SDDup: Confidentiality-Aware Semantic Deduplication

Lucas Mayr[1]**, Wellington Fernandes Silvano**[1]**, Gabriel Holstein**[1]**, Ricardo Custódio**[1]

[1]Instituto de Informática e Estatística – Universidade Federal de Santa Catarina (UFSC)
Florianópolis – SC – Brazil

{lucas.mayr, wellington.fernandes}@posgrad.ufsc.br,

gabriel.h.m@grad.ufsc.br, ricardo.custodio@ufsc.br

Abstract. *Full-document encryption in existing document management systems creates major storage bottlenecks and hampers fine-grained access control. To overcome this, we present SDDup, a novel model that employs semantic-aware segmentation to separate common from unique, sensitive data segments. This allows for efficient deduplication and targeted encryption while maintaining integrity and regulatory compliance. We validate SDDup through theoretical and empirical analysis, security evaluation, and large-scale experiments on Brazilian birth certificates and university degrees. We compare SDDup against EDRStore, a leading system for generic encrypted data reduction. Our findings highlight SDDup as a competitive and scalable solution for document management.*

1. Introduction

The global shift from paper-based records to digital documents, particularly within governmental and large-scale enterprise environments, presents both transformative efficiencies and complex management challenges. While benefits like enhanced interoperability and streamlined workflows are significant, the digital paradigm demands robust solutions for document integrity, efficient processing, and secure, long-term storage, especially for the vast volumes of digitally signed and often sensitive records being generated. This need is further amplified by stringent data protection regulations like Brazil's LGPD [Brasil 2018] and Europe's GDPR [European Union 2018], which mandate protective measures such as encryption for sensitive information. However, implementing such security, particularly through conventional full-document encryption, often exacerbates storage and processing overheads, creating a critical tension between confidentiality and operational efficiency [Zhao et al. 2024]. Consequently, the development of document management systems that can harmoniously ensure integrity, confidentiality, authorization, and efficiency is paramount.

A common method for ensuring the authenticity and integrity of electronic documents is digital signatures; these cryptographic primitives allow documents to be bound to signers, and any modification to the signed document invalidates any previous signature, allowing for the detection of tampering [Stanešić et al. 2025]. Thus, any document management system must avoid breaking these signatures, i.e., procedures must maintain the validity of associated documents while being handled by the system. Conversely, to guard sensitive information against unauthorized access, symmetric encryption is usually used to encrypt the document as a whole, concealing both confidential and non-confidential data before being stored. Consequently, this typically naive solution negatively impacts read-and-write performance for any procedure that does not require access to the whole document. Moreover, procedures that operate on documents would have no granularity; they either have access to the entire document or no access at all.

Usually, documents that stem from the same entity or procedure contain many artifacts and fields in common: text, images, watermarks, and disclaimers, to cite a few [Yan et al. 2016]; when encrypting and storing documents in the naive way, these artifacts are replicated and thus consume processing and storage power that could be dedicated to other operations [Kardaş and Kiraz 2016]. Hence, we argue that there are two clear venues of optimization possible when managing confidential documents: i) to minimize storage of redundant information, and ii) to restrict encryption to confidential information only. While we focus mainly on signed PDF and XML documents to exemplify and show the benefits of our proposal, it can be applied to any other type of document.

There are many entities where secure document management is paramount; however, we argue that governmental institutions that are at the forefront of managing citizen records are a good use-case candidate when evaluating any proposal dealing with efficient and secure document management, such that most lessons taken from a study focused on them could be reasonably applied to other scenarios. Thus, we analyze and measure the benefits of our proposal against the environment of the Brazilian Civil Registry and higher education degree issuance, which are responsible for managing a vast amount of official documentation, ensuring the authenticity, integrity, and confidentiality of these documents in compliance with laws and regulations.

Developing confidentiality-aware, efficient, and secure document management models is critical to ensuring the confidentiality of personal data and documents and can be accomplished by leveraging the semantic characteristics of similar documents. They must ensure personal data integrity, authenticity, and privacy while meeting applicable legal and regulatory standards. This paper describes a semantics-aware model for the efficient management and processing of digital documents that contain similar data. It capitalizes on the characteristics of digital signatures and tightly structured data to enable the extraction and encryption of only sensitive information while minimizing storage by reducing the number of identical artifacts that need to be stored.

Contribution. We present SDDup, a protocol that focuses on efficiently protecting personal data and the confidentiality of documents while adhering to all legal obligations and responsibilities by leveraging the semantic properties of said documents. Additionally, we optimize the storage of these documents, enabling the storage of hundreds of millions of encrypted documents on consumer-grade hardware. This study improves availability and privacy while maintaining the records' integrity and confidentiality at comparable efficiency levels to state-of-the-art research on deduplication mechanisms. We provide a comprehensive analysis of SDDup, including its security implications and a comparative performance evaluation against both naive approaches and advanced systems like EDRStore [Zhao et al. 2024].

2. Background and Related Work

Managing large repositories of standardized documents, which inherently contain significant repetitive content, necessitates storage optimization. Data compression and deduplication are common techniques, with deduplication identifying identical data chunks across files to store only a single copy, offering substantial gains, especially with high content repetition [Yan et al. 2016, Kardaş and Kiraz 2016]. However, challenges arise when naive compression obscures inter-file similarities or when applying these techniques to encrypted data. Conventional encryption typically randomizes ciphertext, destroying redundancy and creating a fundamental tension with storage efficiency goals [Zhao et al. 2024].

