

Exploit the Leak: Understanding Risks in Biometric Matchers

DURBET Axel¹,
axel.durbet@uca.fr

THIRY-ATIGHEHCHI Kevin¹,
kevin.atighehchi@uca.fr

CHAGNON Dorine¹,
dorine.chagnon@uca.fr

GROLLEMUND Paul-Marie²
paul_marie.grollemund@uca.fr

¹LIMOS, Université Clermont-Auvergne, CNRS, Mines Saint-Étienne, Aubière, France

²LMBP, Université Clermont-Auvergne, CNRS, Mines Saint-Étienne, Aubière, France

Abstract

In a biometric authentication or identification system, the matcher compares a stored and a fresh template to determine whether there is a match. This assessment is based on both a similarity score and a predefined threshold. For better compliance with privacy legislation, the matcher can be built upon a privacy-preserving distance. Beyond the binary output (*‘yes’* or *‘no’*), most schemes may perform more precise computations, *e.g.*, the value of the distance. Such precise information is prone to leakage even when not returned by the system. This can occur due to a malware infection or the use of a weakly privacy-preserving distance, exemplified by side channel attacks or partially obfuscated designs. This paper provides an analysis of information leakage during distance evaluation. We provide a catalog of information leakage scenarios with their impacts on data privacy. Each scenario gives rise to unique attacks with impacts quantified in terms of computational costs, thereby providing a better understanding of the security level.

Keywords: Privacy-Preserving Distance, Hamming Distance, Information Leakage, Biometric Security

1 Introduction

Biometric authentication protocols involve the comparison of a fresh biometric template with the reference template. This comparison computes the distance between the newly acquired data and the stored template. If this distance is below a given threshold, access is granted; otherwise, it is denied. While many protocols use standard metrics such as the Hamming distance [1], this process can inadvertently leak information that adversaries might exploit to reconstruct the stored template. Vulnerabilities arise from implementation errors, inherent flaws, and server-level attacks such as malware [2], which can compromise system-wide security. Furthermore, Aydin and Aysu [3] and Hashemi *et al.* [4] have highlighted an increasing prevalence of side-channel attacks. Side-channel techniques, including timing, differential power analysis, cache-based, electromagnetic, acoustic, and thermal attacks, exploit various operational artifacts to extract sensitive information [2]. One of the concerns is the partial or total leakage of distance computation information. Such information leakage poses significant security and privacy risks, especially in sensitive applications like privacy-preserving applications (*e.g.*, biometric recognition systems). In this paper, we focus on the following attacks:

- *Offline exhaustive search attacks* refer to scenarios for which a leaked yet obfuscated database is available for an attacker. The attacker employs the public transformation to verify a candidate vector. This verification may give additional information beyond the minimal information leakage (*‘yes’* or *‘no’*), for example via side-channel attacks.
- *Online exhaustive search attacks* correspond to attacks for which an attacker must interact with the biometric system to infer information about the targeted vector. Then, the attacker needs to force the system to leak additional information beyond the minimal information leakage (*‘yes’* or *‘no’*), for example via a malware infection.

Related Works To the best of our knowledge, two papers investigate information leakage of biometric systems using privacy-preserving distance. Pagnin *et al.* [5] shows that the output of a privacy-preserving distance can be exploited to infer the hidden input. This type of attack is considered the most devastating for such systems, as evidenced by Simoens *et al.* [6]. The work of Pagnin *et al.* takes place in the minimal leakage scenario, wherein only the binary output of the biometric system is given to the attacker. The authors present the *Center Search Attack*, designed to recover the hidden enrolled input for any *‘valid’* biometric template in $\mathbb{Z}_q^n$, where *‘valid’* refers to inputs

Distance-to-Threshold comparison	Leakage	Complexity type	Complexity in Big-Oh	Theorem
Below	Distance	Exponential	$q^{n-\varepsilon} + q\varepsilon$	1
	Positions	Exponential	$q^{n-\varepsilon} + q$	2
	Positions and values	Exponential	$q^{n-\varepsilon}$	3
	Positions and values (accumulation)	Linearithmic/Polynomial	$n^\alpha \log n$	8
Both	Minimal ¹	Exponential	$2^{n-\varepsilon} + n + 2\varepsilon$	4
	Distance	Linear	nq	5
	Positions	Constant	q	6
	Positions and values	Constant	1	7

