Visnyk NTUU KPI Seriia – Radiotekhnika Radioaparatobuduvannia, 2025, Iss. 102, pp. 66–82

UDC 681.3.002

Electronic Identity Documents Security During the Document Transfer Phase

Leliak A. V., Astrakhantsev A. A.

National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", Kyiv, Ukraine

E-mail: *lelyak.andrey@gmail.com*

The subject matter of the article is security of electronic documents. Electronic documents security importance rises with the creation of new international standards for electronic identification (eID) and their adoption across the world. The eID security is crucial for both citizens and the government to ensure mutual trust and provide confidentiality, integrity and availability. Previous researches provide either a too general view on eID security without a required level of technical details, or analyze deeply a single specific domestic solution that doesn't have worldwide interoperability. The goal is to evaluate security for the main electronic documents implementations compliant with the international standards, perform their comparative analysis, and provide security improvements.
The tasks to be solved are to provide the eID security evaluation framework for assessing and comparing different implementation options of electronic identification solutions. Electronic identification security was analyzed only in the document transfer phase, and security targets like the holder documents storage authentication, the eavesdropping avoidance, the document cloning prevention, and verifier authentication were considered. The Electronic Machine Readable Travel Document (eMRTD), according to the ICAO Doc 9303, and the Mobile Driver License (mDL), according to the ISO/IEC 18013-5, solutions' security were analyzed. The main used method is a comparative analysis. Comparative analysis was provided for these implementation options to illustrate differences in reaching the same security targets, evaluate the overall security level, and emphasize existing trade-offs. The following results were obtained: security improvements were proposed to mitigate security threats like post-quantum cryptography attacks, attacks on the Diffie-Hellman key exchange, and hash collision attacks. Conclusions: Study findings can be used to improve the next revisions of ICAO of ISO/IEC specifications for electronic identification or to consider in the security design in a domestic server-based solution.

Keywords: electronic identification; eID; eMRTD; mDL; security; PQC; data security; cyber threats

DOI: 10.20535/RADAP.2025.102.66-82

1 Introduction

1.1 Motivation

Electronic identification (eID) is a digital solution that provides citizens with proof of identity. Most electronic documents are plastic cards with embedded RFID microchips and have a physical format, compliant with the ISO/IEC 7810 specification [1]: ID-1 for most banking cards and ID cards, ID-2 for older-style ID cards and visas, ID-3 for passport booklets, and ID-000 for mini-SIM cards. Also, it's possible to store the digital representation of the identity document in the digital wallet on a personal electronic device like a smartphone or a trusted remote server.

The main implementation options for electronic documents that ensure worldwide interoperability are compliance with the ICAO Doc 9303 specification [2] or with the ISO/IEC 18013-5 specification [3]. The ICAO Doc 9303 standardizes design, issuance, data structures, and security mechanisms for widely used machine-readable foreign passports. The ISO/IEC 18013-5 specification standardizes data and security models for electronic documents that can be fully virtual, without the corresponding previously issued physical card. Also, a lot of countries introduced domestic solutions that manage citizens' digital identity, but are incompatible with alternative solutions from other countries. Security of any implementation of a citizen ID is crucial for both citizens and the government. Confidentiality, integrity, and availability should be guaranteed for mutual trust between the issuing authority and the user. The attack surface for electronic documents is much wider compared to physical documents because more hardware and software are involved in all electronic document lifecycle phases. Also, the most widely spread solution is to generate an electronic document, which is connected to the previously issued physical document, which allows combining attack vectors from both document representations.

1.2 State of the Art

The ISO/IEC 18013-5 specification defines interfaces and protocols for an Mobile Driver License (mDL) holder on the user device, an mDL reader, and an issuing authority infrastructure. Because of this, many national deployments and vendor solutions describe themselves as "ISO-compliant" to provide international interoperability. In North America, the American Association of Motor Vehicle Administrators (AAMVA) coordinates mDL adoption and crossjurisdiction interoperability [4]. As of 2025, multiple U.S. states (e.g., Utah, Maryland, and Virginia) enrolled in AAMVA's Digital Trust Service program, shifting from pilots to early deployment phases [5, 6]. In the European Union, the Directive (EU) 2025/2205 [7] requires member states to issue ISO/IEC 18013-5-compliant mobile driving licenses, which will be legally equivalent to physical permits. European pilot interoperability events – such as a 2025 test in Utrecht [8] – contribute towards cross-border recognition and interoperability. In South Korea, a pilot domestic mobile driving license project was launched in 2022 [9], and now it targets international standardization [10]. Similarly, in the Hong Kong Special Administrative Region, the government announced that ISO-compliant driver licenses will be launched in 2025 [11].

As the electronic documents adoption intensified and became more popular in the last 3-5 years, only papers from the last 2-3 years were considered for this literature review to ensure that all case studies from the countries adopting electronic documents are considered. The most recent and valuable researches are related to domestic national electronic identity (NeID) projects without worldwide interoperability in European countries, such as Denmark, Hungary, and Estonia. The ICAO Doc 9303-compliant solutions were mostly analyzed in previous years, as this is a relatively old and stable specification with a long history of adoption. However, to the author's best knowledge, the security analysis for an ISO/IEC 18013-5-compliant solution is not present yet in the public research field. The main reason for this is that at the time of this paper writing, no country has yet fully adopted the ISO/IEC 18013-5-compliant solution because the specification was released only in 2021, and only a few pilot projects are running at this moment.

Cybersecurity becomes a more interesting each year as new technologies and approaches appear and attackers improve their toolsets [12, 13] and more sophisticated protection mechanisms are suggested [14]. Tsap [15] showed that security is one of the major factors in the eID system's public acceptance. Modern Smart City security also usually depends on the electronic identity documents system security, as eID is used for authorization for certain services [16]. To address common security concerns, the electronic Identification, Authentication and Trust Services (eIDAS) regulation was created in Europe, and a large number of countries are putting significant efforts into being compliant with it [17].

General risks and challenges of national identity documents are described in two related works by Jide Edu et al. [18, 19]. The authors provided the risk assessment framework, comprehensive risk descriptions, and risk mitigation practices, but both papers are descriptive and lack technical details. Similarly, Pöhn et al. [20] modeled common threats to self-sovereign identities (SSI), which include spoofing, tampering, repudiation, information disclosure, denial of service, and elevation of privilege, but provided no technical details or correlations with international identity document standards.

The other side of the coin is publications about domestic electronic document solutions, which usually discuss relevant technical and infrastructure issues with a deep technical background, but their results are directly related only to a single domestic national identity document solution for a specific country. Such papers can be found for Danish, Hungarian, Estonian, and Spanish electronic document solutions. Kingo and Aranha [21] showed how Denmark moved from a legacy NemID system to a modern MitID system for identity management to mitigate identity theft and unavailability issues. Bærentzen et al. [22] also described how the MitID system is compliant with the eIDAS EU security requirements and what security challenges are not yet resolved. Nyári and Kerti [23] performed a comprehensive risk assessment for the Hungarian eID card, which can be reused for other domestic solutions, but still requires significant adaptation. Parsovs [24] discussed some specific security issues in the existing Estonian electronic documents infrastructure and proposed their mitigations, which are mostly relevant only to the Estonian solution. Correa-Marichal et al. [25] analyzed the security of the Spanish electronic documents solution based on the ICAO Doc 9303 security mechanisms, but assessment results are relevant only for this specific implementation.

Also, there is a set of studies that provide a deep analysis of some specific security aspects for electronic documents. Such studies can be reused for other solutions' security assessment, but the whole security landscape analysis and other aspects analysis are still needed. Radutoiu et al. [26] concentrated on document cloning and misuse prevention. Aichinger [27] provided a comprehensive security analysis for the electronic Machine Readable Travel Document (eMRTD) implementations with Extended Access Control (EAC) and Password Authenticated Connection Establishment (PACE) mechanisms. Similarly, Koziel [28] deeply analyzed the security of Password Authentication Connection Establishment (PACE) and Password Authentication Connection Establishment with Chip Authentication Mapping (PACE

CAM) protocols. Fischlin et al. [29] and Alnahawi et al. [30] discussed post-quantum security concerns for electronic documents. Nomis et al. [31] provided face morphing attacks detection approaches for electronic Machine Readable Travel Document (eMRTD) solutions.

As can be seen from the literature review, there is no complex study that covers multiple electronic document standards and compares security risks and mitigations in different implementations. In this work, the authors evaluated the security of the main electronic document implementation options and proposed security improvements to mitigate the security threats identified. Also, the comparative analysis was performed for these implementation options to illustrate differences in reaching the same security targets and emphasize existing tradeoffs.

1.3 Electronic Identity Documents Ecosystem

The electronic identity documents ecosystem (see Fig. 1) consists of three main actors: the issuing authority, the holder, and the verifier. Usually, in a single ecosystem, there is a small number of issuers (e.g., governments of different countries), a medium number of verifiers (various services, which require identification), and a large number of holders (citizens). Issuer verifies identity of individuals applying for an eID and then creates and personalizes eIDs, embedding the individual's personal information in the physical (smart cards) and/or digital formats. Also, the issuer performs Public Key Infrastructure (PKI) certificate management. In most cases, the issuing authority is some government institution, but non-government authorized entities like banks can also act as issuers.