Early attempts to bridge this gap included convergent encryption and Message-Locked Encryption (MLE) [Bellare et al. 2013], which derive encryption keys from the data itself to ensure identical plaintexts yield identical ciphertexts, thus enabling deduplication. However, these deterministic approaches can be vulnerable to guessing attacks, leading to proposals involving auxiliary key servers [Keelveedhi et al. 2013] to mitigate such risks. Further research has explored hardware-assisted secure deduplication, like S2Dedup using Intel SGX [Miranda et al. 2021], and techniques like re-keying [Li et al. 2016] to enhance security against prolonged exposure. While these methods advance secure deduplication for generic data, they often do not specifically address the semantic structure of documents or the need for selective encryption based on content sensitivity.

Another line of work focuses on data integrity and ownership in deduplicated storage, with Proof-of-Ownership protocols [Liu et al. 2020] and schemes for public integrity auditing of encrypted, deduplicated data [Kardaş and Kiraz 2016]. These are crucial for trustworthy outsourced storage but do not inherently solve the efficiency issues for highly structured, partially sensitive documents. Similarly, approaches like Wu et al.'s hierarchical model for encrypted medical records [Wu et al. 2022] provide confidentiality but lack storage optimization through redundancy exploitation and selective encryption, differing from SDDup's goals.

Recent research has focused on secure deduplication with the support of trusted hardware or adaptive cryptography. Esteves et al. (2021) introduced *S2Dedup*, a privacy-preserving deduplication system supported by Intel SGX enclaves [Miranda et al. 2021]. *S2Dedup* enables dynamic adjustment of security and performance levels through multiple encryption schemes. One of *S2Dedup*'s advancements was addressing frequency analysis attacks: it implements a scheme that balances security and performance with the use of *epochs* for key rotation, potentially reducing information leakage through the frequency of duplicate chunks, a vulnerability affecting traditional deterministic deduplication approaches. Another approach is periodic or on-demand re-encryption of deduplicated data (re-keying) to mitigate prolonged exposure of the same ciphertext [Li et al. 2016].

In addition to confidentiality, the integrity of deduplicated data and multi-user access control are significant concerns. *Proof-of-ownership* protocols have been proposed to ensure that only clients who possess a local copy of the data can register duplicates on the server, preventing malicious users from wrongfully claiming a file (attempting to gain access to content they do not possess). For example, Liu et al. implemented *Proof-of-ownership* [Liu et al. 2020] in dynamic sharing scenarios, enabling access revocation and ownership transfer of deduplicated fragments without re-encryption.

Document sanitization is another area of computer security research that faces significant challenges in improving techniques for selecting information within a document that needs to be removed or altered. The effectiveness of such systems relies heavily on the analysis algorithm [Friedlin and McDonald 2008]. Due to the difficulty of manually detecting and eliminating all possibilities, various algorithms have been employed in the literature to automatically sanitize documents, including ERASE [Chakaravarthy et al. 2008], N-Sanitization [Iwendi et al. 2020], PACO2DT [Lin et al. 2021], and Btop [Chakaravarthy et al. 2008]. However, these approaches often permanently alter documents, unlike SDDup, which enables full document reconstruction and, therefore, digital signature preservation.

More recently, systems like EDRStore [Zhao et al. 2024] have aimed for comprehensive data reduction by integrating local compression, deduplication, and delta compression on encrypted data. EDRStore achieves significant storage efficiency for generic data by carefully orchestrating these techniques with specialized encryption schemes. However, its block-oriented process does not inherently preserve or leverage the semantic characteristics of structured documents. This means that the distinction between a shared template and unique fields within a document, which is central to SDDup, is not explicitly managed, potentially limiting optimizations related to selective confidentiality and direct template reuse.

These studies demonstrate the importance of finding alternatives to data compression and deduplication that can guarantee confidentiality and integrity. Building on this line of work, the SDDup model resolves the tension between deduplication and confidentiality in the context of well-structured, signed documents, a reality very present in governmental documents, through semantic awareness of data chunks. While most works in the past decade focus on secure access, redaction, or decentralized distribution, SDDup builds upon these foundations but carves a distinct niche by focusing on *semantic deduplication* for structured, often signed, documents. Unlike systems primarily targeting generic data blocks, SDDup leverages the inherent document structure (templates vs. unique data). This semantic-aware approach directly addresses the tension between confidentiality and storage efficiency in a way that is particularly suited for environments managing large volumes of similar official documents, offering a path to significant storage reduction while respecting data sensitivity and legal requirements for document authenticity.

3. Conceptual Model

The definition of a digital (or electronic) document has been the source of discussion for many years and is usually molded to fit a certain environment, scenario, or regulation. Here, we define digital documents in their broadest sense: digital documents are an ordered set of bits that may be processed through computer systems. We highlight that the terms digital and electronic document can have slightly different meanings depending on the context of the discussion and regulatory body; however, in this paper, *Digital* and *Electronic* are used interchangeably for clarity as the only requirement for our proposal to work is that they must be *structured* in a way that allows the selection of fields or byte ranges that are deemed unique and those that are not.

