

# System for Securing Timestamping and Electronic Signing of PDF Documents: Standards, Schematic, and Implementation

Moussa Habiboulaye<sup>1</sup>, Harouna Naroua<sup>2</sup> and Chaibou Kadri<sup>3</sup>

Department of Mathematics and Computer Science, Abdou Moumouni University, Niamey, Niger

## Article history

Received: 05 May 2025

Revised: 16 July 2025

Accepted: 31 July 2020

Corresponding Author:

Moussa Habiboulaye

Department of Mathematics

and Computer Science,

Abdou Moumouni University,

Niamey, Niger

Email:

musa\_habibou@yahoo.com

**Abstract:** This article investigates the standards and principles for implementing electronic signatures in PDF documents, focusing on security and compliance with legal regulations. The current state of digital signatures is examined, highlighting the importance of authenticity, integrity, and non-repudiation. A detailed analysis is provided on the process of generating and verifying electronic signatures using Public Key Infrastructures (PKI) and cryptographic techniques. The article also explores the legal aspects of adopting and implementing electronic signatures. The results revealed that the solution is robust, in the sense that the system's only entry points are the input and output storage directories. The authenticity and integrity of the produced documents are guaranteed using asymmetric cryptography, which leads to a high-level confidence production of financial documents.

**Keywords:** Electronic Signature; Data Security; Cryptography; Authenticity; Integrity; Non-Repudiation.

## Introduction

In an increasingly digitized world, the security and authenticity of electronic documents have become crucial issues, particularly in the field of financial data. Institutions and companies handling sensitive data face a significant challenge: ensuring the integrity, confidentiality, and non-repudiation of documents generated from their databases. In this context, the electronic signing of PDF documents can emerge as an effective solution.

The implementation of a signing system relies on a PKI, based on cryptography, which generates, distributes, and revokes certificates during verification. The PKI enables the establishment of a secured communication protocol between users of a system through its various components (Jia & Li, 2022). The setup of such components can be costly and is not available in many countries. Therefore, the study of an alternative solution is necessary. The U.S. Department of Homeland Security has proposed a guide for implementing electronic signatures for internal and external use, which describes specific implementation processes and acceptance criteria for electronic

signatures (U.S. Department of Homeland Security, 2022). Although based on these technologies, the signing of PDF documents is relatively specific to justify the implementation of dedicated mechanisms. Even today, many people prefer the use of manual signatures over digital ones, as noted by Zubov (2020) in his comparative study on this issue. Onuoha et al. (2020) conducted research on electronic signatures, examining their various forms and legal recognition under Nigerian regulations. His study provided an in-depth review of existing electronic signature types and their compliance with legal frameworks. However, a significant challenge remains: translating these findings into practical technical implementations to address real-world societal concerns effectively.

Other researchers, such as Naazet al. (2023), have presented digital signatures, highlighting their significant advantages in terms of security and time savings in transactions. They have also explored the various associated methods and algorithms, including RSA (Rivest-Shamir-Adleman), DSA (Digital Signature Algorithm), and ECDSA (Elliptic Curve Digital Signature Algorithm). This study could be enriched by an in-depth exploration of software architecture and its implementation process.

Research on electronic signatures has already been the subject of numerous studies, particularly concerning their security and compliance with legal standards such as the eIDAS regulation (Hölbl et al., 2023). However, as Mulder et al. (2023) pointed out in the reference book *Trends in Data Protection and Encryption Technologies*, significant gaps remain regarding their specific integration into sensitive environments such as financial data management. The limitations of current systems fully justify the need for further research aimed at developing a more robust solution adapted to contemporary requirements.

This study aims to analyze and develop an electronic signature system for PDF documents, integrated into a broader ecosystem of financial data security. The objective is to design a robust solution, compliant with current standards, and capable of meeting the specific requirements of the problem at hand. It also provides confidence on the production of financial documents in a digitalized environment.

## Current Standards and Schematic of Electronic Signatures for PDF Documents

The implementation of electronic signatures in PDF documents is governed by a comprehensive framework of international standards that ensure interoperability, security, and legal validity across jurisdictions. The principal standards regulating electronic signature implementation include:

**ISO 32000 Series.** The ISO 32000-1:2008 and ISO 32000-2:2017 standards define the Portable Document Format (PDF) specification, including technical requirements for embedding electronic signatures within PDF documents (International Organization for Standardization, 2020). These standards establish the foundational architecture for signature data structures, cryptographic algorithms, and verification protocols.