Verifier requests identity document to verify the holder's identity (e.g., airport security officer) or authorize the holder for some action (e.g., online liquor store). After the holder's document is received, the verifier checks transferred data integrity, verifies identity genuineness, and optionally makes a decision if the holder is eligible for performing certain actions (e.g., entering a protected area). A verifier can be represented by a fully automated device like a smart door lock, as well as by an officer with a verifier device, which displays a read document on its screen.

The holder stores the identity document on some electronic device after it was issued by an issuing authority. Then the holder represents the credential to a verifier by request to authenticate themselves. Typically, a holder is a citizen with a mobile phone or smart card that contains an electronic identity document.

1.4 Electronic Identity Document Lifecycle

According to the "Generic system architectures of mobile eID systems" specification [33], there are the following phases in the identity document lifecycle (Fig. 2): initialization phase, installation phase, issuing phase, operation phase, and removal phase.

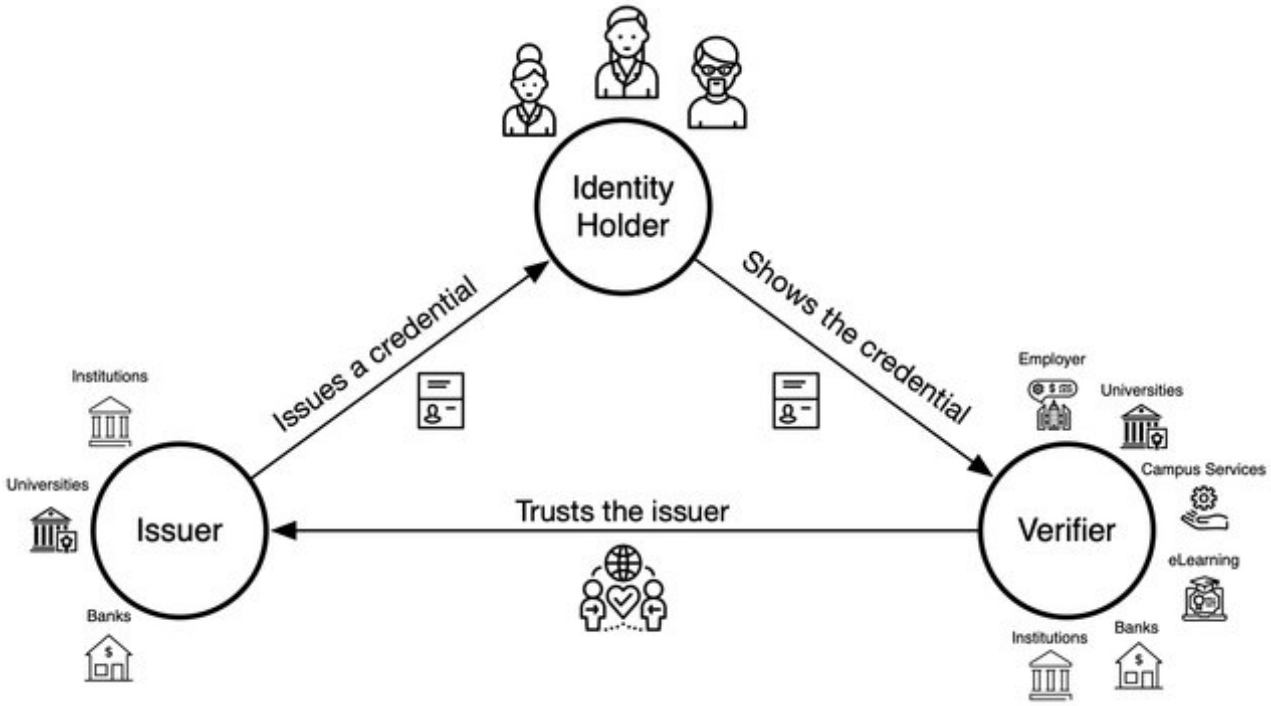

Fig. 1. Electronic identity documents ecosystem [32]

The operational phase is the most vulnerable phase and the greatest interest to the attacker. Firstly, the operational phase is much longer than the other ones, as identity document storage and usage may be performed even for years, meanwhile document issuing and revocation usually are performed during a single day. Secondly, the operational phase contains repeatable actions in each transaction, which are partially similar between document reading iterations. A malicious actor can try to attack the system a large number of times during the phase duration, and also has the possibility to exploit the similarity between document retrieval transactions.

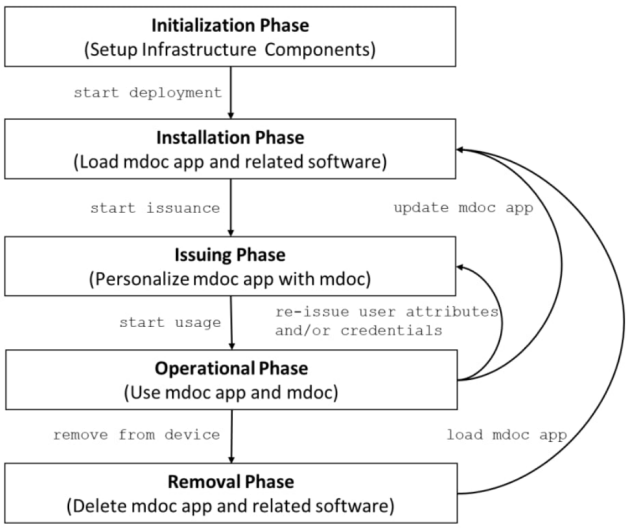

Fig. 2. Electronic identity document lifecycle

So, for this research purpose, let us consider four main phases in the identity document lifecycle:

1. Issuing phase: installing and personalizing the document by an issuer on a holder device;
2. Storage phase: document storage on a holder device;
3. Transfer phase: document transfer from a holder device to a verifier device;
4. Removal phase: document revocation by an issuer from a holder device.

The issuing phase contains verification of the holder's identity and document provisioning using implementation-specific protocols. Also, the holder's device security is evaluated if a personal device (smartphone) is used instead of a standard separate hardware (passport with chip). Document issuing can be performed using online (via the Internet) or offline communications (e.g., via NFC).

The storage phase usually implies the usage of some hardware-backed secure area like an embedded secure element (eSE) to comply with the issuer's security requirements. If a document size is comparable to the secure area available memory, the whole document can be stored in a secure area to reach a maximum security level. Alternatively, if memory restrictions exist, only document cryptographic keys can be stored in a secure area, and document data itself can be stored in a less protected storage in an encrypted form.

The transfer phase includes document data transfer using application-level protocols that are defined by an implementation. These protocols can use NFC, BLE, Wi-Fi Aware for data and key exchange, like [34], and various protocols over the Internet.

The revocation phase is performed by an issuer using implementation-specific protocols, mainly via the Internet. If the holder device is unreachable, the document will anyway be marked as revoked by an issuer, and the next transaction between the holder and verifier will fail.

1.5 Objectives and Approach

The research goal is to describe security threats for the document data transfer phase from the holder device (citizen) to the verifier device (e.g., state officer) and provide mitigation mechanisms for identified security threats.

The main stages of this research are as follows:

- Stage 1. Describe security targets to analyze and main eID implementation options. Analyze how each security target is reached for each implementation option (Section 2);
- Stage 2. Provide security improvements for each implementation option to better reach the security targets (Section 3);
- Stage 3. Perform a comparative analysis of different eID implementation options security (Section 4).

2 Security Analysis

2.1 Security Targets

During the whole identity document lifecycle, the following security targets should be reached:

- Authenticate holder document storage;
- Authenticate document data;
- Authenticate verifier entity;
- Avoid document data skimming;
- Avoid eavesdropping during document transfer;
- Avoid document cloning.

Security targets relevant for the issuing phase are holder document storage authentication, document cloning, and eavesdropping avoidance. Firstly, the issuer should authenticate the holder person and verify that the holder device meets its security requirements. Secondly, the document should be bound to the holder person and to the holder device during the personalization procedure to avoid document cloning. Also, the

document transfer should be carefully protected from eavesdropping during this phase to avoid document cloning and other issues.

During the storage phase, the main security concern is to prevent unauthorized document access, especially data skimming. The attacker can try to access document data either from the same device using another malicious program or using an unauthorized verifier device.

As was mentioned in the previous section, the transfer phase is the most vulnerable one, and most of the security targets are applicable to it. The holder document storage and the document data itself should be authenticated by the verifier to ensure document genuineness. On the other hand, the holder should authenticate the verifier before releasing their personal identifiable information to it. Additionally, most policies explicitly enforce that user should voluntarily give consent for document sharing before sending it to the verifier. Also, the data transfer channel should be protected to avoid eavesdropping during communication between the verifier and the holder.

Security targets relevant for the removal phase are holder document storage authentication, document data authentication, and eavesdropping protection. The issues should authenticate the holder's document storage and document data to ensure that a proper document is revoked. Also, the communication channel should be protected from eavesdropping to avoid fake revocation attacks on other users.

Only the document transfer phase security will be analyzed in this paper.

2.2 eID Implementation Options

As was mentioned earlier, the main eID implementation options are a biometric passport with a chip, which is compliant with ICAO Doc 9303, a mobile driving license, which is compliant with ISO/IEC 18013-5, and domestic solutions. Let us choose the following three specific options for security principles comparison:

1. Biometric passport: TD3 eMRTD implementation according to the ICAO Doc 9303;
2. Mobile driving license on Android phone: mDL implementation according to the ISO/IEC 18013-5;
3. A domestic server-based solution with an application on an Android phone.