Based on the definition of an electronic document described in [Solovyev 2023], a document d can be formally defined as an ordered set of n semantic blocks b, such that $d = (b_1, b_2, \ldots, b_n) = \bigcup_{i=1}^{n} b_i$. Consequently, the set of documents D may be represented as $D = \{d_1, d_2, \ldots, d_n\} = \{(b_{(1,1)}, \ldots, b_{(1,n_1)}), \ldots, (b_{(n,1)}, \ldots, b_{(n,n_n)})\}$. Additionally, as per the ordered requirement a document $d_1 = (b_1, b_2) \neq (b_2, b_1) = d_2$. Lastly, we note that each b_i represents a fraction of the document, whether big or small, and does not necessarily have a fixed size or meaning; each b_i may correspond to a single byte, the whole template, or some part of confidential information. It follows from this definition that the intersection of shared content from a set of documents D containing j documents, such that $d_i \in D$, is defined as the template T, formally defined by Equation 1:

$$T = \bigcap_{i=1}^{n} d_i = \{b_{i,j} \mid 1 \leq i \leq n,\ 1 \leq j \leq |d_i|\} \tag{1}$$

That is, the set T contains every common block among the selection of documents in D. Consequently, the set of unique data $U = (u_1, u_2, \cdots, u_n)$ is composed of the symmetric difference (Δ) between the documents so that $U = d_1\ \Delta\ ...\Delta\ d_n = D - T$. Furthermore, it is straightforward to see that the average size of a document $\bar{D}$ can be measured as the sum of the average size of the set of unique data $\bar{U}$ plus the size of the template T such that $\bar{D} = \bar{U} + T$. The relation between the documents, template, and unique data sets can be seen in Figure 1.

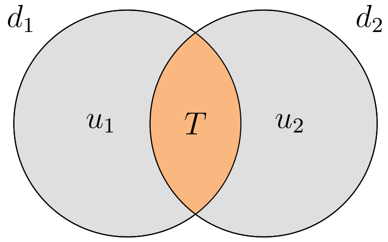


Figure 1: Visual representation of the components that form a document.

The SDDup scheme leverages the natural redundancy often found in collections of similar documents by isolating shared content (hereafter called *template*) from unique information (termed *unique data*). This separation enables efficient storage, retrieval, and encryption of documents while preserving their semantic and structural integrity, which will be discussed in this section. It is worth noting that this proposal requires semantic blocks to be tracked so that the original document can be reconstructed later. Furthermore, the selection of semantic blocks is out of the scope of this proposal as it may vary depending on the format and purpose of the document in question.

SDDup is designed to comply with data protection regulations such as Brazil's LGPD (General Data Protection Law) and the European GDPR. The security properties analyzed in this section support regulatory compliance in several ways: i) data minimization: By encrypting only the unique personal data rather than entire documents, SDDup adheres to the principle of minimizing exposed sensitive information; ii) purpose Limitation: The separation of templates and unique data facilitates access controls that can limit data usage to specific purposes; iii) security by design: The cryptographic assurances for confidentiality and integrity support the requirement for appropriate security measures; iv) data subject rights: The model facilitates the right of access by providing efficient data retrieval while maintaining security.

This separation allows: i) *target encryption*: As an alternative to encrypting the document as a whole, the set of unique data may be encrypted with differing levels of granularity, from the whole set to every block b_i; ii) *data segregation*: Unique data may be stored in distinct databases, such that a data breach does not expose all the data about the user or the whole document, even if the encryption is broken; iii) *storage efficiency*: Blocks that belong to the template T set can be stored as a single entity, increasing storage efficiency, especially for environments with large amounts of documents.

Separating template and unique data can be done in many ways, such as having a previously defined template set. Algorithm 1 exemplifies a generic way of generating a template out of a pre-existing set of documents, removing unique information as it iterates over the set. The semantic blocks that remain belong to the template. Document division may also be performed in different ways. Algorithm 2 exemplifies a generic procedure that iterates over the semantic blocks of a document, selecting any blocks that do not belong to the given template.

Algorithm 1: Template Selection

```
Input  : document set $D$
Output : template $T$
$T = \{\}$
for $d_i \in D$ do
    if $i = 1$ then
        $T \leftarrow d_1$
    else
        $T \leftarrow T \cap d_i$
return $T$
```

Algorithm 2: Document Division

```
Input  : template $T$,
         symmetric key $ek$,
         document set $D$
Output : encrypted data $u$
$u = \{\}$
for $b_i \in d$ do
    if $b_j \notin T$ then
        $u \leftarrow u \cup \mathsf{Enc}_{\mathsf{ek}}(b_i)$
return $u$
```

We note that there is no intrinsic distinction between signed and unsigned documents for this proposal as far as feasibility is concerned, only impacting the overall performance depending on the type of signature present. Any public key infrastructure (PKI) artifact may belong as part of the template, such as certificate revocation lists, or be part of the unique data. Furthermore, SDDup enables the encryption of selective pieces of the document, aligning with the data minimization principle [Biega and Finck 2021] and reducing some cryptographic overhead compared to full-document encryption.

Moreover, by reconstructing the exact original signed document, any signatures that were present maintain their original validity, preserving any assurances given by the underlying PKI and signature scheme. Lastly, while not in the scope of this article, systems may recursively apply SDDup, creating hierarchical templates that create subsets of templates that can be reconstructed before the final document is reassembled. That is, one or more *template of templates* may be generated, exchanging document processing performance for increased storage efficiency.

3.1. Theoretical performance

In this section, we explore the theoretical performance implications of SDDup and the expected behavior from small to large amounts of documents, as well as the difference between having comparable sizes for the template and unique data. Let $\bar{D}$ and $\bar{U}$ denote the average sizes of a full document and unique data set, respectively. We compare the typical naive technique to the SDDup model:

Naive. As previously mentioned, the naive technique involves the encryption of each document d_i as a whole before storage. The total storage S_{Naive} required to store n documents in this way can be expressed as either the sum of the sizes of every document or the average size of documents in D times n, such that:

$$S_{\text{Naive}}(n) = \sum_{i=1}^{n} \mid d_i \mid = n \cdot \bar{D} \tag{2}$$

SDDup. Conversely, the SDDup Model segregates the document template T and unique data u. The total storage requirement for n documents then becomes the sum of the template and unique data for each document, which can also be represented as the average size of unique data times n, consequently:

$$S_{\text{SDDup}}(n) = \mid T \mid + \sum_{i=1}^{n} \mid u_i \mid = \mid T \mid + n \cdot \bar{U} \tag{3}$$

For any set of documents larger than one with any amount of data overlap, it is clear that the SDDup model has increased storage performance, as the overlap is not stored for every document. For sets with few documents and comparable unique data and template sizes, both models start similarly; however, as the number of documents grows, the storage required by SDDup rises significantly slower than the naive technique.

