Johnson and Lindenstrauss. *Random Struct. Algorithms* **2003**, *22*, 60–65. [CrossRef]
40. Achlioptas, D. Database-friendly random projections: Johnson-Lindenstrauss with binary coins. *J. Comput. Syst. Sci.* **2003**, *66*, 671–687. [CrossRef]
41. Dwork, C.; McSherry, F.; Nissim, K.; Smith, A. Calibrating noise to sensitivity in private data analysis. In *Proceedings of the Theory of Cryptography: Third Theory of Cryptography Conference, 2006, NY, USA*; Springer: Berlin/Heidelberg, Germany, 2006; pp. 265–284.
42. Dong, J.; Roth, A.; Su, W.J. Gaussian differential privacy. *J. R. Stat. Soc. Ser. B (Stat. Methodol.)* **2022**, *84*, 3–37. [CrossRef]
43. Wang, R.; Fung, B.C.; Zhu, Y. Heterogeneous data release for cluster analysis with differential privacy. *Knowl.-Based Syst.* **2020**, *201*, 106047. [CrossRef]
44. Nielsen, F. On a variational definition for the Jensen-Shannon symmetrization of distances based on the information radius. *Entropy* **2021**, *23*, 464. [CrossRef]
45. Wang, Q.; Kulkarni, S.R.; Verdú, S. Divergence estimation for multidimensional densities via *k*-nearest-neighbor distances. *IEEE Trans. Inf. Theory* **2009**, *55*, 2392–2405. [CrossRef]
46. Zheng, Z.; Li, Z.; Huang, C.; Long, S.; Li, M.; Shen, X. Data poisoning attacks and defenses to LDP-based privacy-preserving crowdsensing. *IEEE Trans. Dependable Secur. Comput.* **2024**, *21*, 4861–4878. [CrossRef]
47. Demmel, J.W.; Higham, N.J. Improved error bounds for underdetermined system solvers. *SIAM J. Matrix Anal. Appl.* **1993**, *14*, 1–14. [CrossRef]
48. Kraskov, A.; Stögbauer, H.; Grassberger, P. Estimating mutual information. *Phys. Rev. E* **2004**, *69*, 066138. [CrossRef] [PubMed]
49. Gupta, S.; Buriro, A.; Crispo, B. A chimerical dataset combining physiological and behavioral biometric traits for reliable user authentication on smart devices and ecosystems. *Data Brief* **2020**, *28*, 104924. [CrossRef] [PubMed]
50. Chawla, N.V.; Bowyer, K.W.; Hall, L.O.; Kegelmeyer, W.P. SMOTE: Synthetic minority over-sampling technique. *J. Artif. Intell. Res.* **2002**, *16*, 321–357. [CrossRef]
51. Gupta, S.; Buriro, A.; Crispo, B. DriverAuth: A risk-based multi-modal biometric-based driver authentication scheme for ride-sharing platforms. *Comput. Secur.* **2019**, *83*, 122–139. [CrossRef]