Poster: eID in Europe - Password Authentication Revisited

Mirosław Kutyłowski*†, Przemysław Kubiak*, Patryk Kozieł*, Yanmei Cao†
*Wrocław University of Science and Technology, Wrocław, Poland, †Xidian University, China
{miroslaw.kutylowski, przemyslaw.kubiak, patryk.koziel}@pwr.edu.pl, yanmcao@163.com

***Abstract*—A EU Regulation from 2019 aims to achieve a minimal degree of interoperability of personal identity documents in Europe (eID). In particular, eID document verification should be performed according to ICAO protocols, including biometric verification. At the same time, additional functionalities implemented by the Member States must not interfere with the obligatory ICAO part. As presenting personal data from an eID requires an explicit consent of the eID owner, PACE - a password authenticated key exchange (PAKE) protocol, the core part of the ICAO specification – must be used.**

A side effect of the EU Regulation are problems regarding already deployed additional functionalities, such as digital signatures. In this paper we present a pragmatic approach for solving this problem. Instead of installing these functionalities independently – e.g. at a cost of extra code on the eID chip and potential compatibility problems – we may reuse the basic ICAO protocols. As an example of this approach we show a proof-of-presence protocol derived from PACE.

***Index Terms*—eID, ICAO, PAKE, PACE, Authentication, Proof of presence, Privacy**

I. Introduction

Electronic Identity (eID) is one of prerequisites for automatic services that interact with physical persons. Unfortunately, eID is frequently the missing *last mile* of the system. The scope of eID applications is quite wide: the examples range from identity verification at an e-Booth, automatic age verification at a vending machine and proving presence for billing medical transactions, up to cases, where eID is used for pure online activities, like remote server check-in. An eID gives an opportunity to provide a secure cryptographic token, while we benefit from the control system for issuing personal identity documents as well as well trained users' behavior.

A. eID in Europe

Unlike biometric passports, personal ID cards are issued independently by national authorities, with limited international coordination. Consequently, even if the individual design decisions are well motivated, the eID's issued by different countries may be incompatible and cannot serve as a universal ID token (except for eGov services of the issuing country).

A recent EU regulation [1] aims to achieve compatibility of the official ID documents issued by Member States by obligatory compliance with the ICAO standards [2]. From the communication point of view, the obligatory part of this specification is the password-based authentication key exchange (PAKE) protocol called PAssword-based Connection Establishment (PACE), developed by the German Federal Office for Information Security (BSI) [3]. According to [1], PACE will be implemented on all personal ID cards issued in the EU after August 2, 2021.

Annex to ISBN 978-3-903176-39-3© 2021 IFIP

II. PACE

PACE enables creating a connection iff the reader and the chip are using the same password. In this case, the same encryption and MAC session keys are derived. The password may have a low entropy, however the only way to guess it is to play the role of the reader and try all passwords one by one in real interactions with the eID.

A description of PACE can be obtained from Fig. 1 by ignoring all gray boxes. The protocol works as follows:

Phase 1: ***Password dependent transmission of a random*** s**.** The chip chooses s uniformly at random, and then transmits domain parameters $\mathcal{G}$ and a ciphertext of s obtained with the key K_π derived from the password π. The reader recovers s by decrypting with the key K_π. Note that using different passwords by chip and the reader results in different values of the random element s.

Phase 2: ***Establishing a random generator.*** The chip and the reader map s to a random generator $\hat{g}$. In Fig. 1 we proceed with one of the standard options of doing that – so called General Mapping – based on Diffie-Hellman key exchange.

Phase 3: ***Negotiate session keys.*** Just determined random generator $\hat{g}$ is used for the next Diffie-Hellman key agreement protocol yielding a master session key K. The encryption and authentication keys are then derived by hashing K with different parameters.

Phase 4: ***Checking keys and transmitted values.*** The chip and reader exchange the tags T_A and T_B in order to prove that they hold the same keys.