To quantify the efficiency of the SDDup Model compared to the naive technique, we define a growth ratio $R(n) = \frac{S_{\text{SDDup}}}{S_{\text{Naive}}}$, representing the proportion of storage required by the SDDup Model relative to a naive technique and an efficiency ratio $E(n) = 1 - R(n)$ which represents the relative efficiency of SDDup when compared to the naive model. That is, a growth ratio of $R(n) = 0.3$ shows that the SDDup model uses 30% of the storage space used by a naive approach, and an efficiency ratio of $E(n) = 0.7$ shows that the SDDup approach is 70% more efficient than the naive technique. From Equations 2 and 3, the maximum theoretical growth ratio R_{max} and efficiency plateau E_{max} as the number of documents grows indefinitely can be calculated as follows:

$$R_{\text{max}} = \lim_{n \to \infty} \frac{S_{\text{SDDup}}}{S_{\text{naive}}} = \lim_{n \to \infty} \frac{\mid T \mid + n \cdot \bar{U}}{n \cdot \bar{D}} = \frac{\bar{U}}{\bar{D}} = \frac{\bar{U}}{\mid T \mid + \bar{U}} \tag{4}$$

$$E_{\text{max}} = 1 - R_{\text{max}} = 1 - \frac{\bar{U}}{\bar{D}} = 1 - \frac{\bar{U}}{\mid T \mid + \bar{U}} \tag{5}$$

From these equations, we can glean the behavior of E_{max} for distinct sizes of average unique data and template. We note three main scenarios for SDDup from this graph when comparing template and average data size: i) for comparable sizes, the expected maximum efficiency ratio E_{max} should stay near 50%; ii) scenarios where the average unique data is higher than the template ought to show a lower efficiency ratio; and iii) as expected, the best scenario shows that for sets of documents with larger templates, E_{max} rises much faster. That is, SDDup is best suited for environments with high degrees of redundancy and can be expected to have high maximum efficiency ratios when dealing with document types with levels of intrinsic redundant data.

3.2. Security Discussion

The main difference between SDDup and the technique of encrypting data regardless of its semantic meaning is meaningful data granularity and selective encryption. Thus, we analyze what type of information an attacker can glean from different parts of the model, particularly from plain-text template sets. This section analyses the security properties of the SDDup model, examining how the separation of documents into templates and unique data components affects its security properties. We assume the cryptographic primitives used are secure according to their definitions, and that they hold, unless stated otherwise.

Let Enc be an IND-CPA encryption function and ek an encryption key. The presence of the plain-text template T does not help the adversary distinguish between $\mathsf{Enc}_{\mathsf{ek}}(u_1)$ and $\mathsf{Enc}_{\mathsf{ek}}(u_2)$ because T is, by definition, common to all documents and contains no information specific to either u_1 or u_2. However, the template T may inadvertently reveal information about the structure, format, or origin of the documents. Nevertheless, this only happens if there is a link between a particular set U and its corresponding template. Furthermore, for document types where the structure of the template or some sections may be considered sensitive information, the specific semantic block may be encrypted before storage and decrypted when reconstructing the original document. We now analyze some attack vectors explicitly:

Attack 1 (Template Substitution)**.** *An attacker replaces the legitimate template T with a malicious template T' to alter the reconstructed signed document.*

Analysis. This attack is prevented by the digital signature on the original document. Since the signature σ_i was generated over the entire original document $d_i = T \cup u_i$, replacing T with a distinct T' will result in a different document, invalidating the signature. □

Attack 2 (Data Reconstruction)**.** *Given multiple encrypted unique data fragments* $\mathsf{Enc}_{\mathsf{ek}}(u_1), \mathsf{Enc}_{\mathsf{ek}}(u_2), ..., \mathsf{Enc}_{\mathsf{ek}}(u_n)$ *and the template T, an attacker attempts to infer patterns or correlations to reconstruct sensitive information.*

Analysis. This attack is mitigated by the underlying encryption scheme. The ciphertext reveals no information about the plaintext, even if the same field is encrypted multiple times across different documents. □

Attack 3 (Metadata Analysis)**.** *An attacker analyzes metadata such as the size of encrypted unique data blocks to infer sensitive information without breaking encryption.*

Analysis. An attacker might infer some information about the *type* of details of a particular encrypted semantic block by analyzing its size and corresponding template. To mitigate this attack, padding may be employed prior to encryption. □

Furthermore, a security advantage of SDDup over traditional approaches is the reduced attack surface for sensitive data, as only a fraction of the document is encrypted and requires strict security controls. This aligns well with the principle of data minimization and may make document management systems more compliant from a security perspective. Thus, these security properties make SDDup a robust solution for large-scale document management systems that handle sensitive information.

4. Case Studies

In this section, we detail the implementation of our proposed model, the simulation setup, and the empirical results obtained during the evaluation process, and expand the results on the function of n documents. The goal is to compare the efficiency, in real use-cases, of the SDDup model with state-of-the-art alternatives for reducing storage requirements while ensuring data confidentiality and integrity. The experiments investigate Brazilian birth certificates (Brazil's Civil Registry) and Brazilian Degree Certificates (Federal Universities), which both feature high structural redundancy. To evaluate the proposed model, a simulation environment was designed and implemented[1] with *Python 3* and *Poetry*; utilizing AES-256-CBC to encrypt data, *lxml*, *fillpdf*, and *pyhanko* for document manipulation, and the Faker library to generate random citizen information.

