iDSign: A Secure Lightweight NFC-Based Framework to Tap and Digitally Sign Documents

M.Tech Thesis

Submitted in partial fulfillment of
the requirements for the award of the degree
of
Master of Technology
in
Department of Software Engineering
submitted by

Himanshu Sharma (23/SWE/18)
under the guidance of
Dr. Divyashikha Sethia
Associate Professor
Department of Software Engineering

DEPARTMENT OF SOFTWARE ENGINEERING
DELHI TECHNOLOGICAL UNIVERSITY, DELHI
May 2025

DEPARTMENT OF SOFTWARE ENGINEERING
DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Bawana Road, Delhi-110042

CERTIFICATE

This is to certify that M.Tech Thesis entitled **iDSign: A Secure Lightweight NFC-Based Framework to Tap and Digitally Sign Documents** which is submitted by Himanshu Sharma, Roll No - 23/SWE/18, Department of Software Engineering, Delhi Technological University, Delhi in partial fulfilment of the requirement for the award of degree Master Of Technology (Software Engineering) is a record of the candidate work carried out by him under my supervision. To the best of my knowledge this work has not been submitted in part or full for any Degree or Diploma to this University or elsewhere.

Date: 19 May 2025

Place: New Delhi

Divya.

Dr. Divyashikha Sethia
Associate Professor
Department of Software Engineering
Delhi Technological University

DEPARTMENT OF SOFTWARE ENGINEERING
DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Bawana Road, Delhi-110042

CANDIDATE'S DECLARATION

I Himanshu Sharma (23/SWE/18) hereby declare that the work which is being presented in the thesis entitled **iDSign: A Secure Lightweight NFC-Based Framework to Tap and Digitally Sign Documents** in partial fulfilment of the requirements for the award of the degree of Master Of Technology submitted in the Department of Software Engineering, Delhi Technological University is a bonafide record of my own work carried out during the period from August 2023 to June 2025 under the supervision of Dr. Divyashikha Sethia.

The material contained in the thesis has not been submitted by me for the award of any other degree of this or any other institute.

Date: 19 June 2025

Place: New Delhi

Candidate's Signature

This is to certify that the student has incorporated all the corrections suggested by the examiners in the thesis and the statement made by the candidate is correct to the best of our knowledge.

May 21 2025

Signature of Supervisor

24.6.2025

Signature of External Examiner

DEPARTMENT OF SOFTWARE ENGINEERING
DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Bawana Road, Delhi-110042

ACKNOWLEDGEMENT

I am very thankful to **Dr Divyashikha Sethia** (Associate Professor, Department of Software Engineering) and all faculty members of the Software Engineering Department of Delhi Technological University. They all provided me with immense support and guidance for the project.

I would also like to express my gratitude to the University for providing me with the laboratories, infrastructure, testing facilities, and environment, which allowed me to work without obstructions.

I would also like to appreciate the support provided to me by our lab assistants, seniors, and our peer group, who aided us with all the knowledge they had regarding various topics.

Himanshu Sharma

(23/SWE/18)

ABSTRACT

The growing dependence on digital systems for secure document exchanges requires efficient, lightweight digital signature solutions. Conventional methods, frequently dependent on Public Key Infrastructure (PKI) and cloud-based servers, introduce considerable challenges. PKI solutions suffer computational overhead, complex certificate administration, and security weaknesses, whereas cloud-based servers are susceptible to threats like data breaches, man-in-the-middle assaults, and unauthorized access. These risks are especially concerning in resource-limited and mobile environments.

Firstly, this work proposes iDSign, a novel secure and lightweight framework for digital document signatures that utilizes Identity-Based Cryptography (IBC) and a modified lightweight Identity-based Transport Layer Security (iTLS) protocol. iDSign mitigates extensive certificate exchanges and reliance on susceptible cloud infrastructures by improving efficiency with tamper-resistant Secure Elements (SEs). The framework is adaptable for various communication interfaces such as NFC, Bluetooth, and Transmission Control Protocol (TCP/IP), guaranteeing safe offline and proximity-based operations.

Secondly, this work also proposes the design and implementation of the iHDoc application based on the iDSign framework for the mobile-based signing of documents. iHDoc leverages NFC-based Host Card Emulation (HCE) for secure two-way communication and Bluetooth for effective post-authentication data transmission. NFC proximity reduces the risk of man-in-the-middle attacks and ensures the locality of reference. The mutual authentication component of the iDSign framework is compared with an Rivest–Shamir–Adleman (RSA)-based configuration to evaluate performance in real-world mobile scenarios. The results indicate that iHDoc based on iDSign significantly surpasses the RSA-based mutual authentication configuration in terms of execution speed, storage efficiency, and compatibility with mobile devices. This thesis rigorously examines iDSign's security attributes, covering its resistance to assaults and forward secrecy while quantitatively evaluating its communication and computing overhead. The results confirm iDSign as a lightweight, secure solution to digital document signing that is appropriate for offline, resource-limited, and mobile contexts.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Overview

Having the option to safely and securely sign documents digitally has become crucial for various professions, including healthcare, finance, and legal services, in an increasingly digital environment. Digital signatures are essential as it replace physically signed paper documents, which can be easily lost or damaged [1]. By guaranteeing electronic transactions' integrity, authenticity, and non-repudiation, it provide legal validity to digital documents.

Digital signatures employ cryptographic techniques like hash functions and asymmetric encryption [4, 5]. Hash functions generate a unique fingerprint of the document. This fingerprint is digitally signed with the Signer's private key. The recipient verifies the hash, utilising the Signer's public key, and reconciles it against a fresh computation of the document hash. Both are the same, which guarantees document integrity and authenticity and Signer identity (non-repudiation). Apart from rendering paper-based signatures equivalent to the law, digital signatures provide additional benefits:

- *Efficiency:* Enables document processes, eliminating physical transit and storage requirements.
- *Security:* Enhances the integrity of sensitive data by reducing the risk of tampering and unauthorized access.
- *Efficiency in costs:* Saves on costs related to paper, printing, and manual processes.
- *Environmental Friendliness:* Reduces the reliance on paper, contributing to environmental campaigns.

Most operation environments increasingly test the reliability of digital signatures for use today. Resource-limited and mobile environments require cryptographic algorithms to be cost-efficient under limited computation capacity, storage, and connectivity. While secure in regulated environments, conventional Public Key Infrastructure (PKI)-based solutions face challenges in dynamic and decentralized scenarios. Consequently, the necessity for versatile, effective, and secure digital signature systems that facilitate proximity-based and offline functionalities is increasingly significant. The extensive utilization of smartphones and integrated secure features provides opportunities for lightweight cryptographic systems that function independently without ongoing server contact.

1.2 Motivation

Traditionally, digital signature implementations often involved physical tokens (like smart cards) or complex software installations, introducing barriers to adoption and ease of use. While providing numerous advantages, most existing solutions are web-based and susceptible to attacks [4,6,7]. Even though blockchains can also be used [8,9], both web and blockchain solutions require some external remote connection. These suffer from various challenges, such as:

- *Hardware and Software Dependence:* Many existing systems require specialist hardware devices (such as smart cards or USB tokens) or the implementation of related software applications. It creates complexity and potential compatibility difficulties and limits the mobility of signing across various devices and regions.
- *Security Vulnerabilities:* Digital signature systems can be subject to cyberattacks such as phishing attacks, malware, and man-in-the-middle attacks, which can compromise the secrecy of credentials and put signatures at risk [4,6,7].
- *Complex Authentication:* Verification of a Signer's identity with large certificates issued by centralized Certificate Authorities (CAs) for authentication purposes can increase complexity in terms of computational expenses, storage overheads and administrative complexity [10].

Recent improvements in proximity-based communication methods, like Near-Field Communication (NFC) with Host Card Emulation (HCE) [2] and Bluetooth, provide novel opportunities for in-person digital interactions independent of internet connectivity. However, secure mutual authentication over these mediums remains challenging due to transmission speed constraints and processing capabilities. Moreover, legacy systems often lack the mechanisms to guarantee security and usability in mobile-first scenarios. A unified framework integrating lightweight cryptographic operations with modern mobile communication interfaces is essential to meet evolving security and usability demands. The incentive is not cryptographic innovations on paper but to provide practical application through effective protocols that are both performance-oriented and attack vector-resistant.

1.3 Problem Statement

There are numerous problems with current digital signature solutions. Most current systems depend on specialist hardware devices or executing related software packages, producing complexity and lack of mobility. Further, current digital signature solutions are web-based and vulnerable to cyber attacks, breaking the privacy of credentials and exposing signatures to threat. Even, constructing a Signer's identity using large certificates for mutual authentication can incur computation costs, storage overheads, and management complexity.

Aside from the security and technical issues, digital signature systems nowadays are not scalable when used in varied environments, like public offices, hospitals, or government facilities. Such places normally demand dependable signing solutions that will function even if the internet connection is slow or broken. Web-based systems would depend heavily on the availability of constant internet and working servers, which may crash or

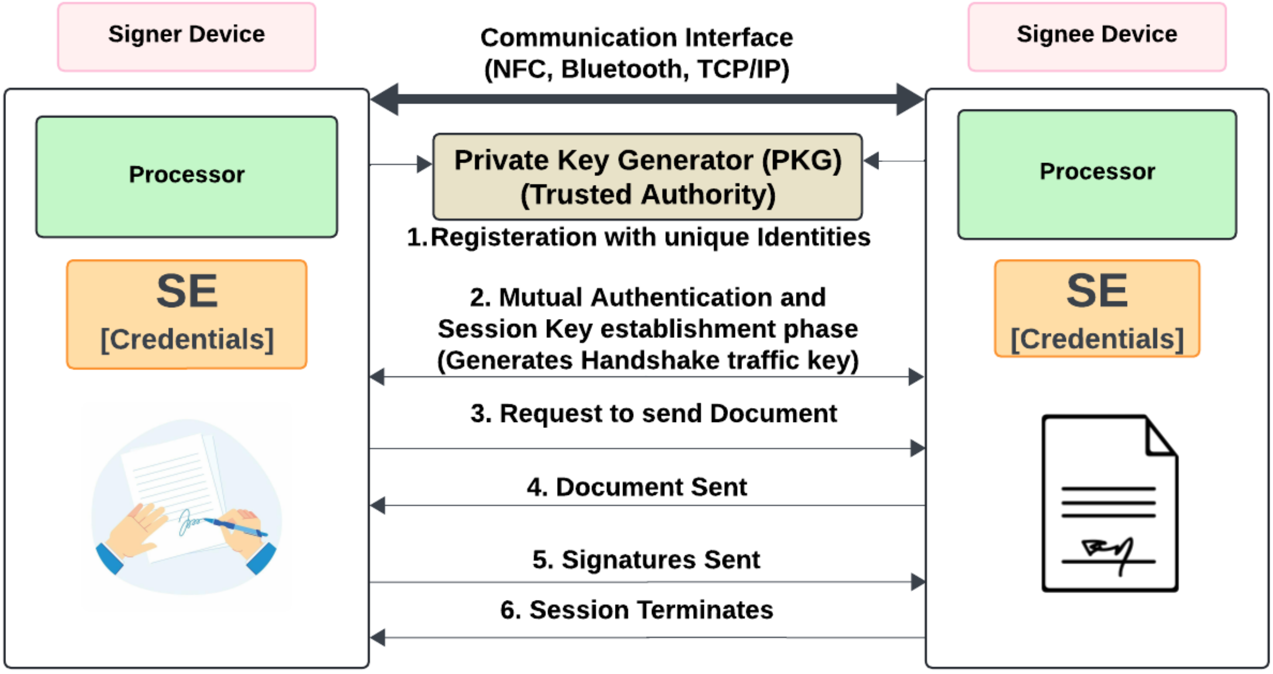

Signer Device
Communication Interface (NFC, Bluetooth, TCP/IP)
Signee Device
Processor
Private Key Generator (PKG) (Trusted Authority)
Processor
1.Registeration with unique Identities
SE [Credentials]
SE [Credentials]
2. Mutual Authentication and Session Key establishment phase (Generates Handshake traffic key)
3. Request to send Document
4. Document Sent
5. Signatures Sent
6. Session Terminates

Figure 1.1: iDSign framework block diagram

become unstable at the most inconvenient time. In addition, most conventional digital signature schemes are cumbersome to operate on contemporary mobile platforms using technologies such as HCE [2] or secure storage elements. These limitations indicate that a more scalable and secure architecture is necessary that can support simple digital signing in low-resource and high-mobility scenarios without compromising key security aspects or multistandard interoperability.

1.4 Proposed Solution

Therefore, there must be digital signature solutions prioritizing robust user authentication, confidentiality, easy cross-platform use, and physical tamper resistance. Design methodologies that rid complexity, simplify authentication, and have an easy-to-use interface are critical. The thesis presents a new iDSign framework overcoming these limitations with a light-weight and secure Identity-Based Cryptography (IBC) based framework [11] and a light-weight identity-based Transport Layer Security (iTLS) Protocol [10] for secure mutual authentication. Figure 1.1 depicts the flow diagram of iDSign. It does not rely on conventional PKI but incorporates IBC, thereby reducing the complexity of certificate management and ensuring robust security.

In order to ensure the applicability of iDSign in everyday mobile environments, this paper also introduces the design and implementation of a prototype application named iHDoc on a mobile-based platform, which combines low-energy NFC-based HCE [2] with the iDSign approach. NFC mobile phones are a perfect choice for face-to-face signing. NFC HCE mode enables a mobile phone to act as a smart card, avoiding extra hardware and being more flexible. iHDoc enables tap-based secure low-energy NFC communication for digital document signing over HCE on mobile devices, and Bluetooth is utilized for low-data-volume post-authentication data transfer. The suggested iDSign flexible solution is a robust one with the capability of significantly enhancing digital signing of documents in both in-person and remote environments.

1.5 Main Contributions of the Thesis

The key contributions of the thesis are listed below:

1. **Proposal of a novel iDSign (Identity-based Digital Document Signatures) Framework**

   a) Introduces *iDSign*, a secure and lightweight digital signature framework for resource-constrained and offline contexts, utilizing IBC [11] and the Hess Identity-Based Signature (Hess-IBS) scheme [12] to eliminate the need for certificate management.

   b) Detailed security and performance evaluation of iDSign through the iHDoc prototype.

   c) Comparative analysis of iDSign with existing digital signature solutions.

2. **Design and implementation of the Proposed iHDoc Mobile Application**

   a) Presents *iHDoc*, a proximity-based Android application based on iDSign, designed to implement and validate the iDSign framework using NFC HCE [2] and Bluetooth.

   b) Detailed performance evaluation of iHDoc mobile application.

1.6 Thesis Layout

The remaining thesis organization is as follows: Chapter 2 outlines the preliminaries. Chapter 3 contains an overview of related work. Chapter 4 presents the proposed iDSign framework. Chapter 5 presents the prototype implementation of the iDSign framework, iHDoc. Chapter 6 discusses the security analysis of the proposed protocol and the performance evaluation of both iDSign and iHDoc. In the end, Chapter 7 closes the work, summarizing key findings and outlining potential directions for further research.

Chapter 2

Technical Background

2.1 Near Field Communication (NFC)

Near Field Communication (NFC) is a short-range wireless technology that enables communication between electronic devices over a short distance (limited to less than 10 centimetres) and operates at a frequency 13.56 MHz. A simple wave or touch can establish a connection between two NFC-enabled devices. Based on radio-frequency identification (RFID), NFC enables seamless and secure data exchange between devices. NFC provides a versatile platform for numerous applications, including contactless payments, access control, data transfer, and device pairing.

The technology offers three primary modes of operation, as shown in Figure 2.1:

1. *Reader/Writer Mode:* An active NFC device (e.g., a smartphone) reads data from or writes data to passive NFC tags embedded in items. This mode enables information retrieval, product authentication, and marketing campaigns.

2. *Peer-to-Peer Mode:* Two NFC-enabled devices initiate direct communication and exchange data, such as contacts, photos, or small files, when in proximity. This mode facilitates quick and convenient data transfer without external infrastructure.

3. *Card Emulation Mode:* An NFC-enabled device emulates a contactless smart card, enabling it to interact with existing contactless card readers for applications like mobile payments [13] and access control [14].

NFC technology offers several advantages:

- *Simplicity:* NFC interactions are often initiated with a simple tap or touch.

- *Security:* NFC incorporates encryption and authentication mechanisms to protect sensitive data.

- *Versatility:* NFC supports many applications.

- *Low Power Consumption:* NFC is suitable for battery-powered devices.

- *Standardization :* NFC is standardized, ensuring device interoperability.

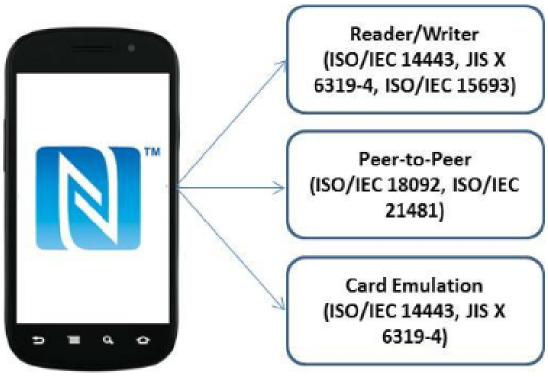

Reader/Writer
(ISO/IEC 14443, JIS X 6319-4, ISO/IEC 15693)
Peer-to-Peer
(ISO/IEC 18092, ISO/IEC 21481)
Card Emulation
(ISO/IEC 14443, JIS X 6319-4)

Figure 2.1: NFC operating modes [2]

2.2 Understanding NFC Host Card Emulation (HCE)

Contactless smart cards, such as those utilized for financial transactions or entry authorization, traditionally depend on embedded Secure Elements (SE) to securely store sensitive data and manage cryptographic processes, as shown in Figure 2.2. However, HCE substitutes the physical secure element with software-based emulation, allowing NFC-enabled devices to carry out transactions and interactions previously limited to physical cards. HCE is a technology that enables smartphones and other devices to act like real contactless smart cards [2]. HCE uses software to mimic smart card communication instead of specialized integrated hardware. HCE implements robust security mechanisms to safeguard confidential information when transmitted wirelessly. This technology is becoming increasingly common for mobile payments and other applications requiring tap-to-interact capability.

Application Protocol Data Units (APDUs) are the fundamental building blocks of communication in the HCE ecosystem [2]. These standardized packets of data, adhering to ISO/IEC 7816-4, are exchanged between the NFC reader and the HCE device. APDUs comprise a header, command data, and optional response data. The HCE service on the device receives command APDUs from the reader, processes the commands, and responds with appropriate response APDUs containing the requested information. HCE's robust security mechanisms ensure data confidentiality and integrity, making it a reliable solution for mobile devices to securely share signing information in the context of iHDoc mobile application, enabling quick and safe signing of documents with just a tap between devices.

2.2.1 ISO Standards for HCE APDU Commands

The ISO/IEC 7816 standard [15] governs the communication between NFC readers and HCE-enabled devices using APDUs. Developers originally designed this standard for physical smart cards but later extended it for HCE use cases. Below are key aspects of the ISO standards relevant to HCE communication:

- **ISO/IEC 7816-4: Organization, Security, and Commands for Interchange [15]**
  - This part of the standard defines the structure of APDUs, including command and response formats.
  - A command APDU typically consists of:

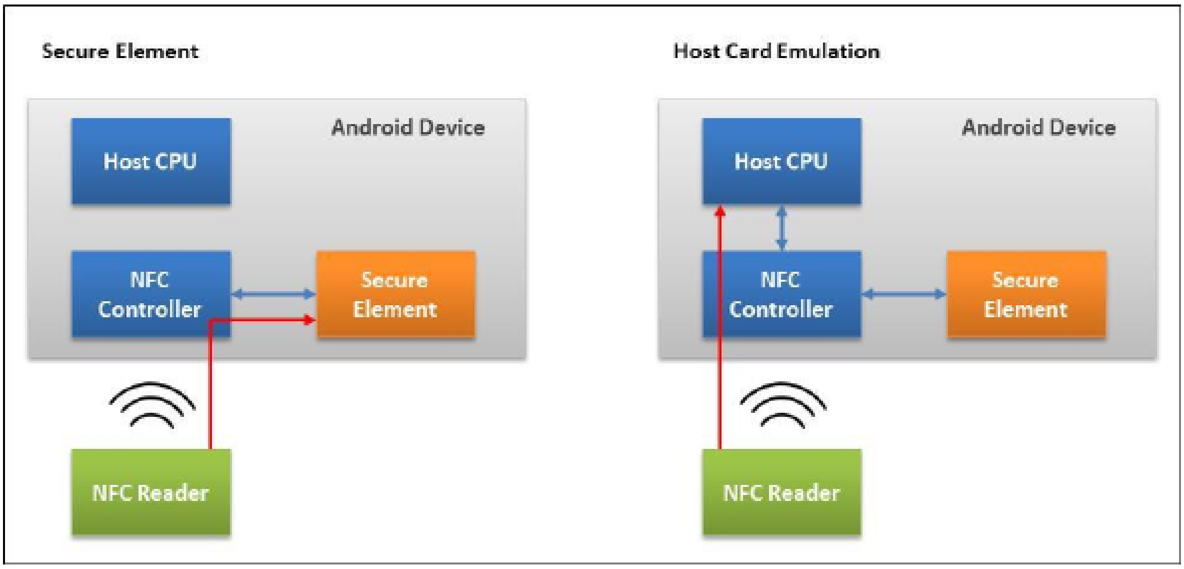


Figure 2.2: NFC communications with the SE (left) or with the host CPU using HCE (right) [2]

    - *CLA (Class Byte):* Identifies the type of command and security level.
    - *INS (Instruction Byte):* Specifies the operation to be performed (e.g., SELECT, READ, WRITE).
    - *P1, P2 (Parameter Bytes):* Provide additional parameters for the command.
    - *Lc (Length of Command Data):* Indicates the length of the data field (if present).
    - *Data Field:* Contains the actual data for the command (optional).
    - *Le (Length of Expected Response):* Specifies the maximum length of the response data expected from the HCE service.
  - A response APDU includes:
    - *Data Field:* Contains the response data (if applicable).
    - *SW1, SW2 (Status Bytes):* Indicate the processing status of the command (e.g., success, error codes).

- **ISO/IEC 7816-3: Electronic Signals and Transmission Protocols**
  - This part specifies the communication protocols between the NFC reader and the device, including the exchange of APDUs over NFC.
  - It ensures reliable data transmission, framing, and error detection mechanisms.
- **ISO/IEC 14443: Proximity Cards**
  - Defines the physical characteristics, radio frequency power and signal interface, and initialization protocols for proximity cards.
  - In HCE's case, this convention makes the NFC reader compatible with the emulated card functionality.

- **ISO/IEC 7816-8: Commands for Security Operations**
  - Offers instructions on how to use cryptographic operations like encryption and digital signatures, which are essential in secure HCE communication.
  - These operations guarantee data confidentiality, integrity, and authenticity in APDU transactions.

2.2.2 Architecture of HCE

The design of an HCE-enabled system will typically have three parts:

- *NFC Interaction*: It is the communication module. External NFC readers talk to the NFC-enabled device via this module. It deals with APDU transmission and reception, which are basic data packets standardized for use in communication.

- *HCE Module*: The HCE service is a software component running on the NFC device with the role of simulating the activity of a contactless smart card. It catches incoming APDUs from external NFC readers, interprets the commands, and reacts accordingly, simulating the activity of a physical smart card.

