COMP 8920: Cryptography Winter 2025

Lecture 12 — Feb 13, 2025

Prof. Curtis Bright *Scribe: Mahzabin Chowdhury*

1 Reducing Collision Finding to Preimage and Second Preimage Problems

We can reduce the collision finding problem to the preimage or second preimage problems. This means that if we can solve the preimage or second preimage problems, then we can solve the collision problem.

1.1 Reduction from Collision to Second Preimage

Suppose we have a procedure that can solve the second preimage problem. Then, we choose a random $x \in \mathcal{X}$ and use the assured algorithm for the second preimage to find (x, x') with $x \neq x'$ and $h(x) = h(x')$. Thus, (x, x') is a collision, and we are done.

(If the second preimage algorithm fails, we output failure.)

1.2 Reduction from Collision to Preimage

We assume that $|\mathcal{X}| \geq 2|\mathcal{Y}|$ for $h : \mathcal{X} \to \mathcal{Y}$. Suppose a preimage oracle solves the preimage problem for h and is an (I, Q) algorithm (uses Q hash queries and always succeeds). Then we construct a $(\frac{1}{2}, Q + 1)$ algorithm to find a collision for h.

1. Choose $x \in \mathcal{X}$ uniformly at random.
2. Compute $y = h(x)$.
3. Use the preimage oracle to find $x' \in h^{-1}(y)$.
4. If $x \neq x'$, output (x, x') as the collision. Otherwise, output failure.

We need to find the probability that $x \neq x'$. Define:

$$[x] = \{x_1 \in \mathcal{X} \mid h(x) = h(x_1)\}$$

This partitions $\mathcal{X}$ into equivalence classes, with all members of $[x]$ in the same class hashing to the same value. For every $[x]$, there is a unique $y \in \mathcal{Y}$ with $h^{-1}(y) = [x]$. The number of equivalence classes is $|\mathcal{Y}|$.

The probability of success is:

$$\frac{1}{|\mathcal{X}|}\sum_{x\in\mathcal{X}}\frac{|[x]|-1}{|[x]|}$$

Rewriting in terms of equivalence classes:

$$=\frac{1}{|\mathcal{X}|}\sum_{E\in\mathcal{E}}\sum_{x\in E}\frac{|E|-1}{|E|}$$

$$=\frac{1}{|\mathcal{X}|}\sum_{E\in\mathcal{E}}(|E|-1)=\frac{1}{|\mathcal{X}|}(|\mathcal{X}|-|\mathcal{E}|)$$

$$=1-\frac{|\mathcal{Y}|}{|\mathcal{X}|}\geq 1-\frac{1}{2}=\frac{1}{2}$$

Thus, the reduction succeeds with probability at least $\frac{1}{2}$.

2 Constructing Hash Functions from Compression Functions

Collision resistance is harder to achieve than making the preimage problem difficult. We now define hash functions for arbitrarily long messages based on a single compression function:

$$C:(\mathbb{Z}_2)^{m+t}\to(\mathbb{Z}_2)^m,\quad \text{for } m,t\geq 1$$

We construct an l-bit hash function:

$$h:\bigcup_{i=m+t+1}^{\infty}(\mathbb{Z}_2)^i\to(\mathbb{Z}_2)^l$$

using the following steps:

1. Given input x with $|x|\geq m+t+1$, construct a new string y whose length is a multiple of t. Commonly, $y=x\|\text{pad}(x)$, where pad incorporates the length of x and adds bits to make $y=y_1\|\ldots\|y_r$.
2. Let IV be a public initial value of length m. Set:

$$z_0=\text{IV}$$

$$z_1=C(z_0\|y_1)$$
$$z_2=C(z_1\|y_2)$$
$$\vdots$$
$$z_r=C(z_{r-1}\|y_r)$$

3. Optionally, define:

$$h(x) = z_r \quad \text{or} \quad h(x) = g(z_r), \quad \text{for some function } g : (\mathbb{Z}_2)^m \to (\mathbb{Z}_2)^l$$

In Step 1, the mapping $x \mapsto y$ must be injective. If there exist distinct x, x' with $x \mapsto y$ and $x' \mapsto y$, then (x, x') would automatically form a collision. This ensures $|y| \geq |x|$. However, the compression function C will not be injective, as its output length is smaller than its input length.

3 Merkle-Damgård Construction

Suppose $C : (\mathbb{Z}_2)^{m+t} \to (\mathbb{Z}_2)^m$ is a collision-resistant compression function, where $t \geq 2$ (though $t = 1$ is also possible, as we will see later).

The Merkle-Damgård construction ensures that if C is collision-resistant, then the resulting hash function h is also collision-resistant.