Advanced Cryptography

Spring Semester 2021

Homework 3

Rolin Nicolas, Leclair Louis

1 A variant of Σ protocol

Question 1. If the Σ protocol fulfills *special soundness* then it means, it exists a polynomially bounded extractor $\mathcal{E}$.
Let $\mathcal{A}$ play the ADVSS_Σ game and let $\mathcal{B}$ play the HARD game defined as:

```
**Game:** $\mathcal{B}(x)$
$(a, e, e', z, z') \leftarrow \mathcal{A}(x)$
$w \leftarrow \mathcal{E}(x, a, e, z, e', z')$
**return** $w$
```

We can see that if $\mathcal{A}$ wins, $\mathcal{B}$ wins. So for a given Σ protocol that has *special soundness* over a hard relation, if the *adversarial special soundness* breaks then the hard relation breaks. Which proves that a Σ protocol that has *special soundness* over a hard relation also fulfills *adversarial special soundness.*

Question 2. The parameters which correspond to the *instance* are (N, t, l, X) and the *witness* is x. The relation R for this protocol is defined as:

$$R := \Big\{\big((N, t, X), x\big) \mid x^t \equiv X \bmod N\Big\}$$

The function R, P, V are polynomially computable since we only have to do multiplication and exponentiation operations which can be done in polynomial time. To prove the correctness, we will show that both sides are equal if and only if $X = x^t \bmod N$.

$$z^t = (x^e r)^t = x^{et} r^t \bmod N$$
$$aX^e = (x^e r)^t = x^{et} r^t \bmod N$$

Since $X = x^t \bmod N$, we show that if both actors act honestly then the last verification pass. Thus proving that Σ_{RSA} is a Σ protocol.

Question 3. First we show that Σ_{RSA} fulfills special soundness. For that we need to show there exists an extractor $\mathcal{E}$ polynomially bounded.

```
Extractor: $\mathcal{E}\big((N,t,X),a,e,z,e',z'\big)$
if $e < e'$ then
    $(e,e') \leftarrow (e',e)$
end
$\alpha,\beta \leftarrow \mathsf{ExtendedEuclidian}(e-e',t)$
$x \leftarrow (\frac{z}{z'})^{\alpha}X^{\beta}$
return $x$
```

Where ExtendedEuclidian stands for the execution of the extended Euclidian algorithm which returns the coefficients α,β when computing the gcd between two numbers. This algorithm works since: $e-e' < t$ and t is prime so we are sure that $\mathsf{gdc}(e-e',t) = 1$ so $\exists(\alpha,\beta) \in \mathbb{Z}^2$ such that $\alpha(e-e')+\beta t = 1$. Thus we have:

$$(\frac{z}{z'})^{\alpha}X^{\beta} = x^{(e-e')\alpha}x^{t\beta} = x$$

Which proves that we can recover a witness in a polynomial time using an extractor.
The relation R is HARD since finding a witness for any X is equivalent to solving the RSA problem, which supposed to be hard in this question. We also know the existence of a polynomially bounded extractor so the protocol fulfills *special soundness* and thanks to question 2, we know that Σ_{RSA} fulfills *adversarial special soundness*. Now, one can see that if an adversary wins in the *strong special soundness* game then he also wins in the *adversarial special soundness* game. If we have $e = e'$ we must have $z = z'$ since $z = x^e r$, $z' = x^{e'}r$ and r and x are determined before then $z = z'$ so the adversary cannot win other than having $e \neq e'$. Therefore, in this Σ_{RSA} protocol, for any $\mathcal{A}$, we have:

only because gcd(t, \phi(N)) = 1 -0.5pt

$$\Pr[\mathsf{STRSS}_{\Sigma_{RSA}}(\mathcal{A}) \Rightarrow \textbf{true}] \leq \Pr[\mathsf{ADVSS}_{\Sigma_{RSA}}(\mathcal{A}) \Rightarrow \textbf{true}]$$

Hence for any ppt $\mathcal{A}$, we have:

$$\mathsf{Adv}^{\mathsf{strss}}_{\mathcal{A},\Sigma_{RSA}} \leq \mathsf{Adv}^{\mathsf{advss}}_{\mathcal{A},\Sigma_{RSA}}$$

Meaning that if the protocol fulfills *adversarial special soundness* then it also fulfills *strong special soundness*. All in all, Σ_{RSA} fulfills *strong special soundness*.

Question 4. First we need to prove that Σ_{RSA} fulfills special HVZK, to do that we have to show the existence of a polynomially bounded simulator. So given an input x, e we can generate a transcript (a, e, z) efficiently.

```
**Simulator:** $\mathcal{S}(x, e)$
$z \leftarrow_{\$} \mathbb{Z}_N^*$
$a \leftarrow z^t X^{-e}$
**return** $(a, e, z)$
```

First, we will prove the correctness of the simulator: $\forall a, e, e', z, z' : S(x, e) = (a, e, z)$, so $e = e'$ then $aX^e = \frac{z^t}{X^e} X^e = z^t$ which is the expected result thus the result is correct. Then we can see that we only used exponentiation and multiplication operations in the algorithm thus making it polynomially bounded. Finally, we need to prove that the transcript from the simulator as the same distribution as a correct execution of the protocol.

$$\Pr[\mathsf{View}_v \to z|x, e] = \Pr[\mathsf{View}_v \to x^e r|x, e]$$

Since r is chosen uniformly at random.

$$\Pr[\mathsf{View}_v \to x^e r|x, e] = \frac{1}{\Phi(N)}$$

And for the simulator, we have:

$$\Pr[\mathbb{S}(x, e) \to z|x, e] = \frac{1}{\Phi(N)} = \Pr[\mathsf{View}_v \to x^e r|x, e]$$

Since they are equal, it means that it is well distributed thus it is a special HVZK. Knowing we have a special HVZK and a simulator, we only have to prove that we have a deterministic StrSim. In our case, we have $\mathsf{StrSim}(x, e, z) = z^t X^{-e}$. Which is a deterministic function since these operations are deterministic thus the Σ_{RSA} protocol is *strong* HVZK.