Table 1 –Summary of all leakage exploits and their complexities with α such that the occurrence of the rarest error is $n^{-\alpha}$ with $\alpha \in \mathbb{R}_{\geq 1}$. The Distance-to-Threshold comparison determines if the leak occurs when $d(x, y) \leq \varepsilon$ (below) or when there is no distance requirement between x and y (both). For all the complexities, x and y are in $\mathbb{Z}_q^n$ with $q \geq 2$ except for the minimal leakage where x and y are in $\mathbb{Z}_2^n$. The provided complexities represent worst-case scenarios, except for the accumulation attack where the result is the expectation.

¹Note that the Big-Oh complexity of the optimal exhaustive search strategy, in the worst-case, is the same as the naive strategy as the minimum of $h(\cdot)$ is 0.

within a ball centered at the enrolled template and with a radius equal to the decision threshold t . To efficiently locate a valid input, the authors also examine the exhaustive search attack, particularly its application on binary templates ($q = 2$). They suggest implementing a *sampling without replacement* strategy using their *Tree algorithm* to streamline the identification of a suitable input for the Center Search Attack. This efficient identification of a proper input requires a number of authentication attempts that is exponential in the space dimension n minus the threshold t . While their work focuses on the minimal leakage scenario, our analysis includes the consideration of multiple additional information leaks that may arise during the matching operation.

Contributions We analyze the impact of potential information leakage in distance evaluations. Our contributions detail various leakage scenarios, their corresponding generic attacks, and the computational costs involved:

- We revisit the exhaustive search attack in the scenario of a minimal (one-bit) information leakage, correcting a previously cited result (see [5]) about the costs of optimal and near-optimal strategies.
- We introduce new attack strategies by malicious clients that exploit various levels of non-minimal information leaks from the system. Our complexity results, which detail the cost of these attacks, apply to both *offline exhaustive search attacks* that leverage a leaked (yet obfuscated) database and *online exhaustive search attacks* involving direct interactions with the server.
- We investigate a novel attack, named accumulation attack, where an *honest-but-curious* server accumulates knowledge during client authentication. This type of attack occurs when there is a minor, yet non-negligible, amount of information leakage.

The complexities of the attacks, relying on different scenarios, are summarized in Figure 1.

Outline Section 2 introduces notations and terminologies and classifies the different types of information leakages. Section 3 begins by revisiting the exhaustive search attack in the minimal (one-bit) information leakage scenario, including a correction of a previously cited result concerning the costs of optimal and near-optimal strategies. It then introduces new strategies for attacks by malicious clients capturing various other types of information leakages, covering both offline and online exhaustive search attacks, with an emphasis on their computational costs. The section concludes by examining accumulation attacks performed by an "honest-but-curious" server during client authentication, detailing the computational cost involved. Section 4 provides a discussion of the presented results.

2 Preliminaries

This section introduces the notations as well as the attacker model and, a list of the considered information leakage scenarios.

2.1 Notations and Attacker Models

Let $\mathbb{Z}_q^n = \{0, \dots, q-1\}^n$ be a metric space equipped with the Hamming distance d and $\varepsilon \in \mathbb{N}$ a threshold. The Hamming distance is defined by

$$d(x, y) = |\{i \in \{1, \dots, n\} | x_i \neq y_i\}|$$

for two vectors $x = (x_1, \dots, x_n)$ and $y = (y_1, \dots, y_n)$ in $\mathbb{Z}_q^n$. Let $\text{Match}_{x,\varepsilon}$ denote the oracle modeling the interaction between the biometric system using a privacy-preserving distance and the attacker. $\text{Match}_{x,\varepsilon}$ receives the template selected by the attacker and compares it with the previously enrolled and stored template. If the distance is below the threshold ε , the oracle returns 1 and 0 otherwise. In a more formal way, $\text{Match}_{x,\varepsilon}$ is a function defined as:

$$\begin{aligned} \text{Match}_{x,\varepsilon} : \mathbb{Z}_q^n &\longrightarrow \{0, 1\} \\ y &\longmapsto \begin{cases} 1 & \text{if } d(x, y) \leq \varepsilon. \\ 0 & \text{otherwise.} \end{cases} \end{aligned}$$