4.1. Case Study: Civil Registry

Brazil's Civil Registry issues critical documents such as birth certificates, which will be the focus of this use case. Managing such documents involves significant challenges: i) *high storage requirements:* over 235 million registered citizens and 5 million new entries annually, there is a need for efficient document management and storage; ii) *ensuring data confidentiality:* sensitive citizen information must remain secure, adhering to regulations like the LGPD (General Data Protection Law); and iii) *document integrity and authenticity:* official documents must be tamper-proof and verifiable.

The process of issuing a birth certificate involves an officially issued PDF template; a notary fills in the personal information of the corresponding citizen, populating the PDF template with citizen information; Finally, the birth certificate is officially signed and issued by a registrar and delivered to the requesting citizen.

SDDup process. Given that there is an official form-fillable PDF template that must be used by the civil registry, the template selection for the SDDup model is straightforward; we set the PDF template as our SDDup template and the filled-in information as the set of unique confidential data. The document is segregated into a shared template T (the official PDF template) and unique data u (which includes personal data, the registrar's signature, and some indexing overhead). Therefore, we begin the division step of SDDup after each birth certificate is issued, analogous to Algorithm 2. Afterward, the information can be processed individually, and the document may be reassembled by combining the template T with the corresponding decrypted unique data u. The birth certificate template was obtained from Brazil's Civil Registry. The final template size is $432\,029$ bytes.

Results and discussions. The evaluation focused on assessing the storage efficiency of the SDDup model, particularly the impact of segregating the document into a shared template T and unique data u. The average size of a birth certificate is $441\,501$ bytes, and the size of the final unique data u stored equals $12\,840$ bytes. In this manner, the Storage required to hold birth certificates, growth rate $R(n)$, and the efficiency ratio $E(n)$, can be obtained

[1] `https://anonymous.4open.science/r/sddup-poc/`

by applying Equation 3, and the subsequent equations described in Section 3.1:

$$R(n) = \frac{S_{\text{SDDup}}(n)}{S_{\text{Naive}}(n)} = \frac{432\,029 + n \cdot 12\,840}{n \cdot 441\,501} \qquad [\text{B}] \tag{6}$$

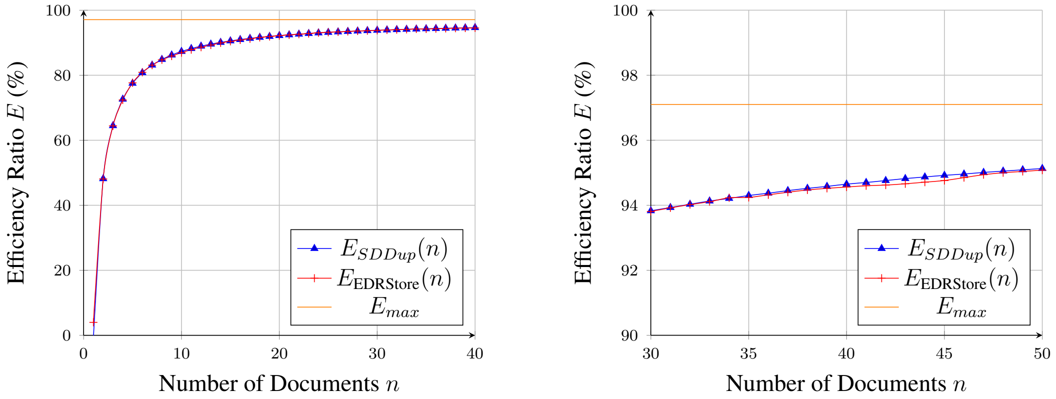
$$E(n) = 1 - R(n) = 1 - \frac{432\,029 + n \cdot 12\,840}{n \cdot 441\,501} \qquad [\text{B}] \tag{7}$$

Find their theoretical limits through Equations 4 and 5 respectively:

$$R_{\max}(n) = \lim_{n\to\infty} \frac{432\,029 + n \cdot 12\,840}{n \cdot 441\,501} = 0.029 = 2.9\% \qquad [\text{B}] \tag{8}$$

$$E_{\max}(n) = \lim_{n\to\infty} 1 - \frac{432\,029 + n \cdot 12\,840}{n \cdot 441\,501} = 0.971 = 97.1\% \qquad [\text{B}] \tag{9}$$

To compare the performance of SDDup to a modern and efficient deduplication approach, EDRStore [Zhao et al. 2024], we generate several birth certificates, extract the unique data, and store them utilizing both solutions. For every document added to the storage, we measure the efficiency of both solutions and plot it in Figure 2.



(a) Efficiency ratio $E_{\text{SDDup}}(n)$, $E_{\text{ERDStore}}(n)$, and $E_{\max}$ for increasing number of documents n.

(b) Zoomed-in view showing the differences between $E_{\text{SDDup}}(n)$ and $E_{\text{ERDStore}}(n)$ at around 40 documents.

Figure 2: Efficiency ratios $E(n)$ for the Brazilian birth certificate use case.