The main differences between these implementation options are cryptography approaches, communication protocols, and the data storage approach. eMRTD and mDL cryptography approaches and communication protocols are defined in the corresponding specifications. For the server-based solution, let us assume an Android app, which acts as a simple client, that communicates via TLS with the remote protected server, where most of the business logic is placed.

In eMRTD implementation, the document data is fully stored in the chip. For the mDL implementation, let us consider the option where the document is stored in limited-access storage in encrypted form, and all keys are stored in the embedded secure element (eSE). In the server-based solution, the original document is stored on the server, and the client holds only its temporary copy. As most of current server-based electronic document deployments are domestic non-standardized solutions with high volatility in implementation specifics, we will not include them in the security comparison.

2.3 Holder Document Storage Authentication

2.3.1 Electronic Machine Readable Travel Document

The eMRTD document storage is authenticated using the Chip Authentication described in the ICAO Doc 9303 specification [2] under section 6.2. The Chip Authentication mechanism is an ephemeral-static Diffie-Hellman key agreement protocol that provides secure communication and unilateral authentication of the eMRTD chip. The protocol description is as follows:

1. The holder's static DH private key is securely stored in the chip's memory;
2. The holder sends its static DH public key PK_{IC} and domain parameters D_{IC} to the verifier. The Passive Authentication mechanism protects the public key integrity and authenticity;
3. The verifier generates an ephemeral DH key pair ($SK_{DH,IFD}$, $PK_{DH,IFD}$, D_{IC}) and sends the ephemeral public key $PK_{DH,IFD}$ to the eMRTD chip;
4. Both the holder and the verifier compute the shared secret
$$K = KA(SK_{IC}, PK_{DH,IFD}, D_{IC}) = \\ = KA(SK_{DH,IFD}, PK_{IC}, D_{IC}) \quad (1)$$
using the key agreement function;
5. Both the holder and the verifier derive session keys
$$KS_{MAC} = KDF_{MAC}(K) \quad (2)$$
$$KS_{Enc} = KDF_{Enc}(K) \quad (3)$$
for secure messaging.

The key agreement algorithm is an anonymous Diffie-Hellman Key Agreement, and the key derivation function is a custom algorithm based on SHA-1. Session keys generation is supported for 3DES and AES algorithms in CBC mode with 112- and 128/192/256-bit key sizes, respectively.

The Chip Authentication flow is shown in Fig. 3.

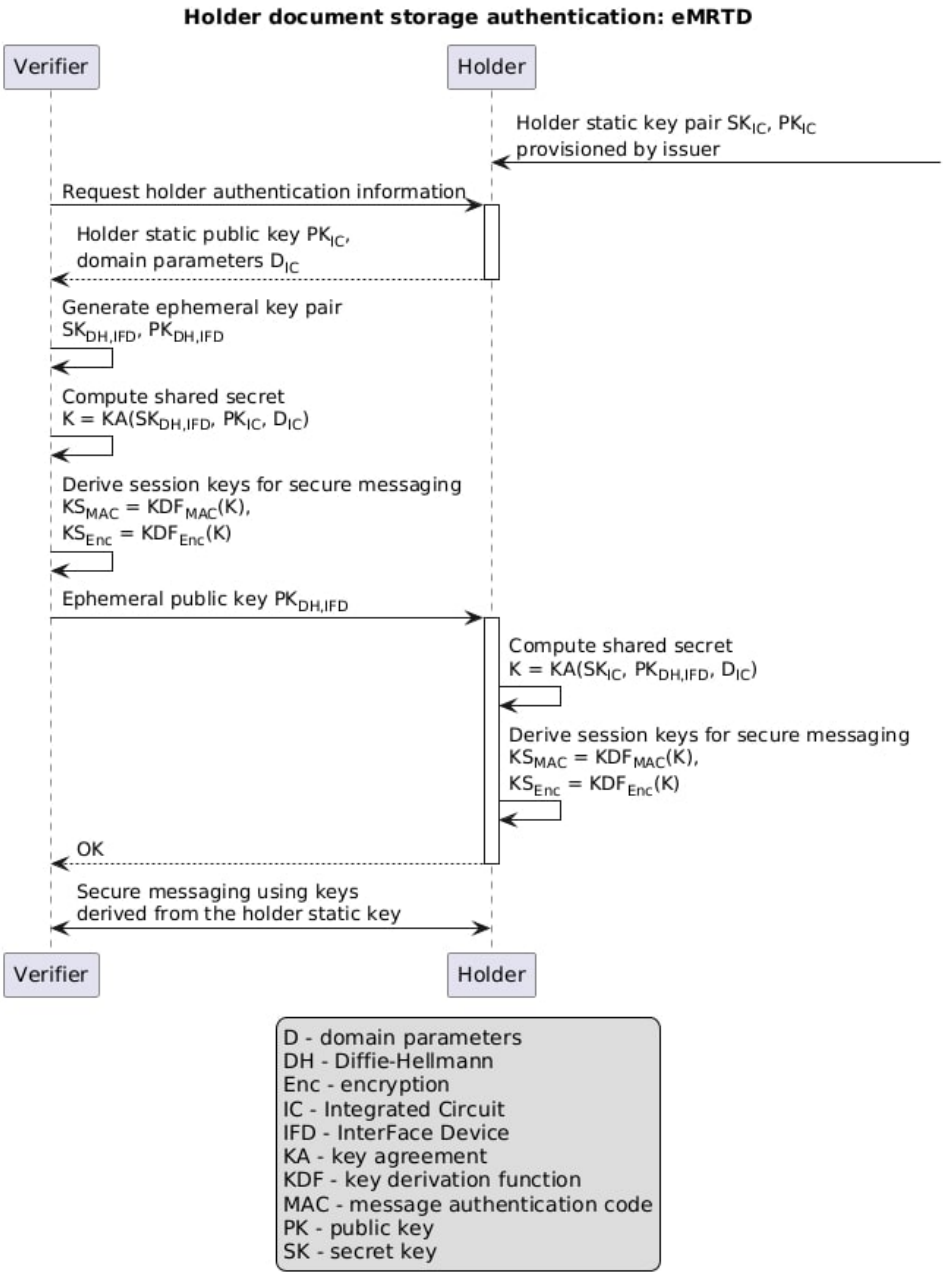

Fig. 3. Holder document storage authentication for eMRTD document

2.3.2 Mobile Driving License

The mDL document storage is authenticated using the mdoc authentication mechanism described in the ISO/IEC 18013-5 specification [3] under section 9.1.3. This mechanism purpose is to prevent cloning of the mdoc and to mitigate man-in-the-middle attacks using ECDH-agreed MAC or ECDSA/EdDSA signature. The protocol description is as follows:

1. The holder's private key SDeviceKey.Priv is securely stored in a device's memory;
2. The holder's public key SDeviceKey.Pub is sent to the verifier and protected by an issuer data authentication mechanism;
3. The verifier's ephemeral public key EReaderKey.Pub is sent to the holder inside the document request message;
4. If ECDH-agreed MAC is used:
   (a) The holder and verifier calculate the shared secret

$$Z_{AB} = KA((SDeviceKey.Priv, EReaderKey.Pub)) = KA((EReaderKey.Priv, SDeviceKey.Pub)) \quad (4)$$

(б) The holder and verifier derive the MAC key using the shared secret and the session context

$$EMacKey = KDF((Z_{AB}, SHA256(SessionTranscriptBytes))) \quad (5)$$

(в) The holder calculates the MAC of the "Device Authentication" session-specific structure, and the verifier validates it:

$$DeviceMac = \\ = MAC(DeviceAuthenticationBytes) \quad (6)$$

5. If ECDSA/EdDSA signature is used

(а) The holder signs the "Device Authentication" session-specific structure, and the verifier validates the signature:

$$DeviceSignature = \\ = Sign(DeviceAuthenticationBytes) \quad (7)$$

The key agreement protocol is Elliptic Curve Key Agreement Algorithm – Diffie-Hellman (ECKA-DH). The key derivation function is HMAC Key Derivation Function (HKDF). The MAC algorithm is HMAC with SHA-256 (HMAC 256/256). The signature algorithm can be one of the following: ECDSA with SHA-256 (ES256), ECDSA with SHA-384 (ES384), ECDSA with SHA-512 (ES512), or EdDSA. The mdoc authentication mechanism is shown in Fig. 4.