A privacy-preserving distance may leak additional information beyond its binary output. Under the specifications of each scenario, the oracle may display this additional information. The objective of the attacker is to find the hidden template x exploiting the oracle outputs. In the context of a biometric system, the objective of the attacker may be relaxed to simply find y that is close to x with respect to d and ε .

2.2 Typology of Information Leakage

In the context of a biometric system, a critical vulnerability arises when information is intercepted between the matcher and the decision module, as illustrated in Figure 1 (point 8). This figure, inspired by Ratha *et al.* [7], provides an overview of the attack points in biometric systems while introducing both the decision module and two additional attack points. Except for the accumulation attack, the attacker exploits points 4 and 8 in all discussed scenarios. Point 4 allows the submission of a chosen template, while point 8 grants access to additional information beyond the binary output. The accumulation attack only necessitates control over the point 8. For detailed insights into the remaining attack points, readers are referred to Ratha *et al.* [7]. There are three main categories of information leakage:

- Below the threshold.
- Above the threshold.
- Both below and above the threshold.

In each of these categories, several sub-settings can be identified. The first one corresponds to the absence of any leakage, resulting in $\text{Match}_{x,\varepsilon}$ yielding only the binary output. Then, the following information leakages are examined:

- The distance.
- The positions of the errors.
- Both the error positions and values.
- Both the distance and the positions of the errors.
- Both the distance and the positions and their corresponding erroneous values.

It is not relevant to consider that additional information is leaked only above the threshold, as no scheme has such behavior. As a consequence, solely scenarios ‘below the threshold’ and ‘below and above the threshold’ are examined. The Hamming distance is a measure of the number of differing coordinates between two templates. Therefore, knowledge of the erroneous coordinates implies knowledge of the distance itself. Hence, we do not consider all possible scenarios.

3 Exploiting the Leakage

3.1 Active Attacks

3.1.1 Attack Complexity for the Minimal (One-bit) Leakage

The exhaustive search may be enhanced for both computationally bounded and unbounded adversaries. The objective is to identify the minimal number of balls of radius ε to partition the space and then, to query the center of each ball in order to remove an entire ball if the query fails. This is an instance of the set covering problem. Pagnin *et al.* [5] proposed to use such a method and claimed that the number of points that the adversary needs to query is only a factor of $\mathcal{O}(\varepsilon \ln(n+1))$ more than the optimal cover. However, the result is imprecise, as detailed below, mainly because the optimal cover is not given.