- *Secure Element Emulation Layer*: Some HCE implementations can involve a secure element emulation layer that emulates the behavior of a physical secure element [2]. The layer offers a trusted environment to store sensitive information, e.g., cryptographic keys and payment data, and perform secure operations, e.g., cryptographic processing and tamper-resistant storage. While the secure element emulation layer is not always needed, it does offer some extra security to HCE-enabled systems in cases where there is high-security demand.

- *Application Layer*: The application layer consists of user applications that use HCE functionality. Examples of these applications are mobile payment apps [13], access control systems [14], ticketing applications, loyalty apps, and others. All apps communicate with the HCE service to start transactions, exchange data with external NFC readers, and download secure resources stored on the device.

2.2.3 Operation of HCE

The HCE process is a sequence of steps [2] that takes place whenever an NFC device enters proximity with an external NFC reader:

- *Initialization*: As an NFC device enters the proximity of an external NFC reader, the HCE service will initialize and prepare itself to accept incoming communication.

- *APDU Exchange*: APDUs are exchanged from the external NFC reader to the HCE service, requesting actions or data. These APDUs often contain commands to start a transaction, authenticate the device, get data, or have some other operation.

- *APDU Processing*: When it receives an APDU, the HCE service executes the command according to the application logic and security regulations. It can include authentication of the request, decrypting the encrypted data, and access of secure resources, before performing the requested action.

- *Response Generation*: The HCE service, after executing the APDU, produces an appropriate response APDU with the requested information or indicator of the executed command. The system returns the response to the external NFC reader.
- *Transmission*: The response of the APDU is sent back to the external NFC reader that finishes the communication cycle. The external NFC reader can then respond and do as needed, be it to complete a payment transaction, provide access, or do some other action.

In HCE operation, the security system utilizes strong security controls to provide confidentiality, integrity, and authenticity of data being communicated between the NFC device and readers that are external. At different levels, encryption, authentication, and access controls, among others, are utilized to protect against unauthorized use, data interception, and tampering.

2.2.4 Advantages of HCE

HCE provides a number of benefits compared to traditional physical secure elements and other NFC-based solutions:

- *Cost-Efficiency*: Through the use of existing hardware and software capabilities, HCE avoids the use of specially created secure element chips, lowering NFC-enabled device production costs.
- *Enhanced Flexibility*: HCE facilitates dynamic provisioning and credential management, supporting easy integration with different applications and services without physical card limitation.
- *User Convenience*: The customers can just perform transactions and interactions using their devices that support NFC without needing to carry numerous physical cards or tokens.
- *Rapid Deployment*: Solutions based on HCE are readily deployable and even remotely updated through software updates, allowing for speedy distribution of new services and functionality without physical card replacement or modification of infrastructure.

2.3 Digital Signatures [1]

In the digital age, digital signatures have become crucial to document security. Digital Signatures act as a digital fingerprint, verifying a document's legitimacy and ensuring it remains unmodified since signing. Digital signatures leverage cryptographic techniques, including public and private keys, as shown in Figure 2.3. The Signer creates the digital signature using its private key to encrypt a distinct hash (a digest) of the document. Anyone can then decrypt the signature with the Signer's public key to confirm its authenticity by comparing the document's computed hash with the hash received by decrypting the signature. Digital signatures are necessary for building confidence and enabling safe transactions in many sectors where document integrity is crucial.

Digital Signatures provide essential security services, including authentication, data integrity, and non-repudiation. Authentication ensures that the claimed sender signed

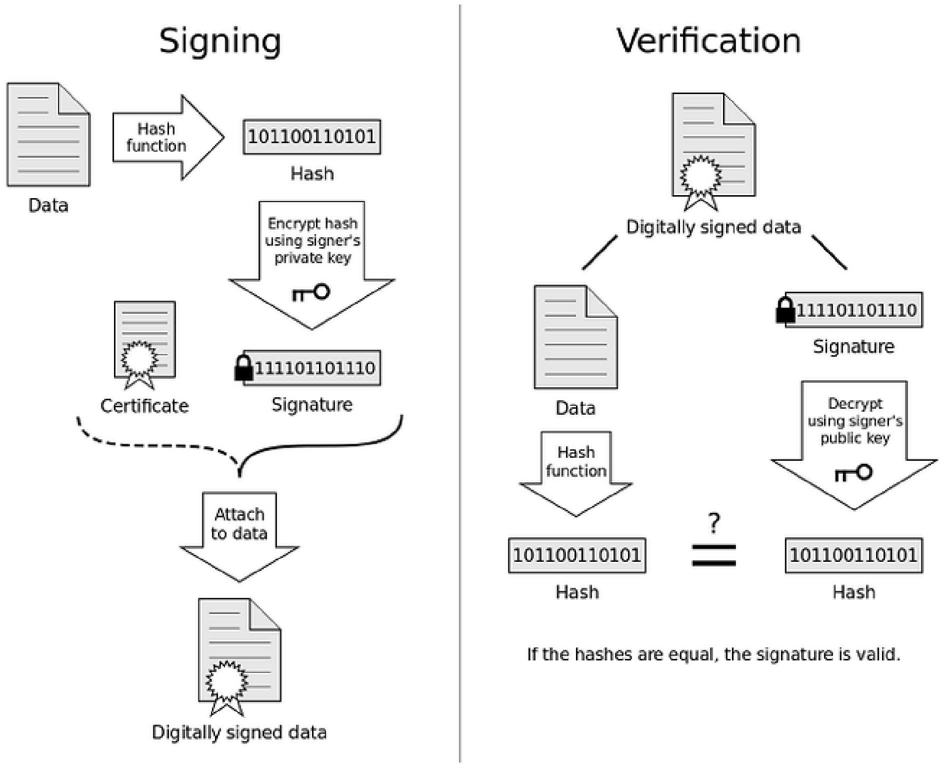


Figure 2.3: Digital signatures [3]

the message or document. Integrity ensures the system has not altered the content during transmission or storage. Non-repudiation ensures that the sender cannot deny having signed the message at a later stage. Several digital signature algorithms exist today, such as Rivest–Shamir–Adleman (RSA) [16], Elliptic Curve Digital Signature Algorithm (ECDSA), and more recently, identity-based schemes like the Hess IBS. Traditional methods like RSA depend on the PKI [16], which involves digital certificates and certificate authorities to validate the keys. However, such systems can introduce overhead, especially in mobile or resource-limited devices. Digital signatures are widely used in applications such as electronic contracts, software distribution, secure email, legal documents, and government records. Their growing importance reflects the need for secure, verifiable communication in a world increasingly dependent on digital interactions.

2.4 Transport Layer Security (TLS 1.3)

TLS is a basic cryptographic method that secures communications over computer networks [16]. It is a de facto implemented standard critical to internet surfing, emailing, and other web-based applications where data confidentiality and integrity are paramount. TLS establishes an encrypted channel of communication between two devices that send messages to each other, usually a client (e.g., web browser) and a server. It employs encryption for the protection of data in transit and authentication mechanisms to ascertain identity of the parties in communication. TLS 1.3, which is the latest version of the protocol, greatly boosts efficiency, security, and privacy by reducing the number of round trips required to create a secure connection [16]. It results in quicker page loading, reduced latency, and improved responsiveness for web applications. TLS 1.3 eliminates the insecure legacy, employs stronger contemporary ciphers, and encrypts a larger majority of the handshake process to deliver more extensive user privacy. Although TLS is extremely secure, it has the potential for high overhead since certificate sizes can be excessively large, which is difficult to accommodate in resource-poor devices.

Forward-secure cipher suites are the norm in TLS 1.3, eliminating legacy features like static RSA key exchange and SHA-1 hash functions. These improvements minimize the attack surface and enhance security for channel communication. TLS 1.3 streamlines the handshake by condensing it to a single round trip, optimized for utmost performance without compromising security. This makes TLS 1.3 quicker and more efficient to establish secure connections compared to previous protocol versions. But even TLS depends on PKI for the authentication of both parties. PKI mandates that certificates have to be signed by Certificate Authorities (CAs), and that adds latency because of the verification of certificates, especially when more than one certificate is used within the certificate chain. That is generating memory and computational cost, which is undesirable on low-resource systems like smartphones, IoT devices, or offline systems. These systems can find it difficult with the computational load of PKI, making TLS less desirable in applications where speed and simplicity and low resource utilization prevail the concern.

2.5 Identity Based Authenticated Key Agreement

Identity-Based Authenticated Key Agreement (IBAKE) is a cryptographic technique by which two users can build a common secret key securely only upon their own identity information. Adi Shamir first proposed the concept of IBC [11] with the entity's identity as its own public key in 1984 [17]. However, building an efficient and secure Identity-Based Encryption (IBE) scheme proved difficult until 2001 when Boneh and Franklin developed a novel scheme based on bilinear pairings on elliptic curves [17]. IBAKE enhances the agreement process since it does not involve the complexity of working with public key certificates. Rather, a reliable third party, the Private Key Generator (PKG), generates private keys for every user from their verified identity and eliminates the burden of a conventional PKI [18]. In the IBAKE protocol, identities are exchanged and independently calculated to derive the same shared secret key with the private key and the other's identity by both individuals. The shared secret key is used for secure communication and authentication.

IBAKE protocol streamlines mutual authentication through the elimination of certificate exchange. Rather than validating certificates generated by third-party authorities, the protocol enables each side to derive the shared session key from identity information and system-generated parameters. The protocol greatly minimizes communication and processing overhead in PKI-based key exchanges. At the core of this configuration is the PKG, which produces private keys using a master key that is secure and a map function to which the user identity is mapped to a point on an elliptic curve. The PKG should be secure and trustworthy since compromise would impact the overall system. When users receive their private keys, users can establish secure sessions without having to handle the PKG again, which is a welcome feature for offline or low-connectivity situations.

2.6 Bilinear Pairing

Bilinear Pairing is a strong cryptography idea [19]. It consists of three sets of prime order 'q' of mathematics: G1 and G2 as sets of addition, and G_{T} as a set of multiplication.

Bilinear Pairing involves one more special set of numbers, denoted as $\mathbb{Z}_q^*$, which contains all positive integers smaller than a large prime number 'q' excluding zero with the property of having a multiplication inverse modulo 'q'. The base of secure communication lies in a function called a pairing, defined as e: G1 x G2 -> G_T. This function requires two elements, each from G1 and G2, and maps the elements to an element in G_T. However, this mapping is not random - it possesses three key properties:

1. *Bilinearity:* Picture multiplying an element in G1 by a number 'a' and doing the same for an element in G2 by a number 'b'. The pairing of these multiplied parts is thus identical to the initial pairing raised to the power of the product 'ab'. The following eq. 2.1 captures this attribute:

$$(e(aP_1, bP_2) = e(P_1, P_2)^{ab}) \tag{2.1}$$

   Here, P1 and P2 are generators of their corresponding groups G1 and G2 respectively.

2. *Non-degeneracy:* There exist special non-trivial elements, P1 in G1 and P2 in G2, such that their pairing, e(P1, P2), is not the identity element in G_T (often denoted as 1 or 0 depending on the group). It ensures the pairing function is non-trivial, providing valuable information for cryptographic applications.

3. *Computable:* For all $S, T \in G, e(S, T)$ is easily computable.

   Assumption 1 [Bilinear Diffie-Hellman (BDH)]: For $x, y, z \in \mathbb{Z}_q^*$, given (P, xP, yP, zP), computing $e(P, P)^{xyz}$ is hard.

   Assumption 2 [Computational Diffie-Hellman (CDH)]: For $x, y \in \mathbb{Z}_q^*$, given (P, xP, yP), computing xyP is hard.

   Assumption 3 [Elliptic Curve Discrete Logarithm (ECDL)]: Given (P, xP), obtaining $x \in \mathbb{Z}_q^*$ is hard.

Bilinear pairings allow identity-based cryptographic protocols to perform digital signatures, encryption, and key agreement operations by enabling relationships between values that traditional discrete logarithm-based methods cannot achieve. The pairing operation creates a mathematical bridge between two elliptic curve groups and a target group, making verifying relationships in identity-based authentication and signature schemes possible.

2.7 Private Key Generator (PKG)

The PKG is the central part of any IBC system [17]. The PKG is a trusted authority that initializes the system parameters and generates user private keys from their identities. The function of the PKG is analogous to Certificate Authority (CA) in conventional PKI but without issuing or possessing digital certificates. In the setup process, the PKG chooses a master secret key and computes corresponding public system parameters. These parameters are something like an elliptic curve generator point, a bilinear pairing function, and secure cryptographic hash functions. After setting up the system, the PKG derives user private keys from its master secret key. It obtains each user's private key from the user's own identity, for example, email address or telephone number, first hashed into a curve point. The security of the entire identity-based scheme depends on the master key of the PKG not being revealed. When the master key is revealed, all derived private keys are insecure. Therefore, the PKG needs to be safeguarded with robust access controls, secure hardware, or in isolation inside a trusted execution environment.

2.8 Secure Elements

Secure Elements (SEs) [20] are security-hardened hardware components meant to safeguard sensitive information and carry out important cryptographic tasks in a highly secure setting. SEs find extensive use in mobile payments, identity authentication, access control, and digital signatures. They provide robust hardware security against attacks at the physical level and unauthorized access, making them compatible for safeguarding digital credentials and private keys. There are a number of different Secure Element types, such as embedded SEs, part of a device's chipset, Universal Integrated Circuit Cards (UICC) employed in SIM cards, and removable microSD cards. Each of these form factors complies with industry security requirements and provides cryptographic functions such as key generation, digital signatures, encryption, and secure storage.

2.9 Hess Identity-Based Signature (Hess-IBS) Scheme

The Hess IBS scheme [12] is an extremely efficient identity-based signature generation scheme, founded on the computation of calculating elliptic curve bilinear pairings. The scheme was first proposed by Florian Hess in 2002 [12]. Digital certificates are circumvented by employing a user's identity as the public key in this scheme. The security of the Hess-IBS scheme is based on intractability of the Bilinear Diffie-Hellman problem in pairing groups and follows the random oracle model. The system has four fundamental algorithms: Setup, Extract, Sign, and Verify. The trusted PKG produces system parameters and master keys in the Setup phase. In the Extract step, the PKG employs the hash representation of the user identity to retrieve the private signing key of the user. Signing consists of computing a pair of values from the identity and message hash through scalar multiplication and pairing operations. Verification is attained through the inspection of pairing relations and the validation of the integrity of the hash.

The scheme has two cyclic groups G and G_T of order of prime l and funtion of bilinear pairing $e : G \times G \to G_T$. Let P be a generator of G, and let H be a hash function that maps user identities to group elements in G. The PKG selects a random master key $t \in \mathbb{Z}_l^*$ and publishes $Q_{\mathrm{TA}} = tP$, keeping t private.

To extract a private key for a user with identity ID, the PKG computes $S_{ID} = t \cdot H(ID)$. To sign a message m, the signer chooses a random scalar k, selects an arbitrary $P_1 \in G$, computes $r = e(P_1, P)^k$, hashes (m, r) to get a scalar v, and finally calculates $u = vS_{ID} + kP_1$. The signature is the pair (u, v).

To verify the signature, the verifier computes $r' = e(u, P) \cdot e(H(ID), -Q_{\mathrm{TA}})^v$ and checks whether $v = h(m, r')$. If the condition holds, the signature is valid.

The scheme provides sound security guarantees, such as existential unforgeability in the presence of chosen-message attacks, assuming that the BDH problem is infeasible. Hess-IBS differs from other identity-based constructions in the best computational cost and signature size. The Signer only does one pairing exponentiation and one scalar multiplication. In the verification process, at most, two pairings and one exponentiation are needed, which can be reduced further if it accesses identities regularly.

Chapter 3

Proposed iDSign Framework

3.1 Introduction

In response to the growing needs for safe, efficient, and lightweight digital signature solutions, the iDSign system has been devised, a new system meant to break current limitations. iDSign is a multipurpose system that is designed to enable strong authentication, data protection, and secure document signing on resource-constrained environments, notably offline environments. The iDSign framework employs an iTLS protocol adaptation [10] for mutual authentication, a lightweight alternative to traditional certificate-based signing schemes. The signing process employs the Hess IBS scheme [12] for cryptographic protection with minimal computational overhead. The iHDoc, the proof-of-concept version of the iDSign framework, employs the low-power NFC HCE mode to enable tap-based in-site digital document signing. iHDoc analyzes the efficiency and practical use of the iDSign system within a real-world scenario in contrast to RSA-based mutual authentication [16]. Including a mobile application, iHDoc facilitates smooth signing and verification activities, facilitating offline and mobile use. The architecture, framework, and prototype design are presented in this section, emphasizing the innovation of iDSign and its ability to rectify the limitations of current digital signature systems.

3.2 Related Work

Chak et al. [5] presented AuthPaper, a method to increase the security and authenticity of paper documents. The process embeds an Authenticated 2D barcode within the document itself containing a digitally signed version of the document content and layout information. Utilizing the JavaScript Object Notation (JSON) data structure format and Quick Response (QR) codes for space-compressed visual encoding, AuthPaper enables effective storage and retrieval of critical document information. To achieve higher verification reliability, the authors adapted the open-source, popular barcode scanning library ZXing [21] into a modified version in order to incorporate extra algorithms to enhance QR code decoding precision even in poor conditions like poor-quality printing or a damaged barcode. AuthPaper's offline nature, independent of extrinsic databases or direct access to the network during verification, provides a complete and simple solution. But material barcode reliance also involves a vulnerability: if the barcode is compromised, it can become unreadable, which makes the authentication process questionable and unverifiable to the document's authenticity.

Gourab et al. 2017dsign introduced DSign, a novel digital signature system to make signing and verification more convenient using paperless business support. DSign uses a centralized database to maintain the critical metadata of a digitally signed document. Rather than including the whole content of the document within the signature, DSign uses the barcode or QR code as a reference to the database record of the document to minimize the computation load upon verification. This design circumvents the complexity of cryptographic processing on the client and facilitates quick verification of document origin. DSign also dispenses with the requirement for special hardware such as USB tokens in a manner that simplifies the signing and verification for users. However, utilization of a central database creates potential security risks, where integrity and confidentiality of the document details stored may be compromised by any unauthorized access, data breaches, or server crashes, thus affecting the trust value of the entire system.

Another new digital signature method using the utilization of biometric, such as fingerprints, for improved security and user authentication was introduced by Saxena et al. [22]. This thesis refers to the method as BioSign. Unique fingerprint characteristics are read by fingerprint scanners and encrypted into a "bio-cryptosystem." The system is combined with the document's hash, an exclusive digital fingerprint of document content, to generate an exclusive digital signature. The signature provides secure authentication of the Signer identity and document originator, since it is hard to replicate or forge. By correlating the biometric data with the signature, BioSign makes the process more secure than traditional digital signatures that depend entirely on cryptographic keys. Yet, biometric systems need to be perfect. They are susceptible to issues of accuracy such as false positives (misidentifying an unauthorized user) or false negatives (misfailing to identify an authorized user), which would lead to issues of security or privacy. Besides, biometric data capture and storage raise ethical concerns regarding the privacy of users and safeguarding of data since sensitive information can be misused in case it is hacked or accessed without users' authority.

Amer et al. [6] proposed a web-based approach to ensure the integrity and authenticity of PDF documents by preventing any unauthorized changes, which is referred to as PDFGuard in this paper. PDFGuard employs a secure cryptographic approach involving the use of the Secure Hash Algorithm 3 (SHA-3) along with the RSA digital signature scheme. SHA-3 generates an immutable and constant-length hash value (cryptographic fingerprint) of the document text, and a value of this kind is encrypted with the RSA algorithm and the sender's private key to generate a digital signature. The signature is appended securely to the PDF document, making it a tamper-evident stamp. Any alteration of the document would cause the recomputed hash and decrypted signature to vary, thus the proof of unauthorized alteration. The web-based character of PDFGuard provides convenient access and transparent integration into current document workflows, allowing users to verify the authenticity and integrity of PDF documents without special software installation. But as it is an internet-based solution, PDFGuard is vulnerable to some common cyber attacks such as man-in-the-middle (MITM) where the attacker can intercept and tamper with communication between the server and the user and phishing attacks where users are manipulated to provide credentials on forged websites. Moreover, PDFGuard's use of self-signed certificates, which do not carry the trust and confidence of their signed popular Certificate Authority (CAs) counterparts, might be inappropriate in high-security environments where stringent authentication and non-repudiation are of greatest concern. The lack of third-party trusted validation for the self-signed certificates

Table 3.1: Proposed iDSign framework: Comparison of existing digital signature schemes

Research	**Primary Method Used.**	**Imp.**	**Encrypt. Key Used**	**Signature Format**	**Verificat.**	**Limitations**
AuthPaper [5]	RSA, Qr Code	Soft.	Document Content + layout	2D barcode	Specialized Scanner, Offline	Physical damage vulnerability
DSign [4]	Centralized Signature Database	Web	Doc metadata (header, subject)	QR Code, Barcode (links to the database)	Web Access, Cloud Database lookup	Dependence on centralized database
BioSign [22]	Hashing biometric data, encryption	Soft.	Biometric Features	Biometric Hash	Fingerprint Scanner, hash checks	Accuracy issues in biometric authentication
PDFGuard [6]	RSA, SHA-3	Web	RSA algorithm	Hex String (Encrypted Hash)	Hash Comparison	Susceptible to web-based attacks
ECDoc [7]	ECDSA	Web	ECDSA algorithm	Encrypted Hash	Hash Comparison	Use of self-signed certificates

leaves room for potential certificate forging and impersonation attacks that compromise the system's overall security.

Rizqi et al. [7] investigated the application of the ECDSA in the protection of the integrity of PDFs. They contrasted their scheme, which has been termed ECDoc here, with other digital signature schemes like RSA and DSA and demonstrated how ECDSA was more secure, computationally efficient, and required less memory, particularly on resource-constrained platforms like the mobile phone. ECDSA is equally secure as RSA but using smaller key sizes and thus better placed to be used on devices with limited resources. The authors developed a web-based prototype which thoroughly exemplifies the feasibility of using ECDSA in PDF document signing and verification operations. The prototype illustrates how ECDSA can be used to reduce digital procedures and make PDF documents more secure. But, as with PDFGuard, the prototype's use of self-signed certificates is an impeding factor. Self-signed certificates do not have the trust and support of certificates drawn from trusted CAs but face improper issuance and misuse of certificates. This lack of trust can prevent the use of ECDoc in situations where it is absolutely necessary for the authenticity and non-repudiation of digital signatures due to the fact that the lack of independent verification would be undesirable in raising questions regarding the signatures' validity.

Existing solutions have limitations, including reliance on centralized databases, physical danger, cumbersome key management, and high security risk, shown in Table 3.1. Most importantly, none of the existing approaches achieves lightweight mutual authentication or employs SE for protecting credentials securely. Therefore, this paper proposes a new paradigm, iDSign (Identity-based Digital Document Signatures), which overcomes these crucial limitations using a secure, light-weight, and adaptive scheme for digitally signing documents.

3.3 Proposed iDSign Architecture

The iDSign (Identity-based Digital Document Signatures) framework enables secure and lightweight digital document signing across diverse devices and communication interfaces, with one device acting as a "Signer" and the other as a "Signee". They may communicate remotely over Transmission Conrol Protocol (TCP/IP) or via an NFC tap, providing flexibility in how they carry out the signing process. At the framework's core is the PKG; it can be imagined as a digital authority in an organization responsible for generating cryptographic keys based on the unique identities of each device registering with PKG. The secure storage and handling of the master secret key by the PKG is of utmost importance, as any compromise could allow unauthorized users to infiltrate the system and falsely register as signers or signees. It requires strong security to protect from such intrusions. After key generation, credentials are stored in secure SE that is tamper-proof on each device utilized for secure storage.