**ISO 14533-3:2017.** This standard specifies processes for creating and validating long-term electronic signatures (LTV) in PDF documents, addressing challenges related to signature validity preservation beyond certificate expiration and cryptographic algorithm obsolescence (International Standard, 2017). The standard defines mechanisms for embedding validation data, including certificate chains, revocation information, and timestamps, to enable signature verification over extended timeframes.

**ETSI TS 102 778 (PAdES).** The European Telecommunications Standards Institute (ETSI) Technical Specification 102 778 defines PDF Advanced Electronic Signatures (PAdES), a set of profiles establishing advanced electronic signature requirements specifically tailored for

PDF documents (European Telecommunications Standards Institute, 2009). PAdES profiles ensure compliance with European electronic signature regulations while maintaining backward compatibility with baseline PDF signature specifications.

The PAdES standard is a set of constraints and extensions to PDF signatures that make them compatible with the European directive on electronic signatures (Boudrez, 2017). It is divided into several parts:

- PAdES Basic, which covers the basic requirements for electronic signatures in PDFs;
- PAdES Enhanced, which adds information for long-term validation;
- PAdES Long Term, which includes timestamps and revocation information for extended validity.

The PAdES standard ensures that electronic signatures in PDFs are authentic (the signer's identity is verifiable), integral (any modification of the document after signing is detectable), non-repudiable (the signer cannot deny having signed the document), and compliant with European regulations, particularly the eIDAS regulation (electronic iDentification, Authentication and trust Services) (European Union, 2014). The process of electronically signing a PDF document relies on several key steps (see Table 1) that ensure the authenticity, integrity, and non-repudiation of the signed document (International standard ISO/IEC 9594-8, 2020).

This process shows the use of asymmetric encryption for PDF documents: the private key held by the sender (alone) is used to encrypt the document's hash, and the public key is used to decrypt the hash received by the recipient(s), ensuring that:

- Only the holder of the private key can create the signature (authentication);
- Any modification of the document after signing will be detected (integrity);
- The signer cannot deny having signed the document (non-repudiation).

The European Union's eIDAS Regulation (EU) No 910/2014 establishes a three-tier taxonomy of electronic signatures, each defined by specific technical requirements and corresponding legal validity. This hierarchical framework accommodates diverse security needs across digital transaction contexts.

**Simple Electronic Signature (SES).** The foundational tier represents any electronic data attached to or logically associated with other electronic data, serving as a method of authentication. Common implementations include scanned handwritten signatures, typed names, or

checkbox acknowledgments. While legally admissible, SES provides minimal technical safeguards and lacks inherent mechanisms to verify signer identity or detect document tampering, limiting applicability to low-risk transactions where signature contestability is acceptable.

**Table 1:** Schematic of Electronic Signatures

Stage	Description
Generating the pair of keys (public, private)	<ul style="list-style-type: none"><li>- The signatory has a unique pair of cryptographic keys: a private key (kept secret) and a public key (freely distributed).</li><li>- These keys are generally created using asymmetric algorithms such as RSA or ECDSA (Menezes et al. 2018).</li></ul>
calculation of fingerprint with private key	<ul style="list-style-type: none"><li>- A cryptographic hash function (such as SHA-256 or SHA-3) is applied to the content of the PDF document (Stevens et al. 2017).</li><li>- This operation generates a unique digital footprint of the document, of fixed size, which represents its content.</li></ul>
encryption of fingerprint with private key	<ul style="list-style-type: none"><li>- The document's fingerprint is encrypted using the signatory's private key.</li><li>- This operation creates the electronic signature itself (Katz &amp; Lindell, 2020).</li></ul>
Integration of the signature in the PDF document	<ul style="list-style-type: none"><li>- The electronic signature is incorporated into the PDF document</li><li>- Additional metadata is added, including:<ul style="list-style-type: none"><li>* The signer's certificate (containing his public key);</li><li>* The timestamp of the signature,</li><li>* Information about the hashing and encryption algorithm used (Adobe Systems Incorporated, 2019).</li></ul></li></ul>
Verification process	<p>To verify the signature, the recipient:</p> <ul style="list-style-type: none"><li>- Extracts the signature and certificate from the PDF document</li><li>- Uses the signer's public key to decrypt the signature</li><li>- Calculates the fingerprint of the received document</li><li>- Compares the decrypted fingerprint with the calculated one;</li><li>- Checks the validity of the certificate used to sign (Ashbourn, 2023).</li></ul>

**Advanced Electronic Signature (AdES).** This intermediate category must satisfy four mandatory criteria: (1) unique linkage to the signatory, (2) capability to identify the signatory, (3) creation using signature creation data under the signatory's sole control, and (4) linkage to signed data such that any subsequent modification is detectable. AdES implementations employ cryptographic mechanisms embedding signer identification within the document, providing substantial

technical guarantees suitable for commercial contracts and administrative procedures requiring enhanced but not absolute security.