EDRStore's demonstrates strong performance, with its efficiency ($E_{\text{EDRStore}}(n)$) exceeding 86% at $n = 10$, reaching 93.81% for $n = 30$, and stabilizing around 96% for $n = 85$. Our experimental results for SDDup show an efficiency ($E_{\text{SDDup}}(n)$) of 87.31% for $n = 10$, 93.83% for $n = 30$, and also stabilizing around 96% for $n = 85$. This aligns with SDDup's theoretical maximum efficiency ($E_{\max}$) of 97.1% for this dataset. From Figure 2, we see that both frameworks achieve high efficiency rapidly and remain very close in performance throughout the experiment.

We show that SDDup's semantic-aware approach is highly competitive in storage efficiency, even against advanced generic data reduction systems (EDRStore), when dealing with structured documents with significant redundant data. Furthermore, the SDDup approach is complemented by its advantages in preserving document semantics and integrity, such as maintaining the validity of signed documents.

4.2. Case Study: Brazilian Degree Certificate

In this scenario, we have slightly different requirements and characteristics when compared to the previous PDF file of birth certificates. Here, degrees are issued in an XML format and contain many signatures and PKI artifacts common to each other; however, there are far fewer degrees being issued than birth certificates. Thus, we now analyze a scenario with fewer documents but far higher redundant data in the form of certificates and certificate revocation lists.

As mentioned, the bulk of the template this time belongs to many public key infrastructure artifacts necessary to validate the signature; these degrees have digital certificates from the signers and the respective certificate authorities issuers, along with their certificate revocation lists, and timestamps to guarantee long-term validation capabilities. For this use case, we simulate degrees issued by a Brazilian federal university as our use case. In total, there are thirteen timestamps and six signatures in such a degree: i) the university's chancellor; ii) the courses' coordinator; iii) twice for the university as an entity; iv) the operator responsible for the degree's registration; and v) the chief entity responsible for the registry; While we simulate and analyze a single batch of documents for brevity and simplicity, we call attention to the fact that these artifacts may be part of multiple distinct batches such that the template T may be deconstructed into smaller templates T_n through repeated applications of SDDup. This is briefly discussed further ahead in Section 5.

In the same manner as the previous use case on birth certificates, Brazilian degree certificates also have similar challenges; i.e., i) high storage requirement, albeit less so when compared to birth certificates, as degrees must be stored for long periods; ii) data confidentiality, as credits and scores can also be considered sensitive information, along with traditional personal information; and iii) document integrity and authenticity.

Furthermore, this use case also has an official XML template that must be followed for degree certificates [Brasil 2020]. Likewise, as degrees are typically issued as a batch process, we may include the certificates used by the signers as part of the degree issuance process in our template. Then, our template T comprises the XML template, the signer's certificates, and any accompanying signature artifacts such as revocation lists and certificates. We note that some artifacts change from degree to degree, such as timestamps and document signatures, and are included in the unique dataset. For these degrees, we have an average unique data size of $231\,724$ bytes and a template size of $6\,437\,253$ bytes, which is composed mostly of PKI artifacts.

SDDup. The SDDup Model separates the personal information and unique PKI artifacts u, such as timestamps and some signatures, from the public information in the XML, along with any redundant PKI artifacts embedded in the degree as the template T. Thus, u contains encrypted student data and some PKI artifacts, while T contains certificates and certificate revocation lists.

Results and discussions. Storage required to hold degrees, growth rate $R(n)$, and the efficiency ratio $E(n)$, can be obtained by applying Equation 3, and subsequent equations described in Section 3.1:

$$R(n) = \frac{S_{\text{SDDup}}(n)}{S_{\text{Naive}}(n)} = \frac{6\,437\,253 + n \cdot 231\,724}{n \cdot 6\,667\,419} \quad [\text{B}] \qquad (10)$$

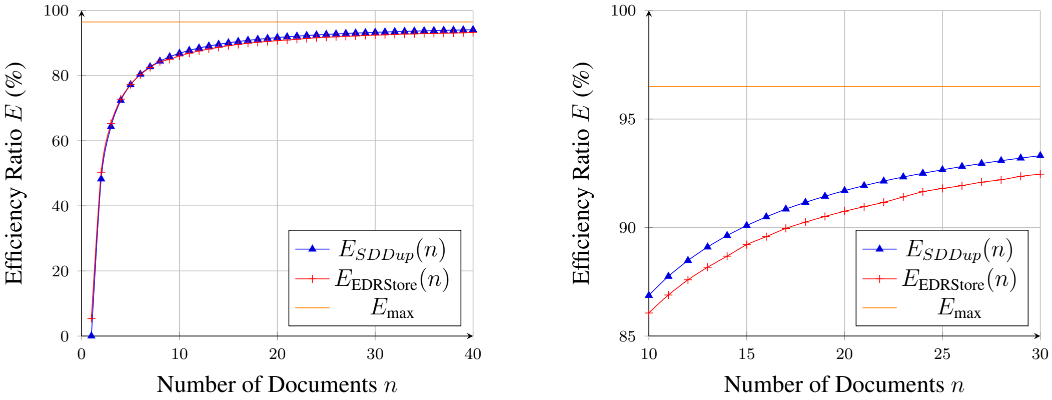
$$E(n) = 1 - R(n) = 1 - \frac{6\,437\,253 + n \cdot 231\,724}{n \cdot 6\,667\,419} \quad [\text{B}] \qquad (11)$$

To measure the growth and efficiency ratios, and their theoretical limits of this use case, we calculate R_{max} and E_{max}:

$$R_{\text{max}}(n) = \lim_{n\to\infty} \frac{6\,437\,253 + n \cdot 231\,724}{n \cdot 6\,667\,419} = 0.035 = 3.5\% \qquad (12)$$

$$E_{\text{max}}(n) = \lim_{n\to\infty} 1 - \frac{6\,437\,253 + n \cdot 231\,724}{n \cdot 6\,667\,419} = 0.965 = 96.5\% \qquad (13)$$

To further assess SDDup's effectiveness, in the same manner as done in Section 4.1, for XML-based degree certificates, we compared its storage efficiency with EDRStore [Zhao et al. 2024]. The results, using the experimental data for this specific use case, are presented in Figure 3.



