International Journal of Scientific Research in Science, Engineering and Technology

Print ISSN - 2395-1990
Online ISSN : 2394-4099

OPEN ACCESS

Available Online at :www.ijsrset.com
doi : https://doi.org/10.32628/IJSRSET

Development of Enhanced Cloud File Sharing Security with eID Token-Based Trust Mechanisms

Umang Chaudhary[1], Ayan Rajput[2]
Research Scholar[1], Assistant Professor[2]
JP Institute of Engineering and Technology, Meerut, India

ARTICLEINFO

Article History:

Accepted : 25 May 2025
Published: 02 July 2025

Publication Issue :
Volume 12, Issue 4
July-August-2025

Page Number :
06-14

ABSTRACT

Numerous systems provide users an electronic identification (eID) for document signing or online service authentication (e.g., governmental eIDs, OpenID). Current solutions, however, fail to provide adequate methods for using them as conventional ID cards that digitally verify their holders to another individual in the actual realm. We foresee a comprehensive mobile eID that offers this capability while safeguarding privacy, meets the stringent security standards for governmental identities (such as driver's licenses and passports), and is applicable in the private sector (e.g., as loyalty cards). This study delineates possible applications for a flexible and privacy-preserving mobile eID and examines the notion of privacy-preserving attribute searches. Additionally, we delineate essential functional, mobility, security, and privacy needs, and provide a concise review of alternative methodologies to address each of them.

Keywords : Electronic identities, privacy, mobile eID, requirements

1. INTRODUCTION

Electronic IDs (eID) enable users to authenticate electronically with service providers or to digitally sign documents. Numerous countries already provide their people electronic identification systems to facilitate administrative functions such as tax filing, subsidy applications, or company registration. The Estonian and Finnish governments equip their people with mobile eIDs incorporated into SIM cards on mobile phones.

OpenID3 and the Fast Identity Online (FIDO)4 protocol exemplify systems in the private domain that operate across system boundaries. OpenID enables user authentication across several services without necessitating separate registrations for each, whereas FIDO offers a more user-friendly authentication experience.

Nonetheless, many current methods depend on the notion of an online service as an identity verification. For instance, eID holders authenticate their identity with an e-government service; users want to access an online mail server, among others. This limitation leads to systems that do not provide identity verification akin to conventional identification cards. An illustrative example is a bouncer at a nightclub who verifies the age of patrons. Although this operation is straightforward with standard identification cards, it

may become problematic with eID tokens (i.e., eID cards). Potential challenges may include the necessity for online connectivity, the requirement of specialised reader equipment for the interface with an eID token connected to a PC, the need for each verifier to obtain specific certification to read the token, and the obligation for the eID holder to input login credentials on the verifying device, among others.

Consequently, current options are inadequate for substituting conventional identification cards as straightforward physical confirmation of identity. The need of Internet access during verification is particularly onerous. We designate these attributes as real-world identification and offline verification.

Moreover, the use of an electronic identification token presents several additional issues regarding user privacy. As data undergoes digital processing, people lose control over their information. They cannot ascertain if their data is sufficiently safeguarded, only used for the stated purpose of identification, and not retained or sent to other parties. Therefore, concerning user privacy, it is essential that little information be sent to verifiers. For instance, it is enough for the bouncer at the nightclub to ascertain if the individual is above 18 years of age. The specific date of birth and further details, like name, address, and social security number, are not pertinent.

The primary objective of this work is to expand upon the concept of an eID system that facilitates both real-world identification and privacy-preserving attribute verification. Our objective is to provide a mobile eID solution that meets the stringent security criteria of government-issued IDs, including driver's licenses and passports. Moreover, we want to develop an eID that serves as a central authentication token for several applications while preserving privacy, including a loyalty card, a public transit ticket, and others.

This study formalises the fundamental criteria for a privacy-preserving eID system that needs excellent security and flexibility. Particular emphasis is placed on the possibility for direct physical engagement during the verification phase (i.e., real-world identification), together with offline and mobile functionalities. We delineate sample use cases that must be addressed by such a system and examine the overarching notion of privacy-preserving attribute enquiries that do not disclose superfluous information about the eID holder. Ultimately, we provide a perspective on forthcoming actions aimed at establishing a system that meets all criteria.

