Solution Sheet #12

Advanced Cryptography 2022

Solution 1 Σ-Protocol for Cubic Residues

1. We use the properties of the Jacobi symbol. Recall that $\left(\frac{-1}{p}\right) = 1$ if $p = 1 \bmod 4$ and -1 otherwise. We have $\left(\frac{-3}{p}\right) = \left(\frac{-1}{p}\right) \times \left(\frac{3}{p}\right) = \left(\frac{p}{3}\right) = \left(\frac{1}{3}\right) = +1$ so -3 is a quadratic residue modulo p.

2. The discriminant of $X^2 + X + 1$ is -3. Let $-3 \equiv u^2 \pmod{p}$. Therefore, $X^2 + X + 1$ has two square roots $(-1 \pm u)/2 \bmod p$.

   Alternately, we have $X^2 + X + 1 = (X + \frac{1}{2})^2 - \frac{u^2}{4} = (X - \frac{-1+u}{2})(X + \frac{-1-u}{2})$ from which we deduce the two roots.

3. The polynomial $X^3 - 1$ cannot have more than 3 roots over the field $\mathbf{Z}_p$. Multiple roots must be roots of its derivative $3X^2$ which has only 0 as a root. So, $X^3 - s$ has no multiple roots when $s \in \mathbf{Z}_p^*$. The polynomial $X^3 - 1$ has root 1 and the roots of $X^2 + X + 1$. So, $X^3 - 1$ has exactly 3 roots.

   We know it cannot have more than 3 roots. Assume it has one root θ. Let $1, \zeta, \zeta'$ be the 3 roots of $X^3 - 1$. We observe that $\theta, \theta\zeta, \theta\zeta'$ are 3 different roots of $X^3 - s$. So we have exactly 3 different roots.

4. A number x is a cubic root of s modulo n iff it is a cubic root modulo p and modulo q. Since 3 is coprime with $\varphi(q)$, every residue has a unique cubic root modulo q. Hence, by using the Chinese remainder theorem we obtain that a number always has the same number of cubic roots modulo n and modulo p.

5. We propose

Prover		**Verifier**
witness: s	**input**: (n, v)	
pick $r \in \mathbf{Z}_n^*$		pick $e \in \{0, 1\}$
$x = r^3 \bmod n$	$\xrightarrow{x}$	
	$\xleftarrow{e}$	
$y = rs^e \bmod n$	$\xrightarrow{y}$	$y^3 v^e \bmod n \stackrel{?}{=} x$

   By going through the checklist, we define:

   - the relation R is already defined

- the first prover function $\mathcal{P}(n, v; r) = r^3 \bmod n$
- the challenge domain $E = \{0, 1\}$
- the second prover function $\mathcal{P}(n, v, e; r) = rs^e \bmod n$
- the verification function $V(n, v, x, e, y) \iff y^3 v^e \bmod n = x$
- the extractor algorithm $\mathcal{E}(n, v, x, e, y, e', y')$: since e and e' are different in $\{0, 1\}$ we denote y_0 resp. y_1 the y or y' value corresponding to the challenge 0 resp. 1. We compute $z = y_1/y_0 \bmod n$.
- the simulator algorithm $\mathcal{S}(n, v, e; r)$: pick $y \in_U \mathbf{Z}_n^*$ form r and set $x = y^3 v^e \bmod n$.

We can now prove all required properties:

- (efficiency) all algorithms are polynomially bounded
- (completeness) for each $((n, v), s)$ in the language and a honestly generated transcript (x, e, y) then $V(n, v, x, e, y)$ holds.
- (special soundness) for each (n, v), if (x, e, y) and (x, e', y') are two accepting transcripts with same x, then $\mathcal{E}$ produces a witness. This comes from

$$\left(\frac{y_1}{y_0}\right)^3 v \equiv \frac{y_1^3 v}{y_0^3} \equiv \frac{x}{x} \equiv 1 \pmod{n}$$

- (honest verifier zero-knowledge) for a honest prover, y is always uniformly distributed (whatever e) and $x = y^3 v^e \bmod n$. For the simulator, this is the same. So, both transcripts have same distribution.

Solution 2 Chameleon Hash Function from Σ-Protocol

This exercise is inspired from Bellare-Ristov, *Hash Functions from Sigma Protocols and Improvements to VSH*, published in the proceedings of ASIACRYPT 2008, LNCS vol. 5350, Springer.

1. *Which objects are missing to define a Σ-protocol?*

   An extractor $E(x, a, e, z, e', z')$ to compute a witness from two accepted transcripts (a, e, z) and (a, e', z') with same commitment a and different challenges $e \neq e'$, and a simulator $S(x, e; r_S)$ to generate a transcript (a, e, z) from x and e with correct distribution.

2. *What is the difference between the hypothesis on E and the special soundness property of Σ-protocols?*

   Now it works whenever $(e, z) \neq (e', z')$ instead of $e \neq e'$. Somehow, the new property for E is stronger than the property of special soundness.

   Show that a strong Σ-protocol is a Σ-protocol.

