ETH Zurich, Department of Computer Science
SS 2018

Prof. Ueli Maurer
Dr. Martin Hirt
Chen-Da Liu Zhang

Cryptographic Protocols
Solution to Exercise 3

3.1 Geometric Zero-Knowledge

a) Given two angles α and β, the angle $\alpha \pm \beta$ can be constructed as follows: Open the compass to an arbitrary angle. Draw a circle around the endpoints of both angles with the resulting radius, which results in four new points $p_\alpha, p'_\alpha, p_\beta, p'_\beta$. Open the compass to the distance between p_α and p'_α. Draw a circle around, say, p_β with the resulting radius and create the line ℓ through p_β and p'_β as well as the intersection points q_β and q'_β of the circle and ℓ. Then, create a line through the endpoint of β and q_β or q'_β, depending on whether $\alpha + \beta$ or $\alpha - \beta$ is to be constructed.

b) A possible protocol for this task is the following one:

Peggy		**Vic**
knows angles α, β s.t. $\beta = 3\alpha$		knows angle β
choose random angle κ create $\tau := 3\kappa$	$\xrightarrow{\quad\tau\quad}$	
	$\xleftarrow{\quad c\quad}$	choose random $c \in_R \{0,1\}$
create $\rho := \kappa + c\alpha$	$\xrightarrow{\quad\rho\quad}$	check $3\rho \stackrel{?}{=} \tau + c\beta$

c) Completeness: One can easily verify that if Peggy is honest and knows α, Vic will always accept.

Soundness (Proof of Knowledge): Here we show that if Peggy knows how to answer both challenges, she actually can compute the trisection α. Assume Peggy knows successful answers ρ, ρ' to both challenges $c = 0$ and $c' = 1$ for the same first message τ. In that case,

$$3\rho = \tau \quad \text{and} \quad 3\rho' = \tau + \beta.$$

Thus, $3\rho' - 3\rho = \beta = 3\alpha$, and, therefore, Peggy may compute the angle α as $\rho' - \rho$.

d) Zero-Knowledge: The protocol is c-simulatable: for a given challenge $c \in \{0,1\}$, choose a uniform random angle ρ and set $\tau := 3\rho - c\beta$, which is easily checked to result in the correct distribution. Moreover, the size of the challenge space is clearly polynomial.

3.2 Honest-Verifier Zero-Knowledge and c-Simulatability

Let (P, V) be a HVZK protocol for R. Let x be the instance. A protocol (P', V') for R can be the following:

1. P' computes the first message t using P, and also chooses a random challenge $c'' \in \mathcal{C}$. Send $t' := (t, c'')$ to V'.

2. V' chooses a random challenge $c' \in \mathcal{C}$ and sends it to P'.
3. P' computes $c = c' + c''$, and a valid answer r to c using P. Send $r' := r$ to V'.
4. V' checks if $(t, c' + c'', r)$ is an accepting transcript for the instance x using V, and accepts/rejects accordingly.

The idea is that the new protocol (P', V') is the same as (P, V), but the challenge is the XOR of challenges chosen by P' and V'.

Completeness: It is easy to verify that the protocol is complete, because the protocol (P, V) is complete.

Soundness (Proof of Knowledge): In this proof we assume that the protocol (P, V) is 2-extractable. That is, that from two accepting triples (t, c_1, r_1) and (t, c_2, r_2) one can extract the witness. Then, the protocol (P', V') is also sound. Let $((t, c''), c_1', r_1')$, $((t, c''), c_2', r_2')$ be two accepting triples in protocol (P', V'). This means that $(t, c'' + c_1', r_1')$ and $(t, c'' + c_2', r_2')$ are two accepting triples in (P, V) and one can extract the witness w from the two triples.

c-Simulatable: The protocol is c-simulatable, because, given c', one can invoke the HVZK simulator for (P, V) which returns (t, c, r), and can choose $c'' = c + c'$, and set $(t', r') = ((t, c''), r)$. Then, the triple (t', c', r') is identically distributed as in the protocol (P', V'), conditioned on the challenge being c'.

3.3 An Interactive Proof

a) A possible protocol, similar to Schnorr's protocol, is the following:

Peggy		**Vic**
knows $x, y \in \mathbb{Z}_{\lvert G \rvert}$		knows $z = g^x h^y$
choose $k, l \in_R \mathbb{Z}_{\lvert G \rvert}$ compute $t := g^k h^l$	$\xrightarrow{\quad t \quad}$	
	$\xleftarrow{\quad c \quad}$	let $c \in_R \mathcal{C} \subseteq \mathbb{Z}_{\lvert G \rvert}$
$r := k + cx$ in $\mathbb{Z}_{\lvert G \rvert}$ $s := l + cy$ in $\mathbb{Z}_{\lvert G \rvert}$	$\xrightarrow{\quad (r, s) \quad}$	check if $g^r h^s \stackrel{?}{=} t z^c$

Completeness: It is easily verified that if Peggy is honest and knows (x, y), then Vic always accepts.

Soundness (Proof of Knowledge): From the prover's replies to two different challenges for the same first message t, one can compute values x' and y' such that $g^{x'} h^{y'} = z$: Let $(t, c, (r, s))$ and $(t, c', (r', s'))$ be two accepting transcripts with $c \neq c'$. That is, $g^r h^s = t z^c$ and $g^{r'} h^{s'} = t z^{c'}$. By dividing the first equation by the second one we get:

$$g^{r - r'} h^{s - s'} = z^{c - c'} = z^{c - c'},$$

which implies that $x' = (r - r')(c - c')^{-1}$ and $x' = (s - s')(c - c')^{-1}$ are values with $g^{x'} h^{y'} = z$. Note that since $|G|$ is prime, $c - c' \neq 0$ has an inverse modulo $|G|$.

b) Zero-Knowledge: Similarly to all previous examples, the protocol is c-simulatable: Choose random $r, s \in \mathbb{Z}_{\lvert G \rvert}$ and set $t := g^r h^s z^{-c}$, which is easily checked to result in the correct distribution. If $\mathcal{C}$ is chosen to be polynomially large, the protocol is zero-knowledge.