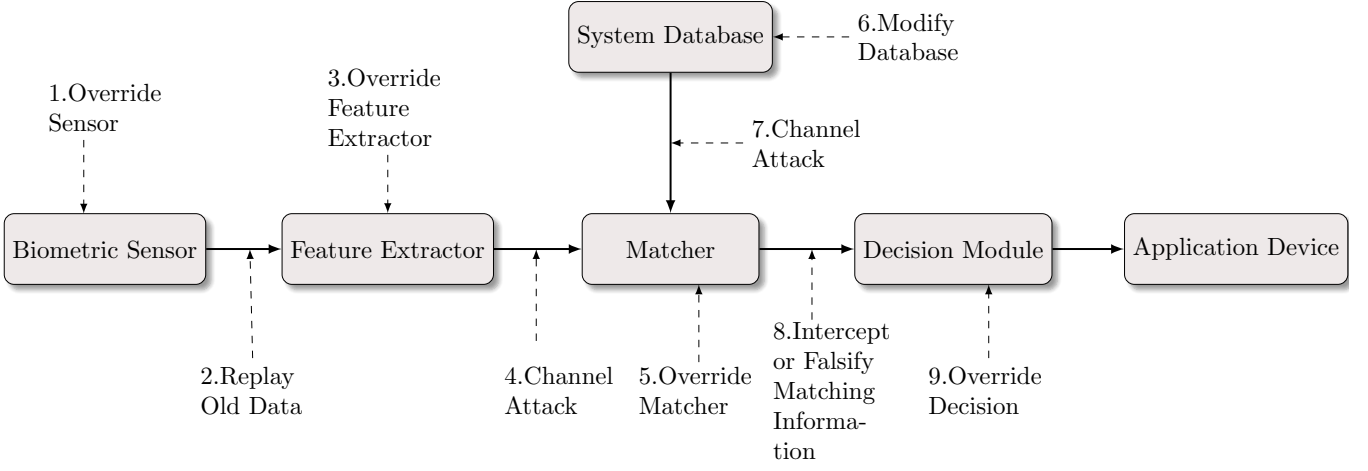


Figure 1 –Attack points in a generic biometric recognition system.

Unbounded Case The adversary solves the NP-hard set covering problem [8] to find the optimal covering of $\mathbb{Z}_q^n$ using balls of radius ε . The adversary exhaustively searches x using at most $q^{n(1-h_q(\varepsilon/n))+o(n)}$ queries to $\text{Match}_{x,\varepsilon}$. The number of vectors involved in a given optimal cover is $\frac{q^n}{|B_{q,\varepsilon}(x)|}$, which can be asymptotically approximated as detailed in what follows. First, recall that the cardinal of $B_{q,\varepsilon}(x)$ is

$$|B_{q,\varepsilon}(x)| = \sum_{i=0}^{\varepsilon} \binom{n}{i} (q-1)^i,$$

and that the q -ary entropy is $h_q(x) = x \log_q(q-1) - x \log_q x - (1-x) \log_q(1-x)$. Then, using bounds on the binomial coefficient (see [9]), the result follows if $\frac{\varepsilon}{n} \leq 1 - \frac{1}{q}$ holds and if n is large enough.

Bounded Case The adversary may use a greedy algorithm to find a non-optimal covering containing $\frac{q^n H(n)}{|B_{q,\varepsilon}|}$ vectors [10] with $H(n) = \sum_{i=1}^n i^{-1}$ the n -th harmonic number. The adversary then finds a solution with an exhaustive search in at most $\frac{q^n H(n)}{|B_{q,\varepsilon}|}$ queries. In order to provide a more intuitive value, notice that $\frac{q^n H(n)}{|B_{q,\varepsilon}|}$ can be bounded up by $\frac{q^n (\ln(n)+1)}{|B_{q,\varepsilon}|}$. As in the unbounded case, using the q -ary entropy and Stirling's approximation, this non-optimal covering leads the attacker to make at most $q^{n(1-h_q(\varepsilon/n))+o(n)}$ queries, as $\log_q(\ln(n)+1) = o(n)$.

3.1.2 Attack Complexities for Leakage Below the Threshold

Leakage below the threshold is considered in this section. Given the hidden target x , querying y such that $d(x, y) \leq \varepsilon$ to the oracle $\text{Match}_{x,\varepsilon}$ provides information beyond the binary output.

Leakage of the Distance The first case occurs when the distance is given to the attacker as extra information.

Theorem 1 *Given ε a threshold, $x \in \mathbb{Z}_q^n$ a vector, and $\text{Match}_{x,\varepsilon}$ leaks the distance below the threshold, an attacker can retrieve x in the worst case in $\mathcal{O}(q^{n-\varepsilon} + q\varepsilon)$ queries to $\text{Match}_{x,\varepsilon}$.*

Proof 1 *The system, using the Hamming distance, requires a minimum of $n - \varepsilon$ accurate coordinates to output 0. Since the attacker specifically targets $n - \varepsilon$ coordinates (the attacker arbitrarily chooses ε coordinates that do not change), an exhaustive search attack is performed in at most $q^{n-\varepsilon}$ steps to get accepted by the system. Then, a hill-climbing attack runs on the remaining ε coordinates to minimize the distance at each step. Coordinate by coordinate, the attacker obtains the right value if the distance decreases. Since there are q different values to test on ε coordinates, determining the correct ones requires a maximum of $(q-1)\varepsilon$ steps. Then, the overall complexity is $\mathcal{O}(q^{n-\varepsilon} + q\varepsilon)$. ■*

Leakage of the Positions The positions of the errors are the extra information given to the attacker, while their values remain secret.

