Introduction to cryptology
TD#4

2018-W09

Exercise 1: Insecure Hash-based MACs

In the following, $\mathscr{H}$ is a Merkle-Damgård hash function $\{0,1\}^* \to \{0,1\}^n$ based on a compression function $f : \{0,1\}^b \times \{0,1\}^n \to \{0,1\}^n$: the hash of a message $m = m_0||m_1||\ldots||m_{\ell-1}$ made of ℓ blocks of length b is given by $f(m_{\ell-1}, f(m_{\ell-2}, \ldots f(m_1, f(m_0, \iota))\ldots))$, where ι is a fixed intialization vector.

For the sake of simplicity, we ignore all padding issues in this exercise.

Q. 1: Let m be an arbitrary message. Call $h = \mathscr{H}(m)$ its hash for the function $\mathscr{H}$. Let m' be any b-bit message. Give a simple expression for $\mathscr{H}(m||m')$ in function of h and f.

Q. 2: Let $\mathscr{M} : \{0,1\}^b \times \{0,1\}^* \to \{0,1\}^n$ be a MAC with b-bit keys. Give a generic existential forgery attack for $\mathscr{M}$ that has advantage 1. What is its complexity?

We define a *prefix-MAC* construction that transforms a hash function into a MAC by having $\mathrm{PM}(k, m) = \mathscr{H}(k||m)$.

Q. 3: Give a simple existential forgery attack for prefix-MAC that succeeds with probability 1. What is the complexity of this attack? Do you think that this attack works for *any* hash function?

We now define a *suffix-MAC* construction that transforms a hash function into a MAC by having $\mathrm{SM}(k, m) = \mathscr{H}(m||k)$.

Q. 4: Explain roughly why your attack on prefix-MAC does not work here.

Q. 5: Assume that m and m' form a collision pair for $\mathscr{H}$ (that is, $\mathscr{H}(m) = \mathscr{H}(m')$. What can you say about $\mathrm{SM}(m)$ and $\mathrm{SM}(m')$? Use this observation to give an existential forgery attack for suffix-MAC with advantage 1. What is the complexity of this attack? Under what conditions on b and n is this attack better than the one of **Q. 2**? Does it work for any hash function?

Exercise 2: Birthday attacks for CBC and CTR modes

In the following, $\mathscr{E} : \{0,1\}^\kappa \times \{0,1\}^n \to \{0,1\}^n$ is a block cipher.

Q. 1: Give the definitions of the CBC and CTR modes.

We make the simplifying hypothesis that if $\mathscr{F} \star \to \{0,1\}^n$ is a "random function" with arbitrary domain (that is, all the outputs of $\mathscr{F}$ are uniformly and independently drawn at random from $\{0,1\}^n$), then the expected number of colliding pairs in the finite sequence $\mathscr{F}(x_0)$, $\mathscr{F}(x_1), \ldots, \mathscr{F}(x_{N-1})$ when all x_is are distinct is $\approx N^2/2^n$.

Q. 2: How long should the above sequence be for one to hope to have one collision among its elements with high probability?

Q. 3: Suppose one wishes to distinguish between the above sequence and the keystream produced by $\mathscr{E}$ in counter mode. Using the fact that $\mathscr{E}(k,\cdot)$ is invertible for any k, what can you say about collisions in this keystream? How long should the sequence be for one to distinguish the two cases with high probability?

Q. 4: Given the sequence of ciphertext blocks of a single message encrypted with CBC, what can be deduced about the plaintext blocks if two ciphertext blocks are equal? Does this property still holds if the sequence is made of the concatenation of encryptions of more than one message?

Q. 5: Assuming that one is encrypting "random" messages, how many blocks need to be encrypted for one to observe two equal ciphertext blocks with high probability?

Q. 6: Based on your answers to the above questions, give a recommendation for the maximal number of blocks that should be encrypted with CTR or CBC with a single key. Explain why changing the key may indeed prevent the attacks.