Pontificia Universidad Católica de Chile
Departamento de Ciencias de la Computación
IIC3253 – Criptografía y Seguridad Computacional
Profesor Marcelo Arenas - Martín Ugarte (Sección 1)
Primer Semestre del 2025

Ayudantía 06 🐷

Funciones de Hash

Ayudante: Diego Rodríguez Cid – darodriguez6@uc.cl

Repaso de conceptos

Funciones de hash

Una función de hash criptográfica H toma una entrada de longitud arbitraria y produce una salida de tamaño fijo (*digest*). Debe cumplir tres propiedades clave:

1. **resistencia a preimagen:** dado y, es difícil encontrar x tal que $H(x) = y$
2. **resistencia a segunda preimagen:** dado x, es difícil hallar otro $x' \neq x$ con $H(x') = H(x)$
3. **resistencia a colisiones:** es difícil encontrar $x \neq x'$ tales que $H(x) = H(x')$

Construcción de Davies–Meyer

Partimos de un esquema criptográfico

$$(Gen, Enc, Dec) \quad \text{sobre} \quad M = K = C = \{0,1\}^*.$$

Para un parámetro de seguridad n, definimos

$$Gen'(1^n) = n,$$

y la función de compresión de bloque fijo

$$h' : \{0,1\}^{2n} \to \{0,1\}^n \quad \text{por} \quad h'(u\|v) = Enc_u(v) \oplus v,$$

En esta construcción:

- El primer bloque $u \in \{0,1\}^n$ se usa como clave de cifrado.
- El segundo bloque $v \in \{0,1\}^n$ es el texto claro.
- Se cifra v con la clave u y luego se aplica XOR con v mismo.

Construcción de Merkle–Damgård

Merkle–Damgård extiende una función de compresión segura a entradas de longitud variable. Se procede así:

1. *Padding*: al mensaje M se le añade un bit '1', ceros y la longitud de M en bits, para que su tamaño sea múltiplo del bloque.
2. *Iteración*: se inicializa con un valor fijo h_0 y se procesa cada bloque l usando la función de compresión: $H_i = (h')^n(m_i \mid\mid H_{i-1})$.
3. *Salida*: el digest final es $h^s(m) := H_l$.

Problema 1. Concepto

¿Por qué el output de una función de Hash debe depender de **todos** los bits de su input? ¿Qué ocurriría en caso contrario?

Problema 2. Resistencia a colisiones

Sean (Gen_1, h_1) y (Gen_2, h_2) dos funciones de hash criptográficas. Se define (Gen, h) como:

$$h^{s_1,s_2}(x) \ = \ h_1^{s_1}(x) \ \| \ h_2^{s_2}(x).$$

Demuestre que si al menos una de las parejas de funciones de hash criptográficas es resistente a colisiones, entonces

$$(\mathsf{Gen}, h)$$

es también resistente a colisiones.

Problema 3

Sea Enc una función de encriptación que cumpla con todos los requisitos necesarios de seguridad (esquema criptográfico ideal). Demuestre que h, una versión de Davies–Meyer modificada, NO es resistente a colisiones (en contraste con la construcción de Davies–Meyer original, que sí lo es).

$$h(u \,\|\, v) \;=\; \mathrm{Enc}_u(v) \;\oplus\; u.$$

Problema 4

Sea un esquema de hash basado en Merkle–Damgård con bloques de $\ell = 8$ bits y un IV fijo $H_0 = 0$. La función de compresión viene dada por

$$h(H, B) \;=\; \bigl(H \oplus B\bigr) + 1 \pmod{256}.$$

Denotamos por $\mathrm{Hash}(M)$ el valor final tras procesar todos los bloques de un mensaje M.

Diseña un procedimiento que, conociendo únicamente $\mathrm{Hash}(M)$ y un nuevo bloque $X \in \{0, \ldots, 255\}$, calcule directamente

$$\mathrm{Hash}(M \,\|\, X)$$

sin necesidad de conocer el contenido completo de M.