Theorem 2 *Given ε a threshold, $x \in \mathbb{Z}_q^n$ a vector, and $\text{Match}_{x,\varepsilon}$ leaks the positions of the errors below the threshold, an attacker can retrieve x in the worst case in $\mathcal{O}(q^{n-\varepsilon} + q)$ queries to $\text{Match}_{x,\varepsilon}$.*

Proof 2 As the leakage occurs solely below the threshold, the first step is to find a vector $y \in \mathbb{Z}_q^n$ such that $d(x, y) \leq \varepsilon$. To identify such a vector, the attacker performs an exhaustive search attack in $q^{n-\varepsilon}$ steps, as previously shown. Since ε coordinates remain unknown, and each coordinate ranges from 0 to $q-1$, every possibility must be examined. By testing all possibilities simultaneously – for instance, testing all coordinates at 0, then all coordinates at 1, and so forth up to $q-2$ while retaining the correct values – the original vector can be identified in no more than $q-1$ queries (refer to the example illustrated in Figure 2). Therefore, the complexity of the attack for recovering x is $\mathcal{O}(q^{n-\varepsilon} + q)$. ■

Figure 2 gives a representation of the attack described above in the case $\mathbb{Z}_4^5$ and the hidden vector or the missing coordinates is $(0, 1, 3, 2, 2)$. Note that the actual complexity is $q-1$ since the final exchange is unnecessary, as the coordinates at $q-1$ become known after $q-1$ queries by inference.

Leakage of the Positions and the Values When a vector below the threshold is given to the oracle $\text{Match}_{x,\varepsilon}$, the attacker gets information about both error positions and their values. This is similar to an error-correction mechanism designed to correct errors below a given threshold. Note that in the binary case, this scenario is the same as the previous one, hence the only considered case is $q > 2$.

Theorem 3 Given ε a threshold, $x \in \mathbb{Z}_q^n$ a vector, and $\text{Match}_{x,\varepsilon}$ leaks the positions and the values of the errors below the threshold, an attacker can retrieve x in $\mathcal{O}(q^{n-\varepsilon})$ queries to $\text{Match}_{x,\varepsilon}$.

Proof 3 First, an exhaustive search is performed to find a vector y for which the distance is below the threshold, for a cost of $\mathcal{O}(q^{n-\varepsilon})$. Then, given the error positions and the corresponding error values, y yields immediately the recovery of x . In the end, the complexity of the attack is $\mathcal{O}(q^{n-\varepsilon})$. ■

3.1.3 Leakage Below and Above the Threshold

The second scenario is considered in this section, which involves a leakage independent of the threshold. In other words, when a hidden vector x is targeted, the queried vector y to the oracle $\text{Match}_{x,\varepsilon}$ results in the leak of additional information.

Minimal Leakage (single bit of information leakage) The basic usage of the system is characterized by the minimal leakage scenario, where the binary output itself is considered a necessary leakage. This minimal leakage is indispensable for the system's work and is consistent across these scenarios as the system always responds. Remark that if the server does not answer above the threshold, the non-answer gives the attacker the wanted information.

Theorem 4 Given ε a threshold, $x \in \mathbb{Z}_2^n$ a vector, and $\text{Match}_{x,\varepsilon}$ that does not leak any extra information, an attacker can retrieve x in $\mathcal{O}(2^{n-\varepsilon} + n + 2\varepsilon)$ queries to $\text{Match}_{x,\varepsilon}$.

Proof 4 As in the previous cases, the attacker seeks a vector y below the threshold. Such a vector is found by exhaustive search in $2^{n-\varepsilon}$ steps. Then, the attacker performs the center search attack [5] to retrieve the original data in at most $n + 2\varepsilon$ queries. The complexity of the attack to find x is $\mathcal{O}(2^{n-\varepsilon} + n + 2\varepsilon)$. ■

Leakage of the Distance In this case, $d(x, y)$ the distance between $y \in \mathbb{Z}_q^n$ the fresh template and $x \in \mathbb{Z}_q^n$ the old template is leaked to the attacker regardless of the threshold.

