Exercise set 11

TMA4160 Cryptography

4. November 2014

In the lectures we have defined a Σ-protocol in the following way. Let $R = \{(h; a)\} \subseteq X \times Y$ be a binary relation, and let $L_R = \{h \in X | \exists a \in Y \text{ s.t. } (h; a) \in R\}$ be the language relating to R. Then we define a Σ-protocol as

P	–	**V**
Input:		Input:
$(h; a) \in R$		$h \in X$
$\alpha \leftarrow A_1(h, a)$		
	$\xrightarrow{\alpha}$	
		$e \in_R C$
	$\xleftarrow{e}$	
$r \rightarrow A_2(h, a, e)$		
		$A_3(h, \alpha, e, r)$?

Fig. 1. Σ-protocol for relation R, where A_1 and A_2 are probabalistic polynomial time algorithms, C is a finite set and A_3 is a polynomial time predicate.

Definition 1. *A Σ-protocol for relation R is a protocol between is a protocol between a prover P and a verifier V of the the form given in figure 1 satisfying the following three properties.*

Completeness. *If P and V follow the protocol, then V always accepts.*

Special soundness. *There exists a probabalistic polynomial time algorithm E which given any $h \in X$ and any pair of accepting conversations (α, e_1, r_1) and (α, e_2, r_2), with $e_1 \neq e_2$, computes a witness a satisfying $(h, a) \in R$.*

Special honest-verifier zero-knowledge. *There exists a probabalistic polynomial time algorithm S which given any $h \in L_R$ and any challenge $e \in C$ produces conversations (α, e, r) with the same probability distribution as conversations between honest P and V on common input h and* *challenge e, where P uses any witness a satisfying $(h, a) \in R$. Furthermore, for $h \in X \setminus L_R$, S is just required to produce arbitrary accepting conversations with challenge e.*

Exercise 1

Consider the following version of the Schnorr protocol (where we work in $G =< g >, |G| = p$ for prime p):

$$
\begin{array}{lcr}
\textbf{P} & - & \textbf{V} \\
\text{Input:} & & \text{Input:} \\
g, h, a & & g, h \\
\text{s.t } h = g^a & & \\
 & & \\
u \in_R \mathbb{Z}_p & & \\
\alpha \leftarrow g^u & & \\
 & \xrightarrow{\alpha} & \\
 & & e \in_R \mathbb{Z}_p \\
 & \xleftarrow{e} & \\
r \rightarrow eu + a \pmod{p-1} & & \\
 & & g^r \stackrel{?}{=} \alpha^e h
\end{array}
$$

Fig. 2. Insecure version of Schnorr's protocol

a) Show that this version of the Schnorr protocol is a Σ-protocol

b) Show that this version of the Schnorr protocol is completly insecure against a cheating verifier, and suggest a fix.

Exercise 2
Let $n = pq$ be a RSA-modulus, and b a large prime (So that $O(b)$ is exponential in the size of n). Consider the Guillou-Quisqater protocol for showing knowledge of bth-roots (i.e knowledge of m such that $m^b = c$ for public c):

P	–	**V**
Input:		Input:
c, m		c
s.t. $c = m^b$		
$u \in_R \mathbb{Z}_n^*$		
$\alpha \leftarrow u^b \pmod n$		
	$\xrightarrow{\alpha}$	
		$e \in_R \mathbb{Z}_b$
	$\xleftarrow{e}$	
$r \rightarrow um^e \pmod n$		
		$r^b \stackrel{?}{=} \alpha c^e$

Fig. 3. The Guillou-Quisqater protocol

Show that the Guillou-Quisqater protocol is a Σ-protocol

Challenge

Exercise 3
You have seen in the lectures how to use Schnorr's protocol to create a protocol for the relation $\{(h_1, h_2; a_1, a_2) \mid h_1 = g^{a_1} \wedge h_2 = g^{a_2}\}$ (AND-protocol). Now use Guillou-Quisqater's protocol to create a protocol for the relation $\{(c_1, c_2; m_1, m_2) \mid c_1 = m_1^b \wedge c_2 = m_2^b\}$.