A. PACE with chip authentication

While initially designed for personal document verification at a local reader, PACE can be used for online authentication on a remote terminal (cf. [3]). In this case the reader is merely a not trusted man-in-the-middle and an end-to-end connection is created between the eID and the terminal. PACE assures that no connection will be established unless the eID holder gives an explicit consent by providing the password. However, the chip is authenticated only through knowledge of the password. As the eID holder enters the password explicitly at the reader, authentication of the chip is very weak.

Bender et al. [4] proposed a PACE—AA protocol, where an eID proves that it holds a private key assigned to it. The idea is to create a digital signature of the eID while reusing some steps of the original PACE. This was the first step towards using eID for witnessing remote presence of the eID. Subsequently, a simplified version of the protocol above was presented independently in [5] and [6]. It was patented by the German government and adopted by ICAO under the name PACE-CAM [2]. It has been extended for Integrated Mapping in [7]. There have been also efforts to couple PACE with biometric authentication where the password is derived from biometrics of the eID holder [8].

B. Extensions' Strategy

The EU regulation [1] enforces implementation of ICAO protocols. Unfortunately, in this way no functionality is supported except for the basic document verification. Having in mind necessity of other eGov applications one can follow two approaches. The first option is that new protocols are developed independently from the ICAO part. The second option is that, as in [4], the functionalities are created via minor changes in the original protocols. This approach has the following advantages:

- **backward compatibility:** with a proper design, a device running a new version can smoothly interact with a device running the basic version,
- **reuse:** the parts of the code and communication may be reused for new purposes.

Size of nonvolatile memory on an eID is very limited, so the executable code on the chip must be deeply optimized. Using the same procedures for different tasks can significantly reduce the resulting code size. In turn, reducing communication improves the runtime and robustness against communication failures. Therefore we strongly support this approach.

The proposed approach may ease resolving the problem of reaching an international consensus. Namely, there are many different incompatible stand-alone schemes already deployed on national eID's in Europe. It would be politically and technically hard to choose one of the solutions for Europe-wide application. Among others, there are fundamental legal problems of advantaging a certain manufacturer or manufacturers. One of principles that must be followed by political decision makers is excluding any kind of unfair competition.

The proposed approach bypasses these problems. We embed new functionalities into an already agreed common platform. Furthermore, as ICAO specification is already implemented in passports in almost all countries, this may provide a platform for a world-wide common solution.

Implicitly, one step in this direction has been already done in [4]. The basic scheme presented there provides an explicit (Schnorr or DSA) signature entangled with the communication transcript. The signature can also encompass any additional message. (In fact, a crucial contribution of [4] was to convert the protocol into an authentication process having no proof value against a third party.)

As a proof-of-concept we present a Proof-of-Presence Protocol based on PACE: it creates a proof for the eID holder that it has interacted with a certain reader. There are many potential applications of such a scheme. For example, an inspector controlling technical installations in the field may be obliged to provide a proof that he has really visited certain physical locations when presenting the bill.

Due to space limitations, we postpone a security analysis of the scheme to a forthcoming technical report. (At this point note that a proof alone for PACE taking into account all active adversary scenarios and all privacy issues is tedious and much longer than originally claimed (see [9], [10]).)

III. Proof of Presence

The protocol PACE-Presence is presented on Fig. 1. Just as in case of PACE-CAM, the initial part of the protocol is the same as for the regular PACE. The difference there are only internal computations. In this case these are the computations on the reader's side. The changes are not observable for the chip until an extra message is sent in the final phase. If an eID is not supporting PACE-Presence, this message can be simply ignored as an unsupported option.

The major component of the scheme is a Schnorr signature (X_B, y_B). Note that thereby we reuse x_B as a random component of the Schnorr signature. There is a subtle issue at this point as y_B is also used for creation of $Y_B = \hat{g}^{y_B}$ and presented in clear. However, $\hat{g}$ is an element for which neither an observer nor any single protocol participant knows the discrete logarithm with respect to g. Therefore, neither the eID nor the observer can find out that the discrete logarithm of Y_B with respect to $\hat{g}$ equals the discrete logarithm of $X_B \cdot Z_B^{H(..)}$, where the hash value is computed as in the protocol description.

