Práctica 5

Puzles hash

Introducción

Recordemos que una función Hash H se dice *amigable para puzles* si dada cualquier posible salida $y \in \mathbb{B}^n$ de n bits, si elegimos k conteniendo una cadena aleatoria en $\mathbb{B}^n$, no es posible encontrar un $x \in \mathbb{B}^*$ tal que $H(k|x)$ en un tiempo significativamente menor que 2^n.
Con una función Hash amigable para puzles es posible realizar un puzle de búsqueda que consiste en a partir de la función Hash $H()$, un valor id conteniendo una cadena aleatoria, y un conjunto objetivo Y, encontrar un valor x tal que $H(id||x) \in Y$.
En nuestro caso, id será la concatenación de un mensaje junto con un nonce de un solo uso.
Lo usual es que el conjunto Y venga dado por una cota, es decir, $y \in Y$ si y sólo si $y \leq y_0$ para cierto y_0. Concretamente, $y_0 = 2^{n-b}$ para un cierto número de bits b. Esto quiere decir que estar en Y es equivalente a comenzar por b ceros.
La resistencia a preimágenes implica ser amigable para puzles, pero no al revés. Las funciones Hash que hemos visto son resistentes a preimágenes, por tanto pueden ser empleadas para puzles de búsqueda.

Tareas a realizar

Elegid una función hash H con salida de n-bits con $n \geq 256$.

1. Para la función H, realizad, en el lenguaje de programación que queráis, una función que tome como entrada un texto y un número de bits b. Creará un id que concatene una cadena aleatoria de n bits con el texto. Pegará a ese id cadenas aleatorias x de n bits hasta lograr que $H(id||x)$ tenga sus primeros b bits a cero. La salida será el id, la cadena x que haya proporcionado el hash requerido, el valor del hash y el número de intentos llevados a cabo hasta encontrar el valor x apropiado.
2. Calculad una tabla/gráfica que vaya calculando el número de intentos para cada valor de b. Con el objeto de que los resultados eviten ciertos sesgos, para cada tamaño b realizad el experimento 10 veces y calculad la media del número de intentos.
3. Repetid la función anterior con el siguiente cambio: Se toma un primer valor aleatorio x y se va incrementando de 1 en 1 hasta obtener el hash requerido.

4. Calculad una nueva tabla/gráfica similar a la obtenida en el punto 2 pero con la función construida en 3.

NOTA: Debéis entregar un PDF describiendo todas las tareas realizadas, incluyendo en él el código desarrollado. No es necesario enviar el material auxiliar ni los posibles ejecutables producidos, pero debéis conservarlos hasta que salga la evaluación de la práctica por si os son requeridos.