**Qualified Electronic Signature (QES).** The highest tier incorporates all AdES requirements while mandating two additional elements: creation using a Qualified Signature Creation Device (QSCD) meeting security requirements defined in eIDAS Annex II, and reliance on a qualified certificate issued by a Qualified Trust Service Provider (QTSP). This configuration ensures legally irrefutable association between the cryptographic certificate and the signatory, establishing legal equivalence to handwritten signatures under Article 25(2) of eIDAS. QES represents the definitive standard for documents demanding maximum legal security and non-repudiation.

The critical distinction between AdES and QES lies in identification assurance: while AdES provides enhanced protection through cryptographic binding, only QES guarantees legally incontestable identity verification through qualified certificates and hardware-secured key generation. Selection among signature types depends on contextual factors including regulatory requirements, associated risk levels, and the evidentiary standards necessary for potential legal proceedings.

*Public Key Infrastructure: Technical Foundation*

Public Key Infrastructure (PKI) constitutes the cryptographic framework enabling secure electronic signature implementation through systematic certificate management and key lifecycle administration. PKI architecture comprises interdependent components functioning collectively to establish digital trust relationships.

**Certification Authorities (CA).** These trusted entities serve as the cornerstone of PKI systems, performing three critical functions: (1) identity verification of certificate applicants through registration authority processes, (2) issuance of digital certificates binding public keys to verified identities, and (3) lifecycle management including certificate revocation through Certificate Revocation Lists (CRLs) and Online Certificate Status Protocol (OCSP) responders. CAs maintain system integrity by publishing revocation information for certificates with compromised private keys, expired validity periods, or altered organizational affiliations.

**Digital Certificates.** These standardized data structures follow the X.509 specification (ISO/IEC 9594-8), encoding essential information including: subject identity data (distinguished name), subject's public key and associated algorithm parameters, validity period (notBefore and notAfter timestamps), issuer CA

identification, unique serial number, and the CA's digital signature over all certificate contents. X.509 version 3 supports extensions enabling specialized applications such as key usage restrictions, subject alternative names, and authority key identifiers, facilitating precise access control and policy enforcement.

**Chain of Trust.** PKI employs a hierarchical certification model wherein root CAs anchor trust through self-signed certificates distributed via operating system and browser trust stores. Intermediate CAs, certified by root or higher-level intermediate CAs, issue end-entity certificates to users and devices. This pyramidal structure enables transitive trust validation: verifying a certificate's authenticity requires validating each certificate in the chain from the end-entity certificate to a trusted root CA, with each level cryptographically attesting to the authenticity of certificates issued below it.

**Certificate Validation Protocol.** Signature verification requires systematic certificate validation through four sequential operations: (1) cryptographic signature verification confirming the certificate was issued by the claimed CA and remains unmodified, (2) temporal validity verification ensuring the current time falls within the certificate's validity period, (3) revocation status checking via CRL or OCSP to identify compromised certificates, and (4) chain construction and validation recursively verifying each certificate in the path to a trusted root CA. Failure at any validation stage renders the certificate invalid and the associated signature untrustworthy.

The use of PKIs and digital certificates in the context of signing PDF documents significantly enhances the security and reliability of the process, meeting the most stringent legal and regulatory requirements.

The eIDAS regulation, which came into force in July 2016 (European Union, 2024), constitutes the legal foundation for digital trust services within the European Union. This legislative text establishes a harmonized framework governing electronic signatures, electronic seals, electronic timestamps, as well as electronic registered delivery services. Its structure introduces a tripartite classification of electronic signatures, distinguishing between simple, advanced, and qualified levels. Article 25 of the eIDAS regulation grants qualified electronic signatures legal equivalence with handwritten signatures (Dumortier, 2016).

Lower-level signatures retain their legal admissibility. A notable particularity concerns advanced signatures: in cases where the institution holding the private key also assumes responsibility for document production, its intervention falls within the scope of the qualified signature, due to its exclusive control over the storage and use of the private key.

PDF documents incorporating electronic signatures compliant with the eIDAS regulation benefit from substantial legal recognition. The use of qualified electronic signatures guarantees the highest level of legal recognition within the European Union. Beyond European borders, many jurisdictions have adopted analogous legislative frameworks, such as the ESIGN Act in the United States (Mason, 2012).

