Advanced Cryptography

Spring Semester 2025

Homework 1

- This homework contains two questions: (1) **Temperamental oracles** (2) **Identity-based encryption**
- You will submit a **report** that will contain all your answers and explanations. The report should be a PDF document. You can use any editor to prepare the report, but Latex is usually the best choice for typesetting math and pseudocode.
- We ask you to **work alone** or in **teams of two**. If you work in a team, please register it on Moodle and **indicate both teammates' name on the report**. Feel free to ask questions to the T.A.
- We expect all answers to be **formally justified** in order to get full points. Please pay attention to the **rigour** and **clarity** of your answers.
- We might announce some typos for this homework on Moodle in the "news" forum. Everybody is subscribed to it and does receive an email as well. If you decided to ignore Moodle emails we recommend that you check the forum regularly.
- The homework is due on Moodle on **March 27th, 23:59**.

1 Temperamental oracles

Group action cryptography is considered as a post-quantum candidate and is a natural extension of the discrete logarithm setting in the classical cryptography realm. We find group action analoguous of the discrete logarithm problem and the Computational Diffie-Hellman (CDH) problem (See Definition 3).

Much like in the classical setting, an interesting question is to relate the hardness of these two problems. Clearly, if one has access to a discrete logarithm oracle, one can solve the Computational Diffie-Hellman problem, both in the classical setting as well as in the setting of group actions. An interesting question is whether the converse is true: is it possible to solve the discrete logarithm problem given a CDH oracle?

In 2021, Galbraith, Panny, Smith and Vercauteren gave a positive answer to this question in the setting of group action. Their reduction is however **quantum** (i.e. it requires use of a quantum algorithm) and requires an oracle with perfect correctness (i.e. an oracle that always returns a correct output). However, in practice it is not realistic to assume access to a perfect oracle, and subsequent work has looked into extending the result to imperfect oracles.

In this exercise, we will look into the notion of imperfect oracles and into the problem of transforming them into perfect ones.

We start with a few reminders and definitions.

Definition 1 (Group action)**.** We say that a group G acts on a set X if there exists a map $\star : G \times X \to X$ such that :

- Identity: If e is the identity element of G, then for any $x \in X$, $e \star x = x$;
- Compatibility: For any $g, h \in G$, and any $x \in X$, $(gh) \star x = g \star (h \star x)$.

Such a group action can be denoted $(G, X, \star)$.

We call a group action *transitive* if for any two elements $x, y \in X$, there exists a group element $g \in G$ mapping x to y, i.e. $y = g \star x$.
We say a group action is *free* if for each $g \in G$ it is the identity element if and only if there exists an element $x \in X$ such that $x = g \star x$.
A group action is called *regular* if it is both transitive and free.
It is called *abelian* if the group that is acting is an abelian group.

Definition 2 (Effective group action)**.** We call a group action $\star : G \times X \to X$ *effective* if the following properties hold :

- G is finite and there exists a PPT algorithm for the following operations : group operation, computation of inverses, membership testing, equality testing and sampling.
- X is finite and there exists a PPT algorithm for membership testing and for computing a unique representation of elements in X.
- There exists a distinguished element $x_0 \in X$ such that its bit-string representation is known. We call this point the *origin*.
- $\star$ can be computed efficiently for any $g \in G, x \in X$.

Definition 3 (GA-CDH)**.** Consider a regular effective group action $(G, X, \star)$ with distinguished element x_0, then the computational Diffie-Hellman problem is hard if for any PPT algorithm $\mathcal{A}$ we have

$$\mathrm{Adv}(\mathcal{A}) = \Pr[GA\text{-}CDH(\mathcal{A}) \to 1] = \mathsf{negl}(\lambda)$$

$\underline{GA\text{-}CDH(\mathcal{A})}$

1: $g_1, g_2 \xleftarrow{\$} G^2$
2: $\mathcal{A}(g_1 \star x_0, g_2 \star x_0) \to x'$
3: **return** $\mathbf{1}_{(x' = (g_1 g_2) \star x_0)}$

Throughout the exercise, we assume that the group action $(G, X, \star)$ is an **abelian regular effective** group action.

1.1 Random self-reduction

Consider a probabilistic oracle $\mathcal{O}$ such that

$$\Pr_{g_1,g_2,\text{coins in } \mathcal{O}}[\mathcal{O}(g_1 \star x_0, g_2 \star x_0) = g_1 g_2 \star x_0] \geq \alpha$$

Reduce it to an oracle $\mathcal{O}'$ such that for any $g_1, g_2 \in G \times G$,

$$\Pr_{\text{coins in } \mathcal{O}'}[\mathcal{O}'(g_1 \star x_0, g_2 \star x_0) = g_1 g_2 \star x_0] \geq \alpha$$

1.2 An example of imperfect oracle

We now turn our attention to the case of probabilistic oracles and study two different examples.

Let $h_2, h_3 \in G$ be group elements such that $h_2^2 = 1 = h_3^3$.

Consider the following imperfect CDH oracle $\mathcal{O}_1$, which on input $y, z = (g_1 \star x_0, g_2 \star x_0)$ returns:

- $g_1 g_2 \star x_0$ with probability $1/4$
- $h_2 g_1 g_2 \star x_0$ with probability $1/4$
- $h_3 g_1 g_2 \star x_0$ with probability $1/4$
- x' for some random element $x' \in X$ with probability $1/4$

i.e. we have

$\underline{\mathcal{O}_1(y, z)}$ ▷ (We have $y = g_1 \star x_0$, $z = g_2 \star x_0$)

1: $b \xleftarrow{\$} \{0, 1, 2, 3\}$
2: **if** $b = 0$ **then return** $g_1 g_2 \star x_0$
3: **end if**
4: **if** $b = 1$ **then return** $h_2 g_1 g_2 \star x_0$
5: **end if**
6: **if** $b = 2$ **then return** $h_3 g_1 g_2 \star x_0$
7: **end if**
8: **if** $b = 3$ **then return** $x' \xleftarrow{\$} X$
9: **end if**

1. Show that for any $y \in X$, there exists a unique $g \in G$ such that $y = g \star x_0$.
2. Give an adversary $\mathcal{B}$, which, given access to the oracle $\mathcal{O}_1$, wins the GA-CDH game with overwhelming probability (i.e. with probability at least $1 - \mathsf{negl}(\lambda)$).