Theorem 5 Given ε a threshold, $x \in \mathbb{Z}_q^n$ a vector, and $\text{Match}_{x,\varepsilon}$ leaks the distance, an attacker can retrieve x in $\mathcal{O}(nq)$ queries to $\text{Match}_{x,\varepsilon}$.

Proof 5 As the attacker has access to the distance, it is possible to perform a hill-climbing attack, trying to minimize the distance at each step. The strategy is to find the vector y , coordinate by coordinate. As each coordinate has q possible values and there are n coordinates, this is done in $\mathcal{O}(nq)$ steps. ■

Leakage of the Positions The extra information given to the attacker is the positions of the errors.

Theorem 6 Given ε a threshold, $x \in \mathbb{Z}_q^n$ a vector, and $\text{Match}_{x,\varepsilon}$ leaks the positions of the errors, an attacker can retrieve x in $\mathcal{O}(q)$ queries to $\text{Match}_{x,\varepsilon}$.

Proof 6 She tries the vector $(0, \dots, 0)$, $(1, \dots, 1)$ up to, $(q-1, \dots, q-1)$ and keep for each coordinate the right value (see Figure 2). Hence, the complexity of the attack to recover x is $\mathcal{O}(q)$. ■

$$\begin{array}{rcccl}
\text{Queries: } & (\boxed{0}, & \underset{\times}{0}, & \underset{\times}{0}, & \underset{\times}{0}, & \underset{\times}{0}) \\
& (\underset{\times}{1}, & \boxed{1}, & \underset{\times}{1}, & \underset{\times}{1}, & \underset{\times}{1}) \\
& (\underset{\times}{2}, & \underset{\times}{2}, & \underset{\times}{2}, & \boxed{2}, & \boxed{2}) \\
& (\underset{\times}{3}, & \underset{\times}{3}, & \boxed{3}, & \underset{\times}{3}, & \underset{\times}{3}) \\
\text{Solution: } & (0, & 1, & 3, & 2, & 2)
\end{array}$$

Figure 2 –Exploiting the error position leaked in the case $\mathbb{Z}_4^5$ and the hidden vector or missing coordinates is $(0, 1, 3, 2, 2)$.

Leakage of the Positions and the Values In this last case, the positions of the errors and corresponding values are leaked. Unlike the scenario of leakage below the threshold, such a leak provides an error-correcting code mechanism that operates irrespective of any distance and threshold.

Theorem 7 *Given ε a threshold, $x \in \mathbb{Z}_q^n$ a vector, and $\text{Match}_{x,\varepsilon}$ leaks the positions of the errors and their values, an attacker can retrieve x in $\mathcal{O}(1)$ queries to $\text{Match}_{x,\varepsilon}$.*

Proof 7 *The submission of any vector gives the position of each error, and how to correct them, yielding a complexity in $\mathcal{O}(1)$.* ■

3.2 Accumulation Attack: A Passive Attack

During the client authentications, the attacker passively gathers information by observing errors leaked by the server. More specifically, the server leaks a list of positions and errors computed over the integers (*i.e.*, $x_i - y_i$) made by a genuine client during each authentication. Such information gathered during one successful authentication attempt is called an observation. The attacker aims to partially or fully reconstruct x by exploiting these observations.

In the binary case (*i.e.*, $q = 2$), the errors precisely yield the bits. If $x_i - y_i = 1$ then $x_i = 1$, and if $x_i - y_i = -1$ then $x_i = 0$. This attack is related to the Coupon Collector’s problem [11], which involves determining the expected number of rounds required to collect a complete set of distinct coupons, with one coupon obtained at each round, and each coupon acquired with equal probability.