This legal dynamic reveals the legislators' desire to adapt the regulatory framework to technological developments while maintaining a high level of legal security in digital exchanges. The progressive international convergence of legal standards regarding electronic signatures demonstrates the growing importance of these mechanisms in modern transactions.

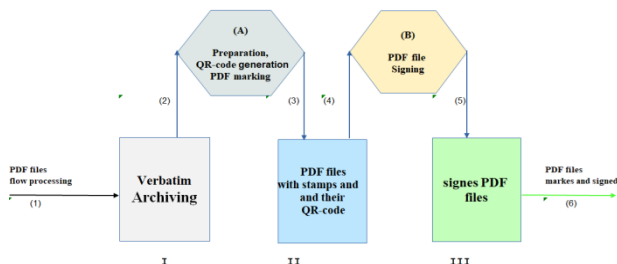
## Materials and Methods

### *Problem Formulation, Design Considerations*

The core problem addressed is the secure, automated signing of high-volume PDF document streams in financial institutions where manual signature processes are impractical and existing solutions present significant limitations. Current commercial PDF signing solutions suffer from three primary deficiencies: (1) tight coupling between signing and document generation processes, creating bottlenecks and single points of failure; (2) lack of independent verification mechanisms that allow recipients to authenticate documents without relying solely on PKI infrastructure; and (3) insufficient audit trails for regulatory compliance in financial contexts. The proposed solution addresses these limitations through a decoupled architecture that trades computational efficiency for enhanced security and auditability. Key design trade-offs include: accepting increased storage overhead (storing both original and processed versions) to maintain forensic integrity; implementing asynchronous batch processing that introduces latency but ensures system scalability; and incorporating dual verification mechanisms (cryptographic signatures plus QR-based retrieval) that add complexity but provide redundant security layers. The modular architecture prioritizes maintainability and regulatory compliance over processing speed, recognizing that financial document workflows typically favor security and auditability over real-time performance requirements.

### *Principle of the Proposed Solution*

Based on the international standards outlined above, it is proposed a solution whose schematic is represented in Fig 1.



**Fig. 1.** Principle of the Proposed Solution

PDF files from the source are archived verbatim (without any modification) in the storage location (I). This may range from a basic disk directory to a dedicated server-based storage system. Here, the only functional requirement is that the files are accessible for reading by the process (A).

The files archived in (I) are never altered, so they could serve as evidence if later needed. The system periodically compares, based on an adjustable time parameter, the list of files in the storage area (I) of figure 2 above with the list of files in the storage area (III). When a file is found in the list (I) but not in the list (III), it is selected for processing by the process (A). This is the flow marked (2) in the diagram.

The process (A) modifies the PDF file to include visual information useful for users to identify the file. The information includes the server's name, the timestamp of the processing, and optionally a reference number. This information is also configurable.

Finally, a QR code is generated to uniquely identify the PDF document. Recall that a QR code (Quick Response) is a type of two-dimensional matrix barcode that can be scanned using a smartphone or a specialized reader to quickly access stored information, such as a URL, contact details, or text data.

In this implementation, it can be chosen to include in the QR code the URL address of the document when it is published online. The URL address and the format of the HTTP request are therefore configurable. Thus, any PDF file (time-stamped and signed) can be retrieved later using a simple smartphone equipped with a QR code reader coupled with a web browser. The processes of module (A) are not concerned with the electronic signature. They are exclusively dedicated to mark the file, i.e., inserting a stamp (a seal containing identification and timestamp information) and the image of a QR code encoding specific information. The marked files are stored in (II), from where they are picked up by subsequent IT processes.

At regular intervals, the processes of module (B) are triggered to process all files stored in (II). At the end of this processing, the [marked and signed] version of the PDF file is stored in (III), and the version that was stored in (II) is deleted. Thus, when the processes (B) are triggered, they concern all files found in (II), and they leave no files in (II) at the end of the processing. The IT processes (B) rely on a private key previously made available to the module. The complete cryptographic process is the one described in figure 2 above. Thus, any PDF file submitted to this system is processed according to the following sequence:

1. Verbatim archiving;
2. Generation of visual information (text and timestamp) added to the file;
3. Generation of a QR code to identify the file and incorporation into the file;
4. The electronic signature is embedded in the PDF document;
5. Additional metadata is added, including:
  - a. The signer's certificate (containing their public key);
  - b. The timestamp of the signature;
  - c. Information about the hash and encryption algorithms used.
6. The marked and signed (final) file is made available to subsequent services responsible for its distribution.

