TD 3 Cryptography Engineering 2024

Léo Colisson Palais

Exercice 1: Poorly secure adaptation of Merkle-Damgård

Let $h : \{0,1\}^{n+t} \to \{0,1\}^n$ be a fixed-length compression function. Suppose we forgot a few of the important features of the Merkle-Damgård transformation, and construct a hash function H from h as follows:

- Let x be the input.
- Split x into pieces $y_0, x_1, x_2, \ldots, x_k$, where y_0 is n bits, and each x_i is t bits. The last piece x_k should be padded with zeroes if necessary.
- For $i = 1$ to k, set $y_i := h(y_{i-1} \| x_i)$.
- Output y_k.

Basically, it is similar to the Merkle-Damgård except we lost the IV and we lost the final padding block.

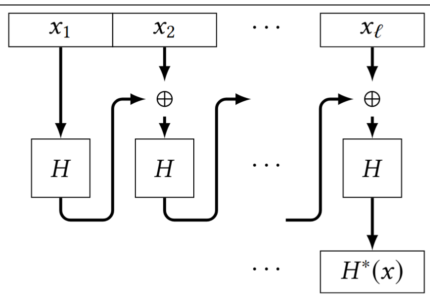
1. Describe an easy way to find two messages that are broken up into the same number of pieces, which have the same hash value under H.
2. Describe an easy way to find two messages that are broken up into different number of pieces, which have the same hash value under H.
   Hint: Pick any string of length $n + 2t$, then find a shorter string that collide with it.

Neither of your collisions above should involve finding a collision in h.

Exercice 2: Attack against "CBC-HASH"

Let H be a collision-resistant hash function with output length n. Let H^* denote iterating H in a manner similar to CBC-MAC:

$\underline{H^*(x_1 \cdots x_\ell):}$
  // *each x_i is n bits*
  $y_0 := 0^n$
  for $i = 1$ to ℓ:
    $y_i := H(x_i \oplus y_{i-1})$
  return y_i



Show that H^* is not collision-resistant. Describe a successful attack.

Exercice 3

1. Suppose a function $H : \{0,1\}^* \to \{0,1\}^n$ has the following property. For all strings x and y of the same length, $H(x \oplus y) = H(x) \oplus H(y)$. Show that H is not collision resistant (describe how to efficiently find a collision in such a function).
2. Show that a bare PRP is not collision resistant. In other words, if F is a secure PRP, then show how to efficiently find collisions in $H(x\|y) := F(x, y)$.

$*$ Exercice 4

Let F be a secure PRF with n-bit inputs, and let H be a collision-resistant (salted) hash function with n-bit outputs. Define the new function $F'((k,s),x) := F(k, H(s,x))$, where we interpret (k,s) to be its key. Prove that F' is a secure PRF with arbitrary-length inputs.