   Computability and completeness are already satisfied by the definition of a partial Σ-protocol. Special soundness is implied by the new definition of E. We construct a simulator $S(x, e; r) = (H_x(e, z), e, z)$ where $z \in Z_x$ is generated with uniform distribution in Z_x given r. The honest execution of the protocol with instance x generates a transcript (a, e, z) with a given distribution such that $V(x, a, e, z)$ holds and e is uniformly distributed in E_x. Due to the definition of strong Σ-protocols, z is uniformly distributed and independent from e and $a = H_x(e, z)$. So, the transcript has the same distribution as the one from the $S(x, e; r)$ when $e \in E_x$ is random.

3. *Show that given x and w such that $R(x, w)$ holds, we can create a collision on the function H_x.*

   With some random r_P and two different $e, e' \in E_x$ we can compute $a = P(x, w; r_P)$, $z = P(x, w, e; r_P)$, and $z' = P(x, w, e'; r_P)$. Since $V(x, a, e, z)$ and $V(x, a, e', z')$ hold, we must have $a = H_x(e, z)$ and $a = H_x(e', z')$, so $H_x(e, z) = H_x(e', z')$. Since $e \neq e'$, this is a collision.

4. *Show that given $x \in L_R$, finding a collision on H_x implies finding a witness for $x \in L_R$.*

   Assume that $a = H_x(e, z) = H_x(e', z')$ with $(e, z) \neq (e', z')$. We know that $V(a, e, z)$ and $V(a, e', z')$ hold due to the property of a strong Σ-protocol. Since $(e, z) \neq (e', z')$, $w = E(x, a, e, z, e', z')$ is a witness for x.

   Deduce that if R is such that given $x \in L_R$ it is hard to find w such that $R(x, w)$ holds, we can define a trapdoor collision resistant hash function by using x as a common reference string.

   We generate x and w such that $R(x, w)$ holds and declare x as being the common reference string. Then, w is a trapdoor. We have shown that making a collision implies recovering the trapdoor so H_x is collision-resistant.

5. *Recall the Goldwasser-Micali-Wigderson Σ-protocol based on graph isomorphism.*

   The relation is $R((G_0, G_1), \varphi)$ where the witness φ is invertible and such that $\varphi(G_0) = G_1$

Prover		**Verifier**
φ st $\varphi(G_0) = G_1$	(G_0, G_1)	
pick π invertible		pick $e \in \{0, 1\}$
$H = \pi(G_0)$	$\xrightarrow{\quad H \quad}$	
	$\xleftarrow{\quad e \quad}$	
$\sigma = \pi \circ \varphi^{-e}$	$\xrightarrow{\quad \sigma \quad}$	$\sigma(G_e) \stackrel{?}{=} H$

   Show that the Golwasser-Micali-Wigderson Σ-protocol is not a strong Σ-protocol.

   If we have a non-trivial automorphism τ of the graph G_e, then if (H, e, σ) is an accepted transcript, then $(H, e, \sigma \circ \tau)$ as well. However, we cannot extract a witness from the two transcripts.

6. *Recall the Fiat-Shamir Σ-protocol.*

   The relation $R((n, v), s)$ holds if and only if $s^2 v \bmod n = 1$.

Prover		**Verifier**
s st $s^2 v \bmod n = 1$	(n, v)	
pick $r \in \mathbf{Z}_n^*$		pick $e \in \{0, 1\}$
$x = r^2 \bmod n$	$\xrightarrow{x}$	
	$\xleftarrow{e}$	
$y = rs^e \bmod n$	$\xrightarrow{y}$	$y^2 v^e \bmod n \stackrel{?}{=} x$

Show that the Fiat Shamir Σ-protocol is not a strong Σ-protocol.

We can have two accepted transcripts (x, e, y) and $(x, e, -y \bmod n)$ with same x which are not enough to extract a witness.

7. *Recall the Schnorr Σ-protocol.*

The relation $R((G, q, g, y), x)$ holds if and only if $g^x = y$ in group G, where q is a prime greater than 2^t, and g has order q in G.

Prover		**Verifier**
x st $g^x = y$	(G, q, g, y)	
pick $k \in \mathbf{Z}_q$		pick $e \in \{1, \ldots, 2^t\}$
$r = g^k$	$\xrightarrow{r}$	q prime $> 2^t$
	$\xleftarrow{e}$	g, y of order q
$s = ex + k \bmod q$	$\xrightarrow{s}$	$ry^e \stackrel{?}{=} g^s$

Show that the Schnorr Σ-protocol is a strong Σ-protocol.

If (r, e, s) and (r, e', s') are accepted transcripts, we have $s, s' \in \mathbf{Z}_q$, $ry^e = g^s$ and $ry^{e'} = g^{s'}$. If $e \neq e'$ we know that we can extract a witness. If $e = e'$, we obtain that $g^s = g^{s'}$. Since g has order q, we must have $s = s'$ in $\mathbf{Z}_q$. This is not possible if $(e, s) \neq (e', s')$.

Furthermore, (r, e, s) is accepted if and only if $r = g^s y^{-e}$ so we can define $H_y(e, s) = g^s y^{-e}$.

Finally, s is uniformly distributed in $\mathbf{Z}_q$. So, we have a strong Σ-protocol.

Deduce a trapdoor hash function based on this protocol. Does it remind you something?

Let x be a trapdoor and $y = g^x$ be a CRS. We define $H_y(e, s) = g^s y^{-e}$ which looks like the Pedersen commitment.