Question 5. First we will define the two algorithm **Gen** and **H**.

```
Generator: $\mathsf{Gen}_H(1^\lambda)$
$(x, w) \leftarrow_\$ \mathsf{Gen}_\Sigma(1^\lambda)$
return $x$
```

```
Hash: $H(k, x_1, x_2)$
return $\mathsf{StrSim(k, x_1, x_2)}$
```

Now we will prove that for any $\mathcal{A}$ one can build $\mathcal{B}$ such that:

$$\mathsf{Adv}^{\mathsf{cr}}_{\mathcal{A},\mathcal{H}} \leq \mathsf{Adv}^{\mathsf{strss}}_{\mathcal{B},\Sigma}$$

Let $\mathcal{A}$ plays the $\mathcal{H}$ game and $\mathcal{B}$ plays the STRSS game be:

```
Game: $\mathcal{B}(x)$
$(e, z, e', z') \leftarrow \mathcal{A}(x)$
$a \leftarrow \mathsf{StrSim(x, e, z)}$
return $(a, e, e', z, z')$
```

Here we have:

$$\begin{aligned}
\mathsf{Adv}^{\mathsf{strss}}_{\mathcal{B},\Sigma} &= \Pr[\mathsf{STRSS}_\Sigma(\mathcal{B} \to \mathbf{true})] = \Pr[(e, z) \neq (e', z') \wedge V(x, a, e, z) \wedge V(x, a, e', z') : \\
&\qquad (e, z, e', z) \leftarrow_\$ \mathcal{A}(x); (x, w) \leftarrow \mathsf{Gen}_\Sigma(1^\lambda); a \leftarrow \mathsf{StrSim}(x, e, z)] \\
&= \Pr[(e, z) \neq (e', z') \wedge V(x, \mathsf{StrSim}(x, e, z), e, z) \wedge V(x, \mathsf{StrSim}(x, e, z), e', z') : \\
&\qquad (e, z, e', z) \leftarrow_\$ \mathcal{A}(x); x \leftarrow \mathsf{Gen}_H(1^\lambda)] \\
&= \Pr[(e, z) \neq (e', z') \wedge V(x, \mathsf{StrSim}(x, e, z), e, z) \wedge V(x, \mathsf{StrSim}(x, e', z'), e', z') \wedge \\
&\qquad \mathsf{StrSim}(x, e, z) = \mathsf{StrSim}(x, e', z') : (e, z, e', z) \leftarrow_\$ \mathcal{A}(x); x \leftarrow \mathsf{Gen}_H(1^\lambda)]
\end{aligned}$$

Sim is a correct simulator for Σ so $V(x, \mathsf{StrSim}(x, e, z), e, z)$ and $V(x, \mathsf{StrSim}(x, e', z'), e', z')$ are always true:

$$\begin{aligned}
&= \Pr[(e, z) \neq (e', z') \wedge H(x, e, z) \wedge H(x, e', z') : (e, z, e', z) \leftarrow_\$ \mathcal{A}(x); x \leftarrow \mathsf{Gen}_H(1^\lambda)] \\
&= \mathsf{Adv}^{\mathsf{cr}}_{\mathcal{A},\mathcal{H}}
\end{aligned}$$

Which proves the initial statement.

Question 6. To show that the Fiat-Shamir Σ protocol doesn't fulfill strong special soundness we will show that given a valid transcript (a, e, z) it is easy to obtain a second valid transcript (a, e', z') such that $(e, z) \neq (e', z')$.
To obtain this second valid transcript we have $(a, e' = e, z' = -z)$. The second transcript still is a valid transcript since we have $(e, z) \neq (e', z')$ and it also passes the validation step at the end since:

$$(z')^2 \bmod n = (-z)^2 \bmod n = z^2 \bmod n$$

So:

$$(z')^2 v^e \bmod n = z^2 v^e \bmod n = x$$

Question 7. The modification we will make will be that if: $y > \frac{n}{2} : y \leftarrow -y \bmod n$.
If we have (x, e, y, e', y') such that $(e, y) \neq (e', y') \wedge V(n, v, x, e, y) \wedge V(n, v, x, e', y')$ then:

$$y^2 v^e \bmod n = x = (y')^2 v^{e'} \bmod n$$
$$(\frac{y}{y'})^2 \bmod n = v^{e'-e} \bmod n$$
$$(\frac{y}{y'})^2 v^{e-e'} \bmod n = 1$$

If we have $e = e'$, then we have a square root of 1 different from 1 (because $y \bmod n \neq y' \bmod n$ and -1 (because $y \bmod n \neq -y' \bmod n$).
And if $e \neq e'$ then $(\frac{y}{y'})^2 v \bmod n = 1$. We found a witness s for (n, v).
So in both cases, finding such a tuple is equivalent to solve a hard problem so this new protocol fulfills *strong special soundness*.

why is it a hard problem? -0.5pt