Homework 1: PAKE_0 – A Password Authenticated Key Exchange protocol

Margot Catinaud `margot.catinaud@lmf.cnrs.fr`

Due date: December 8, 2025

In this homework, we will study a Password Authenticated Key Exchange protocol, namely the PAKE_0 protocol, defined as follows:

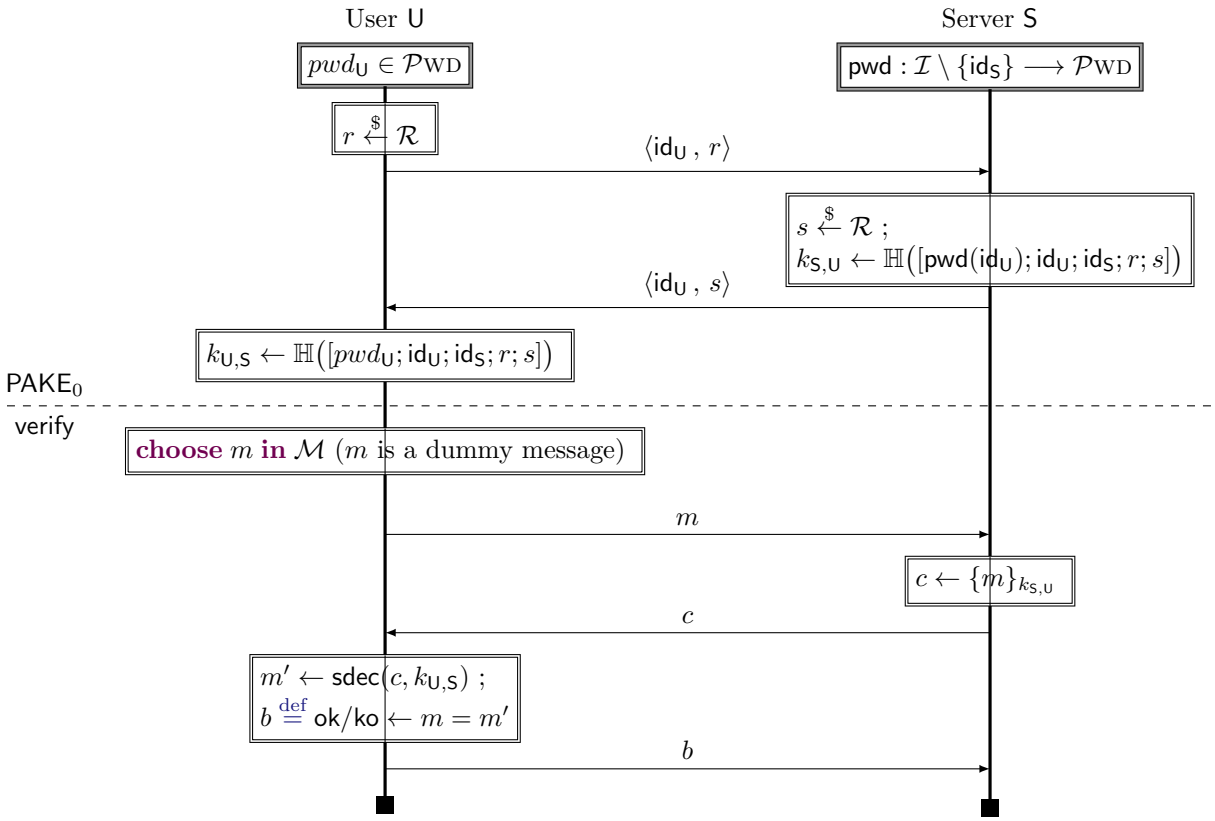

Figure 1: The PAKE_0 protocol.

Notations

- $\mathcal{M}$ is a set of messages ;
- $\mathcal{R}$ is a finite set of random values ("*large enough*") ;
- $\mathcal{I}$ a finite set of identities such that $\mathrm{Card}(\mathcal{I}) \geqslant 2$: we suppose their is at least the server identity $\mathsf{id_S} \in \mathcal{I}$ and an agent $\mathsf{id_U} \in \mathcal{I}$;
- $\mathcal{P}\mathrm{WD} \subset \{0,1\}^*$ a finite set of passwords ;
- $\mathcal{K}$ a set of key ("*large enough*") ;
- $\mathbb{H} : \{0,1\}^* \longrightarrow \mathcal{K}$ a secure[1] hash function ;

[1] By secure we mean that $\mathbb{H}$ is a "*one-way function*", *i.e.* given $\mathbb{H}(x)$ it is hard to retrieve x and is "*collision-resistant*", meaning that, for $x, x' \in \{0,1\}^*$, if $\mathbb{H}(x) = \mathbb{H}(x')$ then $x = x'$. These security properties can be achieved by model $\mathbb{H}$ as a random oracle model such that the two following distributions are equal:

$$\left[\mathbb{H}(x) \mid x \xleftarrow{\$} \{0,1\}^*\right] = \left[k \mid k \xleftarrow{\$} \mathcal{K}\right].$$

Warning: Except for questions where it is **explicitly** said otherwise, we will **not** consider the verify part.

- For all natural numbers $n \in \mathbb{N}$, $n \geqslant 2$, and for all n sets $(X_i)_{i=1}^n$, we define a function

$$[\,\cdot\,;\,\ldots\,;\,\cdot\,]_n : (X_i)_{i=1}^n \longrightarrow \{0,1\}^*$$

  corresponding to the concatenation of elements $x_i \in X_i$, $i \in [\![1;n]\!]$, seen as bitstrings in $\{0,1\}^*$. When n is clear from context, we denote $[\,\cdot\,;\,\ldots\,;\,\cdot\,]$ instead of $[\,\cdot\,;\,\ldots\,;\,\cdot\,]_n$;

- $\{m\}_k$ is a symmetric encryption of message m with key k ; sdec is the corresponding symmetric decryption.