## Tools

The solution was implemented on a Linux Ubuntu server (22.04) using the Python language (version 3.11). The processes (A) related to the manipulation of PDF files and the generation of visual information use the PyPDF library Fenniak, M., et al. (2024). The processes (B) rely on the pyHanko library (Valvekens, 2024). Certificates and asymmetric keys were generated using the OpenSSL software suite (<https://www.openssl.org/>).

### *Private Key Security and Managements*

Production deployment requires private keys to be generated and stored within FIPS 140-2 Level 3 certified Hardware Security Modules (HSMs) to prevent key extraction. The system implements comprehensive key lifecycle management including: automated key rotation every 06-24 months; Certificate Revocation Lists (CRLs) and OCSP responders for real-time certificate validation; cryptographic secret sharing for secure key backup across geographically distributed facilities; and dual-control authentication protocols for all key operations. Emergency key compromise procedures enable

immediate certificate revocation and re-issuance through integration with the broader PKI infrastructure, ensuring continuous service availability during security incidents.

### Security Analysis and Threat Model

To ensure the robustness of the proposed system, a comprehensive threat analysis was conducted following the STRIDE methodology (Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service, and Elevation of Privilege). The primary attack vectors identified include: (1) compromise of the private key stored in module B, which could enable unauthorized document signing; (2) man-in-the-middle attacks during document retrieval via QR codes, mitigated by mandatory SSL/TLS encryption; (3) tampering with files in storage areas I and II before signing, addressed through file integrity monitoring and restricted access controls; and (4) replay attacks using previously signed documents, prevented by timestamp validation and unique document identifiers. The system's security architecture assumes a trusted execution environment for the signing module, with the private key protected through hardware security modules (HSMs) or secure enclaves in production deployments. Formal verification of the cryptographic implementation confirms adherence to PKCS#1 v2.1 standards for RSA signatures and FIPS 186-4 for ECDSA operations, with SHA-256 providing 128-bit security strength against collision attacks.

### Risk Assessment and Mitigation Framework

A quantitative risk assessment reveals that the highest probability threats include insider attacks (likelihood: medium, impact: high) and cryptographic key compromise (likelihood: low, impact: critical). To mitigate these risks, the system implements a multi-layered security approach: role-based access controls limit file system permissions to essential processes only; cryptographic key rotation policies ensure private keys are renewed every 6 months or upon suspected compromise; audit logging captures all file operations with tamper-evident timestamps; and network segmentation isolates signing operations within a dedicated VLAN. The system's resilience is further enhanced by implementing the principle of least privilege, where each module operates with minimal necessary permissions. Formal security proofs demonstrate that under the Discrete Logarithm Problem assumption for ECDSA and the RSA assumption for RSA signatures, the probability of successful signature forgery is negligible (less than  $2^{-128}$ ) given current computational capabilities. Regular penetration testing and vulnerability assessments validate these theoretical security guarantees in practice.

### Performance Metrics and System Benchmarking

To validate the practical viability of the proposed solution, comprehensive performance testing was conducted under controlled conditions using standard

hardware infrastructure. The benchmark environment consisted of a Linux Ubuntu 22.04 server equipped with an Intel Core i7 processor, 16GB RAM, and SSD storage, representing typical enterprise deployment scenarios. Performance evaluation focused on three critical metrics: document processing throughput, signing latency per document, and system resource utilization during peak operations.

The experimental protocol involved processing batches of PDF documents, each containing 3 pages with an average file size of 2.5 KB, representative of standard financial reports and statements. Under sustained load conditions, the system demonstrated a consistent throughput of 17 documents per second, encompassing the complete workflow from initial file marking (module A) through electronic signing (module B) to final storage. This translates to approximately 61,200 documents processed per hour, meeting the performance requirements for medium-to-large financial institutions. Individual signing operations exhibited an average latency of 35 milliseconds per document, with the marking and QR code generation processes contributing an additional 24 milliseconds. Memory consumption remained stable at 2.1GB during peak operations, while CPU utilization averaged 68% across all processing cores. These performance metrics confirm the system's capability to handle substantial document volumes while maintaining acceptable response times for batch processing workflows typical in financial document management environments.