The mutual authentication between 'Signer' and 'Signee' is exchanged using a variant of the Lightweight iTLS protocol [10] such that both are verified to be authentic and does not permit unauthorized access. During this step, a session key is generated by the device such that future communication between the devices is encrypted. Hess IBS scheme is utilized for digital signatures since it is lightweight in nature [12] and therefore can be employed in resource-constrained devices. It uses the hash of the document according to the SHA-256 algorithm, extra variables, and the private key of the Signer to create a secure digital signature. Verification is done by recalculation of the hash of the document and use of some certain variables that are part of the Hess IBS scheme, along with the public key of the Signer, in order to compute the portion of the signature in order to compare it with the initial one.

3.3.1 PKG Security Strategies:

Although iDSign framework relies on the fact that PKG is safe and secure, one should realize that PKG security is of paramount concern in identity-based cryptographic schemes. A compromised PKG may produce spoofing keys or conduct man-in-the-middle (MiTM) attacks, which will undermine the system's overall security. Key Strategies for PKG security:

- *Hardware Security Modules (HSMs):* The PKG can leverage HSMs to securely store and manage cryptographic material within an isolated environment, thereby preventing unauthorized access [23].

- *Access Controls and Auditing:* Strict access controls restrict who can interact with the PKG, while ongoing auditing detects questionable activities, hence reducing the dangers of insider threats [24].

- *Secure Backup Mechanisms:* Encrypted backups of the PKG master key guarantee the key's availability in the event of hardware failures [23].

Table 3.2: Proposed iDSign framework: Symbols, abbreviations, and operators description

Notations	**Definitions**
PKG	Private Key Generator
P	Generator
e	Bilinear pairing
G	Additive Group of Prime order q
G_{T}	Multiplicative Group of Prime order q
s	Master Secret Key
P_{Pub}	Master Public Key
$H()$	Generic Hash Function
SPP	System Public Parameter
$M()$	Mapping Function: maps any string to a point on elliptic curve G
ID_{S}	Signer's Identity
ID_{D}	Signee's Identity
SK_{S}	Signer's Private Key
SK_{D}	Signee's Private Key
PK_{S}	Signer's Public Key
PK_{D}	Signee's Public Key
$\|\|$	Concatenation Operator
SignerHello	Contains parameters for Mutual Auth. relevant to Signer
SigneeHello	Contains parameters for Mutual Auth. relevant to Signee
EK_{S}	Ephemeral Public Key of Signer
EK_{D}	Ephemeral Public Key of Signee
x	Ephemeral Secret randomly chosen by Signer
y	Ephemeral Secret randomly chosen by Signee
N_{S}	Large Random Nonce chosen by Signer
N_{D}	Large Random Nonce chosen by Signee
HTK	Handshake Traffic Key
$HKDF$	HMAC-based Key Derivation Function
H_{DOC}	Hash of Document
SS_{S}	Signer's Shared Secret
SS_{D}	Signee's Shared Secret
EFM	Encrypted Finished Message
$E_{\mathrm{KEY}}()$	Data Encrypted by "Key"
$D_{\mathrm{KEY}}()$	Data Decrypted by "Key"
$ESign$	Encrypted Signatures via HTK : $E_{\mathrm{HTK}}(\mathrm{Sign})$

3.4 iDSign Framework Protocol

This work proposes a secure protocol for iDSign framework, which consists of three phases: *PKG Setup - Devices Registration*, *Mutual Authentication and Session Key Establishment Phase*, *Digital Signature Phase*. Table 3.2 contains a list of notations used in this work. The following assumptions are taken into account when creating the framework:

- This work assumes that the PKG is reliable, safe, and unaffected and will not conduct any aggressive attacks. Given that the master key that the PKG possesses is the source of all private keys.

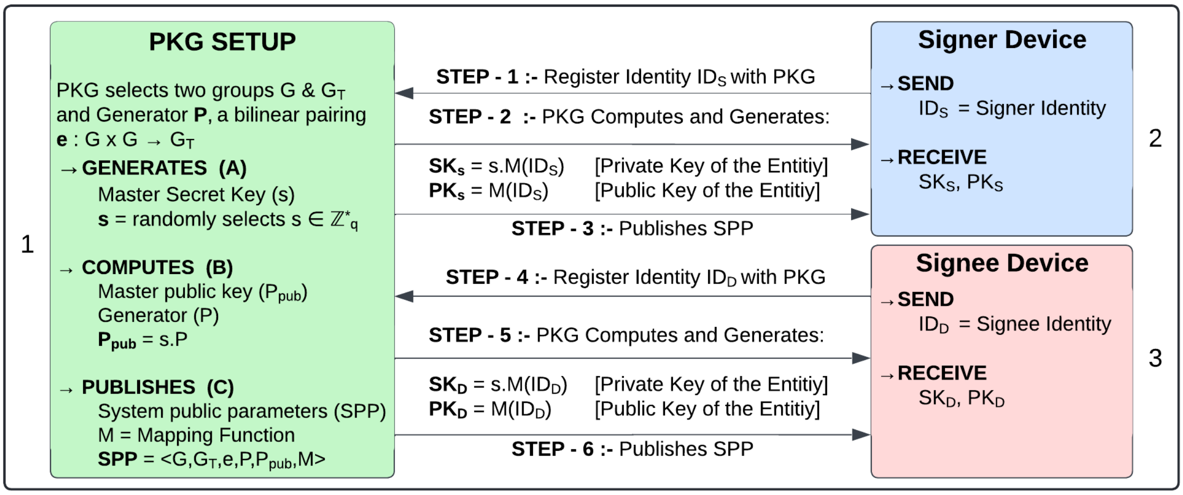

PKG SETUP
PKG selects two groups G & G_T and Generator P, a bilinear pairing e : G x G → G_T
→ GENERATES (A)
Master Secret Key (s)
s = randomly selects s ∈ ℤ*q
→ COMPUTES (B)
Master public key (P_pub)
Generator (P)
P_pub = s.P
→ PUBLISHES (C)
System public parameters (SPP)
M = Mapping Function
SPP = <G,G_T,e,P,P_pub,M>
1
STEP - 1 :- Register Identity ID_S with PKG
STEP - 2 :- PKG Computes and Generates:
SK_s = s.M(ID_S) [Private Key of the Entitiy]
PK_s = M(ID_S) [Public Key of the Entitiy]
STEP - 3 :- Publishes SPP
STEP - 4 :- Register Identity ID_D with PKG
STEP - 5 :- PKG Computes and Generates:
SK_D = s.M(ID_D) [Private Key of the Entitiy]
PK_D = M(ID_D) [Public Key of the Entitiy]
STEP - 6 :- Publishes SPP
Signer Device
→SEND
ID_S = Signer Identity
→RECEIVE
SK_S, PK_S
2
Signee Device
→SEND
ID_D = Signee Identity
→RECEIVE
SK_D, PK_D
3

Figure 3.1: Proposed iDSign architecture: PKG setup phase

- All cryptographic operations and key storage occur within a tamper-resistant SE.
- All the communication with PKG takes place over a secure channel.

The details of the phases are as follows:

3.4.1 PKG Setup Phase

Figure 3.1 illustrates how the Signer and Signee register with PKG and generate their Public and Private keys based on their identities.

- *Step 1:* The PKG picks elliptic curve groups G and G_{T}, a bilinear pairing $e : G \times G \rightarrow G_{\mathrm{T}}$, and a generator P of G forming the core of the pairing-based Identity-Based Key Agreement (IBAKA) mechanism utilized by iTLS for mutual authentication [10].
- *Step 2:* The PKG chooses a random value $s \in \mathbb{Z}_q^*$ as the master secret key. It safeguards this value, as it is essential for computing the private keys for the entities.
- *Step 3:* The PKG computes the master public key $P_{\mathrm{Pub}} = s \cdot P$, where P is the generator, makes it public to allow entities to verify the authenticity of messages from the PKG.
- *Step 4:* The PKG defines a Mapping function $M : \{0,1\}^* \rightarrow G$. This function maps an arbitrary string to a point on the elliptic curve group G and derives entity identities from unique identifiers.
- *Step 5:* The PKG provides the system public parameters $SPP = <G, G_{\mathrm{T}}, e, P, P_{\mathrm{Pub}}, M>$. These parameters are crucial for the framework's ability to work and enable entities to carry out cryptographic operations.
- *Step 6:* Entities enroll their identities with the PKG. An entity's identity (ID) is a unique identifier, such as a device ID or an email address.

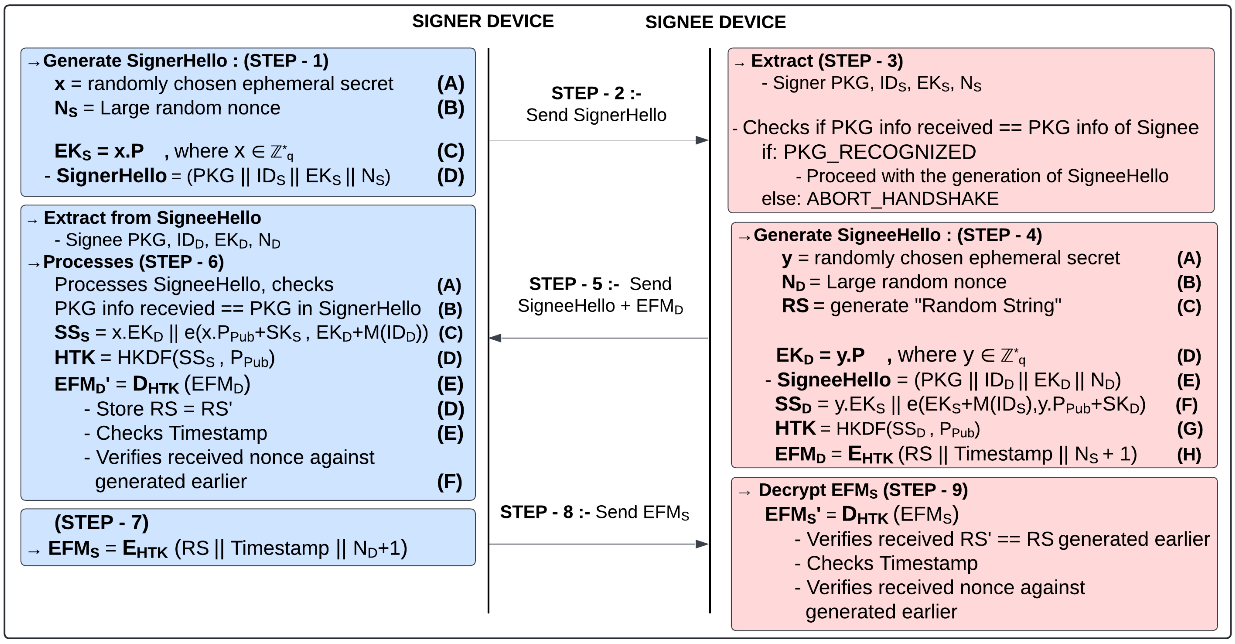

SIGNER DEVICE
SIGNEE DEVICE
→ Generate SignerHello : (STEP - 1)
x = randomly chosen ephemeral secret (A)
N_S = Large random nonce (B)
EK_S = x.P , where x ∈ ℤ*_q (C)
- SignerHello = (PKG || ID_S || EK_S || N_S) (D)
STEP - 2 :- Send SignerHello
→ Extract (STEP - 3)
- Signer PKG, ID_S, EK_S, N_S
- Checks if PKG info received == PKG info of Signee
if: PKG_RECOGNIZED
- Proceed with the generation of SigneeHello
else: ABORT_HANDSHAKE
→ Generate SigneeHello : (STEP - 4)
y = randomly chosen ephemeral secret (A)
N_D = Large random nonce (B)
RS = generate "Random String" (C)
EK_D = y.P , where y ∈ ℤ*_q (D)
- SigneeHello = (PKG || ID_D || EK_D || N_D) (E)
SS_D = y.EK_S || e(EK_S+M(ID_S),y.P_Pub+SK_D) (F)
HTK = HKDF(SS_D , P_Pub) (G)
EFM_D = E_HTK (RS || Timestamp || N_S + 1) (H)
STEP - 5 :- Send SigneeHello + EFM_D
→ Extract from SigneeHello
- Signee PKG, ID_D, EK_D, N_D
→ Processes (STEP - 6)
Processes SigneeHello, checks (A)
PKG info recevied == PKG in SignerHello (B)
SS_S = x.EK_D || e(x.P_Pub+SK_S , EK_D+M(ID_D)) (C)
HTK = HKDF(SS_S , P_Pub) (D)
EFM_D' = D_HTK (EFM_D) (E)
- Store RS = RS' (D)
- Checks Timestamp (E)
- Verifies received nonce against generated earlier (F)
(STEP - 7)
→ EFM_S = E_HTK (RS || Timestamp || N_D+1)
STEP - 8 :- Send EFM_S
→ Decrypt EFM_S (STEP - 9)
EFM_S' = D_HTK (EFM_S)
- Verifies received RS' == RS generated earlier
- Checks Timestamp
- Verifies received nonce against generated earlier

Figure 3.2: Proposed iDSign architecture: Mutual authentication phase

- *Step 7:* The PKG computes the private key $SK_{ID} = s \cdot M(ID)$ by mapping the Identity using the mapping function M and multipying it with the master secret key s.
- *Step 8:* The PKG securely sends the private key SK_{ID} to the respective entities. A mobile Trusted Platform Module (TPM) could establish this secure channel.
- *Step 9:* The associated public key $PK_{ID} = M(ID)$ is used by the entity. Anyone can use the publicly available hash function to compute the public key from the entity's Identity.

3.4.2 Mutual Authentication and Session Key Establishment Phase

Figure 3.2 illustrates the mutual authentication process between devices, leveraging a customized version of iTLS protocol [10], omitting traditional certificates in favour of streamlined ephemeral key exchanges and nonce-based verification. Both SignerHello and SigneeHello messages carry crucial elements such as the entity's unique identification, details about the PKG (which includes PKG's identifier, IBAKA algorithm used, and System Public Parameters), ephemeral public key and random nonce. This approach enables both parties to verify trust and independently compute the same shared secret key for secure communication.

- *Step 1:* Upon initiating the communication, Signer device generates SignerHello (containing the identifier of the PKG where the device is registered, Signer's Identity ID_S, Ephemeral Public Key EK_S and Large random nonce N_S). The Signer computes $EK_\mathrm{S} = x \cdot P$ where $x \in \mathbb{Z}_q^*$ is an ephemeral secret randomly chosen by the Signer device before sending a SignerHello message.
- *Step 2:* Signer initiates the mutual authentication by sending a SignerHello message.

- *Step 3:* Signee receives SignerHello and checks whether the PKG information presented is the same as that of the Signee device. Upon recognition, it generates SigneeHello; otherwise, the handshake terminates with an alert.

- *Step 4:* Upon successful validation, the signee device generates random string RS and SigneeHello (containing the identifier of the PKG where the device is registered, Signee's Identity ID_{D}, Ephemeral Public Key EK_{D} and large random nonce N_{D}). The Signee computes $EK_{\text{D}} = y \cdot P$ where $y \in \mathbb{Z}_q^*$ is an ephemeral secret randomly chosen by the Signee device. With the help of received EK_{S} and ID_{S} of Signer, Signee computes the shared secret $SS_{\text{D}} = y \cdot EK_{\text{S}} || e(EK_{\text{S}} + M(ID_{\text{S}}), y \cdot P_{\text{Pub}} + SK_{\text{D}})$ and the Handshake Traffic key (HTK) by extracting the shared secret through the Hashed Message authentication code (HMAC)-based key derivation function (HKDF) [25] $HTK = HKDF(SS_{\text{D}},\ P_{\text{Pub}})$. Upon successful calculation of HTK, the Signee device generates EFM_{D} by concatenating RS, current timestamp, and incremented nonce N_{S} and then encrypting with HTK, as shown in eq. 3.1.

$$EFM_{\text{D}} = E_{HTK}(RS||Timestamp||N_{\text{S}} + 1) \tag{3.1}$$

- *Step 5:* Signee responds with SigneeHello and EFM_{D}.

- *Step 6:* Upon receiving SigneeHello and EFM_{D}, Signer processes the same and checks if the PKG provided in SigneeHello matches up as that in the SignerHello. With the received ID_{D} and EK_{D} provided in the SigneeHello, Signer can also derive the shared secret $SS_{\text{S}} = x \cdot EK_{\text{D}} || e(x \cdot P_{\text{Pub}} + SK_{\text{S}}, EK_{\text{D}} + M(ID_{\text{D}}))$ and $HTK = HKDF(SS_{\text{S}},\ P_{\text{Pub}})$. Through derived HTK, the Signer decrypts $D_{HTK}(EFM_{\text{D}})$ and firstly stores the random string RS to use it later and then verifies the integrity and authenticity by checking the timestamp and nonce value against those generated earlier in the process by Signer.

- *Step 7:* After a successfully verification of timestamp and random nonce N_{S}, the signer device generates EFM_{S} by concatenating received RS from SigneeHello, current timestamp and incremented nonce N_{D} and then encrypting with HTK, as shown in eq. 3.2.

$$EFM_{\text{S}} = E_{HTK}(RS||Timestamp||N_{\text{D}} + 1) \tag{3.2}$$

- *Step 8:* Signer sends EFM_{S} to Signee device.

- *Step 9:* Signee device decrypts the received EFM_{S} and firstly verifies random string RS against previously generated, checks timestamp and nonce value against those generated earlier in the process by Signee. The authentication is marked as completed upon successful verification. Both devices use HTK to encrypt and secure further communication.

- Proof Of Both Devices Obtaining The Same Shared Secret:

By the bilinearity of the pairing [19], both the devices obtain the same shared secret. The shared secret SS_{S} computed on the Signer side is given by eq. 3.3:

$$SS_{\text{S}} = x.EK_{\text{D}} || e(x.P_{\text{Pub}} + SK_{\text{S}}, EK_{\text{D}} + M(ID_{\text{D}}) \tag{3.3}$$

Eq. 3.4 shows the substitution of identity-based elements:

$$= x.y.P||e(x.s.P + s.PK_{\text{S}}, y.P + PK_{\text{D}}) \tag{3.4}$$

Using pairing properties, eq. 3.5 distributes the pairing function over the sum of elements:

$$= x.y.P||e(x.s.P, y.P)e(x.s.P, PK_{\text{D}})e(s.PK_{\text{S}}, y.P)e(s.PK_{\text{S}}, PK_{\text{D}}) \tag{3.5}$$

Applying bilinearity further, eq. 3.6 expresses the pairings as exponentiated base pairings:

$$= x.y.P||e(P, P)^{x.s.y}e(P, PK_{\text{D}})^{x.s}e(PK_{\text{S}}, P)^{s.y}e(PK_{\text{S}}, PK_{\text{D}})^{s} \tag{3.6}$$

Eq. 3.7 presents the reordered expression computed from the Signee's perspective:

$$= y.x.P||e(x.P, y.s.P)e(x.P, s.PK_{\text{D}})e(PK_{\text{S}}, y.s.P)e(PK_{\text{S}}, s.PK_{\text{D}}) \tag{3.7}$$

Eq. 3.8 simplifies the structure using key definitions:

$$= y.EK_{\text{S}}||e(x.P + PK_{\text{S}}, y.s.P + s.PK_{\text{D}}) \tag{3.8}$$

Using the mapping of identities, eq. 3.9 rewrites the final pairing term:

$$= y.EK_{\text{S}}||e(EK_{\text{S}} + H(ID_{\text{S}}), y.P_{\text{Pub}} + SK_{\text{D}}) \tag{3.9}$$

Hence, both devices compute the same shared secret as shown in eq. 3.10:

$$SS_{\text{S}} = SS_{\text{D}} \tag{3.10}$$

3.4.3 Digital Signature phase

Figure 3.3 illustrates the Digital Signature process, leveraging the Hess IBS scheme [12]. The novelty in this phase resides in employing the Handshake Traffic Key (HTK), established during the mutual authentication phase, combined with an Identity-based Signature scheme for document signing. This strategy not only maintains the confidentiality and integrity of the delivered document but also enhances the overall security of the digital signature procedure. During the Mutual Authentication phase, both devices obtain the same HTK to secure further communication in the iDSign framework. In this phase, the Signee sends the document to the Signer, who then signs it using his identity-based private key via the Hess IBS scheme and sends it back to the Signee for it to integrate it with the document and can verify its authenticity using the Signer's public key.

- *Step 1:* Signer asks for basic document (Packet Number + Total Packets) information before receiving the actual document.
- *Step 2:* The Signee sends the requested document information along with the first packet. This process continues sequentially, and the Signer acknowledges each packet the Signee sends until all packets are received.

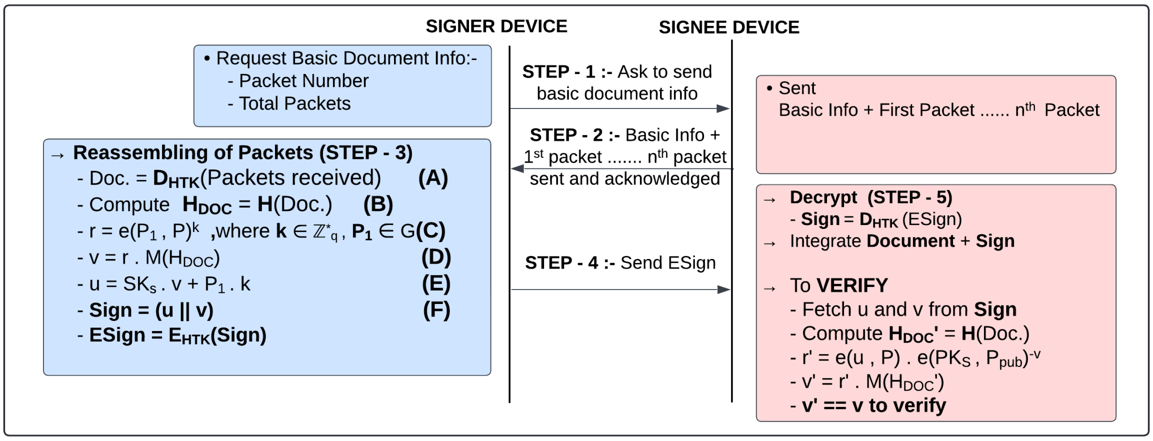

SIGNER DEVICE
SIGNEE DEVICE
• Request Basic Document Info:-
- Packet Number
- Total Packets
STEP - 1 :- Ask to send basic document info
• Sent
Basic Info + First Packet ...... nth Packet
STEP - 2 :- Basic Info + 1st packet .... nth packet sent and acknowledged
→ Reassembling of Packets (STEP - 3)
- Doc. = D_HTK(Packets received) (A)
- Compute H_DOC = H(Doc.) (B)
- r = e(P1 , P)^k ,where k ∈ ℤ*q , P1 ∈ G (C)
- v = r . M(H_DOC) (D)
- u = SK_S . v + P1 . k (E)
- Sign = (u || v) (F)
- ESign = E_HTK(Sign)
STEP - 4 :- Send ESign
→ Decrypt (STEP - 5)
- Sign = D_HTK (ESign)
→ Integrate Document + Sign
→ To VERIFY
- Fetch u and v from Sign
- Compute H_DOC' = H(Doc.)
- r' = e(u , P) . e(PK_S , P_pub)^-v
- v' = r' . M(H_DOC')
- v' == v to verify

Figure 3.3: Proposed iDSign architecture: Digital signature phase

- *Step 3:* The Signer decrypts the packets with HTK, reassembles them, computes the hash of the document extracted using SHA-256 method, denoted as $H_{\text{DOC}} = H(\text{Doc.})$. Using the Hess IBS scheme [12], the Signer computes r, v and u, as shown in eq. 3.11, 3.12 and 3.13:

$$r = e(P_1, P)^k \tag{3.11}$$

  where P is the generator, $P1 \in G$ and $k \in \mathbb{Z}_q^*$

$$v = r \cdot M(H_{\text{DOC}}) \tag{3.12}$$

$$u = SK_{\text{S}} \cdot v + P_1 \cdot k \tag{3.13}$$

- *Step 4:* The Signer forms the complete signature as $Sign = (u||v)$ and encrypts this signature using HTK to produce $ESign = E_{\text{HTK}}(Sign)$. The Signer then sends this encrypted signature to the Signee.