2. Literature Review

We assess numerous cloud storage security solutions' essential features. These programs safeguard Dropbox, Google Drive, and OneDrive files with encryption and access control beyond the cloud service's infrastructure.

The analysis comprises BoxCryptor, Viivo, CloudFogger, Sookasa, TrueCrypt, and CCE [1]. These solutions differ in user control, key management, usability, and interaction with external identity or authentication frameworks for pre-cloud data protection.

This comparison contrasts Protbox's design and operational goals with competitors. Protbox uses national eID infrastructures for robust authentication and functions independently from any external key distribution service, unlike many solutions that employ centrally managed key infrastructures or limited multi-user functionality. Key exchange into data sharing cloud directories ensures end-to-end secrecy and protection from unauthorised deletions or changes while transporting information via cloud services.

This comparison contrasts Protbox with third-party encryption systems in trust, privacy, and architectural autonomy.

Like Protbox, cloud storage uses 256-bit AES encryption and locally generated keys. Protbox uses integrity checking to prevent shared folder corruption, unlike BoxCryptor, TrueCrypt, Viivo, and Sookasa. Viivo and Sookasa solely encrypt cloud service

provider-specific folders, while BoxCryptor and TrueCrypt construct virtual disc drives in the kernel to store plaintext versions of locally encrypted data. Unlike other platforms, Protbox lets users establish unique Protbox Pairs, connect cloud and local directories, and activate numerous pairs in a single folder for concurrent synchronisation across many cloud providers.

These technologies protect local data differently. CloudFogger and Sookasa dynamically encrypt and decrypt data without local copies, while TrueCrypt saves encrypted files in a mountable container only available during runtime. Like Viivo and BoxCryptor, Protbox stores a decrypted view locally but only secures cloud data, not the prot folder. Centralised backend servers manage most system sharing and administration. On BoxCryptor's key server, credentials protect each file's encryption keys. Others may store encryption data locally on their servers using directory-level keys, but without open-source access, this is unverifiable. In Protbox, shared folder transactions distribute keys for safe multi-pair sharing without backend servers.

Many authentication mechanisms exist. Unlike password-based systems, Protbox uses national eID tokens for ownership-factor authentication. CCE [2] requires eID-supported encryption for identity verification, but Protbox merely needs signatures. This works with unencrypted tokens like the Portuguese Citizen Card. Shared protocols differ greatly. Instead of Viivo's moderator-centric architecture, Protbox multicasts requests and requires only one peer's permission and acceptable certificate and signature verifications to assure equal access without central control.

Recent study examines commercial and sophisticated cloud security frameworks. For complete protection, researchers advocate dynamic group sharing with anonymity and traceability [1] and trusted execution environments like Intel SGX [2][3]. Proxy re-encryption, STRIDE threat modelling, blockchain integration, and revocable storage protect data and access. Distributed storage, steganography-enhanced AES, and sender-specified proxy re-encryption improve security. Key-sharing methods like SeDaSC [9] and group key management protocols [2] show how secure cloud storage systems evolve to defend against insider attacks and collusion.

This synthesis shows Protbox's distinct cloud security research perspective. Protbox addresses flexibility, decentralisation, and verification difficulties in the comparative study and offers new enhancements.

Many cloud storage services encrypt data. BoxCryptor, TrueCrypt, Viivo, and Sookasa use 256-bit AES with locally generated keys like Protbox [1][2][3]. Protbox regularly monitors file changes to prevent shared directory corruption and analyses encrypted files for integrity, a feature not mentioned in other systems [4]. Protbox emphasises anonymity and data authenticity, filling a business niche.

Local integration strategies differ by architecture. BoxCryptor and TrueCrypt use OS kernel-dependent virtual disc drives to preserve plaintext replicas, but Viivo and Sookasa can encrypt cloud provider folders. Protbox Pairs lets users map cloud-local folders and synchronise several clouds with one protocol folder [9].