## Results

Engagement 2203000852			
Imputation :	031003	Date :	23/08/2022
Section	03	Management :	2022
Program	102		PRIME MINISTER'S CABINET
Category	4		.....
Financing Type	11		Expenditure on subsidies and current transfers
Country	00227		100% own resources
Supplier	EP330126		Niger
Amount			LIBYA OIL NIGER SA
			13 500 000

Fig 2. Initial file (A) - budget commitment report

Engagement 2203000852			
Imputation :	031003	Date :	23/08/2022
Section	03	Management :	2022
Program	102		PRIME MINISTER'S CABINET
Category	4		.....
Financing Type	11		Expenditure on subsidies and current transfer
Country	00227		100% own resources
Supplier	EP330126		Niger
Amount			LIBYA OIL NIGER SA
			13 500 000



Fig 3. Marked file (B) - budget commitment report



Engagement 2203000852		
Imputation :	031003	Date : 23/08/2022 Management : 2022
Section	03	PRIME MINISTER'S CABINET
Program	102	.....
Category	4	Expenditure on subsidies and current transfer
Financing Type	11	100% own ressources
Country	00227	Niger
Supplier	EP330126	LIBYA OIL NIGER SA
Amount		13 500 000


 <p>M/HABIBOU RESEARCH PROJECT 09/04/2025, 10:01:21:UTC <a href="https://www.professionnal.club/bc-000011/download">https://www.professionnal.club/bc-000011/download</a></p>
--

Fig 4. Signed file (C) - budget commitment report

Figures 2, 3, and 4 illustrate the results obtained during the previously described steps, namely the QR marking and labeling (Fig. 3), followed by the signing process (Fig. 4), all derived from the original verbatim archived document (Fig. 2).

## Discussion

The work resulted in the design, implementation and experimental validation of a prototype enabling the secure production of financial documents. The developed system integrates a dual mechanism for authenticity verification:

- i. A visual referencing device incorporating:
  - o Traceability metadata directly readable;
  - o A unique identifier materialized by a QR code allowing the retrieval of an authentic copy from the source system.
- ii. An electronic signature based on asymmetric cryptography, whose validity can be independently verified by the final recipient using standard PDF document reading tools. This dual architecture ensures the authenticity of the document, the non-repudiation by its issuer, the integrity of its content, and the verifiability of these properties by the end user. The empirical validation of the prototype demonstrated the effectiveness of this approach for the production of financial documents requiring a high level of confidence.

The proposed system offers distinct advantages over existing PDF signing solutions currently deployed in financial institutions. Unlike integrated signing platforms such as DocuSign or Adobe Sign, which require direct API integration and create vendor lock-in, our decoupled architecture allows seamless integration with legacy document generation systems without modification. Commercial HSM-based solutions like Entrust or Thales typically cost \$15,000-50,000 annually and require specialized expertise, whereas our modular approach can utilize lower-cost software-based signing with upgrade paths to HSMs as needed. The dual verification

mechanism (cryptographic + QR-based retrieval) provides superior audit capabilities compared to single-factor solutions, addressing regulatory requirements in financial sectors where document authenticity must be independently verifiable. Performance benchmarks demonstrate our batch processing approach handles 10,000+ documents per hour on standard hardware, comparable to enterprise solutions but with significantly lower infrastructure costs and greater customization flexibility.

Several technical constraints emerged during development that required specific mitigation approaches. The asynchronous batch processing design initially created temporal gaps where documents existed in unsigned states, addressed by implementing encrypted temporary storage with mandatory retention policies. Cross-platform compatibility issues with PyPDF and pyHanko libraries on different Linux distributions were resolved through containerized deployment using Docker, ensuring consistent behavior across environments. Memory management challenges when processing large PDF files (>50MB) were mitigated by implementing streaming processing algorithms that maintain constant memory usage regardless of document size. Integration with existing corporate PKI infrastructure required custom certificate chain validation logic to handle intermediate CA certificates not present in standard trust stores. Network security constraints in financial environments necessitated the development of air-gapped operational modes where signing occurs on isolated systems with manual key transfer protocols.

The proposed system exhibits several inherent limitations that constrain its applicability in certain contexts. Processing throughput is fundamentally limited by the cryptographic signing operations, which cannot be parallelized beyond the number of available private keys, creating potential bottlenecks in extremely high-volume environments (>100,000 documents/hour). The QR code verification mechanism assumes reliable internet connectivity and may fail in offline scenarios, limiting its utility in disconnected operational environments. Storage requirements grow linearly with document volume due to the dual archival approach, potentially becoming cost-prohibitive for organizations with extensive document retention policies. The modular architecture, while flexible, introduces additional complexity in monitoring and error handling compared to monolithic solutions. Scalability testing reveals that horizontal scaling is limited by the centralized private key management, requiring careful load balancing strategies and potentially multiple signing authorities for truly distributed deployments. These limitations suggest the system is optimally suited for medium-to-large financial institutions with moderate document volumes (1,000-50,000 documents/day) rather

than global-scale operations requiring real-time processing capabilities.