- *Step 5:* Upon receiving, Signee decrypts $ESign$ to retrieve $Sign$ and integrate it with the document, hence completing the handshake. For verification, Signee fetches u and v from $Sign$ and recomputes:

  First, the document hash $H_{\text{DOC}}{}'$ is recomputed using SHA-256, as shown in eq. 3.14:

$$H_{\text{DOC}}{}' = H_{\text{SHA256}}(Doc.) \tag{3.14}$$

  Next, r' is calculated using bilinear pairings, as shown in eq. 3.15:

$$r' = e(u, P) \cdot e(PK_{\text{S}}, P_{\text{pub}})^{-v} \tag{3.15}$$

  Then, the value v' is derived using r' and the document hash, as defined in eq. 3.16:

$$v' = r' \cdot M(H_{\text{DOC}}{}') \tag{3.16}$$

Finally, verification is successful if the computed v' matches the received v, as shown in eq. 3.17:

$$v' == v \tag{3.17}$$

3.5 Security Analysis of the iDSign Framework

The PKG is essential within the iDSign framework and is responsible for generating private keys tailored to the unique identities of participating devices. The PKG is presumed to be trusted, protected, not compromised, and will not begin aggressive assaults. Since PKG's secret master key generates all the private keys, a shady PKG or a malicious individual who obtains the secret master key can potentially execute a successful MiTM attack or create unauthorized private keys to impersonate legitimate users. Similar risks are also present in PKI systems, where compromising a certificate authority (CA) can lead to significant consequences.

1. *Resistance against Man-in-the-Middle-Attack:* The iDSign framework efficiently prevents Man-in-the-Middle (MiTM) attacks due to the powerful mutual authentication provided by iTLS [10]. Consider a scenario in which the attacker receives respective EK_{S} and EK_{D}, attacker can create arbitrary $EK_{\text{S}}^{'}$ and $EK_{\text{D}}^{'}$ to replace the original Ephemeral Keys, and transfer them to the Signee and Signer Devices, respectively. For instance, the attacker can arbitrarily choose x' and y', and substitute $EK_{\text{S}} = x.P$ with $EK_{\text{S}}^{'} = x'.P$ and $EK_{\text{D}} = y.P$ with $EK_{\text{D}}^{'} = y'.P$. Following that, the shared secret the Signer computes is $x.EK_{\text{D}}^{'}||e(x.P_{\text{Pub}} + SK_{\text{S}}, EK_{\text{D}}^{'} + H(ID_{\text{D}}))$. However, though the attacker knows EK_{S} and EK_{D}, the attacker will not be able to compute the shared secret without either of the private keys of both devices.

2. *Perfect Forward Secrecy:* This attribute assures that compromising the over time private keys of either of the communication devices never breaks the integrity of the session key generated during the prior interaction, thus the protection of the sensitive information transmitted previously. The computation of the shared secret key relies not merely on the long-term private key of the involved parties but also on the randomly selected ephemeral secrets by each device. Regardless of whether an intruder discovers the private keys, the intruder also needs to figure out $x.y.P$ from $EK_{\text{D}} = y.P$ and $EK_{\text{S}} = x.P$, which is a Computational Diffie-Hellman (CDH) (refer section 3.3) problem and is independent of long-term private keys. Therefore, an intruder with private keys cannot obtain earlier session keys.

3. *Unique Session Keys:* As the device randomly generates the ephemeral public keys in each connection, two distinct communications generate different and independent session keys. Hence, the intruder cannot get relevant information about additional session keys even if someone stoles the session key.

4. *Resilience to Replay Attack:* The keying information utilized for encrypting the data is established by both devices, hence providing replay protection.

5. *Protected from Impersonation Attacks:* The iDSign framework, due to its usage of iTLS, enables protection against impersonation attacks. Even if an attacker succeeds in compromising a device's long-term private key, the attacker will not

be able to establish a connection with either of the devices. Consider a scenario where an attacker impersonates a Signee and knows the private key of Signer SK_{S}, the ephemeral public key EK_{S}, and the ephemeral secret y' attacker chooses. To compute the shared secret $SS_{\mathrm{D}} = y.EK_{\mathrm{S}}||e(EK_{\mathrm{S}}+H(ID_{\mathrm{S}}), y.P_{\mathrm{Pub}}+SK_{\mathrm{D}})$, attacker must compute $e(EK_{\mathrm{S}}, SK_{\mathrm{D}}) = e(P, PK_{\mathrm{D}})^{x.s}$, However, Signee's private key is not known, thus computing $x.s.P$ from $EK_{\mathrm{S}} = x.P$ and $P_{\mathrm{PUB}} = s.P$ is again a CDH problem.

Chapter 4

Design and Implementation of the Proposed iHDoc Mobile Application

4.1 Introduction

The iHDoc mobile application prototype shows a practical implementation of the iDSign framework, as shown in Figure 4.1, enhanced for secure and lightweight digital document signing in proximity-based mobile environments. A key feature is the utilization of NFC in HCE mode [2], allowing mobile devices to replicate smart cards, hence removing the need for supplementary hardware. NFC's proximity-based communication guarantees a safe, easy, and efficient setup optimal for in-person engagements. NFC enhances security for two-way communication by restricting interactions to close-range exchanges, thereby reducing the risks associated with remote cyberattacks, including eavesdropping and replay attacks. The prototype, constructed on Android, utilizes Bluetooth for large file data transmission and the Java Pairing-Based Cryptography (jPBC) library for strong cryptographic functions, providing a complete, reliable solution [26].

4.2 Related Work

In recent years, mobile digital signing solutions have become increasingly popular, with Adobe Sign [27], Visma Sign [28], and DocuSign [29] emerging as prominent choices. These platforms largely depend on cloud-based architecture and PKI, which, although offering strong security, impose constraints that may impact usability and security, especially in mobile and offline environments. Table 4.1 compares existing digital signature solutions.

Adobe Sign [27] employs PKI for authentication and document signing, which cloud-based certificates or hardware tokens can implement. Nevertheless, PKI systems frequently necessitate complex certificate management and incur considerable processing requirements, making them less suitable for resource-constrained devices. The dependence on a centralized authority for certificate validation makes Adobe Sign prone to network-based attacks and restricts offline capabilities.

Visma Sign [28] depends on strong ID verification methods, such as banking or mobile ID, which require continuous internet connectivity for authentication. The reliance on external identity providers and centralized verification creates network weaknesses, making it inappropriate for environments with restricted connectivity. Although Visma

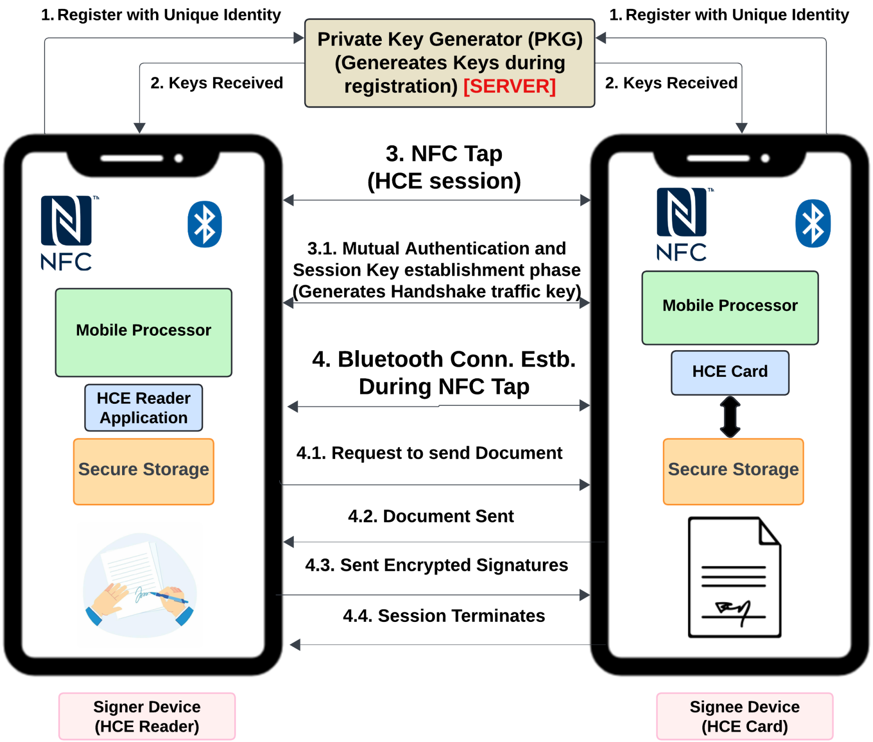

1. Register with Unique Identity
Private Key Generator (PKG) (Genereates Keys during registration) [SERVER]
1. Register with Unique Identity
2. Keys Received
2. Keys Received
3. NFC Tap (HCE session)
NFC
NFC
Mobile Processor
HCE Reader Application
Secure Storage
3.1. Mutual Authentication and Session Key establishment phase (Generates Handshake traffic key)
4. Bluetooth Conn. Estb. During NFC Tap
4.1. Request to send Document
4.2. Document Sent
4.3. Sent Encrypted Signatures
4.4. Session Terminates
Mobile Processor
HCE Card
Secure Storage
Signer Device (HCE Reader)
Signee Device (HCE Card)

Figure 4.1: iHDoc mobile application: Block diagram

Table 4.1: iHDoc mobile application: Comparison of existing digital signature solutions

Existing Solution	**Signature Type**	**Authentication Method**	**Internet Required**	**Encryption Model**	**Limitations**
Adobe Sign [27]	PKI-based digital, Cloud Cert. or tokens	PKI certs, Hardware tokens, Cloud-based tokens	Required	RSA encryption, high computational demand	High processing, Internet dependent
Visma Sign [28]	Strong ID signatures, cloud servers	External ID verification (Banking, Mobile ID)	Mandatory for ID verification	Govt. ID verification	Cloud server dependent, limited offline capability
DocuSign [29]	E-sign, PKI-based digital	SMS, email, third-party integration	Required for auth. and cert. access	RSA encryption, high computational demand	Limited offline access, Cloud dependency

Sign provides secure, regulated authentication, it needs more offline or in-person signing flexibility.

DocuSign [29], a PKI-based service, accommodates several authentication methods like SMS, email, and third-party connections. Nonetheless, its reliance on cloud infrastructure constrains its applicability in offline contexts and presents security vulnerabilities linked to remote authentication.

Although these technologies provide dependable document signing for remote and high-security scenarios, they often rely on constant internet connectivity, making them unsuitable for offline usage. In contrast, the proposed iHDoc mobile application prototype mitigates these limitations by utilizing iDSign framework with low-energy NFC HCE mode [2] for tap-based in-person digital signing. Unlike PKI-based solutions, iHDoc based on iDSign eliminates certificate administration, minimizing computational load and streamlining key management for mobile devices. The NFC HCE mode tap requirement

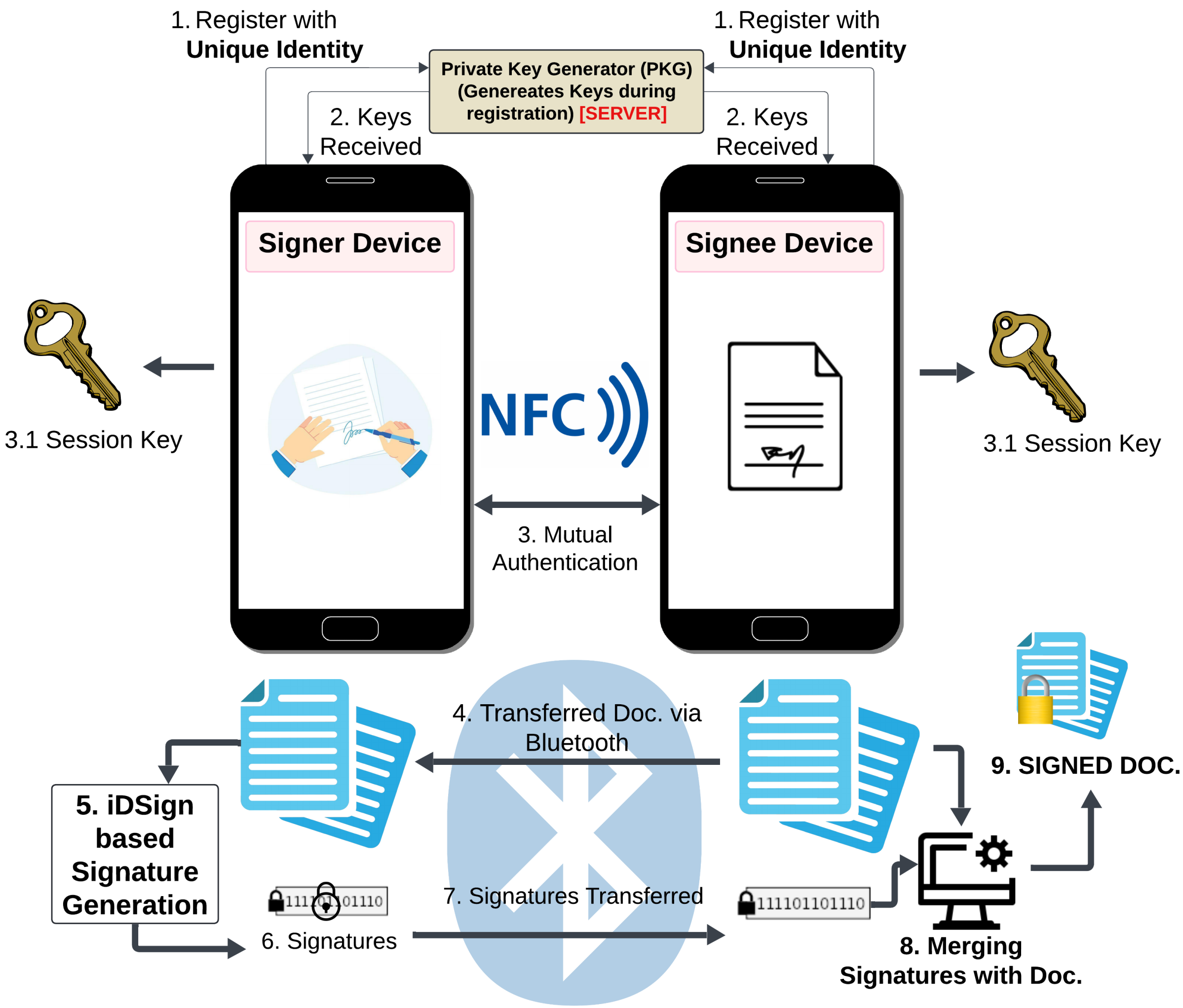


Figure 4.2: iHDoc mobile application: Framework architecture

minimizes proximity attacks such as MiTM attacks. Moreover, iHDoc's offline signing capability makes it well-suited for environments with limited or no connectivity.

4.3 System Architecture and Implementation

The mobile application prototype features an intuitive user interface (see Figure 4.3) designed to streamline the signing process, with distinct Signer and Signee roles in the workflow. This section breaks down the architecture and implementation specifics, demonstrating how each component facilitates secure and efficient in-person digital signing. Figure 4.2 illustrates the process flow for the iHDoc mobile application.

4.3.1 System Components

The iHDoc system consists of the following fundamental components:

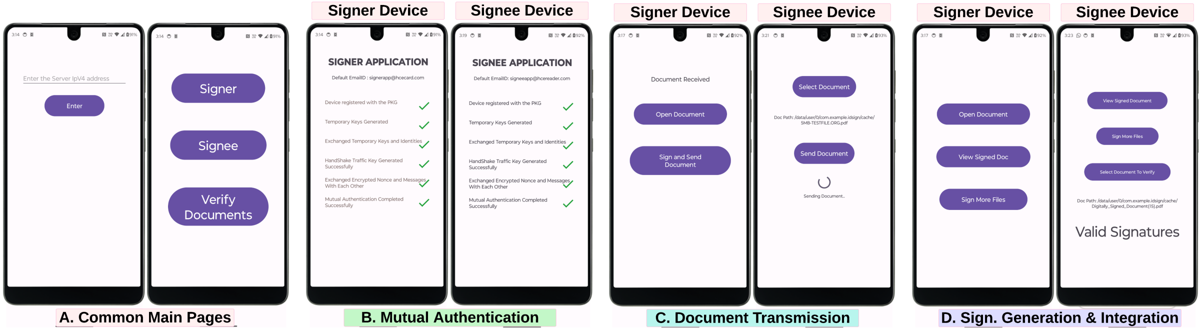
- *Private Key Generator (PKG):* It is a server component that generates and distributes identity-based cryptographic keys.

- *Signer Device:* This device preserves the confidential signing credentials and commences the signing procedure. It functions as the NFC reader in HCE mode [2].
- *Signee Device:* This device requests document signing and receives the signed document. It operates in NFC HCE mode to serve as a virtual card for data transmission.

4.3.2 Workflow Overview

The iHDoc workflow adopts a structured sequence, starting from device registration to completing a digitally signed document. Figure 4.2 shows the details of the iHDoc framework, which utilizes multiple libraries and technologies that work in unison to ensure secure, efficient document signing. The key steps are as follows:

1. *Device Registration with PKG:* The Signer and Signee devices register their unique identities (such as email or phone number) with the PKG. It uses these identities to generate corresponding public and private keys, leveraging IBC.
2. *Keys Reception:* The devices receive their respective keys (private and public) from the PKG.
3. *Mutual Authentication and Session Key Establishment (NFC HCE Session):*
    - Once registered, both devices initiate mutual authentication by tapping the devices, utilizing low-energy NFC HCE mode and iDSign (as shown in Figure 4.3-B).
    - During this tap-based interaction, both devices authenticate each other using iDSign's mutual authentication and exchange ephemeral public keys ($EK_{\mathrm{S}} = x.P$, $EK_{\mathrm{D}} = y.P$), unique identifiers (ID_{S},ID_{D}), nonces (N_{S},N_{D}), and PKG details to establish trust.
    - Both devices independently compute a shared secret SS using bilinear pairing curve operations: $SS_{\mathrm{D}} = y.EK_{\mathrm{S}}||e(EK_{\mathrm{S}} + M(ID_{\mathrm{S}}), y.P_{\mathrm{Pub}} + SK_{\mathrm{D}})$. The shared secret is then processed through an HMAC-based key derivation function (HKDF) to derive the session key $HTK = HKDF(SS, P_{\mathrm{Pub}})$
    - The Signee sends an encrypted message (EFM_{D}) to the Signer, containing a random string (RS), a timestamp, and the incremented nonce (N_{S}+1). The Signer decrypts EFM_{D} using the session key, validates the parameters, and responds with its encrypted message (EFM_{S}) containing similar parameters. Successful validation completes mutual authentication, and the session key secures subsequent communication.
    - In addition to authentication, the Signer's Bluetooth name is shared with the Signee during the NFC tap, facilitating automatic Bluetooth pairing.
4. *Document Transfer via Bluetooth:* The Signee establishes an automated Bluetooth connection and sends the document to the Signer (as shown in Figure 4.3-C).
5. *Digital Signature Generation:* On receiving the document, the Signer uses the Hess IBS [12] scheme to generate the digital signature by utilizing the document's hash, the Signer's private key, and cryptographic parameters managed through the jPBC library [26], which facilitates pairing-based operations for IBC.



Default EmailID : signerapp@hcecard.com

Device registered with the PKG ✓

Temporary Keys Generated ✓

Exchanged Temporary Keys and Identities ✓

HandShake Traffic Key Generated Successfully ✓

Exchanged Encrypted Nonce and Messages With Each Other ✓

Mutual Authentication Completed Successfully ✓

Default EmailID: signeeapp@hcereader.com

Device registered with the PKG ✓

Temporary Keys Generated ✓

Exchanged Temporary Keys and Identities ✓

HandShake Traffic Key Generated Successfully ✓

Exchanged Encrypted Nonce and Messages With Each Other ✓

Mutual Authentication Completed Successfully ✓

Document Received

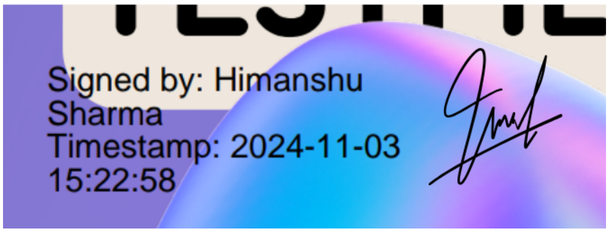
Doc Path: /data/user/0/com.example.idsign/cache/5MB-TESTFILE.ORG.pdf

Sending Document..

Doc Path: /data/user/0/com.example.idsign/cache/Digitally_Signed_Document(15).pdf

Valid Signatures

Figure 4.3: iHDoc mobile application: Screenshots of the application user interface



Signed by: Himanshu Sharma
Timestamp: 2024-11-03 15:22:58

Figure 4.4: iHDoc mobile application: Sample appearance of digital signatures in doc.

6. Signer then encrypts the signatures with the previously generated session key to guarantee secure transmission to the Signee device.

7. *Signature Transmission:* The Signer transmits the encrypted signature to the Signee device over the secure Bluetooth channel.

8. *Signature Integration:* The Signee decrypts signatures using the session key, incorporates it into the original document, and completes the digital signing process. As shown in Figure 4.4, the digital signature includes identifying information and a timestamp for authenticity within the document. The user can continue signing more documents or end the session (as shown in Figure 4.3-D).

9. *Signed Document Storage and Verification:* The final signed document is stored locally and can be verified by any device using the Signer's public key.

4.3.3 Technologies and Libraries Used:

This section breaks down the fundamental technologies and libraries incorporated into the iHDoc prototype, clarifying their functions in secure digital document signing on Android devices.

1. *Android NFC with HCE:* Low-energy NFC technology in HCE mode facilitates proximity, tap-based interactions between Android devices, emulating a physical smart card. This configuration ensures secure session initiation by minimizing vulnerability to man-in-the-middle attacks.

2. *Bluetooth for Data Transfer:* Bluetooth is used for efficient data transmission following iDSign based authentication. This method utilizes Bluetooth's stability and bandwidth for transferring larger files, such as PDF documents, post-authentication.