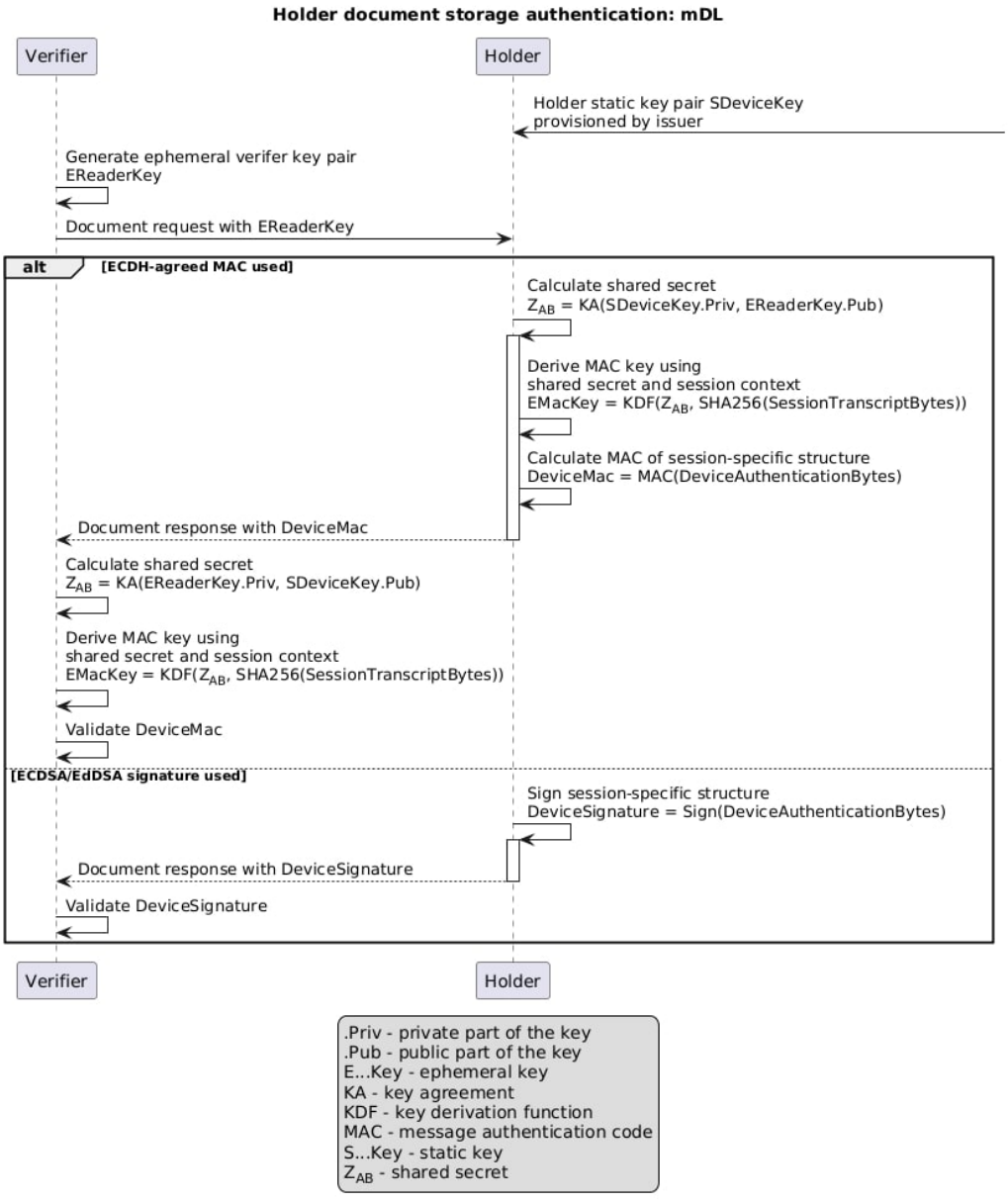

Fig. 4. Holder document storage authentication for mDL document

2.4 Holder document data authentication

All implementation options use roughly the same approach. The document data should be signed by an issuer, the holder should provide the document signature to the verifier, and the verifier should validate that signature using the issuer certificate. The protocol description is as follows:

1. The issuer's private key SK_I is stored securely inside the issuing authority infrastructure.
2. The issuer signs document data or metadata with its private key:

$$Signature = Sign(SK_I, Document) \quad (8)$$

3. The issuer provisions this signed document data with the issuer certificate containing the public key PK_I during the document issuing phase:

$$ProvisioningData = \\ = Document|Signature|PK_I \quad (9)$$

4. Holder sends the ProvisioningData data to verifier in reply to the verifier's request.
5. Verifier validates a certification path from a Trust Anchor to the issuer's certificate containing PK_I.
6. Verifier validates document signature:

$$Verify(PK_I, Signature) \quad (10)$$

The eMRTD document data is authenticated using the Passive Authentication mechanism described in the ICAO Doc 9303 specification [2] under section 5.1. The mDL document data is authenticated using the Issuer Data Authentication mechanism described in the ISO/IEC 18013-5 specification [3] under section 9.1.2.

The document data that the issuer signs can differ. The eMRTD's "Document Security Object" and mDL's "Mobile Security Object" contents differ, but both of them contain signed hashes of document data, which gives the possibility to ensure the document data integrity. The eMRTD ecosystem supports the following signature algorithms: RSA with SHA-1, SHA-224, SHA-256, SHA-384, or SHA-512, and ECDSA with SHA-224, SHA-256, SHA-384, or SHA-512. The mDL ecosystem supports the following signature algorithms: ECDSA with SHA-256 (ES256), ECDSA with SHA-384 (ES384), ECDSA with SHA-512 (ES512), or EdDSA. The document data authentication flow is shown in Fig. 5.

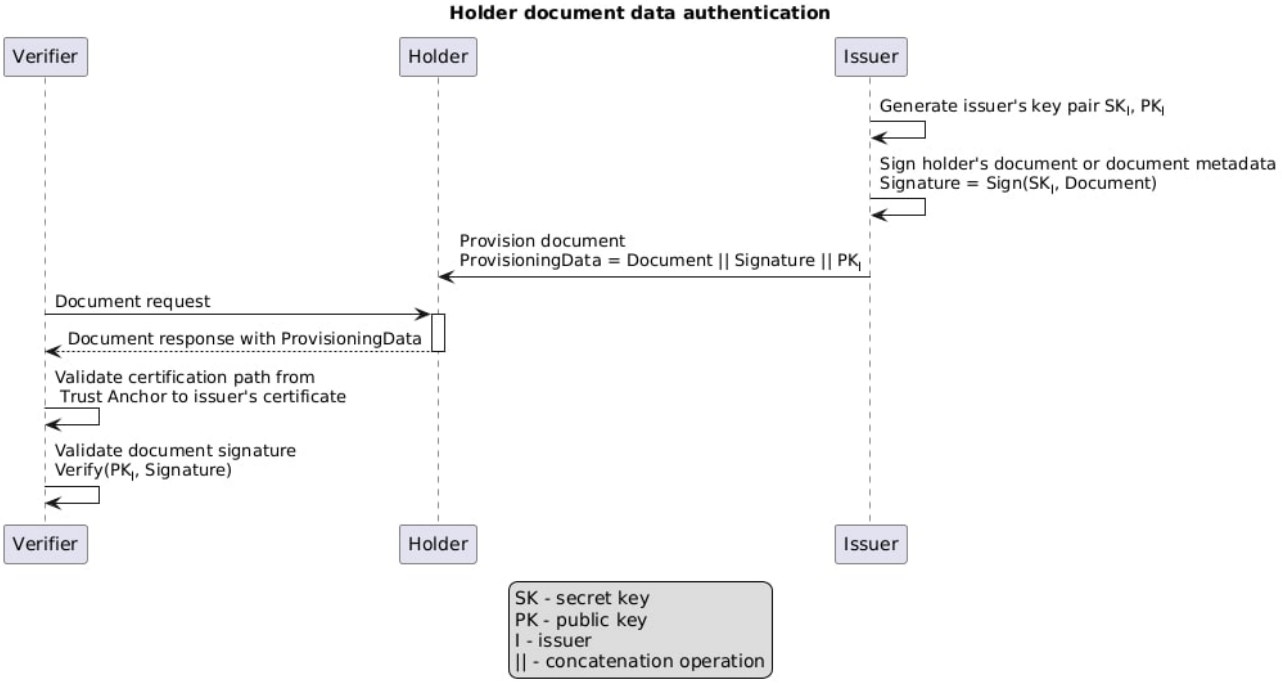

Fig. 5. Holder document data authentication for all document types

2.5 Verifier entity authentication

2.5.1 Electronic Machine Readable Travel Document

The eMRTD terminal device (verifier) is authenticated using the Terminal Authentication described in the ICAO Doc 9303 specification [2] under section 7.1. The Terminal Authentication mechanism is a two-move challenge-response protocol that provides explicit unilateral authentication of the terminal. The protocol description is as follows:

1. The verifier holds its static private key SK_{IFD} securely in the device memory;

2. The verifier sends a certificate chain to the holder. The chain starts with a CA certificate that is verifiable with the public key stored on the holder, and ends with the Terminal Certificate containing PK_{IFD};

3. The holder verifies certificates and extracts the verifier's public key PK_{IFD};

4. The holder randomly chooses a challenge r_{IC} and sends it to the terminal;

5. The verifier composes data to sign from the holder ID ID_{IC}, random challenge r_{IC} and public key $PK_{DH,IFD}$ computed from the previously executed Chip Authenticate protocol as follows. The holder ID should be the document MRZ data if the Basic Access Control mechanism was used, or the PACE ephemeral public key $PK_{DH,IC}$ if the PACE mechanism was used;

$$DataToSign = ID_{IC}||r_{IC}||PK_{IFD} \tag{11}$$

6. The verifier responds to the holder with the signature:

$$S_{IFD} = Sign(SK_{IFD}, DataToSign) \tag{12}$$

7. The holder verifies the signature:

$$Verify(PK_{IFD}, S_{IFD}, DataToSign) \tag{13}$$

Supported signature algorithms are RSA with SHA-256, RSA with SHA-512, ECDSA with SHA-224, ECDSA with SHA-256, ECDSA with SHA-384, and ECDSA with SHA-512.

The Terminal Authentication flow is shown in Fig. 6.