1.2.1 Another imperfect oracle

Let us now modify the above oracle $\mathcal{O}_1$ into an oracle $\mathcal{O}_2$ that behaves in the same way as $\mathcal{O}_1$ except that in the last case instead of returning a random x_0, it returns $h_2 h_3 g_1 g_2 \star x_0$.

$\underline{\mathcal{O}_2(y,z)}$ ▷ (We have $y = g_1 \star x_0$, $z = g_2 \star x_0$)

1: $b \xleftarrow{\$} \{0,1,2,3\}$
2: **if** $b = 0$ **then return** $g_1 g_2 \star x_0$
3: **end if**
4: **if** $b = 1$ **then return** $h_2 g_1 g_2 \star x_0$
5: **end if**
6: **if** $b = 2$ **then return** $h_3 g_1 g_2 \star x_0$
7: **end if**
8: **if** $b = 3$ **then return** $h_2 h_3 g_1 g_2 \star x_0$
9: **end if**

Can you still give an efficient adversary $\mathcal{B}$ that wins the *GA-CDH* game? If yes, give a description of $\mathcal{B}$. If not, argue why.

2 Identity-based encryption

We give below a formal definition of a cryptographic primitive called "identity-based encryption".
Intuitively, it differs from regular public key encryption as there is a master pair of public and secret keys, that allows to derive user specific secret keys. To encrypt a message, one only needs a user id (typically this could be a string such as a username or email address) and the master public key. To derive a user key, one needs the user id and the master secret key. To decrypt a message addressed to a given user, one needs the associated user secret key.

Definition 4 (Identity-Based Encryption)**.** An Identity-Based Encryption (IBE) scheme Π with message space $\mathcal{M}$ and ciphertext space $\mathcal{C}$ is a tuple of algorithms defined in the following way :

- $\mathsf{Setup}(1^\lambda) \to \mathsf{pp}$: On input the security parameter λ, it outputs public parameters pp.
- $\mathsf{KeyGen}(\mathsf{pp}) \to (\mathsf{mpk}, \mathsf{sk})$: On input the public parameters, it outputs a master public key mpk and master secret key sk.
- $\mathsf{Extract}(\mathsf{pp}, \mathsf{sk}, id) \to \mathsf{sk}_{id}$: On input the public parameters, the master secret key and a user id, it returns an associated user secret key sk_{id}.
- $\mathsf{Enc}(\mathsf{pp}, \mathsf{mpk}, id, m) \to c$ on input the public parameters, a user id and a message, it returns a ciphertext c.
- $\mathsf{Dec}(\mathsf{pp}, \mathsf{sk}_{id}, c) \to m$: On input the public parameters, a ciphertext c the user secret key sk_{id}, it returns the plaintext m.

2.1 Formalism

Propose a notion of one-way security for identity-based encryption together with the associated security game.

A concrete proposal Consider the following identity-based encryption algorithm with message space $\mathcal{M} = \{-1, 1\}$. Our scheme does not need any Setup procedure.

Keygen: Choose two random primes p, q such that $gcd(p-1, q-1, 3) = 1$. Let $N = p \cdot q$ and set $\mathsf{mpk} = N$ and $\mathsf{sk} = (p, q)$.

$\mathsf{Extract}(\mathsf{sk}, id)$: Let $H : \{0,1\}^* \to \mathbb{Z}_N^*$ be a hash function. Let $a = H(id)$. Using the knowledge of p, q compute $\mathsf{sk}_{id} = a^{1/3} \mod N$. Return sk_{id}.

$\mathsf{Enc}(\mathsf{mpk}, id, m)$: Let $a = H(id)$. Choose a random v such that

$$\left(\frac{v^3 - a}{N}\right) = m$$

where $(\frac{x}{N})$ denotes the Jacobi symbol.

Return the ciphertext $c = \frac{v(v^3+8a)}{4(v^3-a)} \mod N$.

$\mathsf{Dec}(\mathsf{mpk}, id, \mathsf{sk}_{id}, c)$: Return $(\frac{c-\mathsf{sk}_{id}}{N})$.

2.2 Correctness

Prove that this IBE is correct.

2.3 Attack

Show that you can break the one-way CPA security of this scheme.

Hint: You might want to expand out the identity $(v^6 - 20av^3 - 8a^2)^2$ and try to relate it to some combination of your ciphertext c and a.

2.4 Generalization

Consider now a generalization of this scheme.

Suppose that the master key generation stays the same, but we modify the extract procedure in the following way:

$\mathsf{Extract}(\mathsf{sk}, id)$: Let $H : \{0,1\}^* \to \mathbb{Z}_N^*$ be a hash function. Let $a = H(id)$. Using the knowledge of p, q compute $\mathsf{sk}_{id} = a^{1/k} \mod N$. Return sk_{id}.

Let $f(X)$ be a polynomial of degree at most $k-1$ chosen by the person who encrypts, i.e. they set $c + X = tf(X)^2 \mod (X^k - a)$ with $\left(\frac{t}{N}\right) = m$.

Explain how to decrypt in this general case and show the correctness.

In the case $k = 3$, deduce some constraints on the shape of f and show how we can reduce to the case of the previous section.