3. *jPBC Library [26]:* The Java Pairing-Based Cryptography (jPBC) library is an essential element of the iHDoc system, facilitating all cryptographic functions based on IBC. The jPBC library enables bilinear pairing operations crucial for IBC, facilitating secure connections and digital signatures based on user identities instead of conventional public-key certificates. This pairing-based methodology, crucial to iHDoc, facilitates the protocol's lightweight yet safe cryptographic requirements, from mutual authentication to document signing. The jPBC library minimizes the computational and storage requirements commonly linked to certificate-based systems, enabling iHDoc to provide both security and efficiency, making it a perfect option for mobile applications.

4. *Bouncy Castle Library [30]:* Bouncy Castle provides essential cryptographic operations, including the HKDF (HMAC-based Key Derivation Function), which generates the symmetric session key (HTK) during the mutual authentication phase.

5. *iText Library [31]:* iText facilitates fast and seamless PDF file manipulation within the iHDoc system, offering tools for embedding and validating digital signatures.

6. *Spring Framework for Local Server Setup [32]:* A local server built with the Spring framework operates as the PKG, making and distributing keys based on identity. When devices send identity information (e.g., email or phone number) to the server, it provides the associated identity-based cryptographic key pairs, facilitating iHDoc based on the iDSign framework.

Chapter 5

Results

5.1 Performance Analysis of the iDSign Framework

The performance analysis of the iDSign framework and its prototype implementation, iHDoc, is essential to demonstrate their efficiency, security, and practicality. The evaluation includes qualitative and quantitative assessments, measuring computational overhead, memory usage, and cryptographic strength while comparing them with existing solutions. This section highlights the proposed framework's lightweight nature and the prototype's secure, efficient operations under real-world conditions.

5.1.1 Qualitative Analysis

The design of iDSign addresses several essential areas to boost overall performance and security:

- *Lightweight Authentication:* iDSign's use of IBC removes the need for heavy certificate exchange, reducing communication and computational costs compared to traditional PKI methods. This efficiency is crucial for resource-limited devices.

- *SE Integration:* iDSign framework leverages the Secure Element's hardware acceleration and enclosed environment to securely store cryptographic credentials, accelerating the speed of signing operations while boosting security by rendering cryptographic keys tamper-resistant.

- *Adaptability to Multiple Scenarios:* iDSign's versatile design enables for deployment across many communication interfaces, such as low-energy NFC, Bluetooth, and TCP/IP Protocols, providing a seamless and intuitive user experience while ensuring compatibility with existing infrastructure and supporting both remote and in-person signing use cases.

The added features make iDSign especially appropriate for situations demanding rapid, efficient, and reliable digital signatures.

Table 5.1 compares protection against possible attacks between the proposed framework and other existing schemes. iDSign framework provides a novel combination of mutual authentication, MiTM attack protection, data privacy through robust authentication and integration of SE, key compromise protection with ephemeral keys, and the capability for Perfect Forward Secrecy. This comprehensive approach addresses a wider

Table 5.1: iDSign performance analysis: Qualitative analysis of proposed framework

Scheme	MA	MiTM	DP	SKS	KC	RA	PFS	IA
AuthPaper [5]	✓	✓	✓	*	✓	✓	*	✓
DSign [4]	×	×	×	*	×	*	*	×
BioSign [22]	✓	✓	*	✓	✓	✓	*	✓
PDFGuard [6]	×	×	✓	*	✓	*	*	✓
ECDoc [7]	×	×	✓	*	×	×	*	×
Proposed iDSign	✓	✓	✓	✓	✓	✓	✓	✓

MA: Mutual Auth., MiTM: Man-in-the-Middle Attack, DP: Data Privacy, SKS: Session Key Security, KC: Key Compromise, RA: Replay Attack, PFS: Perfect Forward Secrecy, IA: Impersonation Attack, ✓: Protected, ×: Vulnerable, *: Not Mentioned in the scheme

range of security vulnerabilities than many compared approaches, making iDSign particularly suitable for circumstances where robust authentication and secure communication are vital.

5.1.2 Quantitative Analysis

The quantitative analysis emphasizes two critical elements: communication and computation costs associated with iDSign and other digital signature schemes.

Communication Cost:

This analysis assumes that a 50KB document is submitted to either a central server or portal or directly to the Signer's device based on the type of solution used for generating the digital signature.

Table 5.2 compares the approximate communication cost for data communicated for a document with certain assumptions. It presents the time for the following:

- *MetaData (MD):* This column includes headers, IDs, and other control information for each communication step. The values above are rough approximations.
- *Mutual Authentication (MA):* This column shows the approximate Communication overhead necessary to verify both devices' identities.
- *Document (Optional) + Sign Received:* Some approaches re-transmit the document integrated with a digital signature. iDSign only sends signature records, minimizing overhead.

Table 5.2 displays anticipated data sizes for each communication stage, using typical values for cryptographic primitives and message overhead. It is vital to remember that the transmission cost can vary based on the chosen cryptographic techniques and the size of the signed documents. The iDSign framework leverages lightweight mutual authentication utilizing IBC, delivering substantial reductions in communication overhead compared to the standard certificate-based techniques. Certificate sizes range from hundreds of bytes to several kilobytes [10], generating severe delays during transmission

Table 5.2: iDSign performance analysis: Approximate communication cost comparison for document of size 50KB

Scheme	**Type of Service**	**MD (bytes)**	**MA (bytes)**	**Doc. (Opt) + Sign (bytes)**	**TCC (bytes)**
DSign [4]	Web-Based	200	*	51024	51224
BioSign [22]	Software	1072	200	51300	52572
PDFGuard [6]	Web-Based	100	*	50320	50420
ECDoc [7]	Web-based	100	*	50600	50700
iDSign	Flexible	20	950	300	1270

MD: MetaData, MA: Mutual Auth., Doc. (Opt) + Sign: Document (Optional) + Signature received, TCC: Total communication Cost, *: Not Mentioned in the Scheme

and authentication. In Table 5.2, the analysis indicates that iDSign incurs the lowest total communication cost, amounting to merely 1270 bytes, attributable to its lightweight mutual authentication approach and optimized signature transmission. This reduction achieves a 97% decrease in overhead when compared to DSign and a 75% decrease relative to PDFGuard, demonstrating its effectiveness in environments with constrained bandwidth. iDSign enhances efficiency in low-power environments by transmitting just signature records rather than complete documents, which reduces data transmission without compromising security.

Computation Cost:

Multiprecision Integer and Rational Arithmetic Library (MIRACL) [33] is an open-source library specifically created to simplify the process of testing cryptographic protocols. It provides a robust platform for performing complex arithmetic operations on large integers and rational numbers, essential in various cryptographic applications, such as public key cryptography. This work uses the methodology outlined by Yu et al. [34] based on MIRACL to derive the computation costs. Table 5.3 assumes the values of the following operations: C_{BP}: Computation of Bilinear Pairing ≈ 3.002 ms [34], C_{KH}: Computation of obtaining session key from Diffie-Hellman Algorithm ≈ 2 ms, C_{SED}: Computation of Symmetric Encryption/Decryption ≈ 0.012 ms (for example, Advanced Encryption Standard (AES) [35]), C_{H}: Computation of Hash Function ≈ 0.309 ms (for example, Secure Hash Algorithm (SHA-256) [36]), C_{QR}: Computation of QR generation ≈ 30 ms [37], C_{ET}: Computation of Euler's Totient Function ≈ 2 ms [6], C_{KE}: Computation of ElGamal Encryption ≈ 0.522 ms (It is an asymmetric encryption algorithm with similar computation cost), C_{AED}: Computation of Asymmetric Encryption/Decryption ≈ 0.522 ms (for example, RSA [38]).

Table 5.3 illustrates the computational costs associated with various schemes. Although iDSign has a slightly higher computational time of 12.67 ms due to bilinear pairing and SE operations, it surpasses other schemes in terms of security by integrating mutual authentication and forward secrecy. While ECDoc achieves the lowest computation cost, it lacks robust mutual authentication, making it less secure. The integration of SE in iDSign offers various benefits, such as hardware acceleration and enhanced security through

Table 5.3: iDSign performance analysis: Approximate computation cost comparison

Scheme	Computation Cost	Estimated Time(ms)
DSign [4]	$1C_{\mathrm{H}} + 1C_{\mathrm{QR}}$	30.30
BioSign [22]	$1C_{\mathrm{KH}} + 2C_{\mathrm{KE}} + 2C_{\mathrm{SED}} + 3C_{\mathrm{H}}$	3.99
PDFGuard [6]	$2C_{\mathrm{AED}} + 1C_{\mathrm{ET}} + 2C_{\mathrm{H}}$	3.66
ECDoc [7]	$2C_{\mathrm{AED}} + 2C_{\mathrm{H}}$	1.66
iDSign	$4C_{\mathrm{BP}} + 2C_{\mathrm{H}} + 4C_{\mathrm{SED}}$	12.67

C_{BP}: Computation of Bilinear Pairing, C_{KH}: Computation of obtaining session key from Diffie-Hellman Algo., C_{KE}: Computation of Elgamal Encryption, C_{SED}: Computation of Symmetric Encryption/Decryption, C_{H}: Computation of Hash Function, C_{QR}: Computation of QR generation, C_{ET}: Computation of Euler's Totient Function, C_{AED}: Computation of Asymmetric Encryption/Decryption

tamper-resistant storage, which optimize cryptographic operations and enhance overall performance. While iDSign demands the most computational power, it balances computational efficiency and improved security, justifying the additional resources required to ensure the integrity and trustworthiness of digital signatures.

5.2 Performance Analysis of the iHDoc mobile application prototype

This section evaluates the performance of the iHDoc prototype, designed by integrating the iDSign framework with low-energy NFC HCE technology for proximity-based digital signing of documents. To assess the efficiency, iHDoc's mutual authentication based on the iDSign framework is compared with RSA-based mutual authentication, ensuring a balanced comparison. Key metrics include timing and storage requirements, emphasizing the lightweight nature of iDSign framework for mobile use.

5.2.1 Timing Comparison of Core Components

Table 5.4 compares the approximate time taken for different components of iHDoc based on iDSign and RSA [39] when processing a document of 1MB size. The components analyzed include Mutual Authentication, Bluetooth File Transfer, Signature Generation, and Signature Integration.

- *Mutual Authentication:* iHDoc with iDSign achieves a faster mutual authentication time (0.479 sec) than RSA-based authentication (0.702 sec) due to the absence of certificate verification in iDSign, thus reducing computational overhead.

- *Bluetooth File Transfer:* The time for Bluetooth file transfer remains similar in both configurations, as Bluetooth communication is independent of the cryptographic method.

Table 5.4: iHDoc performance analysis: Approximate timing comparison of diff. components for document of size 1MB

Scheme	**Mutual Auth. (sec)**	**Bluetooth File Transfer (sec)**	**Signature Generation (sec)**	**Signature Integration (sec)**	**Total (sec)**
iHDoc based on iDSign	0.479	11.170	0.207	0.421	**12.277**
RSA [39]	0.778	11.19	0.232	0.428	**12.628**

Table 5.5: iHDoc performance analysis: Approximate credentials storage requirements

Scheme	**Public Key Size (bytes)**	**Private Key Size (bytes)**	**Certificate Size (bytes)**	**Symmetric Session Key Size (bytes)**	**Ppub (bytes)**	**P (bytes)**	**Total (bytes)**
iHDoc based on iDSign	128	20	-	128	128	128	**532**
RSA [39]	256	256	1024	128	-	-	**1664**

Ppub: Master Public Key, P: Generator in iDSign

- *Signature Generation:* iHDoc with iDSign also performs faster signature generation (0.207 sec) compared to RSA-based approach (0.232 sec), reflecting the efficiency of the Hess IBS scheme used in iDSign framework.

- *Signature Integration:* The time required for signature integration is slightly lower in iHDoc with iDSign (0.421 sec) compared to RSA (0.428 sec), benefiting from the streamlined IBC approach used in iDSign framework.

The results in Table 5.4 demonstrate that iHDoc based on iDSign offers reduced processing times in the mutual authentication and signature generation phases, emphasizing the advantages of lightweight cryptography for digitally signing documents on mobile devices.

5.2.2 Storage Requirements

Table 5.5 compares storage requirements for credentials and cryptographic keys in the iHDoc prototype based on the iDSign framework versus RSA [39] certificate-based keys. The RSA keys requires more storage due to the inclusion of public/private key pairs and certificates, whereas iHDoc based on the iDSign framework utilizes IBC which significantly reduces storage requirements by eliminating certificates.

- *RSA Key and Certificate Storage:* In RSA-based approach, each device requires storage for RSA public/private keys and certificates and requires approximately 1664 bytes for storage.

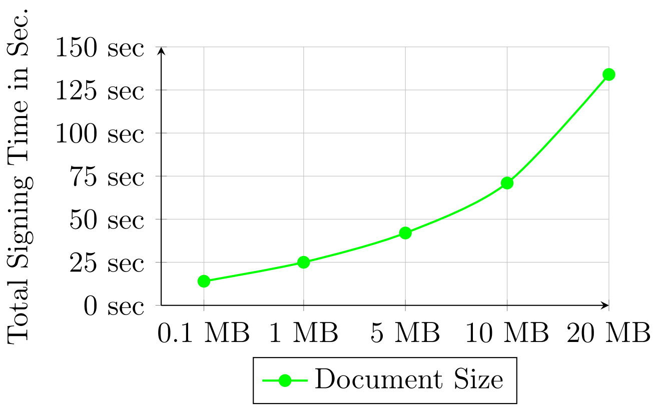


Figure 5.1: iHDoc performance analysis: Relationship between total signing time vs doc. size

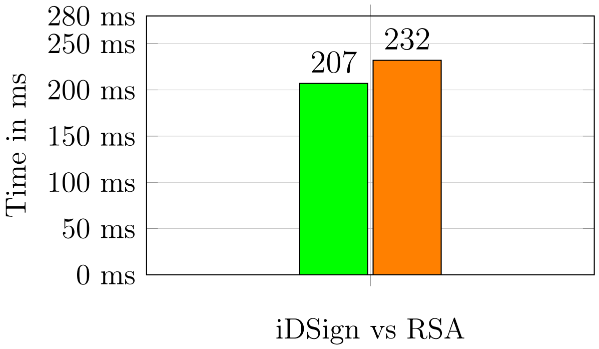


Figure 5.2: iHDoc performance analysis: Comparison of signature generation time with RSA

- *iHDoc Key Storage:* iHDoc due to iDSign integration eliminates certificates, reducing storage needs to only 532 bytes for storing public and private keys, the master public key (Ppub), and the generator (P). Hence, iDSign framework is more suitable for mobile devices with limited storage capacity.

As shown in Table 5.5, the total storage requirement for iHDoc based on iDSign is 532, significantly lower than the 1664 bytes required for the RSA-based approach. This reduction in storage requirements is critical for mobile devices, where storage space is often limited.

5.2.3 Document Size vs. Signing Time

Figure 5.1 illustrates the relationship between document size and total signing time in iHDoc based on iDSign. As document size increases, the time taken for signing also increases, largely due to Bluetooth transfer time. For smaller documents (e.g., 0.1 MB), the signing time is relatively low (around 14 seconds), but for larger documents (e.g., 20 MB), the signing time increases substantially (up to 134 seconds). This graph highlights iHDoc's scalability and the impact of document size on overall performance.

5.2.4 Signature Generation Time Comparison

Figure 5.2 compares signature generation time between iHDoc based on iDSign and RSA-based approach. The signature generation time in iDSign is approximately 207 ms, while RSA requires 232 ms. This performance difference highlights the efficiency of the Hess IBS scheme used in iDSign, making it well-suited for mobile environments with limited computational resources.

Chapter 6

Conclusion and Future Work

This thesis introduces the novel iDSign framework as a secure, lightweight, and flexible solution for the digital signature of documents, compatible with resource-constrained devices. It supports multiple communication interfaces such as NFC, Bluetooth, and TCP/IP. The framework addresses the significant shortcomings of current PKI-based systems, such as complex certificate management, which are prone to several attacks like MiTM and rely on continuous network connectivity. iDSign utilizes IBC and a modified lightweight iTLS protocol to enable certificate-free mutual authentication and efficient cryptographic operations. To validate the iDSign framework in a practical context, the iHDoc mobile application based on iDSign is designed and implemented using NFC HCE technology to enable secure proximity-based communication and mitigate vulnerabilities like MiTM attacks. The application enables tap-to-sign architecture and operates in offline contexts. Performance analysis compares the iHDoc based on iDSign with RSA-based mutual authentication. Results indicate that the iDSign framework significantly reduces mutual authentication and digital signing time compared to the RSA-based approach, thus making it suitable for resource-limited and offline environments.

Future work focuses on exploring the integration of the iDSign framework with smart cards and remote devices. Smart cards offer a secure, portable medium for storing cryptographic keys, enabling digital signing by tapping the card on an NFC-enabled device. Additionally, expanding iDSign to support secure document signing on remote devices will enhance its versatility and enable deployment in enterprise, academic, and governmental workflows, where remote interactions require scalable and robust cryptographic performance.

Bibliography

[1] P. Kaur and N. Arora, “A Comprehensive Study of Cryptography and Digital Signature,” *International Journal of Science, Engineering and Computer Technology*, vol. 5, no. 1, p. 1, 2015.

[2] S. C. Alliance *et al.*, “Host Card Emulation (HCE) 101,” *A Smart Card Alliance Mobile and NFC Council White Paper*, 2014.

[3] CheapSSLsecurity, “Digital signature vs digital certificate: The difference explained,” 2023, accessed: 2025-04-10. [Online]. Available: https://cheapsslsecurity.com/blog/digital-signature-vs-digital-certificate-the-difference-explained/

[4] G. Saha, “DSign Digital Signature System for Paperless Operation,” in *2017 International Conference on Communication and Signal Processing (ICCSP)*. IEEE, 2017, pp. 0324–0328.

[5] C. M. Li, P. Hu, and W. C. Lau, “AuthPaper: Protecting Paper-based Documents and Credentials using Authenticated 2D Barcodes,” in *2015 IEEE International Conference on Communications (ICC)*. IEEE, 2015, pp. 7400–7406.

[6] A. Sharif, D. S. Ginting, and A. D. Dias, “Securing the Integrity of PDF Files using RSA Digital Signature and SHA-3 Hash Function,” in *2021 International Conference on Data Science, Artificial Intelligence, and Business Analytics (DATABIA)*. IEEE, 2021, pp. 154–159.

[7] M. R. Ramadhan, S. Mandala, and F. A. Yulianto, “Analysis and Implementation of Digital Signature Algorithm in PDF Document,” in *2023 11th International Conference on Information and Communication Technology (ICoICT)*. IEEE, 2023, pp. 11–16.

[8] Y. s. Alslman and A. A. Taleb, “Exchanging Digital Documents Using Blockchain Technology,” in *2021 International Conference on Electrical, Communication, and Computer Engineering (ICECCE)*, 2021, pp. 1–6.

[9] K. Pal and C. Kumar, “QR Code based Smart Document implementation using Blockchain and Digital Signature,” in *Data Management, Analytics and Innovation: Proceedings of ICDMAI 2020, Volume 1*. Springer, 2021, pp. 449–465.

[10] P. Li, J. Su, and X. Wang, “iTLS: Lightweight Transport-Layer Security Protocol for IoT With Minimal Latency and Perfect Forward Secrecy,” *IEEE Internet of Things Journal*, vol. 7, no. 8, pp. 6828–6841, 2020.

[11] A. Shamir, “Identity-based cryptosystems and signature schemes,” in *Advances in Cryptology, CRYPTO 1984*, ser. Lecture Notes in Computer Science, G. R. Blakley and D. Chaum, Eds., vol. 196. Springer, Berlin, Heidelberg, 1985, pp. 47–53. [Online]. Available: https://doi.org/10.1007/3-540-39568-7_5

[12] F. Hess, “Efficient identity based signature schemes based on pairings,” in *Selected Areas in Cryptography: 9th Annual International Workshop, SAC 2002, St. John’s, Newfoundland, Canada, August 15–16, 2002, Revised Papers*, ser. Lecture Notes in Computer Science, vol. 2595. Springer, Berlin, Heidelberg, 2003, pp. 310–324. [Online]. Available: https://doi.org/10.1007/3-540-36492-7_20

[13] S. Micallef and I. Konstantinos Markantonakis, “Mobile payments using Host Card Emulation with NFC: security aspects and limitations,” *Royal Holloway Information Security thesis series*, 2018.

[14] T. Molnar and D. Dreiner, “The Campuscard App–A secure solution to the NFC problem,” *Proceedings of European University*, vol. 95, pp. 216–221, 2023.

[15] “ISO/IEC 14443: NFC HCE Overview,” last accessed: August 25, 2024. [Online]. Available: https://developer.android.com/develop/connectivity/nfc/hce

[16] E. Rescorla, “The Transport Layer Security (TLS) Protocol Version 1.3,” Tech. Rep., 2018.

[17] D. Boneh and M. Franklin, “Identity-Based Encryption from the Weil Pairing,” in *Annual international cryptology conference.* Springer, 2001, pp. 213–229.

[18] Q. Yuan and S. Li, “A New Efficient ID-Based Authenticated Key Agreement Protocol,” *Cryptology ePrint Archive*, 2005.

[19] L. Chen, Z. Cheng, and N. P. Smart, “Identity-based Key Agreement Protocols from Pairings,” *International Journal of Information Security*, vol. 6, pp. 213–241, 2007.

[20] D. Sethia, D. Gupta, and H. Saran, “NFC Secure Element-based Mutual Authentication and Attestation for IoT access,” *IEEE Transactions on Consumer Electronics*, vol. 64, no. 4, pp. 470–479, 2018.

[21] “Zxing, open-source, cross-platform 2d Barcode image processing library,” Last visited in March 2014, Last accessed: March 5, 2024. [Online]. Available: https://github.com/zxinglzxingl

[22] S. Saxena and D. Anand, “A Novel Digital Signature Algorithm based on Biometric Hash,” *International Journal of Computer Network and Information Security*, vol. 9, no. 1, p. 12, 2017.

[23] National Institute of Standards and Technology (NIST), “Hardware security modules: Protecting cryptographic keys and operations,” https://csrc.nist.gov/glossary/term/hardware_security_module, last accessed: October 19, 2024.

[24] ——, “Key management guidelines: Key rotation and revocation,” https://doi.org/10.6028/NIST.SP.800-57pt1r5, Special Publication 800-57 Part 1, 2020, last accessed: October 19, 2024.

[25] H. Krawczyk and P. Eronen, "HMAC-based extract-and-expand key derivation function (HKDF)," Tech. Rep., 2010, Last accessed: March 20, 2024. [Online]. Available: https://www.rfc-editor.org/rfc/rfc5869.html

[26] G. Dia, "Java pairing-based cryptography (jpbc)," http://gas.dia.unisa.it/projects/jpbc/, last accessed: November 7, 2024.

[27] Adobe, "Digital signatures with adobe sign," 2024, last accessed: November 7, 2024. [Online]. Available: https://helpx.adobe.com/in/sign/using/digital-signatures.html

[28] Visma, "Visma sign," 2024, last accessed: November 7, 2024. [Online]. Available: https://vismasign.fi/english/

[29] DocuSign, "Electronic signature," 2024, last accessed: November 7, 2024. [Online]. Available: https://www.docusign.com/products/electronic-signature

[30] BouncyCastle, "The bouncycastle crypto apis," https://www.bouncycastle.org/, last accessed: November 7, 2024.

[31] iTextPDF, "itextpdf," https://itextpdf.com, last accessed: November 4, 2024.

[32] Spring, "Spring framework," https://spring.io/, last accessed: November 7, 2024.