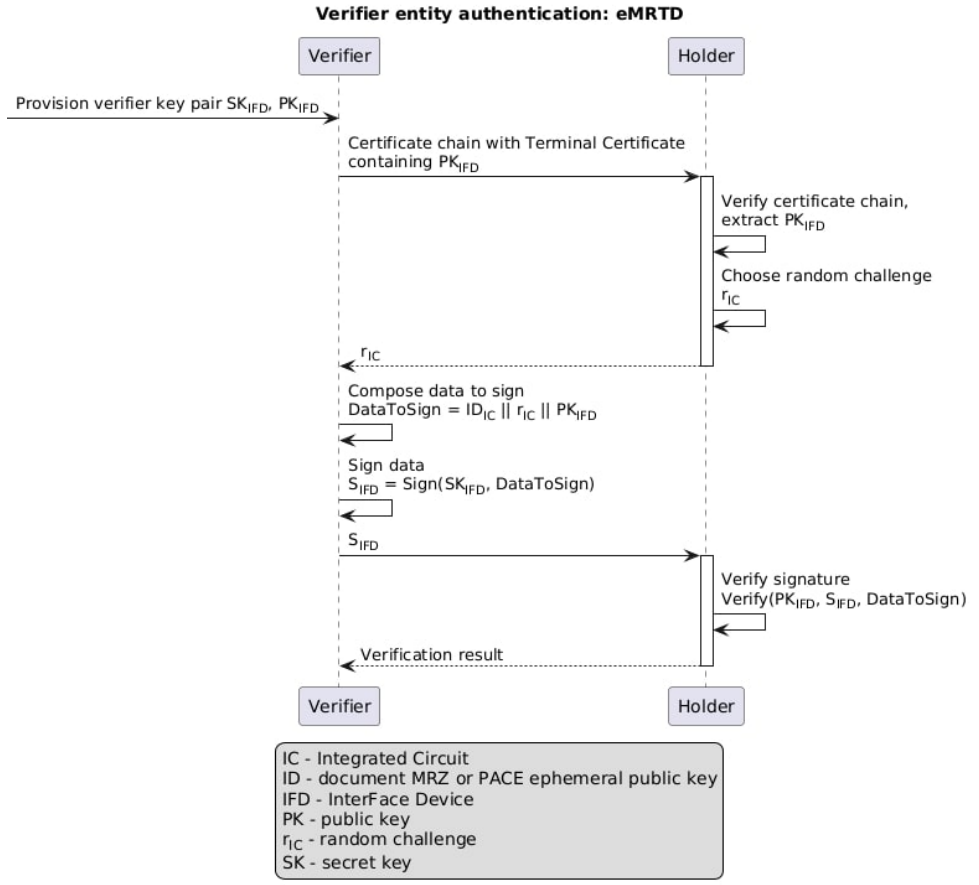

Fig. 6. Verifier entity authentication for eMRTD document

2.5.2 Mobile Driving License

The mDL verifier device is authenticated using the mdoc reader authentication mechanism described in the ISO/IEC 18013-5 specification [3] under section 9.1.4. This mechanism uses information stored in the mDL verifier to confirm that the mDL verifier and its request are authenticated using the ECDSA/EdDSA signature. The protocol description is as follows:

1. The verifier's private key SReaderKey.Priv is securely stored in the verifier's device memory;

2. The verifier builds the ReaderAuthentication structure, which contains the current session data and the verifier's request data;

3. The verifier signs the ReaderAuthentication:

$$ReaderSignature = \\ = Sign(ReaderAuthentication) \quad (14)$$

4. The verifier builds the ReaderAuth structure, which contains the ReaderSignature and the certificate chain with the verifier certificate containing the SReaderKey.Pub, but doesn't contain the ReaderAuthentication structure;

5. The verifier sends the ReaderAuth structure inside the encrypted document request;

6. The holder extracts the certificate chain from the ReaderAuth structure and verifies it;

7. The holder builds the ReaderAuthentication structure, extracts the verifier's signature from the ReaderAuth structure and verify it:

$$Verify(SReaderKey.Pub, \\ ReaderSignature, ReaderAuthentication) \quad (15)$$

Supported signature algorithms are ECDSA with SHA-256, SHA-384, SHA-512, and EdDSA. The mdoc reader authentication flow is shown in Fig. 7.

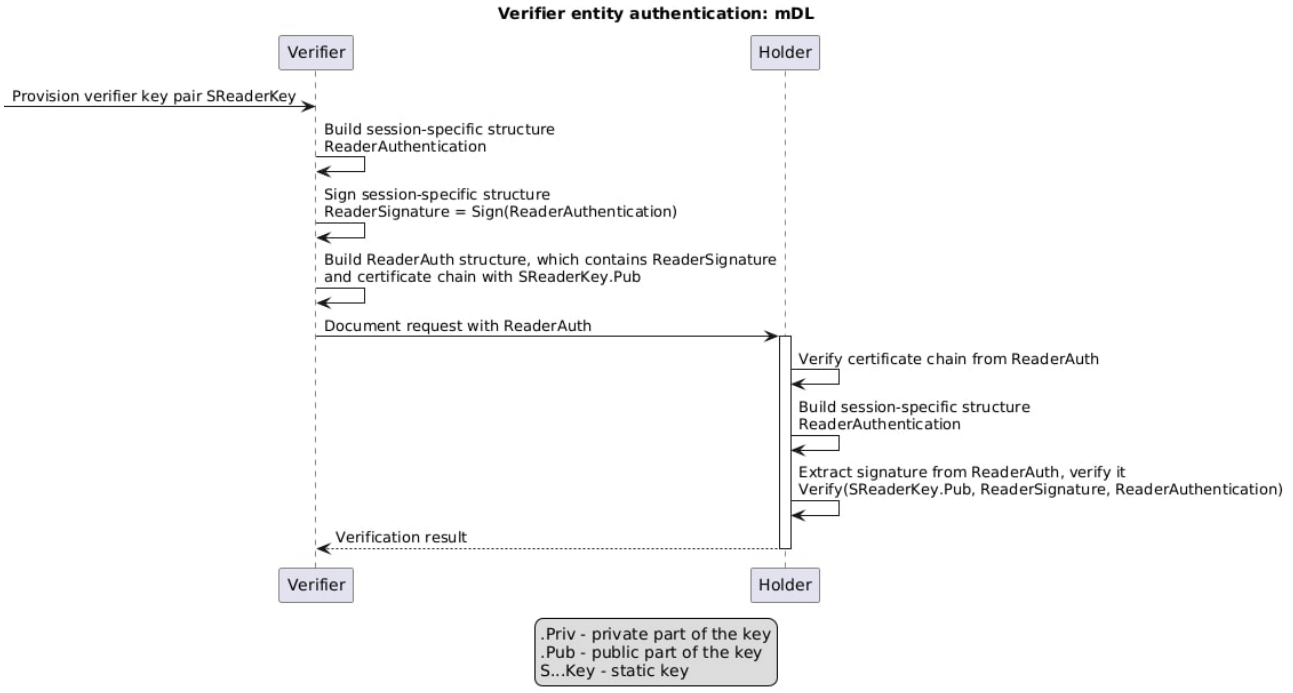

Fig. 7. Verifier entity authentication for mDL document

2.6 Eavesdropping avoidance during document transfer

2.6.1 Electronic Machine Readable Travel Document

The eavesdropping avoidance is ensured by the use of session encryption. The secure messaging is established using the Password Authenticated Connection Establishment (PACE) mechanism described in the ICAO Doc 9303 specification [2] under section 4.4. The PACE is a password-authenticated Diffie-Hellman key agreement protocol that establishes secure messaging based on weak passwords. The protocol description is as follows:

1. The holder static shared password π is available for visual scanning by the verifier. Shared passwords options are Machine Readable Zone (MRZ) or Card Access Number (CAN);

2. Both the holder and the verifier derive a shared key from the shared password:

$$K_\pi = KDF(\pi) \tag{16}$$

3. The holder randomly and uniformly chooses a nonce s, encrypts the nonce, and sends the ciphertext to the verifier:

$$z = E(K_\pi, s) \tag{17}$$

4. The verifier recovers the plaintex using the shared key:

$$s = D(K_\pi, z) \tag{18}$$

5. The holder and verifier exchange additional data required for the mapping of the nonce: ephemeral keys from the generic mapping or additional nonce for the integrated mapping;

6. Both actors compute the ephemeral domain parameters:

$$D = Map(D_{IC}, s, \ldots) \tag{19}$$

7. Both actors perform an anonymous Diffie-Hellman key agreement based on the ephemeral domain parameters and generate the shared secret:

$$\begin{aligned} K = KA(SK_{DH,IC}, PK_{DH,IFD}, D) = \\ = KA(SK_{DH,IFD}, PK_{DH,IC}, D) \end{aligned} \tag{20}$$

8. Both actors derive session keys:

$$KS_{MAC} = KDF(K) \tag{21}$$

$$KS_{MAC} = KDF(K) \tag{22}$$

9. The holder and verifier exchange and verify authentication tokens:

$$T_{IFD} = MAC(KS_{MAC}, PK_{DH,IC}) \tag{23}$$

$$T_{IFD} = MAC(KS_{MAC}, PK_{DH,IC}) \tag{24}$$

10. Both actors use session keys derived in step 8 for further messaging.

The key agreement algorithm is an anonymous Diffie-Hellman Key Agreement, and the key derivation function is a custom algorithm based on SHA-1. The AES-CBC algorithm is used for nonce encryption. AES-CMAC or 3DES in Retail-mode algorithms are used for the authentication token generation. AES-CBC, AES-CMAC, and 3DES-CBC algorithms can be used for secure messaging after successful PACE mechanism completion. The nonce mapping is a custom algorithm based on the anonymous Diffie-Hellman key agreement or on a custom pseudo-random function.

