Advanced Cryptography — Midterm Exam
Solution

Serge Vaudenay

27.4.2023

– duration: 1h45
– any document allowed
– a pocket calculator is allowed
– communication devices are not allowed
– the exam invigilators will **not** answer any technical question during the exam
– readability and style of writing will be part of the grade

The exam grade follows a linear scale in which each question has the same weight.

1 Computing Squares in Exponent Domain

We consider an algorithm $\mathsf{Setup}(1^s) \xrightarrow{\$} \mathsf{pp}$ based on a security parameter s which generates public parameters pp which include a group element g, the order q of g in the group (assumed to be an odd prime), and materials to be able to do group operations. We define the following three games.

Game CDH
1: $\mathsf{Setup}(1^s) \xrightarrow{\$} \mathsf{pp}$
2: pick $x, y \in \mathbf{Z}_q$
3: $X \leftarrow g^x$, $Y \leftarrow g^y$
4: $\mathcal{A}(\mathsf{pp}, X, Y) \xrightarrow{\$} K$
5: **return** $1_{K=g^{xy}}$

Game CDH^*
1: $\mathsf{Setup}(1^s) \xrightarrow{\$} \mathsf{pp}$
2: pick $x, y \in \mathbf{Z}_q^*$
3: $X \leftarrow g^x$, $Y \leftarrow g^y$
4: $\mathcal{A}(\mathsf{pp}, X, Y) \xrightarrow{\$} K$
5: **return** $1_{K=g^{xy}}$

Game Sqr
1: $\mathsf{Setup}(1^s) \xrightarrow{\$} \mathsf{pp}$
2: pick $x \in \mathbf{Z}_q$
3: $X \leftarrow g^x$
4: $\mathcal{A}(\mathsf{pp}, X) \xrightarrow{\$} Y$
5: **return** $1_{Y=g^{x^2}}$

The hardness of a game means that for any PPT algorithm $\mathcal{A}$, the probability that the game returns 1 is a negligible function of s.

Q.1 Prove that the hardness of any of those games imply that $E(\frac{1}{q})$ is a negligible function of s.
HINT: construct an adversary who wins with advantage $E(\frac{1}{q})$.

We consider an adversary $\mathcal{A}$ with input X who picks $x' \in \mathbf{Z}_q$ at random and aborts if $g^{x'} \neq X$. Otherwise, we have $g^{x'} = X$ and either game can be won in a trivial way: CDH or CDH^* by answering $Y^{x'}$, and Sqr by answering $g^{(x')^2}$. Clearly, $\mathcal{A}$ wins with probability $\frac{1}{q}$ with a fixed group. Hence, the advantage is $E(\frac{1}{q})$. The hardness of either game implies that this is negligible.

Q.2 Prove that the hardness of CDH and of CDH* are equivalent.

The difference between CDH *and* CDH* *is obtained by the failure case* $x = 0$ or $y = 0$. *The difference of advantage for an adversary playing both games is bounded by the difference Lemma, hence by the probability that this failure event happens. It is bounded by* $E(\frac{2}{q})$ *which is negligible, thanks to the previous question. Hence, the advantage difference is negligible for any* $\mathcal{A}$. *We deduce that the hardness of one game implies the hardness of the other game.*

Q.3 Prove that the hardness of Sqr implies the hardness of CDH.

HINT: be careful about distributions.

We consider an adversary $\mathcal{A}$ playing CDH. *We construct an adversary $\mathcal{B}$ playing* Sqr *as follows.*

$\mathcal{B}(\mathsf{pp}, X)$:

pick $\lambda \in \mathbf{Z}_q$
$Y \leftarrow X g^{\lambda}$
$K \leftarrow \mathcal{A}(\mathsf{pp}, X, Y)$
$Z \leftarrow K X^{-\lambda}$
return Z

For $x \in \mathbf{Z}_q$ uniform, $(x, x+\lambda)$ is uniformly distributed in $\mathbf{Z}_q^2$. Hence, the input to $\mathcal{A}$ follows the same distribution as in the CDH *game. Consequently, we have $K = g^{x^2+\lambda x}$ with probability $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{CDH}}(s)$, in which case we have $Z = g^{x^2}$. Hence, $\mathsf{Adv}_{\mathcal{B}}^{\mathsf{Sqr}}(s) = \mathsf{Adv}_{\mathcal{A}}^{\mathsf{CDH}}(s)$. Since* Sqr *is hard, we deduce that $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{CDH}}(s)$ is negligible. As this holds for any PPT $\mathcal{A}$, we deduce that* CDH *is hard.*

