TD3: Security Assumptions

Exercise 1.
Let $(\mathsf{Enc}, \mathsf{Dec})$ be an encryption scheme over $K \times P \times \{0,1\}^n$.

1. In this question, we assume that $(\mathsf{Enc}, \mathsf{Dec})$ is smCPA-secure. Prove that there exists a smCPA-secure encryption scheme $(\mathsf{Enc}', \mathsf{Dec}')$ such that $G : k \mapsto \mathsf{Enc}'(k, 0)$ is not a secure PRG. *Hint: try to concatenate constant bits to every ciphertext.*

Exercise 2. *Attacking the DLG problem*
Let $\mathbb{G}$ be a cyclic group generated by g, of (known) prime order p, and let h be an element of $\mathbb{G}$. Let $F : \mathbb{G} \to \mathbb{Z}_p$ be a nonzero function, and let us define the function $H : \mathbb{G} \to \mathbb{G}$ by $H(\alpha) = \alpha \cdot h \cdot g^{F(\alpha)}$. We consider the following algorithm (called *Pollard ρ Algorithm*).

Pollard ρ Algorithm

Input: $h, g \in \mathbb{G}$

Output: $x \in \{0, \dots, p-1\}$ such that $h = g^x$ or FAIL.

1. $i \leftarrow 1$
2. $x \leftarrow 0$, $\alpha \leftarrow h$
3. $y \leftarrow F(\alpha)$; $\beta \leftarrow H(\alpha)$
4. **while** $\alpha \neq \beta$ do
5. $\quad x \leftarrow x + F(\alpha) \bmod p; \alpha \leftarrow H(\alpha)$
6. $\quad y \leftarrow y + F(\beta) \bmod p; \beta \leftarrow H(\beta)$
7. $\quad y \leftarrow y + F(\beta) \bmod p; \beta \leftarrow H(\beta)$
8. $\quad i \leftarrow i + 1$
9. **end while**
10. **if** $i < p$ **then**
11. $\quad$ **return** $(x - y)/i \bmod p$
12. **else**
13. $\quad$ **return** FAIL
14. **end if**

To study this algorithm, we define the sequence (γ_i) by $\gamma_1 = h$ and $\gamma_{i+1} = H(\gamma_i)$ for $i \geqslant 1$.

1. Show that in the **while** loop from Steps 4 to 9 of the algorithm, we have $\alpha = \gamma_i = g^x h^i$ and $\beta = \gamma_{2i} = g^y h^{2i}$.

2. Show that if this loop terminates with $i < p$, then the algorithm returns the discrete logarithm of h in basis g.

3. Let j be the smallest integer such that there exists $k < j$ such that $\gamma_j = \gamma_k$. Show that $j \leqslant p + 1$ and that the loop ends with $i < j$.

4. Show that if F is a random function, then the average execution time of the algorithm is in $O(p^{1/2})$ multiplications in $\mathbb{G}$.