ETH Zurich, Department of Computer Science
SS 2021

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

Cryptographic Protocols
Solution to Exercise 5

5.1 Perfectly Binding/Hiding Commitments

We consider *perfectly correct* commitment schemes with a *non-interactive* COMMIT *phase*. Such a commitment scheme can be characterized by a function $C : \mathcal{X} \times \mathcal{R} \to \mathcal{B}$ that maps a value $x \in \mathcal{X}$ and a randomness string r from some randomness space $\mathcal{R}$ to a blob $b = C(x, r)$ in some blob space $\mathcal{B}$. The OPEN phase simply consists of the prover's sending (x, r) to the verifier, who checks that $C(x, r) = b$.

In the following, denote by $\mathcal{B}_x := \operatorname{im} C(x, \cdot)$ for $x \in \mathcal{X}$.

a) Let $x \neq x'$. Perfectly binding means that $\mathcal{B}_x \cap \mathcal{B}_{x'} = \emptyset$, whereas perfectly hiding means that $C(x, R)$ and $C(x', R)$ are identically distributed random variables for $R \in_R \mathcal{R}$. This requires in particular that $\mathcal{B}_x = \mathcal{B}_{x'}$, which contradicts $\mathcal{B}_x \cap \mathcal{B}_{x'} = \emptyset$.

b) Subtasks **b)** and **c)** are discussed simultaneously in **c)**.

c) Note that in all cases, the combined scheme is a string commitment $C(x, (r_1, r_2))$.

1. HIDING: The computational hiding property of C_B cannot be broken by additionally adding the blob of the perfectly hiding scheme C_H.[1]
   BINDING: As C_B is perfectly binding, this is also true for the combined scheme $(C_H(x, r_1), C_B(x, r_2))$, since $C(x, (r_1, r_2)) = C(x', (r'_1, r'_2))$ implies that $C_B(x, r_2) = C_B(x', r'_2)$.
2. HIDING: Clearly, the scheme is perfectly hiding as $C_H(C_B(x, r_1), r_2)$ perfectly hides $C_B(x, r_1)$ and thereby x.
   BINDING: Assume for contradiction one could efficiently come up with $x \neq x'$, (r_1, r_2), and (r'_1, r'_2) such that $C(x, (r_1, r_2)) = C(x', (r'_1, r'_2))$. Then, by the fact that C_B is perfectly binding, $y := C_B(x, r_1) \neq C_B(x', r'_1) =: y'$, one can efficiently come up with $y \neq y'$, r_2, and r'_2 such that $C_H(y, r_2) = C_H(y', r'_2)$, which breaks the (computational) binding property of C_H.
3. HIDING: Clearly, the scheme is perfectly hiding as $C_H(x, r_1)$ perfectly hides x.
   BINDING: Assume for contradiction one could efficiently come up with $x \neq x'$, (r_1, r_2), and (r'_1, r'_2) such that $C(x, (r_1, r_2)) = C(x', (r'_1, r'_2))$. Then, by the fact that C_B is perfectly binding, $y := C_H(x, r_1) = C_H(x', r'_1) =: y'$, one can efficiently come up with $x \neq x'$, r_1, and r'_1 such that $C_H(x, r_1) = y = C_H(x', r'_1)$, which breaks the (computational) binding property of C_H.

[1] Formally, this would have to be proved via a reduction.

5.2 Homomorphic Commitments

Note that a blob committing to 0 is a quadratic residue, and, since t is a quadratic non-residue with $\left(\frac{t}{m}\right) = +1$, a blob committing to 1 is a quadratic non-residue b with $\left(\frac{b}{m}\right) = +1$. Thus, the scheme is of type B, where the computational hiding property relies on the QR assumption, which states that modulo an RSA prime m it is hard to distinguish quadratic residues from quadratic non-residues with $\left(\frac{b}{m}\right) = +1$.

a) Denote by $b_0 = r_0^2 t^{x_0}$ and $b_1 = r_1^2 t^{x_1}$ two blobs committing to bits x_0 and x_1, respectively. By multiplying b_0 and b_1, one obtains

$$b = b_0 \cdot b_1 = r_0^2 \cdot r_1^2 \cdot t^{x_0 + x_1}.$$

This is a commitment to $x_0 \oplus x_1$: If $x_0 = x_1$ (i.e., $x_0 \oplus x_1 = 0$), then b is a quadratic residue (with square root $r = r_0 r_1$ if $x_0 = x_1 = 0$ and $r = r_0 r_1 t$ if $x_0 = x_1 = 1$). If $x_0 \neq x_1$ (i.e., $x_0 \oplus x_1 = 1$), then b is a quadratic non-residue with $\left(\frac{b}{m}\right) = +1$ and can be opened using $r = r_0 r_1$.

b) Let $b = r^2 t^x$ be the blob committing to x. By multiplying b by t one obtains

$$b' = b \cdot t = r^2 \cdot t^{x+1}.$$

If $x = 0$, b' is a quadratic non-residue and thus a commitment to 1. In this case, b' can be opened using randomness $r' = r$.

If $x = 1$, b' is a quadratic residue and thus a commitment to 0. In this case, b' can be opened using randomness $r' = rt$.

c) As shown in **a)**, if $x_0 = x_1$, $b_0 \cdot b_1$ is a quadratic residue, a fact that Peggy can prove using the Fiat-Shamir protocol. Moreover, if $x_0 \neq x_1$, then $b := b_0 \cdot b_1$ is a quadratic non-residue with $\left(\frac{b}{m}\right) = +1$ and thus $b_0 \cdot b_1 \cdot t$ is a quadratic residue, which, again, can be proved using the Fiat-Shamir protocol.

5.3 Graph Coloring

The protocol is a proof of statement, it shows that $\mathcal{G}$ has a 3-coloring. Let $V = \{1, \ldots, n\}$, and the 3-coloring be defined as a function $f : V \to \{1, 2, 3\}$.

Peggy		**Vic**
knows a 3-coloring f for $\mathcal{G} := (V, E)$		knows $\mathcal{G}$
choose a random permutation of the colors π let $f' = \pi \circ f$ $\forall i \in V$, commit to $f'(i)$ as C_i	$C_1, \ldots, C_n$ →	
	← (i, j)	let $(i, j) \in_R E$
open colors of vertices i and j	d_i, d_j →	check if $f'(i), f'(j) \in \{1, 2, 3\}$ and $f'(i) \neq f'(j)$

Completeness: It is easily verified that if G has a 3-coloring, then Vic always accepts. Peggy can answer all the Vic's queries correctly such that Vic is convinced as long as the commitment scheme is binding.

Soundness: The scheme has soundness $\frac{1}{|E|}$: if $\mathcal{G}$ does not have a 3-coloring, a cheating prover must commit to a coloring that has at least one edge whose vertices have the same color, or to colors that are not in $\{1, 2, 3\}$. Hence, with probability $\frac{1}{|E|}$, the verifier catches him, assuming the commitments are perfectly binding. When doing $n|E|$ sequential repetitions of the protocol, the soundness error is down to $(1 - \frac{1}{|E|})^{n|E|} \leq e^{-n}$.

Zero-Knowledge: The protocol is c-simulatable: Given (i, j), choose random colors σ_i, σ_j, and compute the commitments C_i, C_j. Since $|E|$ is polynomially large the protocol is zero-knowledge., assuming that the commitments are perfectly hiding.