Management practices show philosophical disparities. Most solutions (BoxCryptor, Viivo, Sookasa) store and distribute keys on central backend servers, creating single points of failure. Protbox's fully decentralised key distribution employs shared files, necessitating advanced conflict resolution during simultaneous writes. SeDaSC's split-key technique [19] and GKMP's group key management [2] are recent innovations, but Protbox's operational simplicity and independence from trusted third parties make it unique.

Similar to authentication, methods differ. Protbox improves CCE's core work [ZSTD13] over password-based authentication [4][5] with national eID tokens. CCE required eID encryption, which Portugal's Citizen Card lacks, whereas Protbox simply needs

signature functionality, boosting interoperability. The change highlights how Protbox balances security and deployability.

Recent studies improve cloud security. ABE offers user attribute-based access control [3], whereas proxy re-encryption allows secure data delegation [4][7]. Intel SGX-based systems like SeGShare provide hardware-enforced security, while blockchain connectors enable decentralised integrity verification. CSSM utilises dispersion and encryption for secure storage [8], while RS-IBE uses ciphertext modifications for revocable storage [7]. Protbox's decentralised sharing method supports these improvements, especially zero-trust [8]. This detailed analysis situates Protbox in cloud security, stressing its distinctive contributions and appreciating key past work [1-20].[ZSTD13]. Decentralisation, cryptographic agility, and practical application can lead to personal cloud security solutions, however key revocation and quantum resistance remain unsolved. [9][19].

2.1. Research Deficiencies

The following is the revised tabular representation of research gaps:

Table 2.1: Research Gaps

Research Gap	Contribution of this Thesis	References
Overreliance on Centralised Servers for Key Management: Many commercial and academic systems maintain and distribute encryption keys using trusted, centralised backend servers. Users must trust the service provider's infrastructure because of this single point of failure.	This thesis offers a cloud folder-only, decentralised, peer-to-peer key distribution mechanism. Using eID-signed files, it exchanges cryptographic material without a trusted key management server.	[2]
Many secure sharing tools employ passwords for user authentication, which is weak or inconvenient. This strategy fails to protect sensitive data from phishing, theft, and impersonation.	The proposed system uses national electronic identification (eID) tokens for robust two-factor authentication. This immediately links file access to a cryptographically secure, physical credential, improving user identity verification and non-repudiation.	[9], [8]
Existing systems rarely provide transparent, user-verifiable integrity tests for encrypted files. They also trust the cloud provider for file recovery, leaving data exposed to silent corruption or malicious deletion that the user cannot recover.	All shared files have HMAC-SHA1 local integrity verification. The system's local, user-controlled backup and versioning registry (PReg) allows file recovery independent of the cloud provider's restrictions and capabilities.	[4], [8]
Some solutions deeply interact with cloud provider APIs or need deep kernel-level integration (e.g., virtual disc drives). This restricts their portability, flexibility, and deployment across cloud services and operating systems.	This thesis provides a cloud provider-agnostic architectural solution. It runs at the file system level, requiring only a basic folder synchronisation capability, ensuring broad compatibility and making the system transparent and adaptable for users.	[1], [5]

2.2 RELATED WORK

Numerous nations already provide eID cards to their residents. In European nations, this is often achieved with smart cards that enable the creation of qualified and legally binding signatures. A study conducted by Lehman et al. [8] on the governmental eIDs accessible in the European Union reveals that none provide anonymous and privacy-preserving verification procedures. Only the Austrian and German eID cards provide significant functionalities for safeguarding users' privacy via pseudonym creation and selective attribute disclosure. The predominant strategies used for privacy-preserving eID systems in the literature are pseudonym-based signatures [1, 2] and group signature protocols [4, 5, 11]. The latter enables group members to endorse communications on behalf of the whole group while maintaining their anonymity inside that group. Pseudonym-based signatures use public-key cryptography (e.g., RSA, ECC) and provide each member with a collection of pseudonyms for signing communications.

characteristic-based signatures, as proposed by Maji et al. [10], represent an adaptation of group signatures, enabling a signer to affirm possession of a certain characteristic. Consequently, the verifier does not get the actual property but just receives confirmation of whether the characteristic has a certain value.

