ETH Zürich, D-INFK
Spring 2018

Prof. Ueli Maurer
Christian Badertscher
Fabio Banfi

Cryptography Foundations

Exercise 5

5.1 The Lamport One-Time Signature Scheme

Goal*: We explore how to devise a one-time signature scheme based on one-way functions.*

A *one-time signature scheme* is a digital signature scheme (according to Definition 2.21), for which no feasible adversary can win the signature forgery game for 1 message (according to Definition 2.22)[1] with non-negligible probability. A *one-way function* is a function $f\colon \mathcal{X} \to \mathcal{Y}$ such that an efficient algorithm to compute f is given but no feasible algorithm has non-negligible success probability in the following *inversion game*:

1. $x \in \mathcal{X}$ is chosen uniformly at random and $y \coloneqq f(x) \in \mathcal{Y}$ is given to the algorithm.
2. The algorithm outputs a value $x' \in \mathcal{X}$ and wins the game if $f(x') = y$.

Let $f\colon \mathcal{X} \to \mathcal{Y}$ be a function and let $n > 0$. Further let the message space be $\mathcal{M} \coloneqq \{0,1\}^n$, let the signature set be $\mathcal{S} \coloneqq \mathcal{X}^n$, let the verification-key set be $\mathcal{V} \coloneqq \mathcal{Y}^{2n}$, and let the signing-key set be $\mathcal{Z} \coloneqq \mathcal{X}^{2n}$. Devise a one-time signature scheme that is secure if f is a one-way function. More precisely, show how any adversary for the signature forgery game for 1 message with success probability α can be turned into an algorithm with success probability at least $\frac{\alpha}{2n}$ in the inversion game for f.

5.2 Signature Schemes from Trapdoor One-Way Permutations

Goal*: We learn that the security of TOWP-based signature schemes crucially depends on the strength of the underlying hash-function and that it is possible to prove their security in the* random oracle *model.*

Recall Definition 2.18 of a TOWP which consists of functions $f : \mathcal{X} \times \mathcal{P} \to \mathcal{Y}$ and $g : \mathcal{Y} \times \mathcal{T} \to \mathcal{X}$, as well as a parameter-trapdoor distribution over $\mathcal{P} \times \mathcal{T}$, and assume that $\mathcal{X}$ and $\mathcal{Y}$ are finite sets of equal cardinality. Also, consider a hash-function $h : \mathcal{M} \to \mathcal{Y}$ mapping a message to the codomain of the TOWP. A TOWP-based signature scheme for messages over $\mathcal{M}$ and signatures over $\mathcal{X}$ can be defined (cf. Definition 2.21 and Section 2.8.3 of the lecture notes) by

$$\sigma : \mathcal{M} \times \mathcal{T} \to \mathcal{X} : (m,t) \mapsto g(h(m),t),$$

where the trapdoor t corresponds to the signing-key, and

$$\tau : \mathcal{M} \times \mathcal{X} \times \mathcal{P} \to \{0,1\} : (m,s,p) \mapsto f(s,p) \stackrel{?}{=} h(m),$$

where the parameter p corresponds to the verification-key (and the distribution over $\mathcal{P} \times \mathcal{T}$ remains the same as the one of the underlying TOWP).

Recall that for the specific instantiation of the RSA TOWP we have $\mathcal{X} = \mathcal{Y} = \mathbb{Z}_n^*$ and $\mathcal{P} = \mathcal{T} = \mathbb{N} \times \mathbb{Z}_{\varphi(n)}$, $f(x,(n,e)) \coloneqq [m^e \mod n]$, and $g(y,(n,d)) \coloneqq [y^d \mod n]$. One then obtains the so-called *FDH-RSA signature scheme* by basing the above described scheme on the RSA TOWP and by using an appropriate hash function $h : \mathcal{M} \to \mathbb{Z}_n^*$.