The session encryption flow is shown in Fig. 8.

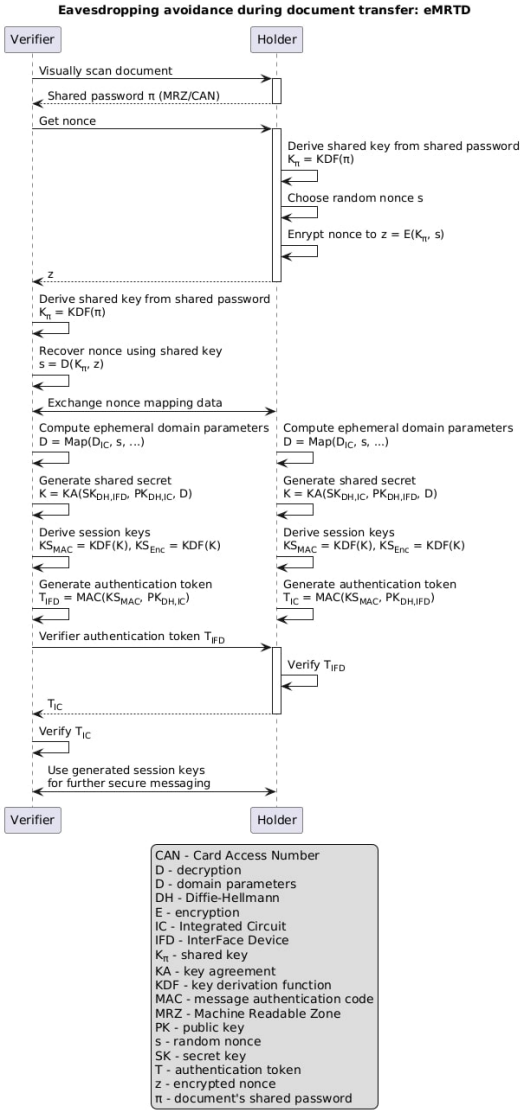

Fig. 8. Eavesdropping avoidance during document transfer for eMRTD document

2.6.2 Mobile Driving License

The eavesdropping avoidance in the mDL ecosystem is also ensured by introducing a session encryption, which is described in the ISO/IEC 18013-5 specification [3] under section 9.1.1. This mechanism uses standard ephemeral key ECDH to establish session keys for authenticated symmetric encryption. The protocol description is as follows:

1. The holder generates an ephemeral key pair (EDeviceKey.Priv,EDeviceKey.Pub) and sends the public key to the verifier during the device engagement procedure;

2. The verifier retrieves the elliptic curve from the device engagement structure and generates its own ephemeral key pair;

3. The verifier derives symmetric session keys SKReader and SKDevice using a key derivation function on the current session data, which contains both ephemeral public keys;

4. The verifier encrypts the document request using SKReader and sends an encrypted request with EReaderKey.Pub to the holder;

5. The holder derives symmetric session keys SKReader and SKDevice as was done in step 3, and decrypts the message with the SKReader key;

6. The holder encrypts its response using SKDevice, sends it to the verifier, and the verifier decrypts it using the same session key.

The key derivation function is HMAC-based Key Derivation Functions (HKDF), which uses SHA-256. The AES-256-GCM algorithm is used for the messages encryption.

The session encryption flow is shown in Fig. 9.

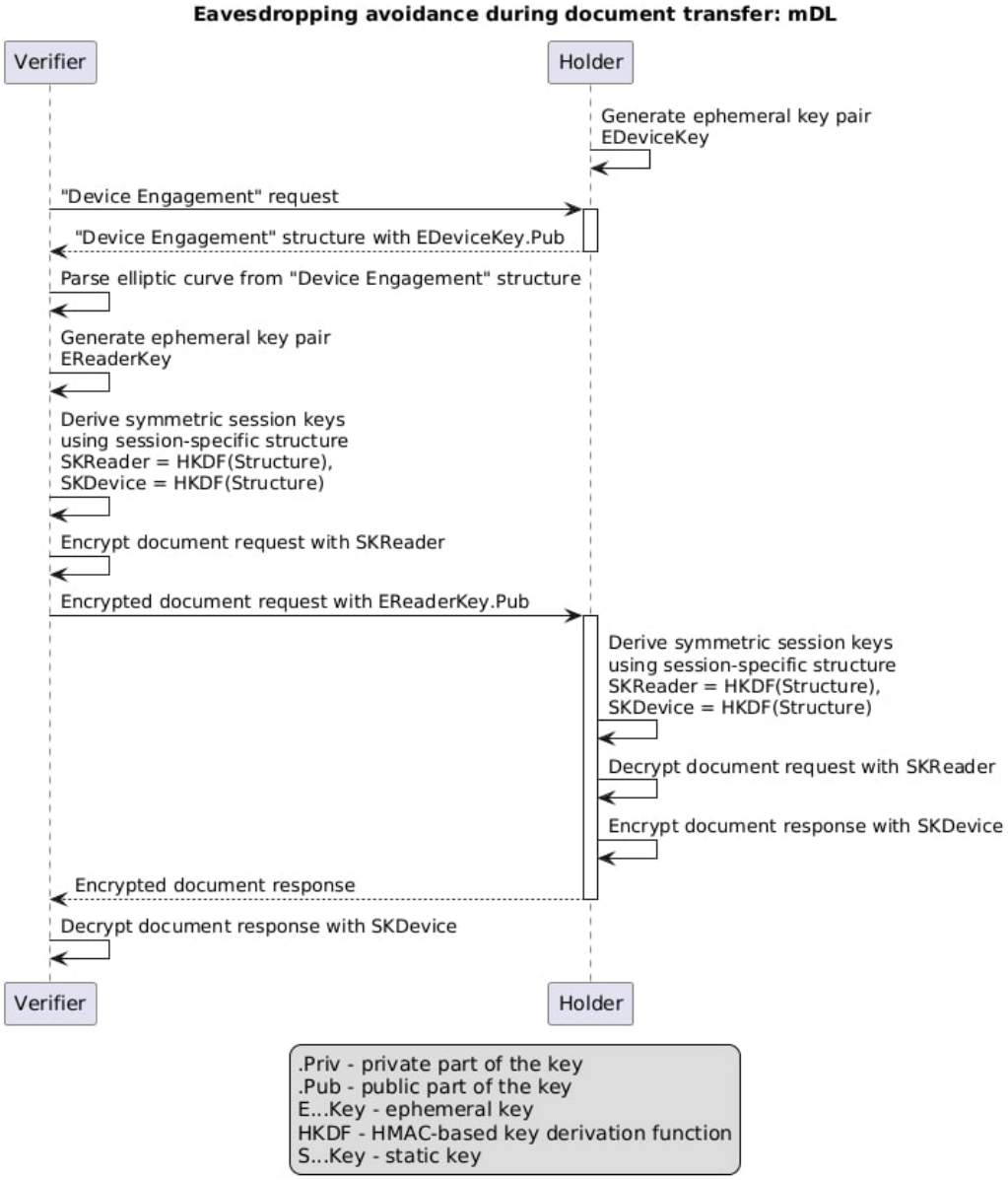

Fig. 9. Eavesdropping avoidance during document transfer for mDL document

3 Security Improvements

Possible attacks for the eMRTD and mDL documents security are:

- Man-in-the-middle [35] and various other attacks [36] on the Diffie-Hellman Key Exchange protocol: holder document storage authentication in eMRTD and mDL, and eavesdropping avoidance during document transfer in mDL;
- Collision attacks on the SHA-1 protocol [37, 38]: holder document storage authentication and eavesdropping avoidance during document transfer in eMRTD;
- Various attacks on the deprecated 3DES protocol: holder document storage authentication and eavesdropping avoidance during document transfer in eMRTD;
- "Chosen plaintext", "Chosen ciphertext", and "Padding oracle" attacks on AES-CBC algorithm: holder document storage authentication, and eavesdropping avoidance during document transfer in eMRTD;
- PQC attack, which breaks Diffie-Hellman using Shor's algorithm: holder document storage authentication in eMRTD and mDL, holder document data authentication in eMRTD and mDL, verifier entity authentication in eMRTD and mDL, and eavesdropping avoidance during document transfer in eMRTD;
- Various attacks on the custom pseudo-random function, which is typically less secure than state-of-the-art alternatives: eavesdropping avoidance during document transfer in eMRTD;
- User tracking between different verifiers because the same document data hashes are used: holder document data authentication in eMRTD and mDL.

Possible security improvements for the eMRTD and mDL documents security are:

- To mitigate MITM attack: use one of the enhanced versions of the Diffie-Hellman Key Exchange protocol (two verification stages [39], modular arithmetic equations [40], new shared secret key for each message [41], entities authentication [42], string comparison [43]);
- To mitigate 3DES-related attacks: remove 3DES protocol from the supported protocols list. Replace the 3DES protocol with ChaCha20 if more than one encryption protocol support is needed for interoperability reasons;
- To mitigate SHA-1 collision attacks: replace SHA-1 with SHA-2 or SHA-3 algorithms;
- To mitigate AES-CBC-related attacks: replace AES-CBC protocol with AES-GSM protocol to avoid AES-CBC vulnerabilities;
- To mitigate PQC attacks on asymmetric cryptography: replace Diffie-Hellman protocol with post-quantum key agreement protocols like ML-KEM, recommended by NIST. Replace ECDSA/EdDSA protocols with post-quantum digital signature algorithms like ML-DSA, recommended by NIST;
- To mitigate PQC attacks on symmetric cryptography: increase key size for AES and HKDF algorithms to keep required security level;
- To mitigate pseudo-random-related attacks: replace custom pseudo-random function with any of state-of-the-art algorithms: any AES-based PRF, ChaCha20, HMAC or HKDF;
- To mitigate user tracking attack: provide a mechanism to use a different document signature at each transaction to avoid user tracking.