Another possible solution was to use

$\mathcal{B}(\mathsf{pp}, X)$:

if $X = 1$ **then return** 1
pick $\lambda \in \mathbf{Z}_q^*$
$Y \leftarrow X^{\lambda}$
$K \leftarrow \mathcal{A}(\mathsf{pp}, X, Y)$
$Z \leftarrow K^{\frac{1}{\lambda}}$
return Z

but it required to be careful with distributions: to treat the $X = 1$ case separately and to use $\mathcal{A}$ playing CDH*. *Then, we could use the previous question.*

A common mistake is to define $\mathcal{B}(\mathsf{pp}, X) = \mathcal{A}(\mathsf{pp}, X, X)$. This solution does not work because we can only say how successful $\mathcal{A}$ is with uniformly distributed input (X, Y). In this solution, the input (X, X) is not uniform. It could be the case that $\mathcal{A}$ works very well on average over (X, Y) (so break CDH*) but always fail when $X = Y$.*

A more subtle mistake was to use

$\mathcal{B}(\mathsf{pp}, X)$:

if $X = 1$ **then return** 1
pick $\lambda \in \mathbf{Z}_q^*$
$X' \leftarrow X^{\lambda}$
$Y \leftarrow X^{\frac{1}{\lambda}}$
$K \leftarrow \mathcal{A}(\mathsf{pp}, X, Y)$
return K

and arguing that (X', Y) is uniformly distributed, which is wrong. Indeed, althrough X' and Y are both uniform, they are not independent as $X'Y = g^{x^2}$ is the exponential of a quadratic residue, so $X'Y$ is constained to be in half of the group.

Q.4 Prove that the hardness of CDH implies the hardness of Sqr.
HINT: be careful about distributions.

We consider an adversary $\mathcal{A}$ *playing* Sqr. *We construct an adversary* $\mathcal{B}$ *playing* CDH *as follows.*

$\mathcal{B}(\mathsf{pp}, X, Y)$:

$U \leftarrow \mathcal{A}(\mathsf{pp}, XY)$
$V \leftarrow \mathcal{A}(\mathsf{pp}, X/Y)$
$Z \leftarrow (U/V)^{\frac{1}{4}}$
return Z

Since 2 is invertible modulo q, (x, y) *uniform in* $\mathbf{Z}_q^2$ *is equivalent to* $(x+y, x-y)$ *uniform in* $\mathbf{Z}_q^2$. *Hence,* XY *and* X/Y *are independent and uniform in the group. For any value of* pp, *let* p_{pp} *be the success probability of* $\mathcal{A}$ *in* Sqr *conditioned to the value of* pp *being selected. We have that* $\mathcal{B}$ *succeeds with probability* p_{pp}^2 *conditionned to* pp. *Hence,* $\mathsf{Adv}_{\mathcal{B}}^{\mathsf{CDH}}(s) = E(p_{\mathsf{pp}}^2)$ *while* $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{Sqr}}(s) = E(p_{\mathsf{pp}})$. *Due to the Jensen inequality, we have* $E(p_{\mathsf{pp}}^2) \geq E(p_{\mathsf{pp}})^2$. *So,* $\mathsf{Adv}_{\mathcal{B}}^{\mathsf{CDH}}(s) \geq \mathsf{Adv}_{\mathcal{A}}^{\mathsf{Sqr}}(s)^2$. *By assumption, we know that* $\mathsf{Adv}_{\mathcal{B}}^{\mathsf{CDH}}(s)$ *is negligible. So,* $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{Sqr}}(s)^2$ *is negligible too, and so is* $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{Sqr}}(s)$. *As this holds for any PPT* $\mathcal{A}$, *we deduce that* Sqr *is hard.*

A common mistake was to answer something like

$\mathcal{B}(\mathsf{pp}, X, Y)$:

$U \leftarrow \mathcal{A}(\mathsf{pp}, XY)$
$V \leftarrow \mathcal{A}(\mathsf{pp}, X)$
$W \leftarrow \mathcal{A}(\mathsf{pp}, Y)$
$Z \leftarrow (U/(VW))^{\frac{1}{2}}$
return Z

The problem here is that the 3 executions of $\mathcal{A}$ *are not done with independent inputs. So we cannot say this succeeds with probability* p_{pp}^3. *It could be the case that* $\mathcal{A}$ *never succeed at the same time on* X, Y, *and* XY *but still succeeds well on average. For instance, if* $\mathcal{A}$ *succeeds on all* $X = g^x$ *such that* x *is odd, then it succeeds on half of the group but never at the same time on* X, Y, *and* XY.

