Skanda Koppula
Collaborators: Jonathan Frankle, Sam DeLaughter

Problem 2.1. UOWHF is the stronger primitive. In the definition for UOWHF, the adversary must commit to an x before the key for the hash function is picked. In a CRHF construction, the adversary has knowledge of the keyed hash function when determining the set of colliding inputs. The adversary for UOWHF is stronger, and correspondingly, the UOWHF is the stronger primitive.

Problem 2.2.a We show this by contradiction. Suppose there existed a PPT A that could non-negligibly extract claws from f_0 and f_1. We construct PPT B to factor N:

1. Query A for the claw (α_i, α_j). By definition, $f_0(\alpha_i) = f_1(\alpha_j)$, or equivalently, $\alpha_i^2 = 4\alpha_j^2 \mod N$.
2. B now performs the factorization. We simplify:

$$\alpha_i^2 - 4\alpha_j^2 = \mod N$$
$$(\alpha_i - 2\alpha_j)(\alpha_i + 2\alpha_j) = 0 \mod N$$

Note that $\left(\frac{\alpha_i}{N}\right) = 1$ (because $f_0 : \mathsf{QR} \to \mathsf{QR}$). Furthermore, $\left(\frac{2\alpha_j}{N}\right) = \left(\frac{2}{N}\right)\left(\frac{\alpha_j}{N}\right) = (-1)(1) = -1$ (because $N = 21 = 5 \mod 8$, implying $\left(\frac{2}{N}\right) = -1$ by Fact 1, and $f_1 : \mathsf{QR} \to \mathsf{QR}$).

Because the Jacobi symbol of α_i and $2\alpha_j$ differ, they cannot be the same value, so $\alpha_i - 2\alpha_j$ must be a non-zero quantity. Similarly, $\alpha_i + 2\alpha_j$ must be a non-zero quantity. We have have thus found two numbers, when taken $\mod N$, must be divisors of N.

We reach a contradiction with the hardness of factorization assumption, so no such B can exist. The family (f_o, f_1) is claw-free.

Problem 2.2.b Suppose there existed a PPT A that produced colliding (x, x') with non-negligible probability. We construct PPT B to generate claws with non-negligible probability:

1. Query A for a colliding x and x' such that $x \neq x'$.
2. Suppose x and x' are the same for bits $t+1$ to m. The value of $z_{t+1} = f_{x_{t+1}}(\ldots(f_m(\alpha)))$ is the same for x and x' at this point. Denote the point at which x', x first differ as bit t; at this point, the value of z_t diverges. Because z from x and x' eventually collide again (from being diverging), we know there must be a point $1 \leq i \leq t$ such that $f_{x_i}(z_{i+1}) == f_{x'_i}(z_{i+1})$, but for which z'_i, z_i are different. This is our claw, which B returns. We can find this point i, and the corresponding z'_i and z_i by iterating through all i and computing the forward evaluation of H.

From 2.2.a., not such B can exist, so no such A can exist, showing H is a CRHF as desired.

Problem 2.3 For contradiction, suppose there was a PPT A that could compute a pre-image for H with non-negligible probability. We construct a PPT B that solves the discrete log x, on input $\gamma = 4^x \mod N$:

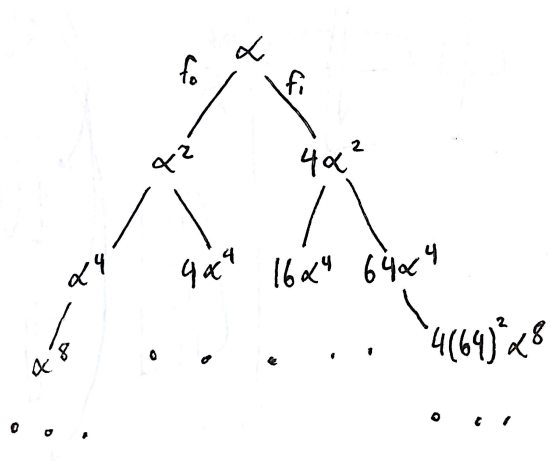

Figure 1: The final hash value with initial seed α. Every distinct path through the tree corresponds to a unique value of input x.

1. Note that on input x, the output of H is of the form $4^x\alpha^{2^{|x|}}$. Randomly sample $\alpha \xleftarrow{\$} \mathsf{QR_N}$, and compute a faux hash output $\beta = H_{\alpha,N}(x) = \gamma\alpha^{2^{|x|}}$. We know $|x|$, which is a fixed m.
2. Feed β to A, and retrieve output x. Output x as the discrete log.

B returns the correct answer by construction: it only returns x such that $4^x \mod N$ equals the challenge γ. This contradicts the discrete log assumption, so no such B can exist, and H must have pre-image computational hardness.

Problem 2.4

Repeated application of f_0 and f_1 with initial seed α results in a tree depicted in Figure 1.

Two colliding inputs correspond to two equal values on the same level of the tree. The values on the same level of the tree are of the form $4^i\alpha^{2^l}$. Finding colliding inputs simplifies to finding a, b such that:

$$4^a\alpha^{l'} = 4^b\alpha^{l'} \mod N$$

$$4^a = 4^b \mod N$$

This implies we are looking for a, b such that $a = b \mod \phi(N)$ but also $a \neq b \mod N$. Because N factors into two p, q that we know (our trapdoor), we are able to compute $\phi(N)$. The bounds on $a, b < N$ means that we can sample a, b that satisfy these conditions in poly-time. Valid a, b (identifying two different nodes along a level in the tree) correspond exactly to a colliding input x, x' (corresponding to the from the root to reach those input).