4 Solutions Comparative Analysis

As was written in Section 2, the security of the three main implementation options was analyzed. It was assumed that eMRTD and mDL solutions are implemented strictly according to the corresponding specifications, following all security considerations mentioned there. Also, it's assumed that state-of-the-art crypto libraries were used in each solution to avoid implementation errors and timing attacks.

As shown in Table 1, there is a set of security threats that are applicable to any implementation option. The potential PQC attacks on asymmetric cryptography will be a serious threat soon for most systems. Attacks on Diffie-Hellman key exchange are also common for all solutions, as it's a standard way for secure generation of a symmetric key over a public channel.

Also, user depersonalization on repeated communications using issuer signature on document data and metadata is a domain-specific issue, and any implementation option will suffer from it until an improved mechanism is provided.

5 Discussion

The eMRTD solution has a larger number of possible attacks, mostly because of the wider range of protocols used. As the ICAO eMRTD specification is an older standard, it should maintain backward compatibility with previously used protocols like 3DES, SHA-1, and AES-CBC, which are considered less secure nowadays and have better alternatives. Also, as the eMRTD holder solution hardware is a biometric

Table 1 Security comparative analysis

Attack		Is the solution vulnerable?	
Category	Target	eMRTD	mDL
PQC	DH key exchange	Yes	Yes
	ECDSA or RSA digital signatures	Yes	Yes
Specific protocols	DH key exchange: MITM and other	Yes	Yes
	SHA-1 collision	Yes	No
	Various 3DES attacks	Yes	No
	Various AES-CBC attacks	Yes	No
Other	Various AES-CBC attacks	Yes	Yes
	Custom PRF attack	Yes	No
	Side-channel attacks	Yes	No

password, it's more vulnerable to various side-channel attacks like timing or power analysis attacks because of the possibility of direct chip access.

The mDL shows better results mostly because of usage of modern crypto protocols and much harder realization of side-channel attacks. Also, the mDL ecosystem allows the "online retrieval" approach using communication via WebAPI, Rest API or OIDC, which allows usage of mDL as a server-based solution.

The most effective improvements for any eID system security will be the use of post-quantum crypto algorithms and protection from MITM attacks on the Diffie-Hellman key exchange. The eMRTD solution can be improved by deprecating outdated, insecure algorithms, e.g., by introducing a new security profile with state-of-the-art algorithms that should be used for any new deployment. As far as the authors know, there is no standardized solution to avoid user tracking via issuer signatures on a document's metadata, so this is a field for further research.

Consclusions

The security of electronic identity documents during the transfer phase between holder and verifier was analyzed across three main implementation options: Electronic Machine Readable Travel Document (eMRTD) and Mobile Driving License (mDL). The following security targets were analyzed: holder document storage authentication, document data authentication, verifier entity authentication, document data skimming avoidance, eavesdropping avoidance, and document cloning avoidance. For each security target, existing security mechanisms were described for each implementation option, security threats were identified, and security improvements were proposed. To the best of the authors' knowledge, this is the first comprehensive comparative analysis of the eIDs implementation options mentioned above. This study presents a novel evaluation framework, which allows to analyze document transfer security for both the document holder, verifier, and document transportation between them.

Our findings reveal several underreported security threats to the various eID implementations. In all implementation options, all security mechanisms that use the Diffie-Hellman key exchange protocol or ECDSA/RSA digital signatures are vulnerable to post-quantum cryptography attacks. Similarly, the Diffie-Hellman key exchange protocol makes mechanisms that use it vulnerable to man-in-the-middle attacks. Also, user tracking via issuer signature on the document data is an issue for all implementation options. The Electronic Machine Readable Travel Document implementation suffers from outdated protocols like 3DES and SHA-1 usage, which are left in the current specification version for backward compatibility. Also, the eMRTD solution is more vulnerable to side-channel attacks because of its form factor.

This work proposes a set of security recommendations to avoid the security threats listed above. Security improvements include NIST-recommended PQC cryptographic algorithms for the key agreement protocol and digital signatures, or at least the usage of the enhanced version of the Diffie-Hellman key agreement protocol to avoid MITM attacks. Outdated protocols like 3DES, SHA-1, and AES-CBC should be replaced with AES, SHA-2/3, and AES-GCM, respectively. Similarly, the custom pseudo-random functions should be replaced with the state-of-the-art PRFs.

The comparative analysis of different implementation options security advances the current understanding of trade-offs between different implementation types. It shows that eMRTD and mDL solutions provide roughly the same level of security if the eMRTD security profile can forbid outdated protocols. The eMRTD solution is more vulnerable to side-channel attacks than the mDL solution, but it pays off with full issuer control of the holder's document, including the hardware part.

Overall, this work contributes to the electronic documents security by providing a systematical comparison of three most widely used eIDs implementation options, identification of major security threats for them, and recommending a novel set of security enhancements for future eIDs infrastructure resilience.

Further research direction are as follows:

- Avoid user tracking via issuer signatures on a document's metadata;
- Perform comparative security analysis for the document storage phase;
- Perform comparative security analysis for the document issuing and removal phases.

Contributions of authors: conceptualization, methodology, analysis, writing – Andrii Leliak; scientific supervising, review and editing – Andrii Astrakhantsev.

All the authors have read and agreed to the published version of this manuscript.

Conflict of Interest

The authors declare that they have no conflict of interest in relation to this research, whether financial, personal, author ship or otherwise, that could affect the research and its results presented in this paper.

Financing

This study was conducted without financial support.

Data Availability

The manuscript has no associated data.

Use of Artificial Intelligence

The authors confirm that they did not use artificial intelligence methods while creating the presented work.

References

[1] ISO/IEC 7810:2019 Identification cards — Physical characteristics. (2019). *International Organization for Standardization (ISO)*.

[2] Doc 9303 Machine Readable Travel Documents. (2021). *International Civil Aviation Organization (ICAO)*.

[3] ISO/IEC 18013-5:2021 Personal identification — ISO-compliant driving licence. Part 5: Mobile driving licence (mDL) application. (2021). *International Organization for Standardization (ISO)*.

[4] Mobile Driver License. *American Association of Motor Vehicle Administrators (AAMVA)*.

[5] AAMVA's Mobile Driver License Digital Trust Service is Now Live. (2025). *AAMVA News*.

[6] Virginia Added to AAMVA's Digital Trust Service. (2025). *AAMVA News*.

[7] Directive (EU) 2025/2205 — Union standard specifications on driving licences and mutual recognition. 22 Oct., (2025). *Official Journal of the European Union*.

[8] February 2025: acceleration for the European digital driving licence. (2025). *Digital-Identity-Wallet.eu*.

[9] Mee-yoo, K. (2022). Korea launches mobile driver's license trial. *The Korea Times*.

[10] Liang, L.-H. (2025). South Korea's ETRI sets sights on international standard for digital ID wallets. *Biometric Update*.

[11] Mobile drivers licenses to launch in Hong Kong in 2025. (2024). *Biometric Update*.

[12] Astrakhantsev, A., & Pedan, S. (2024). Improving user security during a call. *Radioelectronic and Computer Systems*, Vol. 2024, No. 2, pp. 173–185. doi:10.32620/reks.2024.2.14.

[13] Akouhar, M., Abarda, A., El Fatini, M., & Ouhssini M. (2025). Enhancing credit card fraud detection: the impact of oversampling rates and ensemble methods with diverse feature selection. *Radioelectronic and Computer Systems*, Vol. 2025, No. 1, pp. 85–101. doi:10.32620/reks.2025.1.06.

[14] Benadjila, R., Feneuil T., & Rivain, M. (2024). MQ on my Mind: Post-Quantum Signatures from the Non-Structured Multivariate Quadratic Problem. *2024 IEEE 9th European Symposium on Security and Privacy (EuroS&P)*, pp. 468-485, doi: 10.1109/EuroSP60621.2024.00032.

[15] Tsap, V. (2022). eID Public Acceptance: Success Factors, Citizen Perception, and Impact of Electronic Identity, doctoral thesis. *TALLINN UNIVERSITY OF TECHNOLOGY*, 175 p.

[16] Tok, Y. C., & Chattopadhyay, S. (2023). Identifying threats, cybercrime and digital forensic opportunities in Smart City Infrastructure via threat modeling. *Forensic Science International: Digital Investigation*, Vol. 45, 301540. doi:10.1016/j.fsidi.2023.301540.

[17] Sharif, A., Ranzi, M., Carbone, R., Sciarretta, G., Marino, F. A., & Ranise, S. (2022). The eIDAS Regulation: A Survey of Technological Trends for European Electronic Identity Schemes. *Applied Sciences*, Vol. 12, Iss. 24, 12679. doi:10.3390/app122412679.