The Fast Identity Online (FIDO) consortium has just published a widely recognised definition for electronic identities (eIDs). They constitute an industry collaboration aimed at enhancing the usability of user authentication on the Internet by diminishing dependence on passwords. By offering one standard for passwordless authentication and another for second-factor authentication, they provide frameworks for safe identity verification across all online services.

Nyman et al. [12] delineate an eID architecture predicated on the utilisation of Trusted Platform Modules (TPM). They expand upon version 2.0 of the TPM standard and assess its viability as an identification token on both PC and mobile systems. They also provide a comprehensive specification of standards for electronic identification systems, emphasising online services. We expand upon the definitions established in prior research, particularly those concerning eID systems by Nyman et al. [12] and the privacy attributes delineated by Camenisch et al. [4]. We further augment them with needs derived from actual applications of real-world identification using mobile eIDs, as detailed in the subsequent sections of this work.

3. PRIVACY-PRESERVING MOBILE eID

This section delineates an eID system that facilitates real-world identification while ensuring privacy-preserving attribute verification. Specifically, we enhance the stakeholder and attribute verification processes that will be feasible under such a system.

3.1 Stakeholders

We consider four main stakeholders of an eID system:

The eID issuer serves as the primary authority overseeing eID enrolment and offers an interface for verifiers to get system information.

The prover is the legitimate possessor of an eID and employs methods to validate her identity to a verifier. The architecture of an eID system may need the prover to own a physical eID token (e.g., smart card, mobile device, etc.) that has data characteristics (e.g., name, address, etc.) pertaining to the holder.

The verifier seeks to identify and authenticate eID bearers. This may be another individual seeking to verify that the eID holder has certain features. A verifier need not need include a person, such as an automated vending machine.

A verifier group may consist of an online service or any domain that offers services to provers. The group may append characteristics to the eID, and all group verifiers can authenticate them. Nonmembers must

not have access to any information on these qualities. For instance, the point-of-sale at a retail establishment (i.e., verifier group member) seeks to authenticate that an eID belongs to a customer loyalty program member. A rival must not have access to such information.

3.2 Privacy-preserving Attribute Queries

To safeguard the privacy of eID holders, an attribute query must not disclose information that is irrelevant to a particular purpose. To this end, we delineate three privacy-preserving enquiries that will only provide a binary outcome:

- Query about attribute equivalence. The verifier determines if an attribute on the eID token has a certain reference value using this sort of query. If the verifier lacks knowledge of the real value, it cannot be ascertained using such a question.
- Input: characteristic, benchmark value Result: 1 | 0
- Query about attribute inequality. This query enables the verifier to ascertain if a certain property on the eID exceeds or falls short of a designated reference value.
- Input: attribute, reference value, operator (<, ≤, >, ≥) Result: 1 | 0
- Verification of group affiliation. This inquiry enables the verifier to ascertain if a holder belongs to a certain group (e.g., loyalty program).
- Group identifier Result: 1 | 0

Exemplary Use Cases

A privacy-preserving mobile eID has several possible real-world applications. This is a scenario in which privacy-preserving attribute searches are advantageous.

- Verification of parcel collecting address. The package of an eID holder was redirected to the post office due to non-delivery. Upon collecting the item, the eID holder must confirm their identity as the legitimate receiver and authenticate their residence at the delivery address. To confirm this, the verifier (i.e., postal officer) use a mobile device running the eID application to transmit an attribute equality query including the specified name and address to the eID token. The postal officer receives a binary outcome whether the address and name correspond with the eID characteristics.
- Verification of age. A bouncer at a nightclub permits entry just to those above the age of 18. To verify the age of the eID holder, the bouncer employs a mobile device to transmit an attribute inequality query, using the date 18 years prior to the current date as a reference value together with the ≤ operator. If the date of birth recorded on the eID token is less than or equal to the reference value, the holder is above 18, then the query result is 1.
- Membership in the loyalty program. A store provides exclusive discounts for members of its loyalty program. eID holders may participate in this initiative using the eID application on their mobile devices to share all pertinent information (e.g., group identification). The point-of-sale terminal at the store transmits the group identifier during a group membership verification and receives a binary outcome. Consequently, the terminal ascertains if the consumer is a program member without requiring further data (e.g., name, age, etc.).