Example 3.2.1 *Suppose a setting with a metric space $\mathbb{Z}_2^n$ equipped with the Hamming distance. A client seeks to authenticate to an honest-but-curious server that uses a scheme leaking $d(x, y)$ and the corresponding errors if $d(x, y) \leq \varepsilon$. As the client is legitimate, *i.e.*, $d(x, y) \leq \varepsilon$ with a high probability, the attacker recovers the values of at most ε erroneous bits. The attacker needs to collect all the bits of the client, turning this problem into a Coupon Collector problem. For example, let assume $x = (0, 0, 1, 1, 0, 1, 0)$, $\varepsilon = 3$. The attacker sets $z = (?, ?, ?, ?, ?, ?, ?)$. Session 1: The client authenticates with $y = (1, 1, 0, 1, 0, 1, 0)$. In this case, $d(x, y) = 3 \leq \varepsilon$. The values of the erroneous bits of the client are obtained, yielding $z = (0, 0, 1, ?, ?, ?, ?)$. Session 2: the client authenticates with $y = (0, 0, 0, 0, 1, 1, 0)$. In this case, $d(x, y) = 3 \leq \varepsilon$, and the attacker obtains the value of the erroneous bits of the client and updates $z = (0, 0, 1, 1, 0, ?, ?)$. At this point, replacing the unknown values with random bits gives a vector that lies inside the acceptance ball as the number of unknown coordinates is smaller than the threshold ε .*

In biometrics, some errors happen more frequently than others. In this setup, the Weighted Coupon Collector’s Problem must be considered. Each coupon (*i.e.*, each error) has a probability p_i to occur. Suppose that $p_1 \leq p_2 \leq \dots \leq p_n$ and $\sum_{i=1}^n p_i \leq 1$ then, according to Berenbrink and Sauerwald [12] (Lemma 3.2), the expected number of round E is such that:

$$\frac{1}{p_1} \leq E \leq \frac{H(n)}{p_1} \quad (1)$$

with $H(n)$ the n -th harmonic number. The upper bound on $H(n)$ is $1 + \log n$, which yields the expected number of rounds required to complete the collection:

$$\frac{1}{p_1} \leq E \leq \frac{\ln(n) + 1}{p_1}. \quad (2)$$

However, while in the original problem one coupon is obtained at each round, the number of errors made by a client during an authentication session is variable, *i.e.*, between 1 and ε . In this case, the expected number of rounds required before all the errors have been observed is smaller than in the case where only one error occurs at each round. Consequently, the expected number of rounds required to collect all the errors is still in $\mathcal{O}(\log n / p_1)$.

Theorem 8 *Given ε a threshold, $x \in \mathbb{Z}_2^n$ a vector, $\text{Match}_{x,\varepsilon}$ leaks the positions of the errors and their values below the threshold, and assuming that the rarest coupon is obtained with probability $p_1 = n^{-\alpha}$ with $\alpha \in \mathbb{R}_{\geq 1}$ an attacker can retrieve x in $\mathcal{O}(n^\alpha \log n)$.*

Proof 8 According to the Weighted Coupon Collector’s problem and assuming that the rarest coupon is obtained with probability $p_1 = n^{-\alpha}$ with $\alpha \in \mathbb{R}_{\geq 1}$, the vector x is recovered in $\mathcal{O}(n^\alpha \log n)$ observations. ■

It is worth noting that in this scenario, the attacker does not control the error. If the attacker controls the error locations, then it is possible to obtain x in $\lceil n/\varepsilon \rceil$ queries. This can happen during a fault attack, akin to side-channel attacks. It should also be noted that some coordinates of a biometric data may be non-variable and, as a consequence, an attacker cannot recover them. This partial recovery attack is therefore a privacy attack, and leads to an authentication attack if the number of variable coordinates is sufficiently large (at least $n - \varepsilon$ in the binary case).

Remark 3.2.1 In the non-binary case, the value $x_i - y_i$ does not provide enough information. The exact value of x_i can be determined in two cases. First, if $x_i - y_i = -q + 1$, then $x_i = 0$. Second, if $x_i - y_i = 2(q - 1)$, then $x_i = q - 1$. For all other cases, there is an ambiguity regarding the value of x_i as y_i is unknown. However, by knowing the distribution of x_i and y_i , repeating observations yields a statistical attack.

Attacks for each type of leakage along with their complexities are summarized in Figure 1.