[18] Edu, J., Hooper, M., Maple, C., & Crowcroft, J. (2023). Exploring the Risks and Challenges of National Electronic Identity (NeID) System. *IET Conference Proceedings CP846*, Vol. 2023, No. 14, pp. 118-123, doi:10.48550/arXiv.2310.15813.

[19] Edu, J., Hooper, M., Maple, C., & Crowcroft, J. (2023). An Impact and Risk Assessment Framework for National Electronic Identity (eID) Systems. *International Conference on AI and the Digital Economy*, Vol. 2023, pp. 124-133, doi:10.48550/arXiv.2310.15784.

[20] Pöhn, D., Grabatin, M., & Hommel, W. (2023). Modeling the Threats to Self-Sovereign Identities. *Open Identity Summit 2023*, pp. 85-96, doi:10.18420/OID2023_07.

[21] Kingo, T., & Aranha, D. F. (2023). User-centric security analysis of MITID: The Danish passwordless Digital Identity Solution. *Computers & Security*, Vol. 132, 103376, doi:10.1016/j.cose.2023.103376.

[22] Bærentzen, M. S., Ulstrand, C., & Andersen, B. (2023). MitID: A Security Investigation of eID Deployment in a Modern Society. *ResearchGate*, preprint, doi:10.13140/RG.2.2.12472.11521.

[23] Nyári, N., & Kerti, A. (2024). A Risk Assessment of the Hungarian Eid Card. *Scientific Bulletin*, Vol. 29, Iss. 1, pp. 91–102. doi:10.2478/bsaft-2024-0010.

[24] Parsovs, A. (2022). Security improvements for the Estonian ID card. *Estonian Cyber Security News Aggregator*.

[25] Correa-Marichal, J., Caballero-Gil, P., Rosa-Remedios, C., & Sarwat-Shaker, R. (2022). Study and security analysis of the Spanish identity card. *World Congress in Computer Science, Computer Engineering, and Applied Computing. Book of Abstracts CSCE 22. American Council on Science and Education*, doi:arXiv.2210.04064.

[26] Radutoiu, A.-T., Bassit, A., Veldhuis, R., & Busch, C. (2024). A Study on the Next Generation of Digital Travel Credentials. *Christoph Busch's website*.

[27] Aichinger, T. (2022). Security Target - ACOS-IDv2.1 eMRTD (A) EAC/PACE Configuration. Common Criteria for Information Technology Security Evaluation. *www.commoncriteriaportal.org*.

[28] Koziel, P. (2023). PACE and PACE CAM: Security Issues and Protocol Extensions, doctoral dissertation. *Wrocław University of sience and Technology*.

[29] Fischlin, M., von der Heyden, J., Margraf, M., Morgner, F., Wallner, A., & Bock, H. (2023). Post-quantum security for the Extended Access Control Protocol. Part of the book series: Lecture Notes in Computer Science ((LNCS,volume 13895)). *Security Standardisation Research*, pp. 22–52, Springer. doi:10.1007/978-3-031-30731-7_2.

[30] Alnahawi, N., Schmitt, N., Wiesmaier, A., & Zok, C.-M. (2024). Toward Next Generation Quantum-Safe eIDs and eMRTDs: A Survey. *ACM Transactions on Embedded Computing Systems*, Vol. 23, No. 2, pp. 1–28. doi:10.1145/3585517.

[31] Nomis, E. M., Jasim, K. S., & Al-Janabi, S. (2024). Face Morphing Attacks Detection Approaches: A Review. *Mesopotamian Journal of Big Data*, pp. 82–101. doi:10.58496/mjbd/2024/007.

[32] Yildiz, H., Küpper, A., Thatmann, D., Göndör, S., & Herbke, P. (2022). A Tutorial on the Interoperability of Self-sovereign Identities. *TechRxiv*, preprint, doi: 10.36227/techrxiv.20430825.v1.

[33] ISO/IEC 23220-1:2023 Cards and security devices for personal identification — Building blocks for identity management via mobile devices, Part 1: Generic system architectures of mobile eID systems. (2023). *International Organization for Standardization (ISO)*.

[34] Ackermann E., Bober K.L., Jungnickel V., & Lehmann A. (2024). SEKA: Secretless Key Exchange and Authentication in LiFi Networks. *2024 IEEE 9th European Symposium on Security and Privacy (EuroS&P)*, pp. 633-657, doi: 10.1109/EuroSP60621.2024.00041.

[35] Mitra, S., Das, S., & Kule, M. (2020). Prevention of the man-in-the-middle attack on Diffie–Hellman Key Exchange Algorithm: A Review. Part of the book series: Advances in Intelligent Systems and Computing ((AISC,volume 1255)). *Proceedings of International Conference on Frontiers in Computing and Systems*, pp. 625–635, Springer, doi:10.1007/978-981-15-7834-2_58.

[36] Raymond, J.-F., & Stiglic, A. (2002). Security Issues in the Diffie-Hellman Key Agreement Protocol. *IEEE Transactions on Information Theory*, 22.

[37] Stevens, M., Bursztein, E., Karpman, P., Albertini, A., & Markov, Y. (2017). The First Collision for Full SHA-1. Part of the book series: Lecture Notes in Computer Science ((LNSC,volume 10401)). *Advances in Cryptology – CRYPTO 2017*, pp. 570–596, Springer, doi: 10.1007/978-3-319-63688-7_19.

[38] Leurent, G., & Peyrin, T. (2019). From Collisions to Chosen-Prefix Collisions Application to Full SHA-1. Part of the book series: Lecture Notes in Computer Science ((LNSC,volume 11478)). *Advances in Cryptology – EUROCRYPT 2019*, pp. 527–555, Springer. doi:10.1007/978-3-030-17659-4_18.

[39] Kara, M., Laouid, A., AlShaikh, M., Bounceur, A., & Hammoudeh, M. (2021). Secure Key Exchange Against Man-in-the-Middle Attack: Modified Diffie-Hellman Protocol. *Jurnal Ilmiah Teknik Elektro Komputer dan Informatika*, Vol. 7, No. 3, pp. 380-387. doi:10.26555/jiteki.v7i3.22210.

[40] Rawat A. S., & Deshmukh M. (2019). Efficient Extended Diffie-Hellman Key Exchange Protocol. *2019 International Conference on Computing, Power and Communication Technologies (GUCON)*.

[41] Aryan, Kumar, C., & Durai Raj Vincent, P. M. (2017). Enhanced Diffie-Hellman algorithm for reliable key exchange. *IOP Conference Series: Materials Science and Engineering*, 263, 042015. doi:10.1088/1757-899x/263/4/042015.

[42] Pal, O., Alam, B. (2017). Diffie-Hellman key exchange protocol with entities authentication. *International Journal Of Engineering And Computer Science*, Vol. 6, Iss. 4, pp. 20831-20839. doi:10.18535/ijecs/v6i4.06.

[43] Taparia, A., Panigrahy, S. K., & Jena, S. K. (2017). Secure Key Exchange using enhanced Diffie-Hellman protocol based on string comparison. *2017 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET)*, pp. 722–726. doi:10.1109/wispnet.2017.8299856.

Безпека електронних посвідчень особи під час фази передачі документу

Леляк А. В., Астраханцев А. А.

Предметом вивчення в статті є безпека електронних документів. Важливість безпеки електронних документів зростає зі створенням нових міжнародних стандартів електронної ідентифікації та їх впровадженням у всьому світі. Безпека електронної ідентифікації має вирішальне значення як для громадян, так і для уряду, щоб забезпечити взаємну довіру та конфіденційність, цілісність та доступність. Попередні дослідження надають або занадто загальний погляд на безпеку електронної ідентифікації без необхідного рівня технічних деталей, або глибоко аналізують окреме конкретне вітчизняне рішення, яке не має міжнародної сумісності. Метою є оцінка безпеки основних реалізацій електронних документів, що відповідають міжнародним стандартам, проведення їхнього порівняльного аналізу та забезпечення покращень безпеки.

Завдання, які необхідно вирішити, полягають у створенні системи оцінки безпеки електронної ідентифікації для оцінки та порівняння різних варіантів реалізації рішень електронної ідентифікації. Безпека електронної ідентифікації аналізувалася лише на етапі передачі

документів, і розглядалися такі цілі безпеки, як автентифікація зберігання документів власника, запобігання прослуховуванню, запобігання клонуванню документів та автентифікація верифікатора. Було проаналізовано безпеку електронного машинозчитуваного проїзного документа (eMRTD) відповідно до документа ICAO 9303, мобільного посвідчення водія (mDL) відповідно до стандарту ISO/IEC 18013-5 та серверних рішень. Основним використаним методом є порівняльний аналіз. Порівняльний аналіз було надано для цих варіантів впровадження, щоб проілюструвати відмінності в досягненні однакових цілей безпеки, оцінити загальний рівень безпеки та підкреслити існуючі компроміси. Були отримані такі результати: запропоновано покращення безпеки для пом'якшення загроз безпеці, таких як атаки постквантової криптографії, атаки на обмін ключами Діффі-Хеллмана та атаки з використанням колізій хеш-процесів. Висновки: результати дослідження можуть бути використані для покращення наступних переглядів специфікацій ISO/IEC ICAO для електронної ідентифікації або для врахування під час розробки безпеки у внутрішньому серверному рішенні.

Ключові слова: електронна ідентифікація особи; електронний машиннозчитуваний паспорт; мобільне посвідчення водія; захист електронних документів; атаки під час передачі інформації; постквантова криптографія; інформаційна безпека; системи шифрування