4. FORMAL REQUIREMENTS

The criteria for a mobile eID system are derived from the definitions provided by Nyman et al. [12]. They may be categorised into three classifications: functional, security, and privacy needs. In addition to these criteria, we established several supplementary ones that specifically address real-world identification and mobility contexts. The primary criteria outlined in [12] that we use are the one-to-many connection,

secrecy of identification keys, code separation, and cryptographic specifications. Furthermore, we examine the category of mobility needs.

4.1 Functional Requirements

The eID should provide identity verification analogous to conventional identification papers. This real-world identification between the prover and verifier should be feasible using commonplace technologies, such as mobile phones and tablets. An illustrative scenario is the police officer verifying the driver's licence saved on the eID inside a mobile phone of the individual or the bouncer at a nightclub assessing the age of patrons using a tablet.

One-to-many relationship: A single individual should be permitted to register for many domains. The user must be permitted to belong to several verifier groups (e.g., loyalty programs) using the same eID. These organisations should therefore have the capability to augment properties in the eID tokens (e.g., save information about a public transport ticket, include loyalty program data, etc.)

Revocation must be permissible for the eID owner (e.g., user has lost the identity token), the eID issuer (e.g., citizen is dead), and the verifier group (e.g., service provider terminating membership). Given a governmental identity capable of executing sensitive duties, revocation methods must be almost instantaneous. The approach must also account for scalability, since revocation may occur often. According to [9], a national Belgian eID records 375,000 cancelled IDs among a population of 10 million individuals. Consequently, a straightforward certificate revocation list retrieved by each verifier may be impractical.

Scalability: In addition to revocation, the system must scale across all essential components of the eID architecture, including enrolment and verification.

4.2 Mobility Requirements

Verification must be feasible with offline devices on both the prover and verifier sides. In other words, neither should need internet access to a central server for verification purposes. Nevertheless, they may periodically connect to get system updates. Furthermore, the revocation tests (Req. 1.c) must be feasible in mobile and offline contexts. For instance, law enforcement officials should not need network connection to authenticate the legitimacy of a driver's license.

Power-off: Verification must not need the prover's gadget to be switched on. Therefore, the accessibility of an eID, such as one stored on a mobile device, should be unaffected by a depleted battery.

Scalability: Analogous to the functional need in Req. 1.d, scalability is a fundamental criterion for the mobility of both prover and verifier. To ensure the system remains functional on mobile devices, the volume of data handled must remain manageable on resource-constrained devices, even with a substantial user base.

4.3. Security Requirements

Essential confidentiality and code segregation: The cryptographic keys associated with an identity must be safeguarded using hardware engineered to provide elevated confidentiality and integrity guarantees (e.g., smart cards). All operations using these keys must be conducted inside this environment.

Unforgeability and attribute authenticity: Only eID tokens registered in the system are permitted to provide valid identity proofs, and eID attributes may only be altered by the issuer. The verifier must be capable of identifying both fraudulent identity assertions and altered characteristics.

Data transmission security: The confidentiality and integrity of data characteristics must be safeguarded during transmission between the eID and the verifier/issuer.

Advanced cryptography: For the specified data protection standards, advanced cryptographic algorithms and key sizes must be used. As per [3], the present minimal standards are equal to 256-bit elliptic curve cryptography with SHA-256. The system must be capable of adapting to future developments in cryptographic primitives and key sizes. This attribute is particularly significant in relation to official identification, such as driver's licenses, which often remain valid for more than a decade. Therefore, the security of identity tokens with extended validity will be resilient to future threats.

4.4. Privacy Requirements

Privacy-preserving signatures: As already elaborated in the related work section, group signatures are a good candidate for providing privacy-protective techniques to create signatures in an eID system. As discussed in [4], a well designed scheme should thereby provide the principles of anonymity, unforgeability, and unlinkability. Note that unforgeability is already listed as a security requirement in Req. 3.b. Furthermore, we consider backward unlinkability as defined by Nakanishi et al. [11] as an essential requirement:

Anonymity: Users' identities must remain indiscernible among the whole population (k-anonymity, where k is the population size). A signature generated by an eID must not disclose the user's identify.

Unlinkability: Signatures generated during a verification procedure or revocation data pertaining to the same user must be untraceable to one another. An eID holder must remain untraceable throughout verifications.