No student noticed the problem of computing the probabilities over a fixed pp *then using the Jensen inequality.*

2 Proof of DDH

We consider a PPT algorithm $\mathsf{Setup}(1^s) \xrightarrow{\$} \mathsf{pp} = (\ldots, g, q)$ based on a security parameter s which generates public parameters pp which include a group element g, the order q of g in the group (assumed to be prime), and materials to be able to do group operations. We consider the two following relations:

$$R((\mathsf{pp}, X, Y, K), y) : Y = g^y \wedge K = X^y$$
$$R'((\mathsf{pp}, X, Y, K), (x, y)) : X = g^x \wedge Y = g^y \wedge K = g^{xy}$$

Q.1 Construct a Σ-protocol for the relation R. Carefully specify all elements required in a Σ protocol.

We use the discrete log equality protocol from the generalized Schnorr protocol for $\varphi(y) = (g^y, X^y)$:

- *The challenge set is $\mathbf{Z}_q$.*
- *The prover picks $k \in \mathbf{Z}_q$ and sends $(R_1, R_2) = \varphi(k) = (g^k, X^k)$. After getting $e \in \mathbf{Z}_q$, the prover sends $s = ey + k \bmod q$.*
- *The verifier checks $\varphi(s) = (R_1 + eY, R_2 + eK)$.*
- *The extractor with $(R_1, R_2, e_1, s_1, e_2, s_2)$ with $e_1 \neq e_2$ computes $y = \frac{s_2 - s_1}{e_2 - e_1} \bmod q$.*
- *The simulator picks $s \in \mathbf{Z}_q$ at random and computes $(R_1, R_2) = \varphi(s) - e(Y, K)$.*

All algorithms are clearly PPT. Completeness comes from the homomorphic property of φ:

$$\varphi(s) = e\varphi(y) + \varphi(k) = e(R_1, R_2) + (Y, K)$$

Extraction comes from

$$\varphi(s_2 - s_1) = (R_1 + e_2Y, R_2 + e_2K) - (R_1 + e_1Y, R_2 + e_1K) = (e_2 - e_1).(Y, K)$$

so $\varphi(y) = (Y, K)$. The simulation property comes from the usual trick: in the honest protocol, we observe that $s = ey + k$ is uniformly distributed in $\mathbf{Z}_q$ since k is uniform and e is independent. Then, (R_1, R_2) uniquely follows from e and s.

Q.2 Construct a Σ-protocol for the relation R'. Carefully specify all elements required in a Σ protocol.

We define $R''((\mathsf{pp}, X, Y, K), x) \Leftrightarrow X = g^x$. *We have*

$$R'((\mathsf{pp}, X, Y, K), (x, y)) \Leftrightarrow R(\mathsf{pp}, X, Y, K), y) \wedge R''((\mathsf{pp}, X, Y, K), x)$$

Hence, we can use an AND proof between the previous protocol and the Schnorr protocol. The result of this AND proof is as follows.

- *The challenge set is* $\mathbf{Z}_q$.
- *The prover picks* $k, k' \in \mathbf{Z}_q$ *and sends* $R_1 = g^k$, $R_2 = X^k$, *and* $R_3 = g^{k'}$. *After getting* $e \in \mathbf{Z}_q$, *the prover sends* $s = ey + k \bmod q$ *and* $s' = ex + k' \bmod q$.
- *The verifier checks* $g^s = R_1 Y^e$, $X^s = R_2 K^e$, *and* $g^{s'} = R_3 X^e$.
- *The extractor with* $(R_1, R_2, R_3, e_1, s_1, s'_1, e_2, s_2, s'_2)$ *with* $e_1 \neq e_2$ *computes* $y = \frac{s_2 - s_1}{e_2 - e_1} \bmod q$ *and* $x = \frac{s'_2 - s'_1}{e_2 - e_1} \bmod q$.
- *The simulator picks* $s, s' \in \mathbf{Z}_q$ *at random and computes* $R_1 = g^s Y^{-e}$, $R_2 = X^s K^{-e}$, *and* $R_3 = g^{s'} X^{-e}$.

The properties satisfied by the Σ *protocol follow from the AND construction.*

A mistake which was observed several times was to use $k = k'$ *in the above construction. The problem occurred in the simulator who could not enforce a good distribution. Actually,* (s, s') *is not uniformly distributed because* $s' - s = e(y - x)$. *As a matter of fact, such solution leaks* $y - x = \frac{s - s'}{e}$ *so is not zero-knowledge.*