[33] Dhilip, "Miracl cryptographic sdk: Multiprecision integer and rational arithmetic cryptographic library," https://github.com/miracl/MIRACL, 2024, last accessed: August 26, 2024.

[34] S. Yu, A. K. Das, Y. Park, and P. Lorenz, "Slap-iod: Secure and lightweight authentication protocol using physical unclonable functions for internet of drones in smart city environments," *IEEE Transactions on Vehicular Technology*, vol. 71, no. 10, pp. 10 374–10 388, 2022. [Online]. Available: https://doi.org/10.1109/TVT.2022.3188769

[35] FIPS PUB, "Advanced encryption standard," https://csrc.nist.gov/files/pubs/fips/197/final/docs/fips-197.pdf, 2001, last accessed: August 26, 2024.

[36] F. PUB, "Secure hash standard," https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.180-4.pdf, 2015, last accessed: August 26, 2024.

[37] alf.io, "alf.io issue 698," https://github.com/alfio-event/alf.io/issues/698, Last Accessed: April 7, 2024.

[38] S. Azad, "Practical cryptography," https://www.oreilly.com/library/view/practical-cryptography/9781482228892/ch08.html, 2014, last accessed: August 26, 2024.

[39] H. Jaiswal, R. Mishra, H. Sharma, and D. Sethia, "SecureXfer - NFC HCE Mobile Framework using RSA for Secure Digital Transfer and Signing," 2025, under submission to the IEEE International Conference on Next Generation Information System Engineering 2025.

DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Shahbad Daulatpur, Main Bawana Road, Delhi-42

PLAGIARISM VERIFICATION

Title of the Thesis iDSign : A Secure Lightweight NFC-Based Framework to Tap and Digitally Sign Documents

Total Pages 54 Name of the Scholar Himanshu Sharma

Supervisor (s)

(1) Dr. Divyashikha Sethia

(2)

(3)

Department of Software Engineering

This is to report that the above thesis was scanned for similarity detection. Process and outcome is given below:

Software used: Turnitin Similarity Index: 11%, Total Word Count: 15,559

Date: 18 May 2025

Candidate's Signature

Divya.

May 21 2025

Signature of Supervisor(s)

Himanshu_Mtech_thesis-iDSign.pdf

Delhi Technological University

Document Details

Submission ID

trn:oid:::27535:96534335

Submission Date

May 18, 2025, 9:38 PM GMT+5:30

Download Date

May 18, 2025, 9:41 PM GMT+5:30

File Name

Himanshu_Mtech_thesis-iDSign.pdf

File Size

3.2 MB

54 Pages

15,559 Words

91,913 Characters

11% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

Devya.

May 20, 2025

Filtered from the Report

- Bibliography
- Quoted Text
- Cited Text
- Small Matches (less than 8 words)

Match Groups

134 Not Cited or Quoted 11%
Matches with neither in-text citation nor quotation marks

0 Missing Quotations 0%
Matches that are still very similar to source material

0 Missing Citation 0%
Matches that have quotation marks, but no in-text citation

0 Cited and Quoted 0%
Matches with in-text citation present, but no quotation marks

Top Sources

8%	Internet sources
6%	Publications
8%	Submitted works (Student Papers)

Integrity Flags

0 Integrity Flags for Review

No suspicious text manipulations found.

Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we notice something strange, we flag it for you to review.

A Flag is not necessarily an indicator of a problem. However, we'd recommend you focus your attention there for further review.

Match Groups

- **134**Not Cited or Quoted 11%
  Matches with neither in-text citation nor quotation marks
- **0** Missing Quotations 0%
  Matches that are still very similar to source material
- **0** Missing Citation 0%
  Matches that have quotation marks, but no in-text citation
- **0** Cited and Quoted 0%
  Matches with in-text citation present, but no quotation marks

Top Sources

8%	Internet sources
6%	Publications
8%	Submitted works (Student Papers)

Top Sources

The sources with the highest number of matches within the submission. Overlapping sources will not be displayed.

1	Internet	**dspace.dtu.ac.in:8080**	**2%**
2	Publication	**Pengkun Li, Jinshu Su, Xiaofeng Wang. "iTLS: Lightweight Transport-Layer Securit...**	**<1%**
3	Publication	**Kang, Bao Yuan. "New Types of Verii¬ably Encrypted Signature Schemes", Advanc...**	**<1%**
4	Submitted works	**Delhi Technological University on 2019-06-28**	**<1%**
5	Submitted works	**Coventry University on 2023-04-18**	**<1%**
6	Internet	**export.arxiv.org**	**<1%**
7	Internet	**link.springer.com**	**<1%**
8	Submitted works	**University of Bristol on 2007-06-05**	**<1%**
9	Internet	**m.moam.info**	**<1%**
10	Internet	**www.globalsign.com**	**<1%**

11	Internet	hdl.handle.net	<1%
12	Submitted works	University of Alabama at Birmingham on 2024-09-23	<1%
13	Internet	dokumen.pub	<1%
14	Submitted works	Higher Education Commission Pakistan on 2015-09-11	<1%
15	Internet	cyber-gateway.net	<1%
16	Submitted works	Indian Institute of Technology, Kanpur on 2009-07-08	<1%
17	Submitted works	University of Bristol on 2015-10-09	<1%
18	Internet	pdffox.com	<1%
19	Internet	www.frontiersin.org	<1%
20	Submitted works	Centre for Distance and Online Education Galgotias University on 2024-12-29	<1%
21	Submitted works	Higher Education Commission Pakistan on 2013-01-10	<1%
22	Publication	Huan Zhou, Xiaofeng Wang, Jinshu Su. "An efficient identity-based key agreemen...	<1%
23	Submitted works	Oxford Brookes University on 2023-05-12	<1%
24	Submitted works	Universiti Sains Malaysia on 2012-02-08	<1%

25	Submitted works	Cranfield University on 2023-08-16	<1%
26	Publication	Hadeal Abdulaziz Al Hamid, Sk Md Mizanur Rahman, M. Shamim Hossain, Ahmad ...	<1%
27	Internet	openaccess.altinbas.edu.tr	<1%
28	Internet	www.diva-portal.org	<1%
29	Submitted works	Asia Pacific University College of Technology and Innovation (UCTI) on 2024-06-12	<1%
30	Submitted works	Indian Institute of Management, Indore on 2015-03-30	<1%
31	Publication	Junjian Chen. "IBE Applied to Identity Authentication for Object-Based Storage Sy...	<1%
32	Submitted works	Queen's University of Belfast on 2024-07-08	<1%
33	Internet	ceur-ws.org	<1%
34	Internet	www.securetechalliance.org	<1%
35	Publication	Aina Sui, Bo Peng, Yongbin Wang. "Self-certified identity-based signcryption in D...	<1%
36	Submitted works	Colorado State University, Global Campus on 2024-03-03	<1%
37	Submitted works	ESoft Metro Campus, Sri Lanka on 2025-04-04	<1%
38	Submitted works	Indian Institute of Technology, Madras on 2016-03-09	<1%

39	Submitted works	University of Wales Swansea on 2025-03-17	<1%
40	Internet	researchspace.ukzn.ac.za	<1%
41	Internet	www.geeky-gadgets.com	<1%
42	Submitted works	Higher Education Commission Pakistan on 2016-09-14	<1%
43	Publication	Renu Mary Daniel, Elijah Blessing Rajsingh, Salaja Silas. "An efficient eCK secure i...	<1%
44	Submitted works	University of Queensland on 2015-08-27	<1%
45	Internet	ir.unimas.my	<1%
46	Internet	koreascience.kr	<1%
47	Submitted works	Kingston University on 2025-04-10	<1%
48	Submitted works	Universiti Tunku Abdul Rahman on 2016-03-07	<1%
49	Internet	arxiv.org	<1%
50	Internet	csrc.nist.gov	<1%
51	Internet	downloads.hindawi.com	<1%
52	Internet	icact.org	<1%

53	Internet	
rainbow.essi.fr		<1%
54	Publication	
"Chapter 300246 Data Breach Impacts", Springer Science and Business Media LLC...		<1%
55	Submitted works	
Ajman University of Science and Technology on 2014-06-02		<1%
56	Submitted works	
La Trobe University on 2007-06-26		<1%
57	Submitted works	
Pepperdine University on 2009-12-08		<1%
58	Submitted works	
Rocky Mountain High School on 2019-01-20		<1%
59	Submitted works	
University of St Andrews on 2025-04-25		<1%
60	Publication	
Zhiyan Xu, Libing Wu, Muhammad Khurram Khan, Kim-Kwang Raymond Choo, De...		<1%
61	Internet	
core.ac.uk		<1%
62	Internet	
fdocuments.in		<1%
63	Publication	
"Smart Cards, Tokens, Security and Applications", Springer Science and Business ...		<1%
64	Publication	
Diego Galar Pascual, Pasquale Daponte, Uday Kumar. "Handbook of Industry 4.0 ...		<1%
65	Publication	
SK Hafizul Islam, G.P. Biswas. "A pairing-free identity-based two-party authentica...		<1%
66	Submitted works	
University of New South Wales on 2023-06-08		<1%

67	Submitted works	University of Wales, Lampeter on 2023-09-14	<1%
68	Internet	apps.dtic.mil	<1%
69	Internet	art.tools.ietf.org	<1%
70	Internet	artemis.cslab.ece.ntua.gr:8080	<1%
71	Internet	lucris.lub.lu.se	<1%
72	Publication	"Data Management, Analytics and Innovation", Springer Science and Business Me...	<1%
73	Publication	"Information Security and Cryptology – ICISC 2004", Springer Science and Busines...	<1%
74	Submitted works	Asia Pacific University College of Technology and Innovation (UCTI) on 2023-11-08	<1%
75	Publication	David T. Gray. "Pseudonym management using mediated identity-based cryptogr...	<1%
76	Submitted works	Delhi Technological University on 2019-01-23	<1%
77	Publication	Jana, Amit. "Cryptanalysis of Selected SPN and NLFSR-Based Symmetric-Key Ciphe...	<1%
78	Submitted works	Kingston University on 2022-11-25	<1%
79	Submitted works	Liverpool Community College on 2021-10-29	<1%
80	Publication	Meijiao Duan, Jing Xu, Dengguo Feng. "Efficient identity-based strong designated ...	<1%

81	Submitted works	
	North South University on 2023-08-04	<1%
82	Submitted works	
	Queen Mary and Westfield College on 2025-05-06	<1%
83	Submitted works	
	Royal Holloway and Bedford New College on 2007-04-19	<1%
84	Submitted works	
	Royal Holloway and Bedford New College on 2012-09-13	<1%
85	Publication	
	Subhabrata Rana, Fatemeh Khoda Parast, Brett Kelly, Yang Wang, Kenneth B. Ken...	<1%
86	Submitted works	
	Texas A&M University, Central Texas on 2025-04-20	<1%
87	Submitted works	
	University of Bristol on 2005-10-10	<1%
88	Submitted works	
	University of Hong Kong on 2011-06-27	<1%
89	Submitted works	
	University of Melbourne on 2023-10-22	<1%
90	Submitted works	
	University of Nebraska at Omaha on 2024-12-14	<1%
91	Submitted works	
	VIT University on 2025-03-25	<1%
92	Publication	
	Xinyi Huang. "Certificateless Signature Revisited", Lecture Notes in Computer Sci...	<1%
93	Internet	
	cybok.org	<1%
94	Internet	
	ijns.jalaxy.com.tw	<1%

95	Internet	journals.mesopotamian.press	<1%
96	Internet	repository.ubn.ru.nl	<1%
97	Internet	vdoc.pub	<1%
98	Internet	www2.ia-engineers.org	<1%

Himanshu_Mtech_Thesis_iDSign.pdf

Delhi Technological University

Document Details

Submission ID

trn:oid:::27535:97207381

Submission Date

May 22, 2025, 5:57 PM GMT+5:30

Download Date

May 22, 2025, 6:03 PM GMT+5:30

File Name

Himanshu_Mtech_Thesis_iDSign.pdf

File Size

3.2 MB

54 Pages

15,729 Words

91,617 Characters

0% detected as AI

The percentage indicates the combined amount of likely AI-generated text as well as likely AI-generated text that was also likely AI-paraphrased.

Caution: Review required.

It is essential to understand the limitations of AI detection before making decisions about a student's work. We encourage you to learn more about Turnitin's AI detection capabilities before using the tool.

Devya.

may 21 2025

Detection Groups

0 AI-generated only 0%
Likely AI-generated text from a large-language model.

0 AI-generated text that was AI-paraphrased 0%
Likely AI-generated text that was likely revised using an AI-paraphrase tool or word spinner.

Disclaimer
Our AI writing assessment is designed to help educators identify text that might be prepared by a generative AI tool. Our AI writing assessment may not always be accurate (it may misidentify writing that is likely AI generated as AI generated and AI paraphrased or likely AI generated and AI paraphrased writing as only AI generated) so it should not be used as the sole basis for adverse actions against a student. It takes further scrutiny and human judgment in conjunction with an organization's application of its specific academic policies to determine whether any academic misconduct has occurred.

Frequently Asked Questions

How should I interpret Turnitin's AI writing percentage and false positives?
The percentage shown in the AI writing report is the amount of qualifying text within the submission that Turnitin's AI writing detection model determines was either likely AI-generated text from a large-language model or likely AI-generated text that was likely revised using an AI-paraphrase tool or word spinner.

False positives (incorrectly flagging human-written text as AI-generated) are a possibility in AI models.

AI detection scores under 20%, which we do not surface in new reports, have a higher likelihood of false positives. To reduce the likelihood of misinterpretation, no score or highlights are attributed and are indicated with an asterisk in the report (*%).

The AI writing percentage should not be the sole basis to determine whether misconduct has occurred. The reviewer/instructor should use the percentage as a means to start a formative conversation with their student and/or use it to examine the submitted assignment in accordance with their school's policies.

What does 'qualifying text' mean?
Our model only processes qualifying text in the form of long-form writing. Long-form writing means individual sentences contained in paragraphs that make up a longer piece of written work, such as an essay, a dissertation, or an article, etc. Qualifying text that has been determined to be likely AI-generated will be highlighted in cyan in the submission, and likely AI-generated and then likely AI-paraphrased will be highlighted purple.

Non-qualifying text, such as bullet points, annotated bibliographies, etc., will not be processed and can create disparity between the submission highlights and the percentage shown.

DECLARATION

I hereby certify that the work which is presented in the Major Project-II entitled **iDSign: A Secure Lightweight NFC-Based Framework to Tap and Digitally Sign Documents** in fulfillment of the requirement for the award of the Degree of Master of Technology in Software Engineering and submitted to the Department of Software Engineering, Delhi Technological University, Delhi is an authentic record of my own, carried out during a period from January to May 2025 under the supervision of **Dr. Divyashikha Sethia**.

The mater presented in this report has not been submitted by me for the award of any other degree of this or any other Institute/University. The work has been published/accepted/communicated in SCI/SCI expanded/SSCI/Scopus indexed journal OR peer reviewed Scopus indexed conference with the following details.

Title of the Paper: **iDSign: Secure Lightweight Digital Document Signatures Framework with Mutual Authentication and Identity-Based Cryptography**
Author names (in sequence): Himanshu Sharma, Dr. Divyashikha Sethia
Name of Conference/Journal: 6th INTERNATIONAL CONFERENCE ON COMMUNICATION AND INTELLIGENT SYSTEMS (ICCIS 2024)
Conference dates with venue: 08-09th November 2024, Bhopal, Madhya Pradesh, India
Status of paper (Communicated/Accepted/Published): Accepted
Date of paper communication: August 31, 2024
Date of paper acceptance: October 11, 2024
Date of paper publication: N/A

Himanshu Sharma
Roll No. 23/SWE/18

Student Roll No., Name and Signature

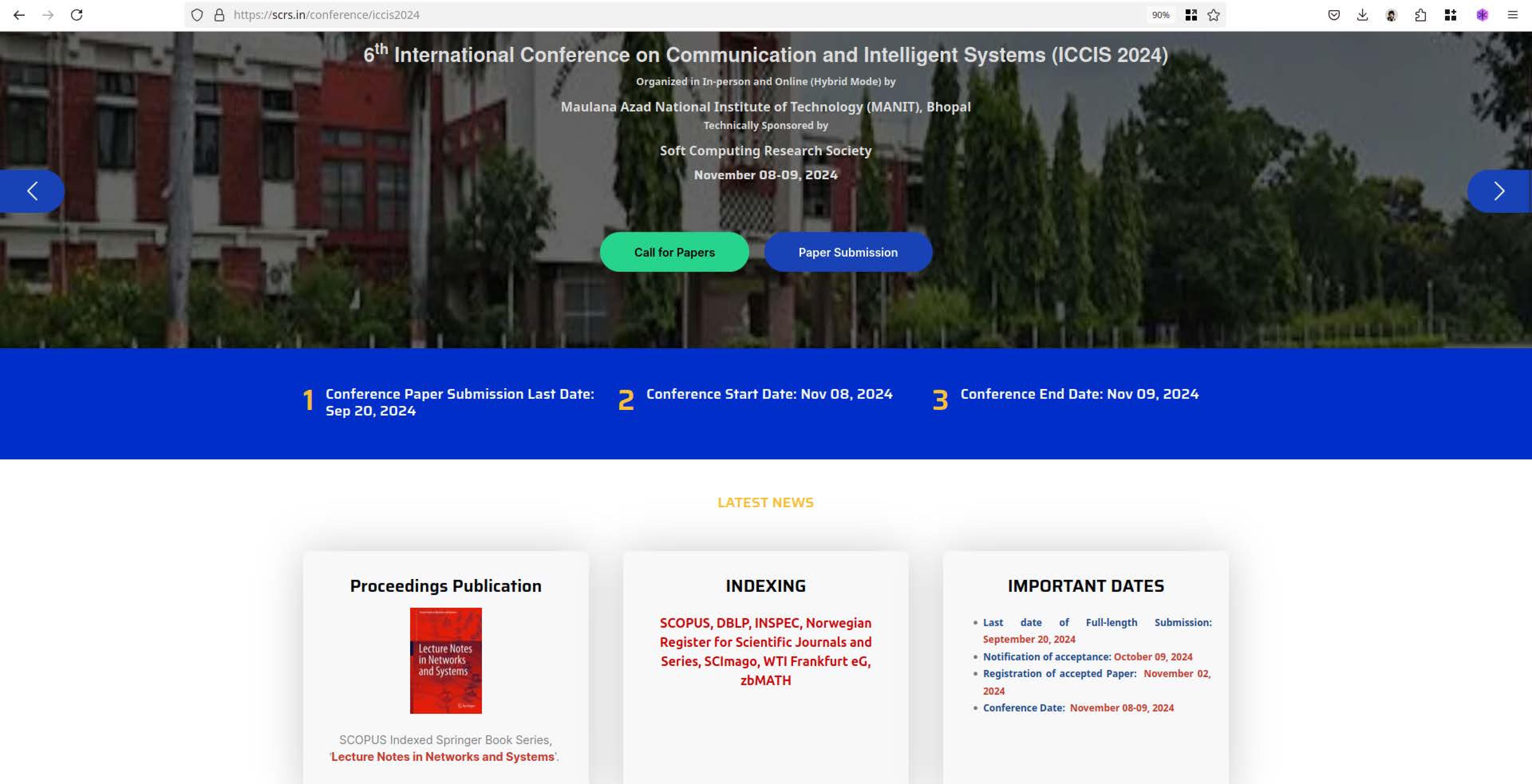
SUPERVISOR CERTIFICATE

To the best of my knowledge, the above work has not been submitted in part or full for any Degree or Diploma to this University or elsewhere. I further certify that the publication and indexing information given by the students is correct.

Divya.

Date: May 19, 2025 **Dr. Divyashikha Sethia**

Place: New Delhi **Supervisor Name and Signature**



6th International Conference on Communication and Intelligent Systems (ICCIS 2024)

Organized in In-person and Online (Hybrid Mode) by

Maulana Azad National Institute of Technology (MANIT), Bhopal

Technically Sponsored by

Soft Computing Research Society

November 08-09, 2024

1 Conference Paper Submission Last Date: Sep 20, 2024

2 Conference Start Date: Nov 08, 2024

3 Conference End Date: Nov 09, 2024

LATEST NEWS

Proceedings Publication

SCOPUS Indexed Springer Book Series, 'Lecture Notes in Networks and Systems'.

INDEXING

SCOPUS, DBLP, INSPEC, Norwegian Register for Scientific Journals and Series, SCImago, WTI Frankfurt eG, zbMATH

IMPORTANT DATES

- Last date of Full-length Submission: September 20, 2024
- Notification of acceptance: October 09, 2024
- Registration of accepted Paper: November 02, 2024
- Conference Date: November 08-09, 2024

23/SWE/18 HIMANSHU SHARMA <himanshusharma_23swe18@dtu.ac.in>

International Conference on Intelligent control, Computing and Communications-2025 : Submission (60) has been created.

Microsoft CMT <email@msr-cmt.org> Sun, Oct 20, 2024 at 11:59 PM
Reply-To: Microsoft CMT - Do Not Reply <noreply@msr-cmt.org>
To: himanshusharma_23swe18@dtu.ac.in

Hello,

The following submission has been created.

Track Name: Communications and Networking

Paper ID: 60

Paper Title: iHDoc: Secure Mobile-based Digital Document Signatures Framework using Host Card Emulation Tap and Lightweight Mutual Authentication

Abstract:
This paper introduces the prototype implementation and assessment of iHDoc, a lightweight framework for secure digital document signing on mobile devices, which is based on the previously proposed iDSign framework (accepted, to be published). The original study laid the theoretical foundation for a versatile framework for safe digital document signing utilizing identity-based cryptography and lightweight mutual authentication through a modified lightweight identity-based Transport Layer Security (iTLS) protocol. This paper builds upon the prior research by employing the framework on mobile devices through native Android development and utilizing NFC-based host card emulation (HCE) for secure session initiation via NFC tap and Bluetooth for effective data transfer. The prototype utilizes the Hess Identity-Based Signature (IBS) technique for speedy and efficient signing, making it suitable for resource-constrained devices. The comparative performance assessment with the RSA-based counterpart, iHDoc_RSA, demonstrates that the iHDoc framework attains faster execution times and decreased storage requirements. The iText library facilitates PDF manipulation and enables the smooth integration of digital signatures. The findings indicate that iHDoc provides both security and performance benefits, making it an effective solution for mobile document signing. Future endeavours will focus on real-world deployment and improved compatibility among platforms.

Created on: Sun, 20 Oct 2024 18:29:30 GMT

Last Modified: Sun, 20 Oct 2024 18:29:30 GMT

Authors:

- himanshusharma_23swe18@dtu.ac.in (Primary)

Secondary Subject Areas: Not Entered

Submission Files: Not Uploaded

Submission Questions Response: Not Entered

Thanks,
CMT team.

To stop receiving conference emails, you can check the 'Do not send me conference email' box from your User Profile.

Microsoft respects your privacy. To learn more, please read our Privacy Statement.

Microsoft Corporation
One Microsoft Way
Redmond, WA 98052

23/SWE/18 HIMANSHU SHARMA <himanshusharma_23swe18@dtu.ac.in>

ICCIS 2024: Notification of your paper ID 391: Acceptance

Microsoft CMT <email@msr-cmt.org> Thu, Oct 10, 2024 at 10:58 PM
Reply-To: ICCIS SCRS <scrs.iccis@gmail.com>
To: Himanshu Sharma <himanshusharma_23swe18@dtu.ac.in>

Dear Himanshu Sharma,

Greetings!

