0.1 Building a Variable Length Collision Resistance hash Function

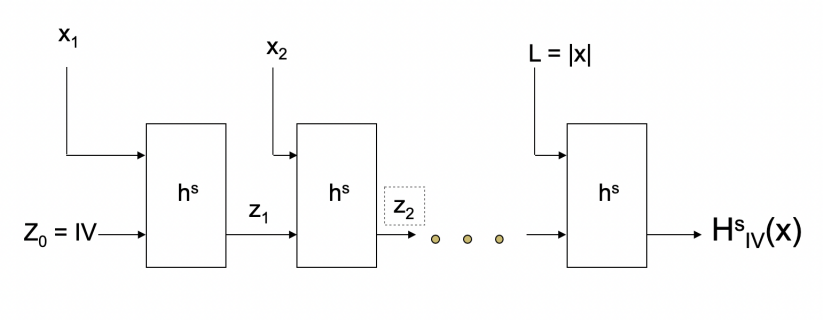

Figure 1: Merkle Damgard Transform

Let (Gen, h) be a fixed-length collision-resistant hash function for inputs of length $2\ell(n)$ and with output length $\ell(n)$. Construct a variable-length hash function (Gen, H) as follows: - Gen: remains unchanged. - H : on input a key s and a string $x \in \{0,1\}^*$ of length $L < 2^{\ell(n)}$, do the following (set $\ell = \ell(n)$ in what follows):

1. Set $B := \lceil \frac{L}{e} \rceil$ (i.e., the number of blocks in x). Pad x with zeroes so its length is a multiple of ℓ. Parse the padded result as the sequence of ℓ-bit blocks $x_1, \ldots, x_B$. Set $x_{B+1} := L$, where L is encoded using exactly ℓ bits.

2. Set $z_0 := 0^R$.

3. For $i = 1, \ldots, B+1$, compute $z_i := h^s(z_{i-1} \| x_i)$.

4. Output z_{B+1}.

This is known as the Merkle-Damgard Transform