Backward unlinkability: The anonymity and unlinkability attributes of an eID should remain intact even after its revocation.
The eID holder must retain authority over their data. Thus, the user will possess the authority to choose which data attributes may be viewed by a verifier and which will stay secret.

Enquiries about characteristics: The eID system must include specific privacy-preserving queries to prevent the revelation of extraneous attributes. Binary outcomes for designated searches are preferable than supplying the actual content (see to Section 3.2). It is essential to recognise that while these searches provide certain information, they may still be subject to further human oversight (Req. 4.b).

5. CONCLUSIONS

This article presents the concept of a privacy-preserving mobile eID designed for real-world identification akin to conventional ID cards, while also offering flexibility for use in private sectors. As a first step towards that aim, we delineated the overarching framework of such a system and introduced some actual applications. We presented the notion of privacy-preserving attribute searches and outlined the essential needs for a system that meets the stringent security standards of governmental electronic IDs, such as driver's licenses and passports.

To our knowledge, no solution presently exists that meets all these criteria. A viable solution would likely integrate various techniques: NFC secure elements (SE) for confidentiality, code isolation, and power-off support; group signatures for privacy-preserving identity verification; a scheme as referenced in [7] for scalable, unlinkable, and offline revocation checks; and a secure channel protocol as outlined in [6] for ensuring confidentiality and integrity during communication with the SE, among others.

REFERENCES

[1]. J. Bringer, H. Chabanne, R. Lescuyer, and A. Patey. Efficient and Strongly Secure Dynamic Domain-Specific Pseudonymous Signatures for ID Documents. In Financial Cryptography and Data Security, pages 255–272. Springer, 2014.

[2]. J. Bringer, H. Chabanne, R. Lescuyer, and A. Patey. Hierarchical Identities from Group Signatures and Pseudonymous Signatures. In The New Codebreakers, pages 457–469. Springer, 2016.

[3]. BSI. Kryptographische Verfahren: Empfehlungen und Schlu¨ssella¨ngen. Technical Report TR-02102-1 v2016-1, Feb. 2016.

[4]. J. Camenisch and M. Stadler. Efficient group signature schemes for large groups. In Advances in Cryptology, CRYPTO ’97, pages 410–424. Springer, Aug. 1997.

[5]. D. Chaum and E. Van Heyst. Group signatures. In Advances in Cryptology, EUROCRYPT ’91, pages 257–265. Springer, 1991.

[6]. M. H¨olzl, E. Asnake, R. Mayrhofer, and M. Roland. A Password-authenticated Secure Channel for App to Java Card Applet Communication. International Journal of Pervasive Computing and Communications (IJPCC), 11:374–397, Oct. 2015.

[7]. V. Kumar, H. Li, J.-M. J. Park, K. Bian, and Y. Yang. Group Signatures with Probabilistic Revocation: A Computationally-Scalable Approach for Providing Privacy-Preserving Authentication. In Proc. CCS 2015, pages 1334–1345. ACM, 2015.

[8]. A. Lehmann et al. Survey and Analysis of Existing eID and Credential Systems. FutureID Deliverable D32.1, Apr. 2013.

[9]. W. Lueks, G. Alpa´r, J.-H. Hoepman, and P. Vullers. Fast revocation of attribute-based credentials for both users and verifiers. In ICT Systems Security and Privacy Protection, pages 463–478. Springer, 2015.

[10]. H. K. Maji, M. Prabhakaran, and M. Rosulek. Attribute-Based Signatures. In Topics in Cryptology, CT-RSA 2011, pages 376–392. Springer, Feb. 2011.

[11]. T. Nakanishi and N. Funabiki. Verifier-Local Revocation Group Signature Schemes with Backward Unlinkability from Bilinear Maps. In Advances in Cryptology, ASIACRYPT 2005, pages 533–548.

[12]. Springer, Dec. 2005.

[13]. T. Nyman, J.-E. Ekberg, and N. Asokan. Citizen electronic identities using TPM 2.0. In Proceedings of the 4th International Workshop on Trustworthy Embedded Devices, pages 37–48. ACM, 2014.