Thank you for submitting your research article to the 6th International Conference on Communication and Intelligent Systems (ICCIS 2024) to be held on November 08-09, 2024 at Maulana Azad National Institute of Technology (MANIT), Bhopal, India in Hybrid Mode.

We are pleased to inform you that based on reviewers’ comments, your paper titled "iDSign: Secure Lightweight Digital Document Signatures Framework with Mutual Authentication and Identity-Based Cryptography" has been accepted for presentation during ICCIS 2024, and publication in the proceedings to be published in Scopus-indexed Springer Book Series "Lecture Notes in Networks and Systems" subject to the condition that you submit a revised version as per the comments, available at Authors CMT account. It is also required that you prepare a response to each comment from the reviewer and upload it as a separate file along with the revised paper.

The similarity index in the final paper must be less than 20%. Please note that the high plagiarism and any kind of multiple submissions of this paper to other conferences or journals will lead to rejection at any stage. Please note that the publisher, i.e. Springer Nature may ask for any other changes during the production which are supposed to be implemented. The publisher has the final right to exclude the paper from the proceedings if they found it unsuitable for publication.

Please carry out the steps to submit the camera-ready paper and online registration (Under "Regular Author" Category) as per the instructions available at

https://scrs.in/conference/iccis2024/page/Camera_Ready_Paper_Submission

In order to register in the SCRS member category (subsidized registration fees), you can first become a member at https://www.scrs.in/register and then register for the conference OR you may register as a Regular Author Category.

Please note that the Last date for submission of the camera-ready paper, payment of the registration fee, and online registration is October 24, 2024.

Feel free to write to the "General Chairs, ICCIS 2024" at scrs.iccis@gmail.com, should you have any questions or concerns. Please remember to always include your Paper ID- 391, whenever inquiring about your paper.

Looking forward to meeting you during the conference.

With Regards
Team ICCIS 2024 (https://scrs.in/conference/iccis2024)
Maulana Azad National Institute of Technology (MANIT), Bhopal, India

To stop receiving conference emails, you can check the 'Do not send me conference email' box from your User Profile.

Microsoft respects your privacy. To learn more, please read our Privacy Statement.

Microsoft Corporation
One Microsoft Way
Redmond, WA 98052

CONFERENCE SERVICES

Gurunanak Nagar, Gwalior,
Madhya Pradesh, India - 474012
Phone: +91 76928 04154
Email: info@confservices.in

TAX-INVOICE

Conference Services Website: https://confservices.in/ GSTIN: 23BGUPS7760Q1Z6	**Invoice Date**	24 October 2024
	Invoice Number	CS/OCT/2024/193

BILL ISSUED TO
Himanshu Sharma A-15 Ganga Vihar Delhi -110094 sdothimanshu@gmail.com **For Paper ID**: 391 **Paper Title**: iDSign: Secure Lightweight Digital Document Signatures Framework with Mutual Authentication and Identity-Based Cryptography

DESCRIPTION	
Event Name	6th International Conference on Communication and Intelligent Systems (ICCIS 2024) https://scrs.in/conference/iccis2024
Event Dates	08 Nov 2024 - 09 Nov 2024
Category of Registration	Regular Author
Registration Fee	INR 9000
Extra Page Charges	INR 4000
GST (18%)	INR 2340
Total	**INR 15340.00** **(FIFTEEN THOUSANDS THREE HUNDRED FORTY RUPEES ZERO PAISE)**

Payment Transaction Id	Date Time	Mode of Payment
ICCIS2024-391-20241024210620000056347421779175428	24 Oct 2024 23:07:29	Paytm

Note: Whether tax is payable under reverse charge - No

This is a computer generated invoice and needs no signature.

Licence to Publish Proceedings Papers

SPRINGER NATURE

Licensee	Springer Nature Singapore Pte Ltd.	(the 'Licensee')
Title of the Proceedings Volume/Edited Book or Conference Name:	Communication and Intelligent Systems: Proceedings of ICCIS 2024, Volume 3	(the 'Volume')
Volume Editor(s) Name(s):	Harish Sharma, Vivek Shrivastava, Ashish Kumar Tripathi, Lipo Wang	
Proposed Title of the Contribution:	iDSign: Secure Lightweight Digital Document Signatures Framework with Mutual Authentication and Identity-Based Cryptography	(the 'Contribution')
Series: The Contribution may be published in the following series	A Springer book series Lecture Notes in Networks and Systems	
Author(s) Full Name(s):	Himanshu Sharma, Divyashikha Sethia	(the 'Author')

When Author is more than one person the expression "Author" as used in this Agreement will apply collectively unless otherwise indicated.

Corresponding Author Name:	Himanshu Sharma	
Instructions for Authors	https://www.springer.com/gp/authors-editors/conference-proceedings/conference-proceedings-guidelines	(the 'Instructions for Authors')

1 Grant of Rights

a) For good and valuable consideration, the Author hereby grants to the Licensee the perpetual, exclusive, world-wide, assignable, sublicensable and unlimited right to: publish, reproduce, copy, distribute, communicate, display publicly, sell, rent and/or otherwise make available the contribution identified above, including any supplementary information and graphic elements therein (e.g. illustrations, charts, moving images) (the 'Contribution') in any language, in any versions or editions in any and all forms and/or media of expression (including without limitation in connection with any and all end-user devices), whether now known or developed in the future. Without limitation, the above grant includes: (i) the right to edit, alter, adapt, adjust and prepare derivative works; (ii) all advertising and marketing rights including without limitation in relation to social media; (iii) rights for any training, educational and/or instructional purposes; (iv) the right to add and/or remove links or combinations with other media/works; and (v) the right to create, use and/or license and/or sublicense content data or metadata of any kind in relation to the Contribution (including abstracts and summaries) without restriction. The above rights are granted in relation to the Contribution as a whole or any part and with or in relation to any other works.

b) Without limiting the rights granted above, Licensee is granted the rights to use the Contribution for the purposes of analysis, testing, and development of publishing- and research-related workflows, systems, products, projects, and services; to confidentially share the Contribution with select third parties to do the same; and to retain and store the Contribution and any associated correspondence/files/forms to maintain the historical record, and to facilitate research integrity investigations. The grant of rights set forth in this clause (b) is irrevocable.

c) If the Licensee elects not to publish the Contribution for any reason, all publishing rights under this Agreement as set forth in clause 1a above will revert to the Author.

2 Copyright

Ownership of copyright in the Contribution will be vested in the name of the Author. When reproducing the Contribution or extracts from it, the Author will acknowledge and reference first publication in the Volume.

3 Use of Contribution Versions

a) For purposes of this Agreement: (i) references to the "Contribution" include all versions of the Contribution; (ii) "Submitted Manuscript" means the version of the Contribution as first submitted by the Author prior to peer review; (iii) "Accepted Manuscript" means the version of the Contribution accepted for publication, but prior to copy-editing and typesetting; and (iv) "Version of Record" means the version of the Contribution published by the Licensee, after copy-editing and typesetting. Rights to all versions of the Manuscript are granted on an exclusive basis, except for the Submitted Manuscript, to which rights are granted on a non-exclusive basis.

b) The Author may make the Submitted Manuscript available at any time and under any terms (including, but not limited to, under a CC BY licence), at the Author's discretion. Once the Contribution has been published, the Author will include an acknowledgement and provide a link to the Version of Record on the publisher's website: "This preprint has not undergone peer review (when applicable) or any post-submission improvements or corrections. The Version of Record of this contribution is published in [insert volume title], and is available online at https://doi.org/[insert DOI]".

c) The Licensee grants to the Author (i) the right to make the Accepted Manuscript available on their own personal, self-maintained website immediately on acceptance, (ii) the right to make the Accepted Manuscript available for public release on any of the following twelve (12) months after first publication (the "Embargo Period"): their employer's internal website; their institutional and/or funder repositories. Accepted Manuscripts may be deposited in such repositories immediately upon acceptance, provided they are not made publicly available until after the Embargo Period.
The rights granted to the Author with respect to the Accepted Manuscript are subject to the conditions that (i) the Accepted Manuscript is not enhanced or substantially reformatted by the Author or any third party, and (ii) the Author includes on the Accepted Manuscript an acknowledgement in the following form, together with a link to the published version on the publisher's website: "This version of the contribution has been accepted for publication, after peer review (when applicable) but is not the Version of Record and does not reflect post-acceptance improvements, or any corrections. The Version of Record is available online at: http://dx.doi.org/[insert DOI]. Use of this Accepted Version is subject to the publisher's Accepted Manuscript terms of use https://www.springernature.com/gp/open-research/policies/accepted-manuscript-terms".
Under no circumstances may an Accepted Manuscript be shared or distributed under a Creative Commons or other form of open access licence.
Any use of the Accepted Manuscript not expressly permitted under this subclause (c) is

subject to the Licensee's prior consent.

d) The Licensee grants to Author the following non-exclusive rights to the Version of Record, provided that, when reproducing the Version of Record or extracts from it, the Author acknowledges and references first publication in the Volume according to current citation standards. As a minimum, the acknowledgement must state: "First published in [Volume, page number, year] by Springer Nature".

i. to reuse graphic elements created by the Author and contained in the Contribution, in presentations and other works created by them;

ii. the Author and any academic institution where they work at the time may reproduce the Contribution for the purpose of course teaching (but not for inclusion in course pack material for onward sale by libraries and institutions);

iii. to reuse the Version of Record or any part in a thesis written by the same Author, and to make a copy of that thesis available in a repository of the Author(s)' awarding academic institution, or other repository required by the awarding academic institution. An acknowledgement should be included in the citation: "Reproduced with permission from Springer Nature";

iv. to reproduce, or to allow a third party to reproduce the Contribution, in whole or in part, in any other type of work (other than thesis) written by the Author for distribution by a publisher after an embargo period of 12 months; and

v. to publish an expanded version of their Contribution provided the expanded version (i) includes at least 30% new material (ii) includes an express statement specifying the incremental change in the expanded version (e.g., new results, better description of materials, etc.).

4 Warranties & Representations

Author warrants and represents that:

a)

i. the Author is the sole copyright owner or has been authorised by any additional copyright owner(s) to grant the rights defined in clause 1,

ii. the Contribution does not infringe any intellectual property rights (including without limitation copyright, database rights or trade mark rights) or other third party rights and no licence from or payments to a third party are required to publish the Contribution,

iii. the Contribution has not been previously published or licensed, nor has the Author committed to licensing any version of the Contribution under a licence inconsistent with the terms of this Agreement,

iv. if the Contribution contains materials from other sources (e.g. illustrations, tables, text quotations), Author has obtained written permissions to the extent necessary from the copyright holder(s), to license to the Licensee the same rights as set out in clause 1 but on a non-exclusive basis and without the right to use any graphic elements on a stand-alone basis and has cited any such materials correctly;

b) all of the facts contained in the Contribution are according to the current body of research true and accurate;

c) nothing in the Contribution is obscene, defamatory, violates any right of privacy or publicity, infringes any other human, personal or other rights of any person or entity or is otherwise unlawful and that informed consent to publish has been obtained for any research participants;

d) nothing in the Contribution infringes any duty of confidentiality owed to any third party or violates any contract, express or implied, of the Author;

e) all institutional, governmental, and/or other approvals which may be required in connection with the research reflected in the Contribution have been obtained and continue in effect;

f) all statements and declarations made by the Author in connection with the Contribution are true and correct;

g) the signatory who has signed this Agreement has full right, power and authority to enter into this Agreement on behalf of all of the Authors; and

h) the Author complies in full with: i. all instructions and policies in the Instructions for Authors, ii. the Licensee's ethics rules (available at https://www.springernature.com/gp/authors/book-authors-code-of-conduct), as may be updated by the Licensee at any time in its sole discretion.

5 Cooperation

a) The Author will cooperate fully with the Licensee in relation to any legal action that might arise from the publication of the Contribution, and the Author will give the Licensee access at reasonable times to any relevant accounts, documents and records within the power or control of the Author. The Author agrees that any Licensee affiliate through which the Licensee exercises any rights or performs any obligations under this Agreement is intended to have the benefit of and will have the right to enforce the terms of this Agreement.

b) Author authorises the Licensee to take such steps as it considers necessary at its own expense in the Author's name(s) and on their behalf if the Licensee believes that a third party is infringing or is likely to infringe copyright in the Contribution including but not limited to initiating legal proceedings.

6 Author List

Changes of authorship, including, but not limited to, changes in the corresponding author or the sequence of authors, are not permitted after acceptance of a manuscript.

7 Post Publication Actions

The Author agrees that the Licensee may remove or retract the Contribution or publish a correction or other notice in relation to the Contribution if the Licensee determines that such actions are appropriate from an editorial, research integrity, or legal perspective.

8 Controlling Terms

The terms of this Agreement will supersede any other terms that the Author or any third party may assert apply to any version of the Contribution.

9 Governing Law

This Agreement shall be governed by, and shall be construed in accordance with, the laws of the Republic of Singapore. The courts of Singapore, Singapore shall have the exclusive jurisdiction.

Signed for and on behalf of the Author	Print Name:	Date:
Sharma	Himanshu Sharma	11 December 2024

Address:	A-15 Ganga Vihar Delhi - 110094
Email:	sdothimanshu@gmail.com

6th International Conference on Communication and Intelligent Systems (ICCIS 2024)

Organized by

Maulana Azad National Institute of Technology (MANIT), Bhopal, India

Technically Sponsored by

Soft Computing Research Society

November 08-09, 2024

Certificate of Presentation

This is to certify that **Himanshu Sharma** has presented the paper titled **IDsign: Secure Lightweight Digital Document Signatures Framework with Mutual Authentication and Identity-Based Cryptography** authored by **Himanshu Sharma, Divyashikha Sethia** in the 6th International Conference on Communication and Intelligent Systems (ICCIS 2024) held during November 08-09, 2024.

SCRS\ICCIS2024\PC\391

Prof. Sanjay Sharma
(General Chair)

Dr. Harish Sharma
(General Chair)

https://scrs.in/conference/iccis2024

DECLARATION

I hereby certify that the work which is presented in the Major Project-II entitled **iDSign: A Secure Lightweight NFC-Based Framework to Tap and Digitally Sign Documents** in fulfillment of the requirement for the award of the Degree of Master of Technology in Software Engineering and submitted to the Department of Software Engineering, Delhi Technological University, Delhi is an authentic record of my own, carried out during a period from January to May 2025 under the supervision of **Dr. Divyashikha Sethia**.

The mater presented in this report has not been submitted by me for the award of any other degree of this or any other Institute/University. The work has been published/accepted/communicated in SCI/SCI expanded/SSCI/Scopus indexed journal OR peer reviewed Scopus indexed conference with the following details.

Title of the Paper: **iHDoc: Secure NFC Mobile-based Digital Document Signatures Prototype**
Author names (in sequence): Himanshu Sharma, Dr. Divyashikha Sethia
Name of Conference/Journal: 2025 INTERNATIONAL CONFERENCE ON INTELLIGENT CONTROL, COMPUTING AND COMMUNICATIONS (IC3 2025)
Conference dates with venue: 13-14th February 2025, Mathura, Uttar Pradesh, India
Status of paper (Communicated/Accepted/Published): Published
Date of paper communication: November 20, 2024
Date of paper acceptance: January 14, 2025
Date of paper publication: April 16, 2025

Himanshu Sharma
Roll No. 23/SWE/18

Student Roll No., Name and Signature

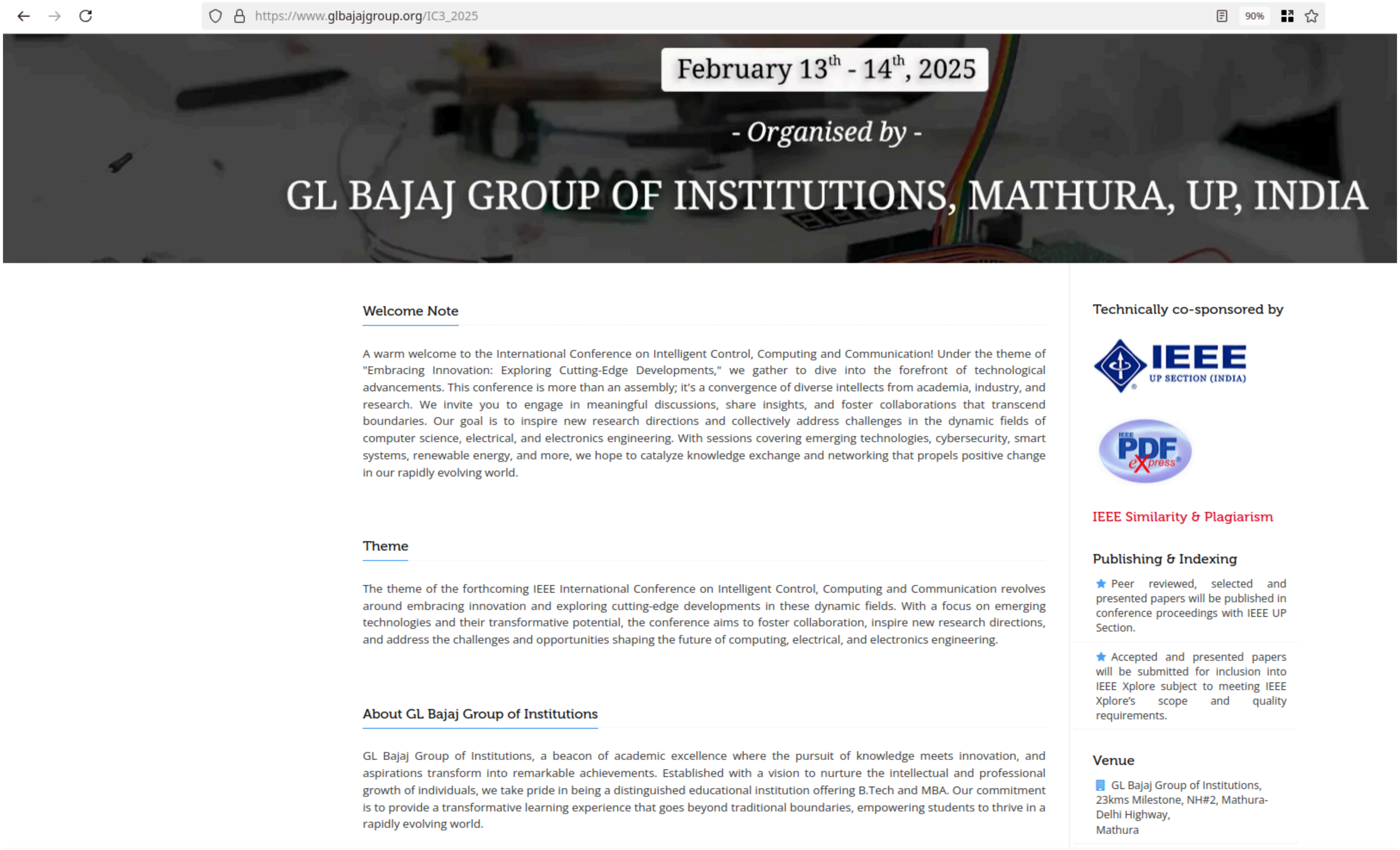
SUPERVISOR CERTIFICATE

To the best of my knowledge, the above work has not been submitted in part or full for any Degree or Diploma to this University or elsewhere. I further certify that the publication and indexing information given by the students is correct.

Divya.

Date: May 19, 2025 **Dr. Divyashikha Sethia**

Place: New Delhi **Supervisor Name and Signature**



February 13th - 14th, 2025

- Organised by -

GL BAJAJ GROUP OF INSTITUTIONS, MATHURA, UP, INDIA

Welcome Note

A warm welcome to the International Conference on Intelligent Control, Computing and Communication! Under the theme of "Embracing Innovation: Exploring Cutting-Edge Developments," we gather to dive into the forefront of technological advancements. This conference is more than an assembly; it's a convergence of diverse intellects from academia, industry, and research. We invite you to engage in meaningful discussions, share insights, and foster collaborations that transcend boundaries. Our goal is to inspire new research directions and collectively address challenges in the dynamic fields of computer science, electrical, and electronics engineering. With sessions covering emerging technologies, cybersecurity, smart systems, renewable energy, and more, we hope to catalyze knowledge exchange and networking that propels positive change in our rapidly evolving world.

Theme

The theme of the forthcoming IEEE International Conference on Intelligent Control, Computing and Communication revolves around embracing innovation and exploring cutting-edge developments in these dynamic fields. With a focus on emerging technologies and their transformative potential, the conference aims to foster collaboration, inspire new research directions, and address the challenges and opportunities shaping the future of computing, electrical, and electronics engineering.

About GL Bajaj Group of Institutions

GL Bajaj Group of Institutions, a beacon of academic excellence where the pursuit of knowledge meets innovation, and aspirations transform into remarkable achievements. Established with a vision to nurture the intellectual and professional growth of individuals, we take pride in being a distinguished educational institution offering B.Tech and MBA. Our commitment is to provide a transformative learning experience that goes beyond traditional boundaries, empowering students to thrive in a rapidly evolving world.

Technically co-sponsored by

IEEE UP SECTION (INDIA)

IEEE PDF eXpress®

IEEE Similarity & Plagiarism

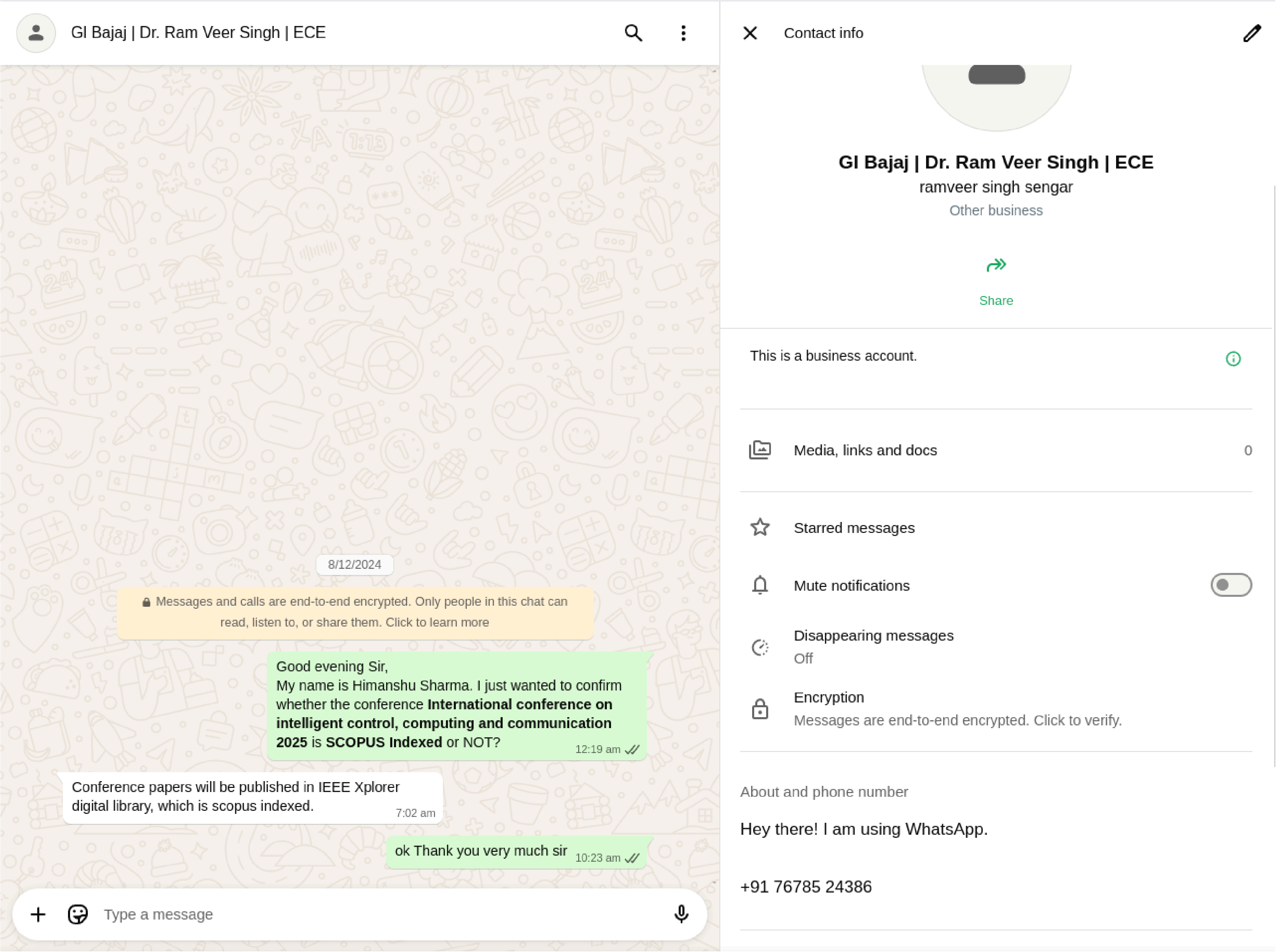
Publishing & Indexing

★ Peer reviewed, selected and presented papers will be published in conference proceedings with IEEE UP Section.