At the end of an interaction of the PAKE_0 protocol between an agent U (with identity $\mathsf{id}_\mathsf{U} \in \mathcal{I}$) and the server S, both agents U and S are agreed on a same session key k, *i.e.* $k_{\mathsf{U},\mathsf{S}} = k_{\mathsf{S},\mathsf{U}} \stackrel{\text{def}}{=} k$. Notice that in the PAKE_0 protocol, we simplify the knowledge of passwords by the server as a map $\mathsf{pwd} : \mathcal{I} \setminus \{\mathsf{id}_\mathsf{S}\} \longrightarrow \mathcal{P}\text{WD}$ between agent identities and their respective clear password.

Question 1

Provide a trace $\tau_{\mathsf{U} \rightleftharpoons \mathsf{S}}$ (of the PAKE_0 protocol with the verify *part) that leads to* U *accepting.*

Question 2

Formally write the PAKE_0 protocol in the process calculus for the case of an interaction between only one user U *and the server* S *(in this question, we have* $\text{Card}(\mathcal{I}) = 2$*).*

Question 3

Formally write the PAKE_0 protocol in the process calculus for the case of an interaction between multiple users and the server (in this question, we have $\text{Card}(\mathcal{I}) \stackrel{def}{=} n + 1 > 2$*).*

In this homework, we consider these two following security properties, and we will study them with perfectly secure cryptography[2], *i.e.* the security analysis is performed in the *symbolic model*. Besides, we suppose we are in a presence of a *honest-but-curious* adversary $\mathcal{A}$: $\mathcal{A}$ sees all messages exchanged on the network and can compute any functions but **can not** participate to the protocol : $\mathsf{id}_\mathcal{A} \notin \mathcal{I}$.

- **(Authentication)** The key k, if it is shared with anyone, is shared between an instance of the server S and an instance of user U ; and this instance of S *should think* he is talking to an instance of user U.
- **(Secrecy of key k)** Any adversary *can not* retrieve the session key k between an instance of user U and one of the server S.

Question 4

Express the two intended security properties.

Question 5

Show that the authentication *property is always satisfied by the PAKE_0 protocol.*

Question 6

Suppose that we have *strong* passwords: that is adversary $\mathcal{A}$ can not guess any password $pwd \in \mathcal{P}\text{WD}$.

Show that in this setting, secrecy *of key k is ensured.*

Question 7

Suppose that we have *weak* passwords: the set of passwords $\mathcal{P}\text{WD}$ is a subset of some relatively small dictionary $\mathcal{D}$ of common passwords. Moreover, suppose that after the PAKE_0 protocol, agent U chooses a message $m \in \mathcal{M}$ and sends it along with its encryption $c \leftarrow \{m\}_k$ under the session key k.

Show how to compromise the secrecy *property of the key k with a honest-but-curious adversary $\mathcal{A}$ in this setting.*

[2]Study them without this assumption leads to more concrete analysis, for example with the interpretation we seen in Exercise 2 – TD2.

Bonus question – A fix for the Question 7 attack

In this bonus section, we will propose a fix to the attack you have found in Question 7. Consider the PAKE_1 protocol, defined as follows:

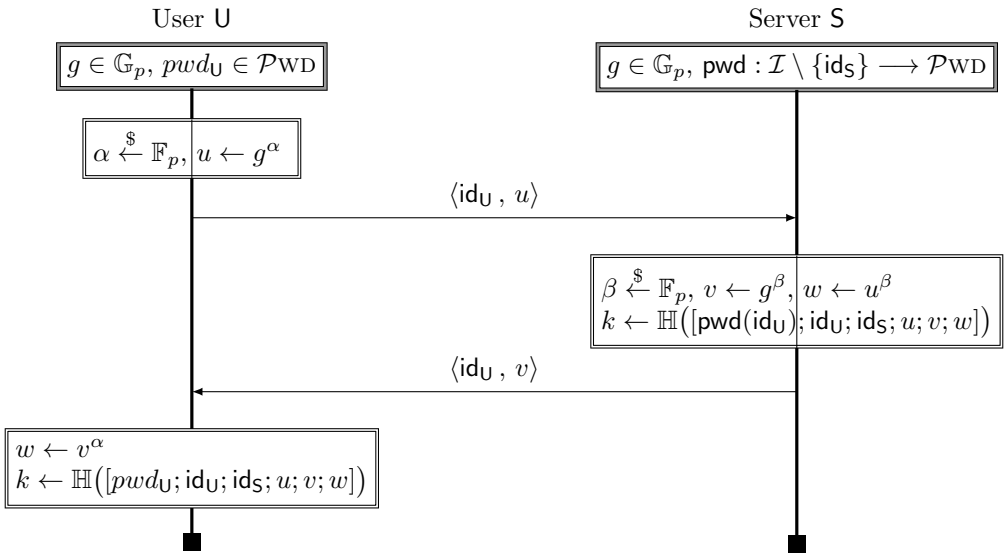

Figure 2: The PAKE_1 protocol.

Notations

- $p \in \mathbb{N}$ a prime number ;
- $\mathbb{F}_p$ the finite field of p elements ;
- $\mathbb{G}_p$ a cyclic group of prime order p where $g \in \mathbb{G}_p$ is a generator of $\mathbb{G}_p$.

Bonus question 1

We suppose same settings as in Question 7 but on the PAKE_1 *protocol. Try to find the property the* secrecy *of key k reduces to in the case of a* honest-but-curious *adversary $\mathcal{A}$.*