Pontificia Universidad Católica de Chile
Escuela de Ingeniería
Departamento de Ciencia de la Computación
IIC3253 – Lógica para Ciencias de la Computación – 2022-1
Estudiante: Diego Emilio Bustamante Henríquez

Tarea 1

Pregunta 4

a) Se dice que una función de Hash tiene Resistencia a Preimagen si no existe un algoritmo eficiente que, dado $x \in \mathcal{H}$, encuentra $m \in \mathcal{M}$ tal que $h(m) = x$. Se define el juego $Hash\text{–}PM(n)$ como sigue:

i) Verificador genera $s = Gen(1^n)$ y genera aleatoriamente un mensaje m sobre $\mathcal{M}$ con distribución uniforme.

ii) Verificador le entrega a Adversario $x = h^s(m)$, m y s.

iii) Adversario elige mensaje m' tal que $m' \neq m$.

iv) Adversario gana si $h^s(m') = x$, en caso contrario pierde.

Una función de hash (Gen, h) se dice resistente a preimagen si para todo adversario que funciona como un algoritmo aleatorizado en tiempo polinomial, existe una función despreciable $f(n)$ tal que:

$$Pr(Adversario\ gana\ Hash\text{–}PM(n)) \leq f(n)$$

b) Por demostrar, usando contrapositivo, que si (Gen, h) no es resistente a preimagen, entonces (Gen, h) no es resistente a colisiones.

Sea (Gen, h) un hash no resistente a preimagen, entonces existe un adversario que funciona como un algoritmo aleatorizado polinomial tal que su probabilidad de ganar $Hash\text{–}PM(n)$ no es despreciable. Sea $\mathcal{A}$ el algoritmo utilizó el Adversario para ganar $Hash\text{–}PM(n)$ que eficientemente encuentra un m' tal que $h^s(m') = x$ para un s y m dados.

Así, se utilizará $\mathcal{A}$ al jugar $Hash\text{–}Col(n)$ para encontrar una colisión:

i) Verificador genera $s = Gen(1^n)$ y se lo entrega a Adversario.

ii) Adversario elige un m_1 arbitrario como 0^n y calcula $h^s(m_1) = x$. Luego, le entrega a $\mathcal{A}$ los valores m_1, x y s. El algoritmo $\mathcal{A}$ retorna m' y Adversario lo elige como m_2.

iii) Adversario gana porque $h^s(m_1) = x = h^s(m_2)$.

La probabilidad de ganar no es despreciable debido a que $\mathcal{A}$ haga $Hash\text{–}PM(n)$. Además, el algoritmo del Adversario se basa en $\mathcal{A}$ aleatorio y polinomial en n por lo que este es completamente aleatorio y polinomial en n. Por lo tanto, (Gen, h) no es resistente a colisiones.