As needed, the proof obtained by an eID and later presented could not be forged by the eID. Such forgery would effectively mean forgery of Schnorr signature for some specific messages.

On the other hand, one can ask whether a reader can convince a third party that it has interacted with a given eID? The point is that the reader can create a valid communication transcript with all values generated on the side of the reader without any interaction with the eID. As such a transcript can be forged, it has no proof value for the third parties.

If we regard a passive observer, then the situation is similar as for PACE-CAM: the key K'_{Enc} can be replaced by a random one and the observer would not recognize the difference. Consequently, C_B can be replaced by random value and for the observer the any cryptanalytic attack against the protocol reduces essentially to an attack against PACE.

Let us remark that for the dual proof-of-presence protocol (proving an interaction with eID by the terminal) one can apply the initial protocol from [4].

IV. Final Remarks and Conclusion

We claim that there is a big potential for creating an ecosystem of fundamental protocols for interaction between physical

eID(A)		Reader(B)
holds:		**holds:**
π - password		π password (e.g. entered by the user)
		$z_B, Z_B = g^{z_B}$ - private and public key
		cert(Z_B) - certificate for Z_B
$\mathcal{G}$ - parameters of a group of order q		arbitrary message M, e.g. the current time
	Protocol execution	
$K_\pi := H(\pi \Vert 0)$		$K_\pi := H(\pi \Vert 0)$
choose $s \leftarrow \mathbb{Z}_q \setminus \{0\}$ at random		
$z := \mathrm{Enc}(K_\pi, s)$	$\xrightarrow{\mathcal{G},z}$	abort if $\mathcal{G}$ incorrect, decrypt z
		choose $x_B \leftarrow \mathbb{Z}_q \setminus \{0\}$ at random
abort if $X_B \notin \langle g \rangle \setminus \{1\}$	$\xleftarrow{X_B}$	$X_B := g^{x_B}$
choose $x_A \leftarrow \mathbb{Z}_q \setminus \{0\}$ at random		
$X_A := g^{x_A}$	$\xrightarrow{X_A}$	
$h := X_B^{x_A}$ (abort if $h = 1$)		$h := X_A^{x_B}$ (abort if $h = 1$)
$\hat{g} := h \cdot g^s$		$\hat{g} := h \cdot g^s$
choose $y_A \leftarrow \mathbb{Z}_q \setminus \{0\}$ at random		$\boxed{y_B := x_B + z_B \cdot H(M, X_B, X_A) \bmod q}$
$Y_A := \hat{g}^{y_A}$	$\xleftarrow{Y_B}$	$Y_B := \hat{g}^{y_B}$
	$\xrightarrow{Y_A}$	
abort if $Y_B = X_B$		abort if $Y_A = X_A$
$K := {Y_B}^{y_A}$		$K := {Y_A}^{y_B}$
$K_{\mathrm{Enc}} := H(K \Vert 1)$, $K_{\mathrm{MAC}} := H(K \Vert 2)$		$K_{\mathrm{Enc}} := H(K \Vert 1)$, $K_{\mathrm{MAC}} := H(K \Vert 2)$
$K'_{\mathrm{MAC}} := H(K \Vert 3)$, $\boxed{K'_{\mathrm{Enc}} := H(K \Vert 4)}$		$K'_{\mathrm{MAC}} := H(K \Vert 3)$, $\boxed{K'_{\mathrm{Enc}} := H(K \Vert 4)}$
$T_A := \mathrm{MAC}(K'_{\mathrm{MAC}}, (Y_B, \mathcal{G}))$		$T_B := \mathrm{MAC}(K'_{\mathrm{MAC}}, (Y_A, \mathcal{G}))$
	$\xleftarrow{T_B}$	
abort if T_B incorrect	$\xrightarrow{T_A}$	abort if T_A incorrect
	Terminal's Signature	
$\boxed{\text{abort if cert}(Z_B) \text{ invalid or}}$	$\boxed{\xleftarrow{C_B}}$	$\boxed{C_B := \mathrm{Enc}(K'_{\mathrm{Enc}}, (M, y_B, \mathrm{cert}(Z_B)))}$
$\boxed{g^{y_B} \neq X_B \cdot Z_B^{H(M,X_B,X_A)} \text{ or } Y_B \neq \hat{g}^{y_B}}$		
$\boxed{\text{output Schnorr signature } (X_B, y_B) \text{ together with } X_A, M}$		