★ Accepted and presented papers will be submitted for inclusion into IEEE Xplore subject to meeting IEEE Xplore's scope and quality requirements.

Venue

GL Bajaj Group of Institutions,
23kms Milestone, NH#2, Mathura-Delhi Highway,
Mathura



GI Bajaj | Dr. Ram Veer Singh | ECE

8/12/2024

Messages and calls are end-to-end encrypted. Only people in this chat can read, listen to, or share them. Click to learn more

Good evening Sir,
My name is Himanshu Sharma. I just wanted to confirm whether the conference **International conference on intelligent control, computing and communication 2025** is **SCOPUS Indexed** or NOT? 12:19 am

Conference papers will be published in IEEE Xplorer digital library, which is scopus indexed. 7:02 am

ok Thank you very much sir 10:23 am

Contact info

GI Bajaj | Dr. Ram Veer Singh | ECE

ramveer singh sengar

Other business

This is a business account.

Media, links and docs 0

Starred messages

Mute notifications

Disappearing messages
Off

Encryption
Messages are end-to-end encrypted. Click to verify.

About and phone number

Hey there! I am using WhatsApp.

+91 76785 24386

23/SWE/18 HIMANSHU SHARMA <himanshusharma_23swe18@dtu.ac.in>

International Conference on Intelligent control, Computing and Communications-2025 : Submission (60) has been created.

Microsoft CMT <email@msr-cmt.org>
Reply-To: Microsoft CMT - Do Not Reply <noreply@msr-cmt.org>
To: himanshusharma_23swe18@dtu.ac.in

Sun, Oct 20, 2024 at 11:59 PM

Hello,

The following submission has been created.

Track Name: Communications and Networking

Paper ID: 60

Paper Title: iHDoc: Secure Mobile-based Digital Document Signatures Framework using Host Card Emulation Tap and Lightweight Mutual Authentication

Abstract:

This paper introduces the prototype implementation and assessment of iHDoc, a lightweight framework for secure digital document signing on mobile devices, which is based on the previously proposed iDSign framework (accepted, to be published). The original study laid the theoretical foundation for a versatile framework for safe digital document signing utilizing identity-based cryptography and lightweight mutual authentication through a modified lightweight identity-based Transport Layer Security (iTLS) protocol. This paper builds upon the prior research by employing the framework on mobile devices through native Android development and utilizing NFC-based host card emulation (HCE) for secure session initiation via NFC tap and Bluetooth for effective data transfer. The prototype utilizes the Hess Identity-Based Signature (IBS) technique for speedy and efficient signing, making it suitable for resource-constrained devices. The comparative performance assessment with the RSA-based counterpart, iHDoc_RSA, demonstrates that the iHDoc framework attains faster execution times and decreased storage requirements. The iText library facilitates PDF manipulation and enables the smooth integration of digital signatures. The findings indicate that iHDoc provides both security and performance benefits, making it an effective solution for mobile document signing. Future endeavours will focus on real-world deployment and improved compatibility among platforms.

Created on: Sun, 20 Oct 2024 18:29:30 GMT

Last Modified: Sun, 20 Oct 2024 18:29:30 GMT

Authors:

- himanshusharma_23swe18@dtu.ac.in (Primary)

Secondary Subject Areas: Not Entered

Submission Files: Not Uploaded

Submission Questions Response: Not Entered

Thanks,
CMT team.

23/SWE/18 HIMANSHU SHARMA <himanshusharma_23swe18@dtu.ac.in>

Paper Acceptance Notification for International Conference on Intelligent control, Computing and Communications-2025

Microsoft CMT <email@msr-cmt.org> Sun, Jan 12, 2025 at 4:25 PM
Reply-To: "Prof. Neeta Awasthy" <mandhir.verma@glbitm.ac.in>
To: Himanshu Sharma <himanshusharma_23swe18@dtu.ac.in>
Cc: mandhir.verma@glbitm.ac.in

Dear Himanshu Sharma,

Congratulations! On behalf of the "2025 International Conference on Intelligent Control, Computing and Communications (IC3)" Technical Program Committee (TPC), we are pleased to inform you that your paper entitled (Paper Id- 60)"iHDoc: Secure NFC Mobile-based Digital Document Signatures Prototype" has been accepted as a REGULAR paper for presentation at the IC3-2025 dated 13-14 February, 2025. All accepted and presented papers will be submitted to IEEE for possible inclusion on IEEE Xplore.

The steps you have to follow in submitting the final version of your paper (Camera Ready Paper):
1. Please be specific in every contents of your manuscript. Every paragraph must be meaningful and should have connections with prior and after sentences.
2. Make sure that all figures/table/equation shown/given in the manuscript are drawn/written originally, if any figure/table/equation are copied from other sources, please cite them and write the references.
3. All text of figure/Table must be readable and should be as per IEEE standard. Make sure Equations & Results are self drawn and clear- visible to readers.
4. If the reviewers have provided comments, ensure all their suggestions are addressed thoroughly and incorporated into the manuscript.
5. Please check your paper once as it is sole responsibility of authors for making it quality work and representing their work in good manner.

After incorporating above points/suggestions and checking your manuscript thoroughly, Kindly proceed for registration process through given link till 15 January 2025.

Registration Link:
https://docs.google.com/forms/d/e/1FAIpQLSdYKuDcUP_DTHRZVqPuTmzl_ZfBbgXudJvmO3QWP3QIid_Y1w/viewform

Registration Fee Details:
https://www.glbajajgroup.org/IC3_2025/registration

Also, please email following files at iccc@glbajajgroup.org
1. Payment receipt.
2. Word file of final revised paper (It is required to write header & footer as per IEEE standard.)
3. Valid IEEE Member card (if applicable)

Note: While submitting revised manuscript, Pl. check format as per IEEE Xplore & ready manuscript carefully to avoid plagiarism (must be lass than 15%), punctuation, grammatical error if any. As per IEEE policy plagiarism is not allowed at any stage of quality check..

Important Note: Your paper will be excluded from proceedings at any level of quality checks carried out by IEEE. Conference organizers are not responsible if your paper detects plagiarism or any lack in quality reported by IEEE after submission of final proceedings (camera ready copies) to IEEE. Conference organizers are responsible for the technical quality of paper on the basis of reviews obtained. It is sole responsibility of author/s if their paper detects plagiarism even in later stages of quality check. No registration fee will be refunded in any discrepancies if your paper is excluded from proceedings by the IEEE.

Best regards,
TPC IC3-2025
2025 International Conference on Intelligent Control, Computing and Communications (IC3)
iccc@glbajajgroup.org

To stop receiving conference emails, you can check the 'Do not send me conference email' box from your User

Profile.

Microsoft respects your privacy. To learn more, please read our Privacy Statement.

Microsoft Corporation
One Microsoft Way
Redmond, WA 98052

Acknowledgement

Fund Transfer is Successful.

From Account	3509482247
Remitter Name	Mr. HIMANSHU SHARMA
To Account	14772011000890
Transaction Date	16-01-2025 11:15:47
Transfer Amount	10000.00
Commission	0.0
GST	0.0
Beneficiary Name	GL BAJAJ GROUP OF INSTITUTIONS
Beneficiary Address	Mathura Uttar pradesh
Remarks	For my second conference
Bank Name	PUNJAB NATIONAL BANK
Branch Name	AKBARPUR DIST MATHURA
IFSC	PUNB0147710
UTR NO	CBINN52025011668183700

Print

IEEE COPYRIGHT AND CONSENT FORM

To ensure uniformity of treatment among all contributors, other forms may not be substituted for this form, nor may any wording of the form be changed. This form is intended for original material submitted to the IEEE and must accompany any such material in order to be published by the IEEE. Please read the form carefully and keep a copy for your files.

iHDoc: Secure NFC Mobile-based Digital Document Signatures Prototype

Himanshu Sharma, Divyashikha Sethia

2025 International Conference on Intelligent Control, Computing and Communications (IC3)

COPYRIGHT TRANSFER

The undersigned hereby assigns to The Institute of Electrical and Electronics Engineers, Incorporated (the "IEEE") all rights under copyright that may exist in and to: (a) the Work, including any revised or expanded derivative works submitted to the IEEE by the undersigned based on the Work; and (b) any associated written or multimedia components or other enhancements accompanying the Work.

GENERAL TERMS

1. The undersigned represents that he/she has the power and authority to make and execute this form.
2. The undersigned agrees to indemnify and hold harmless the IEEE from any damage or expense that may arise in the event of a breach of any of the warranties set forth above.
3. The undersigned agrees that publication with IEEE is subject to the policies and procedures of the IEEE PSPB Operations Manual.
4. In the event the above work is not accepted and published by the IEEE or is withdrawn by the author(s) before acceptance by the IEEE, the foregoing copyright transfer shall be null and void. In this case, IEEE will retain a copy of the manuscript for internal administrative/record-keeping purposes.
5. For jointly authored Works, all joint authors should sign, or one of the authors should sign as authorized agent for the others.
6. The author hereby warrants that the Work and Presentation (collectively, the "Materials") are original and that he/she is the author of the Materials. To the extent the Materials incorporate text passages, figures, data or other material from the works of others, the author has obtained any necessary permissions. Where necessary, the author has obtained all third party permissions and consents to grant the license above and has provided copies of such permissions and consents to IEEE

You have indicated that you DO wish to have video/audio recordings made of your conference presentation under terms and conditions set forth in "Consent and Release."

CONSENT AND RELEASE

1. ln the event the author makes a presentation based upon the Work at a conference hosted or sponsored in whole or in part by the IEEE, the author, in consideration for his/her participation in the conference, hereby grants the IEEE the unlimited, worldwide, irrevocable

permission to use, distribute, publish, license, exhibit, record, digitize, broadcast, reproduce and archive, in any format or medium, whether now known or hereafter developed: (a) his/her presentation and comments at the conference; (b) any written materials or multimedia files used in connection with his/her presentation; and (c) any recorded interviews of him/her (collectively, the "Presentation"). The permission granted includes the transcription and reproduction of the Presentation for inclusion in products sold or distributed by IEEE and live or recorded broadcast of the Presentation during or after the conference.

2. In connection with the permission granted in Section 1, the author hereby grants IEEE the unlimited, worldwide, irrevocable right to use his/her name, picture, likeness, voice and biographical information as part of the advertisement, distribution and sale of products incorporating the Work or Presentation, and releases IEEE from any claim based on right of privacy or publicity.

BY TYPING IN YOUR FULL NAME BELOW AND CLICKING THE SUBMIT BUTTON, YOU CERTIFY THAT SUCH ACTION CONSTITUTES YOUR ELECTRONIC SIGNATURE TO THIS FORM IN ACCORDANCE WITH UNITED STATES LAW, WHICH AUTHORIZES ELECTRONIC SIGNATURE BY AUTHENTICATED REQUEST FROM A USER OVER THE INTERNET AS A VALID SUBSTITUTE FOR A WRITTEN SIGNATURE.

Himanshu Sharma	21-02-2025
Signature	Date (dd-mm-yyyy)

Information for Authors

AUTHOR RESPONSIBILITIES

The IEEE distributes its technical publications throughout the world and wants to ensure that the material submitted to its publications is properly available to the readership of those publications. Authors must ensure that their Work meets the requirements as stated in section 8.2.1 of the IEEE PSPB Operations Manual, including provisions covering originality, authorship, author responsibilities and author misconduct. More information on IEEE's publishing policies may be found at http://www.ieee.org/publications_standards/publications/rights/authorrightsresponsibilities.html Authors are advised especially of IEEE PSPB Operations Manual section 8.2.1.B12: "It is the responsibility of the authors, not the IEEE, to determine whether disclosure of their material requires the prior consent of other parties and, if so, to obtain it." Authors are also advised of IEEE PSPB Operations Manual section 8.1.1B: "Statements and opinions given in work published by the IEEE are the expression of the authors."

RETAINED RIGHTS/TERMS AND CONDITIONS

- Authors/employers retain all proprietary rights in any process, procedure, or article of manufacture described in the Work.
- Authors/employers may reproduce or authorize others to reproduce the Work, material extracted verbatim from the Work, or derivative works for the author's personal use or for company use, provided that the source and the IEEE copyright notice are indicated, the copies are not used in any way that implies IEEE endorsement of a product or service of any employer, and the copies themselves are not offered for sale.
- Although authors are permitted to re-use all or portions of the Work in other works, this does not include granting third-party requests for reprinting, republishing, or other types of re-use.The IEEE Intellectual Property Rights office must handle all such third-party requests.
- Authors whose work was performed under a grant from a government funding agency are free to fulfill any deposit mandates from

that funding agency.

AUTHOR ONLINE USE

- **Personal Servers**. Authors and/or their employers shall have the right to post the accepted version of IEEE-copyrighted articles on their own personal servers or the servers of their institutions or employers without permission from IEEE, provided that the posted version includes a prominently displayed IEEE copyright notice and, when published, a full citation to the original IEEE publication, including a link to the article abstract in IEEE Xplore. Authors shall not post the final, published versions of their papers.
- **Classroom or Internal Training Use.** An author is expressly permitted to post any portion of the accepted version of his/her own IEEE-copyrighted articles on the author's personal web site or the servers of the author's institution or company in connection with the author's teaching, training, or work responsibilities, provided that the appropriate copyright, credit, and reuse notices appear prominently with the posted material. Examples of permitted uses are lecture materials, course packs, e-reserves, conference presentations, or in-house training courses.
- **Electronic Preprints.** Before submitting an article to an IEEE publication, authors frequently post their manuscripts to their own web site, their employer's site, or to another server that invites constructive comment from colleagues. Upon submission of an article to IEEE, an author is required to transfer copyright in the article to IEEE, and the author must update any previously posted version of the article with a prominently displayed IEEE copyright notice. Upon publication of an article by the IEEE, the author must replace any previously posted electronic versions of the article with either (1) the full citation to the IEEE work with a Digital Object Identifier (DOI) or link to the article abstract in IEEE Xplore, or (2) the accepted version only (not the IEEE-published version), including the IEEE copyright notice and full citation, with a link to the final, published article in IEEE Xplore.

Questions about the submission of the form or manuscript must be sent to the publication's editor.
Please direct all questions about IEEE copyright policy to:
IEEE Intellectual Property Rights Office, copyrights@ieee.org, +1-732-562-3966

IEEE
UP SECTION (INDIA)

GL BAJAJ
Group of Institutions
Mathura
FIND YOUR SPARK
Approved by AICTE & Affiliated to AKTU

GLBM/IC3-2025/63308/ 69

GL BAJAJ GROUP OF INSTITUTIONS

Mathura, Uttar Pradesh, India

CERTIFICATE OF ACKNOWLEDGEMENT

Presented to

Himanshu Sharma

Delhi Technological University

In recognition and appreciation of valuable contribution of paper entitled

iHDoc: Secure NFC Mobile-based Digital Document Signatures Prototype

as

Presenter

in International Conference on

Intelligent Control, Computing and Communications (IC3-2025)

at GL Bajaj Group of Institutions, Mathura on February 13-14, 2025

Ramveer Singh
Dr. Ram Veer Singh Sengar
Conference Convener (IC3-2025)

Prof. (Dr.) V. K. Singh
Conference Chair (IC3-2025)

Prof. Neeta Awasthy
General Chair (IC3-2025)

23/SWE/18 HIMANSHU SHARMA <himanshusharma_23swe18@dtu.ac.in>

IC3-2025 Accepted Papers are Published on IEEE Explore digital library

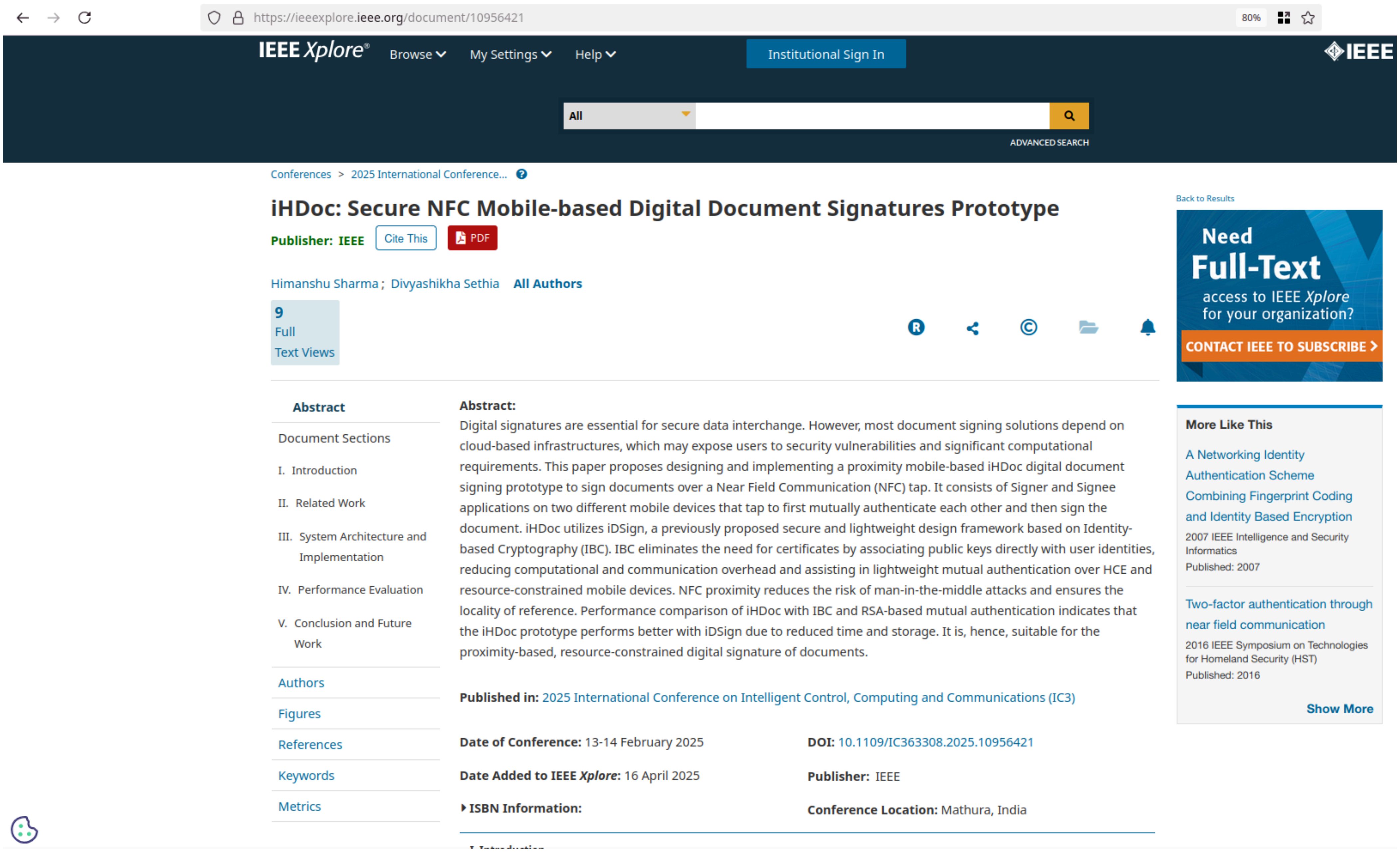
Microsoft CMT <noreply@msr-cmt.org> Thu, Apr 17, 2025 at 9:36 PM
To: Himanshu Sharma <himanshusharma_23swe18@dtu.ac.in>
Cc: mandhir.verma@glbitm.ac.in

Dear All Authors,

Congratulations to you All !!

All the accepted and presented papers in IEEE International Conference on Intelligent Control, Computing and Communications (IC3-2025) has been posted to the IEEE Xplore digital library on April 16, 2025.
Kindly download your article IEEE Xplore digital library. Write your name and affiliation in search box of IEEE Xplore digital Library and download the full paper.
Thanks for your kind support.

Prof V K Singh
Conference Chair(IC3-2025)



iHDoc: Secure NFC Mobile-based Digital Document Signatures Prototype

Publisher: IEEE

Himanshu Sharma ; Divyashikha Sethia **All Authors**

9 Full Text Views

Abstract
Document Sections
I. Introduction
II. Related Work
III. System Architecture and Implementation
IV. Performance Evaluation
V. Conclusion and Future Work
Authors
Figures
References
Keywords
Metrics

Abstract:
Digital signatures are essential for secure data interchange. However, most document signing solutions depend on cloud-based infrastructures, which may expose users to security vulnerabilities and significant computational requirements. This paper proposes designing and implementing a proximity mobile-based iHDoc digital document signing prototype to sign documents over a Near Field Communication (NFC) tap. It consists of Signer and Signee applications on two different mobile devices that tap to first mutually authenticate each other and then sign the document. iHDoc utilizes iDSign, a previously proposed secure and lightweight design framework based on Identity-based Cryptography (IBC). IBC eliminates the need for certificates by associating public keys directly with user identities, reducing computational and communication overhead and assisting in lightweight mutual authentication over HCE and resource-constrained mobile devices. NFC proximity reduces the risk of man-in-the-middle attacks and ensures the locality of reference. Performance comparison of iHDoc with IBC and RSA-based mutual authentication indicates that the iHDoc prototype performs better with iDSign due to reduced time and storage. It is, hence, suitable for the proximity-based, resource-constrained digital signature of documents.

Published in: 2025 International Conference on Intelligent Control, Computing and Communications (IC3)

Date of Conference: 13-14 February 2025

DOI: 10.1109/IC363308.2025.10956421

Date Added to IEEE *Xplore*: 16 April 2025

Publisher: IEEE

▸ **ISBN Information:**

Conference Location: Mathura, India

More Like This

A Networking Identity Authentication Scheme Combining Fingerprint Coding and Identity Based Encryption
2007 IEEE Intelligence and Security Informatics
Published: 2007

Two-factor authentication through near field communication
2016 IEEE Symposium on Technologies for Homeland Security (HST)
Published: 2016