4 Concluding Remarks

Our investigation into the information leakage of a biometric system using privacy-preserving distance has shed light on critical security vulnerabilities that arise under various scenarios. By evaluating the impact of different types of leakage, including distance, error position, and error value, we have examined the potential risks posed to data privacy and security in practical applications.

Our analysis encompasses ‘below the threshold’ and ‘below and above the threshold’ setups, allowing us to identify specific conditions under which information leakage can have a substantial effect on the overall security of the system.

It is important to highlight that the leakage ‘below the threshold’ does not significantly hurt the security of the system, while the leakage of ‘both below and above the threshold’ significantly decreases the security. Indeed, the attacks exploiting the leakage ‘below the threshold’ are primarily exponential, while those exploiting information ‘below and above the threshold’ are mainly constant.

The accumulation attack we investigated is based on the assumption of errors uniformly distributed throughout each authentication session. The result of the accumulation attack could be further refined by considering a variable number of coupons, randomly drawn between 0 and ε in each round, while acknowledging the actual distribution of the errors. As far as we are aware, no previous studies provide an analysis of the distribution of the errors for any systems. In practical scenarios, certain errors may occur more frequently than others, while some may never occur. A skewed distribution of errors substantially increases the expected number of authentications required from the legitimate user so that the server recovers the hidden template. Further research involves refining the accumulation attack as suggested above and exploring other distance metrics, such as L_1 (Manhattan) and L_2 .

References

- [1] Juan Carlos Bernal-Romero, Juan Manuel Ramirez-Cortes, Jose De Jesus Rangel-Magdaleno, Pilar Gomez-Gil, Hayde Peregrina-Barreto, and Israel Cruz-Vega. A review on protection and cancelable techniques in biometric systems. *IEEE Access*, 11:8531–8568, 2023.
- [2] Shreyansh Sharma, Anil Saini, and Santanu Chaudhury. A survey on biometric cryptosystems and their applications. *Computers & Security*, page 103458, 2023.
- [3] Furkan Aydin and Aydin Aysu. Leaking secrets in homomorphic encryption with side-channel attacks. *Journal of Cryptographic Engineering*, pages 1–11, 2024.
- [4] Mohammad Hashemi, Domenic Forte, and Fatemeh Ganji. Time is money, friend! timing side-channel attack against garbled circuit constructions. In *International Conference on Applied Cryptography and Network Security*, pages 325–354. Springer, 2024.
- [5] Elena Pagnin, Christos Dimitrakakis, Aysajan Abidin, and Aikaterini Mitrokotsa. On the leakage of information in biometric authentication. In *International Conference on Cryptology in India*, pages 265–280. Springer, 2014.
- [6] Koen Simoons, Julien Bringer, Hervé Chabanne, and Stefaan Seys. A framework for analyzing template security and privacy in biometric authentication systems. *IEEE Transactions on Information Forensics and Security*, 7:833–841, 2012.

- [7] Nalini K. Ratha, Jonathan H. Connell, and Ruud M. Bolle. An analysis of minutiae matching strength. In Josef Bigun and Fabrizio Smeraldi, editors, *Audio- and Video-Based Biometric Person Authentication*, pages 223–228, Berlin, Heidelberg, 2001. Springer Berlin Heidelberg.
- [8] Bernhard H Korte, Jens Vygen, B Korte, and J Vygen. *Combinatorial optimization*, volume 1. Springer, 2011.
- [9] MTCAJ Thomas and A Thomas Joy. *Elements of information theory*. Wiley-Interscience, 2006.
- [10] Vasek Chvatal. A greedy heuristic for the set-covering problem. *Mathematics of operations research*, 4(3):233–235, 1979.
- [11] Marco Ferrante and Monica Saltalamacchia. The coupon collector’s problem. *MATerials MATemàtics*, 2014:35, 05 2014.
- [12] Petra Berenbrink and Thomas Sauerwald. The weighted coupon collector’s problem and applications. In Hung Q. Ngo, editor, *Computing and Combinatorics*, pages 449–458, Berlin, Heidelberg, 2009. Springer Berlin Heidelberg.