Fig. 1. PACE-Presence – a proof of presence for the eID. H stands for hash functions where the choice of the function depends on the context – the target image of the hash function.

persons and IT systems based on the ICAO protocol chosen as a common platform for personal ID documents in the EU. In this way we may expand the original concept of identity verification of international travelers to a concept of a universal identity token.

We may benefit from fragility features of PACE: any change of the protocol by an active adversary results in a connection failure. Thereby the scheme enjoys high resilience against active adversaries and thereby follow the strict rules of GDPR.

REFERENCES

[1] The European Parliament and the Council of the European Union, "Regulation (EU) 2019/1157 - strengthening the security of identity cards and of residence documents issued to EU citizens and their family members exercising their right of free movement," *Official Journal of the European Union*, vol. 188, no. 67, 2019.

[2] ICAO, "Machine Readable Travel Documents - Part 11: Security Mechanism for MRTDs." Doc 9303, 2015.

[3] BSI, "Advanced security mechanism for machine readable travel documents extended access control (EAC)," in *Technical Report (BSI-TR-03110)*, Bundesamt für Sicherheit in der Informationstechnik (BSI), 2010. Current version: " Technical Guideline Advanced Security Mechanisms for Machine Readable Travel Documents and eIDAS Token", 2015, https://www.bsi.bund.de/SharedDocs/Downloads/EN/BSI/Publications/TechGuidelines/TR03110/BSI_TR-03110_Part-1_V2-2.pdf;jsessionid=D0F668D383C36806A7F9B3D3F3E792A5.internet461?__blob=publicationFile&v=1.

[4] J. Bender, Ö. Dagdelen, M. Fischlin, and D. Kügler, "The PACE|AA protocol for machine readable travel documents, and its security," in *Proc. of 16th International Conference Financial Cryptography and Data Security*, pp. 344–358, 2012.

[5] L. Hanzlik, Ł. Krzywiecki, and M. Kutyłowski, "Simplified PACE|AA protocol," in *Prof. of 9th International Conference Information Security Practice and Experience*, pp. 218–232, 2013.

[6] J. Bender, M. Fischlin, and D. Kügler, "The PACE|CA protocol for machine readable travel documents," in *Prof. of 5th International Conference Trusted Systems*, pp. 17–35, 2013.

[7] L. Hanzlik and M. Kutyłowski, "Chip authentication for e-passports: PACE with chip authentication mapping v2," in *Prof. of 19th International Conference Information Security*, pp. 115–129, 2016.

[8] N. Buchmann, R. Peeters, H. Baier, and A. Pashalidis, "Security considerations on extending PACE to a biometric-based connection establishment," in *Proc. of the 12th International Conference of Biometrics Special Interest Group*, pp. 15–26, 2013.

[9] J. Bender, M. Fischlin, and D. Kügler, "Security analysis of the PACE key-agreement protocol," in *Prof. of 12th International Conference Information Security*, pp. 33–48, 2009.

[10] M. Kutyłowski and P. Kubiak, "Privacy and security analysis of PACE GM protocol," in *Prof. of 18th IEEE International Conference On Trust, Security And Privacy In Computing And Communications / 13th IEEE International Conference On Big Data Science And Engineering*, pp. 763–768, 2019.