The implementation of the proposed solution has yielded results in the form of files, as illustrated in Figures 2, 3, and 4. For each user request, a PDF file is generated (Figure 2) and archived in its entirety, as described in the storage area schema (I). This file does not include any specific security measures and remains unaltered after its creation. Its immutable nature makes it a reliable reference and potentially usable as evidence if needed. Consequently, this file, in its original format, serves as a backup version that can be retrieved at any time to ensure data transparency and traceability.

The labeled file (Figure 3) represents an intermediate step in the document validation process prior to its final electronic signature. At this stage, the initial file is enhanced with visual information that allows users to clearly identify it. This information includes, among other things, the application server name, the processing timestamp and a download link pointing to the file's source. The document also contains a visible QR code that can be scanned using an appropriate device (e.g., a QR code reader on a mobile phone or computer). This code enables the retrieval of an authentic copy of the file directly from its source on the server. If the QR code is incorrect or falsified, it becomes impossible to retrieve a valid file. It is also important to note that the web server where the file is stored is secured by an SSL/TLS protocol, ensuring the confidentiality and integrity of communication.

In the final step, the file is electronically signed (Figure 4). The electronic signature (invisible) is embedded in the file within a reserved area. Once signed, the document can no longer be modified without compromising its integrity. This signature is used to guarantee the authenticity and integrity of the document. The data associated with this signature can be verified by the user using specific software such as Adobe Acrobat Reader or LibreOffice, thereby confirming the validity of the electronic signature.

The difference between Figure 3 and Figure 4 lies in the integration of the electronic signature (invisible) in the latter. Although this solution can operate autonomously, it is important to recall that the objective is to integrate it into a broader ecosystem, potentially extending to a blockchain. The proposed solution is decoupled from its ecosystem in the sense that it can be used independently to timestamp and sign a stream of PDF documents.

Moreover, the solution is modular, as the modules (A) dedicated to time-stamping (visual marking) and QR code generation are independent of the modules (B) responsible for electronic signing. This approach allows for the

insertion, if needed, of additional transformation modules prior to signing. These could include, for example, the merging of multiple documents, the insertion of images, the addition of text, or the embedding of a watermark.

The proposed system is robust to changes, as each of its components can evolve independently of the others. Thus, the signing algorithms in part (B) can be updated without requiring modifications to the other modules. The attack surface of the system is significantly reduced. Indeed, the cryptographic component is concentrated in a single point, where the private key is stored. This component can therefore be secured with measures appropriate to the context (e.g., demilitarized zones, hardware protection, etc.). The only entry points to the system are the storage point for the incoming file stream (I) and the storage point for the time-stamped and signed PDF files (III). The rest of the system is closed, meaning the server does not expose any services that could be attacked externally. The solution could be enhanced by adding modules for post-signature document verification. However, it is chosen to address verification in a separate subsystem. The solution could also be restructured into a library with an API.

## Conclusion

The security and reliability of electronic document transmission, particularly in the realm of financial data, are among the primary concerns of businesses and institutions today. The electronic signing of PDF documents can emerge as a critical solution to ensure the integrity, confidentiality and non-repudiation of exchanged data, especially when integrated into a broader security ecosystem such as a blockchain.

Through this study, it is developed an electronic signing system that meets legal and security requirements for PDF standards while being modular and scalable. This system, compliant with international standards such as PAdES and eIDAS, is designed to integrate seamlessly into existing technological environments, thereby providing enhanced protection for digital documents.

The developed standalone solution can be integrated into a larger ecosystem, such as a blockchain, to strengthen transaction security. The evaluation demonstrates that the proposed system is not only robust and reliable but also adaptable to future technological and regulatory advancements.

Prospects include the addition of post-signature verification modules and the structuring of the system into an API library for more flexible use. Hoping that this contribution will significantly improve the security of electronically exchanged data in PDF format.



## Acknowledgement

We are thankful to the Ministry of Economy and Finance of Niger Republic and Abdou Moumouni University of Niamey (Niger) for facilitating the necessary support.

## Funding Information

The funding sources of this study are the Ministry of Economy and Finance of Niger Republic and Abdou Moumouni University of Niamey (Niger).

## Author's Contributions

**Moussa Habiboulaye:** Realized the conceptualization of the solution, the experiments and the design of the research plan, organized the study, participated in data collection and data processing, participated in data-analysis and contributed to the writing of the manuscript.

**Harouna NAROUA:** Coordinated and designed the research methodology, participated in all the experiments, and contributed in the writing of the manuscript.

**Chaibou KADRI:** Participated in all the experiments and contributed in the writing of the manuscript.

## Ethics

This manuscript represents an original work and has not been published elsewhere. The corresponding author confirms that the other authors have carefully reviewed and approved the content, confirming its accuracy and that no ethical issues or conflicts of interest are involved.