(a) Efficiency ratio $E_{\text{SDDup}}(n)$, $E_{\text{ERDStore}}(n)$, and E_{max} for increasing number of documents n.

(b) Zoomed-in view showing the differences between $E_{\text{SDDup}}(n)$ and $E_{\text{ERDStore}}(n)$ at around 20 documents.

Figure 3: Efficiency ratios $E(n)$ for the Brazilian degree certificate use case.

For this use case, our experiments show EDRStore's efficiency ($E_{\text{EDRStore}}(n)$) rapidly growing to 86.06% for $n = 10$, reaching 92.46% at $n = 30$, and plateauing at 93.95% at $n = 60$. Our experimental data for SDDup shows comparable initial growth to EDRStore, surpassing it after just six documents and rising to 86.87% for $n = 10$, 93.31% for $n = 30$, and 94.92% for $n = 60$. SDDup's theoretical maximum efficiency (E_{max}) for these degree certificates is 96.5%. This represents around 1.5% differential in efficiency at 60 documents, owing to its staggering performance due to the large amount of redundant and sizable PKI artifacts in this use case. In contrast to the previous scenario, EDRStore demonstrates lower initial optimization efficiency for XML Degree certificate files. The compression system's performance when processing a single XML file was substantially less effective, potentially due to the inherently compact structure of XML and EDRStore's approach to chunk processing. The curves observed in Figure 3 demonstrate a more pronounced separation between approaches for multiple files. The results indicate that SDDup offers superior performance as the number of files increases, establishing an increasing advantage as more documents are added.

SDDup's semantic-aware approach is highly competitive in storage efficiency, even against advanced generic data reduction systems (EDRStore), when dealing with structured documents with significant redundant data. Despite the minor differences in efficiency, the SDDup's advantages in preserving document semantics and additional characteristics are elaborated upon in the subsequent section.

5. General Discussion

The performance evaluations in Section 4 demonstrated that SDDup achieves high storage efficiency, comparable and in some scenarios theoretically superior to EDRStore [Zhao et al. 2024], a state-of-the-art system for generic encrypted data reduction. Beyond raw storage savings, SDDup's semantic-aware design offers distinct advantages. It addresses different aspects of secure document management while maintaining a simple design that can be adapted to existing software in a straightforward manner. Table 1 provides a consolidated comparison of SDDup and EDRStore, highlighting their fundamental architectural and philosophical differences, which become particularly relevant when considering the specific requirements of structured, usually signed, official documents.

Table 1: Comparative Analysis: SDDup vs. EDRStore

Characteristic	SDDup (Proposed)	EDRStore ([Zhao et al. 2024])
Primary Granularity	Document: semantic level: template + unique data fields	Data block: generic chunks after Content-Defined Chunking
Semantic Awareness	High: distinguishes template from unique data fields.	Low: Identifies similarity between generic data blocks via features.
Encryption Strategy	Selective encryption of data.	Encryption of all blocks.
Confidentiality	Semantic block.	All data.
Client-Side Complexity	Semantic segregation; unique data encryption and metadata for reconstruction.	Feature generation (twice), selective local compression, two-phase encryption, and key server interaction.
Cloud-Side Complexity	Storage and retrieval of templates and unique data.	Deduplication (dual-fingerprint), delta compression, management of multiple indexes.
Primary Focus	Efficient and secure management of *structured and signed documents*, emphasizing selective confidentiality and semantics.	Redundancy reduction in *generic encrypted data* for outsourced storage.

The distinctions outlined in Table 1 emphasize that while EDRStore excels at optimizing generic encrypted byte streams, SDDup is tailored for environments where document structure, semantic meaning, selective confidentiality, and the integrity of embedded digital signatures are paramount. For instance, SDDup's ability to treat the template as a separate, potentially public entity, while only encrypting sensitive fields, aligns closely with data minimization principles and facilitates more granular access control and specialized processing (e.g., analysis over non-sensitive template data) that are not primary concerns for generic encrypted deduplication systems. Furthermore, the explicit preservation of the original document structure by SDDup simplifies the lifecycle management of signed documents.

6. Conclusion

This paper shows how a Confidentiality-Aware Semantic Deduplication model can be designed to address critical challenges in large-scale document management concerning storage efficiency, data security, and integrity. SDDup's core innovation lies in its semantic segmentation of structured documents into shared templates and unique, potentially sensitive data. This allows for highly efficient single-instance template storage while enabling targeted, robust encryption only for the sensitive, unique portions of each document.

We achieve substantial storage reductions, up to 97.1% for Brazilian birth certificates and around 94.9% for Brazilian degree certificates compared to the naive traditional technique. The model's scalability highlights its applicability to diverse domains beyond civil registries. This research demonstrated that the SDDup model is feasible and theoretically sound in real-world applications while improving upon the expected security of critical documents. Beyond sheer storage metrics, the fundamental architectural differences between SDDup and generic systems like EDRStore yield distinct impacts on security and document lifecycle management. SDDup's semantic-aware design, which treats templates and unique data as distinct entities, inherently supports principles of data minimization by allowing the encryption of only what is necessary. This selective confidentiality reduces the attack surface for sensitive information and facilitates more granular access controls. Furthermore, unlike block-level generic systems that may obscure document structure after encryption and complex transformations, SDDup's explicit preservation of the original document's structural integrity through its template-based approach directly simplifies the management and long-term verifiability of signed digital documents, a paramount concern for official and legal documents.

