eID in Europe - Password Authentication Revisited

Mirosław Kutyłowski, Yanmei Cao, Patryk Kozieł, Przemysław Kubiak

Wrocaw University Of Science and Technology, Wrocław, Poland
Xidian University, Xi'an, People's Republic of China

Background

- EU Regulation makes PACE password authentication key exchange obligatory on official personal ID cards issued after Aug. 2, 2021
- additional (compatible) protocols allowed explicitly by EU

Background

- EU Regulation makes PACE password authentication key exchange obligatory on official personal ID cards issued after Aug. 2, 2021
- additional (compatible) protocols allowed explicitly by EU
- our contribution – an extension of PACE:
  - **PACE Proof-of-Presence** protocol
  - **added functionality:** eID gets a proof of interaction with a terminal that can be checked by third parties
  - strong authentication of the terminal during the session based on possession of a secret key

Design features

- **backwards compatibility**: connection should be established even if the terminal or the eID runs the plain PACE version
- **minimal changes**: just fine tune the original protocol
- **reuse** the code and expensive cryptographic operations
- guarantee that the **security arguments** for the plain version are **still valid**

Changes to original PACE (in grey boxes)

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	
$K_\pi := H(\pi \vert\vert 0)$		$K_\pi := H(\pi \vert\vert 0)$
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		$y_B := x_B + z_B \cdot H(M, X_B, X_A) \bmod q$
$Y_A := \hat{g}^{y_A}$	$\xleftarrow{Y_B}$ $\xrightarrow{Y_A}$	$Y_B := \hat{g}^{y_B}$
abort if $Y_B = X_B$		abort if $Y_A = X_A$
$K := {Y_B}^{y_A}$		$K := {Y_A}^{y_B}$
$K_{\mathrm{Enc}} := H(K \vert\vert 1)$, $K_{\mathrm{MAC}} := H(K \vert\vert 2)$		$K_{\mathrm{Enc}} := H(K \vert\vert 1)$, $K_{\mathrm{MAC}} := H(K \vert\vert 2)$
$K'_{\mathrm{MAC}} := H(K \vert\vert 3)$, $K'_{\mathrm{Enc}} := H(K \vert\vert 4)$		$K'_{\mathrm{MAC}} := H(K \vert\vert 3)$, $K'_{\mathrm{Enc}} := H(K \vert\vert 4)$
$T_A := \mathrm{MAC}(K'_{\mathrm{MAC}}, (Y_B, \mathcal{G}))$		$T_B := \mathrm{MAC}(K'_{\mathrm{MAC}}, (Y_A, \mathcal{G}))$
abort if T_B incorrect	$\xleftarrow{T_B}$ $\xrightarrow{T_A}$	abort if T_A incorrect
	Terminal's Signature	
abort if cert(Z_B) invalid or	$\xleftarrow{C_B}$	$C_B := \mathrm{Enc}(K'_{\mathrm{Enc}}, (M, y_B, \mathrm{cert}(Z_B)))$
$g^{y_B} \neq X_B \cdot Z_B^{H(M, X_B, X_A)}$ or $Y_B \neq \hat{g}^{y_B}$		
output Schnorr signature (X_B, y_B) together with X_A, M		

Comments

- **reduction**: security of PACE Proof of Presence reduced to security of the original PACE
- **fragility** of PACE Proof-of-presence - any manipulation of the messages exchanged results in a connection failure (not the case for the original PACE)
- **other such extensions** are possible (strong mutual authentication, eID's proof of presence, signature, . . .)