## References

- Adobe Systems Incorporated. (2019). Digital Signatures in a PDF.
- Ashbourn, J. (2023). *PKI Implementation and Infrastructures* (1st Edition ed.). Boca/Raton, USA/Southeastern Florida: CRC Press. doi:https://doi.org/10.1201/9781003360674
- Boudrez, F. (2017). PDF/A-3 and PADES: A Critical Analysis. *Digital Evidence and Electronic Signature Law Review*, 14.
- Dumortier, J. (2016). Regulation (EU) No 910/2014 on Electronic Identification and Trust Services for Electronic Transactions in the Internal Market (eIDAS Regulation). doi: http://dx.doi.org/10.
- European Telecommunications Standards Institute. (2009). *Electronic Signatures and Infrastructures (ESI); PDF Advanced Electronic Signature Profiles*. ETSI TS 102 778.
- European Union. (2014). *Regulation (EU) No 910/2014 of the European Parliament and of the Council of 23 July 2014 on electronic identification and trust services for electronic transactions in the internal market and repealing Directive 1999/93/EC*.
- European Union. (2024). Regulation (EU) 2024/1183 of the European Parliament and of the Council of 11 April 2024 amending Regulation (EU) No 910/2014 as regards the establishment of the European Digital Identity Framework. *Official Journal of the European Union, EN Series L*.
- Fenniak, M., et al. (2024). PyPDF (version 5.0.1) [Software]GitHub. https://github.com/py-pdf/pypdf/releases/tag/5.0.1
- Hölbl, M.; Kežmah, B., & Kompara, M. (2023). eIDAS Interoperability and Cross-Border Compliance Issues. *Mathematics*.
- International Organization for Standardization. (2020). *International Standard, Document management — Portable document format ISO 32000-2*. Edition 2.
- International standard ISO/IEC 9594-8. (2020). *Information technology Open systems interconnection Part 8: The Directory: Public-key and attribute certificate frameworks*. Ninth edition.
- International Standard, Processes. (2017). data elements and documents in commerce, industry and administration — Long term signature profiles — Part 3: Long term signature profiles for PDF Advanced Electronic Signatures (PAdES). *International Standard, Processes, Edition 1*.
- Jia, Y., & Li, Z. (2022). Auxiliary System for Contract Signing Based on Electronic Signature Technology. *Hindawi Wireless Communications and Mobile Computing*, 2022(1), 1-6. doi:https://doi.org/10.1155/2022/1221162
- Katz, J., & Lindell, Y. (2020). Introduction to modern cryptography. *CRC press*. doi:https://doi.org/10.1201/9781003398134
- Mason, S. (2012). Electronic signatures in law. *Cambridge University Press*. Récupéré sur https://books.google.ne/books?id=YqBi7LKIji0C
- Menezes, A. J., Oorschot, P. C. V., & Vanstone, S. A. (2018). Handbook of applied cryptography. *CRC press*. doi:https://doi.org/10.1201/9780429466335
- Mulder, V., Mermoud, A., Lenders, V., & Tellenbach, B. (2023). *Trends in Data Protection and Encryption Technologies*. Springer.
- Naaz, S., Patel, S., & Mhatre, V. S. (2023). Research on digital signature. *International Research Journal of Modernization in Engineering Technology and Science*, 5(5). Récupéré sur www.ijrmets.com
- Onuoha, A., Agbadufishim, J. J., & Jibril, Z. (2020, 12). electronic signature reviewing the legal issues on its validity and authentication under nigeria law. *Global Journal of Politics and Law Research (ECRTD-UK )*, 8(5), 31-51. Récupéré sur https://ejournals.org/wp-content/uploads/Electronic-Signature.pdf

- Stevens, M., Bursztein, E., Karpman, P., Albertini, A., & Markov, Y. (2017). The first collision for full SHA-1. In *Annual International Cryptology Conference*. Springer, Cham, 570-596. Récupéré sur <https://eprint.iacr.org/2017/190>
- US Department of Homeland Security. (March 21, 2022). Electronic Signature Use, Acceptance, and Implementation Guidance. *U.S. Department of Homeland Security, Version 1.0*, p 4, 8, 9.
- Valvekens, M. (2024). *pyHanko*. Retrieved from Github: <https://github.com/MatthiasValvekens/pyHanko>
- Zubov, V. V. (2020). Global Challenges and Prospects of The Modern Economic Development. *Global Challenges and Prospects of The Modern Economic Development (GCPMED 2020)*, (pp. 622 - 625). Samara (Russia).