Future work. The ability to reduce storage costs, secure sensitive data, and preserve document signatures positions it as a valuable tool for government agencies, financial institutions, and large-scale document repositories. Future works may explore extending the model to different domains, assessing its efficiency under different document characteristics and regulatory frameworks, such as medical records. Particularly, we expect SDDup to fit well into InterPlanetary File System (IPFS) protocols, leading to large reductions in storage requirements for environments with notorious storage issues. Furthermore, additional performance enhancements may be studied for different use cases where there is redundancy between multiple templates, leading to the multiple uses of SDDup to enhance the storage of the templates themselves. Moreover, analyzing and comparing different semantic classification techniques, such as machine learning, and how they impact the semantic deduplication process overall, may show interesting results. Finally, exploring the properties gained by the newfound granularity of the unique data may lead to some interesting findings.

Acknowledgments

This study was financed in part by the *Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil* (CAPES) – Finance Code 001 and *Fundação de Amparo a Pesquisa e Inovação do Estado de Santa Catarina* (FAPESC), *Edital* 19/2024.

References

[Bellare et al. 2013] Bellare, M., Keelveedhi, S., and Ristenpart, T. (2013). Message-locked encryption and secure deduplication. In *Annual international conference on the theory and applications of cryptographic techniques*, pages 296–312. Springer.

[Biega and Finck 2021] Biega, A. J. and Finck, M. (2021). Reviving purpose limitation and data minimisation in data-driven systems. *arXiv preprint arXiv:2101.06203*.

[Brasil 2018] Brasil (2018). Lei nº 13.079, de 14 de agosto de 2018. Lei Geral de Proteção de Dados Pessoais (LGPD). *Diário Oficial da União*, 157(1):59–64.

[Brasil 2020] Brasil (2020). INSTRUÇÃO NORMATIVA Nº 1, DE 15 DE DEZEMBRO DE 2020.

[Chakaravarthy et al. 2008] Chakaravarthy, V. T., Gupta, H., Roy, P., and Mohania, M. K. (2008). Efficient techniques for document sanitization. In *Proceedings of the 17th ACM conference on Information and knowledge management*, pages 843–852.

[European Union 2018] European Union (2018). General data protection regulation, regulation (eu) 2016/679.

[Friedlin and McDonald 2008] Friedlin, F. J. and McDonald, C. J. (2008). A Software Tool for Removing Patient Identifying Information from Clinical Documents. *Journal of the American Medical Informatics Association*, 15(5):601–610.

[Iwendi et al. 2020] Iwendi, C., Moqurrab, S. A., Anjum, A., Khan, S., Mohan, S., and Srivastava, G. (2020). N-sanitization: A semantic privacy-preserving framework for unstructured medical datasets. *Computer Communications*, 161:160–171.

[Kardaş and Kiraz 2016] Kardaş, S. and Kiraz, M. S. (2016). Solving the secure storage dilemma: An efficient scheme for secure deduplication with privacy-preserving public auditing. *Cryptology ePrint Archive*.

[Keelveedhi et al. 2013] Keelveedhi, S., Bellare, M., and Ristenpart, T. (2013). {DupLESS}:{Server-Aided} encryption for deduplicated storage. In *22nd USENIX security symposium (USENIX security 13)*, pages 179–194.

[Li et al. 2016] Li, J., Qin, C., Lee, P. P., and Li, J. (2016). Rekeying for encrypted deduplication storage. In *2016 46th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*, pages 618–629. IEEE.

[Lin et al. 2021] Lin, J. C.-W., Srivastava, G., Zhang, Y., Djenouri, Y., and Aloqaily, M. (2021). Privacy-preserving multiobjective sanitization model in 6g iot environments. *IEEE Internet of Things Journal*, 8(7):5340–5349.

[Liu et al. 2020] Liu, X., Yang, G., Susilo, W., Tonien, J., Chen, R., and Lv, X. (2020). Message-locked searchable encryption: A new versatile tool for secure cloud storage. *IEEE Transactions on Services Computing*, 15(3):1664–1677.

[Miranda et al. 2021] Miranda, M., Esteves, T., Portela, B., and Paulo, J. (2021). S2dedup: Sgx-enabled secure deduplication. In *Proceedings of the 14th ACM international conference on systems and storage*, pages 1–12.

[Solovyev 2023] Solovyev, A. V. (2023). The problem of defining the concept of "electronic document for long-term storage". In Silhavy, R., Silhavy, P., and Prokopova, Z., editors, *Data Science and Algorithms in Systems*, pages 326–333, Cham. Springer International Publishing.

[Stanešić et al. 2025] Stanešić, J., Morić, Z., Regvart, D., and Bencarić, I. (2025). Digital signatures and their legal significance. *Edelweiss applied science and technology*, 9(1):403–412.

[Wu et al. 2022] Wu, Z., Xuan, S., Xie, J., Lin, C., and Lu, C. (2022). How to ensure the confidentiality of electronic medical records on the cloud: A technical perspective. *Computers in biology and medicine*, 147:105726.

[Yan et al. 2016] Yan, Z., Jiang, H., Tan, Y., and Luo, H. (2016). Deduplicating compressed contents in cloud storage environment. In *8th USENIX Workshop on Hot Topics in Storage and File Systems (HotStorage 16)*.

[Zhao et al. 2024] Zhao, J., Yang, Z., Li, J., and Lee, P. P. (2024). Encrypted data reduction: Removing redundancy from encrypted data in outsourced storage. *ACM Transactions on Storage*, 20(4):1–30.