[1] Note that in step 2. of the game, the adversary should be restricted to at most t queries.

a) Show that for the FDH-RSA signature scheme, if $\mathcal{M} = \mathbb{Z}_n^*$ and h is the identity function, it is easy to find a valid pair (m, s) (i.e., an existential forgery), only knowing the public key but no other message-signature pair.

b) Again for the FDH-RSA signature scheme, show that under the same conditions on h as in **a)**, given any message m, it is easy to find a valid forgery s for this m if the adversary has access to a signing oracle. As in the signature forgery game, a forgery for m is only considered valid if m was not asked as a query to the signing oracle.

c) Let us now consider an arbitrary TOWP-based signature scheme and let $h : \mathcal{M} \to \mathcal{Y}$ be modeled as a truly random function—a so-called *random oracle*. This actually means that instead of thinking of h as a function with a certain concrete description, we assume that an additional system $\mathbf{H}$ is available in the random experiments that behaves as follows: on input x to the system $\mathbf{H}$, if x has not been queried before, a value y from the output domain is chosen uniformly at random and the system internally defines the function value $h(x) := y$. Finally, y is output as the response to this query. If x has been queried before to $\mathbf{H}$, the already defined value $y = h(x)$ is returned by the system.

For completeness, we provide below the signature forgery game for t messages of Definition 2.22 for the concrete case of a TOWP-based signature scheme in the random oracle model:

1. The challenger samples the parameter-trapdoor pair (p, t) according to the distribution specified by the underlying TOWP, and outputs as verification key p to the adversary. In the following, let $\mathbf{H}$ be a random oracle modeling a truly random hash function $h : \mathcal{M} \to \mathcal{Y}$, as described above.
2. The adversary can ask at most t queries of the following two kinds:
   - The adversary can obtain signatures to messages, i.e., query a message $m \in \mathcal{M}$ and obtain $s := g(h(m), t)$, where $h(m)$ is obtained by querying the random oracle $\mathbf{H}$ on input m.
   - The adversary can obtain function values by querying the random oracle $\mathbf{H}$ on arbitrary inputs x and receive the result.
3. The adversary chooses a message $\hat{m}$ and a signature $\hat{s}$. He wins the game if $\hat{m}$ was not asked as a signing-query in step 2 and if $h(\hat{m}) = f(\hat{s}, p)$, where $h(\hat{m})$ is again obtained by querying the random oracle $\mathbf{H}$ on input $\hat{m}$.

Note that since the adversary is allowed to ask at most t queries, in the experiment at most $t + 1$ distinct inputs are queried to $\mathbf{H}$. Given a winner W with advantage α in the above forgery game, design a new winner W' (which internally uses W) with advantage least $\frac{\alpha}{t+1}$ in the TOWP inversion game.

Hint: W' receives an image $y = f(x, p)$ to invert, where x is chosen uniformly at random from $\mathcal{X}$. Recall that $f(\cdot, p)$ is a bijective mapping from set $\mathcal{X}$ to set $\mathcal{Y}$ (with $g(\cdot, t)$ corresponding to the inverse mapping for (p, t) in the support of the parameter-trapdoor distribution) and try to "program" the uniformly random function table describing h in a clever way for the replies to W (you can assume that sampling uniformly from sets $\mathcal{X}$ and $\mathcal{Y}$ is easy). Figure out when your reduction is successful.

5.3 The Merkle-Damgård Hash-Function Construction

Goal*: Learn about the Merkle-Damgård construction of a hash function from a compression function.*

Let $f\colon \{0,1\}^{m+n+1} \to \{0,1\}^n$ with $m, n \geq 1$. f is called a *compression function*. We want to use f to construct a hash function $h : \{0,1\}^* \to \{0,1\}^n$. The idea is to split a message $x \in \{0,1\}^*$ into blocks of length m and iteratively feed the concatenation of the current block and the output of the previous stage into the compression function. Then we can use the output of the last stage as the hash of x. Hence we must first pad x to a multiple of m bits. So let

$$x \mapsto \hat{x} : \{0,1\}^* \to \{0,1\}^*$$

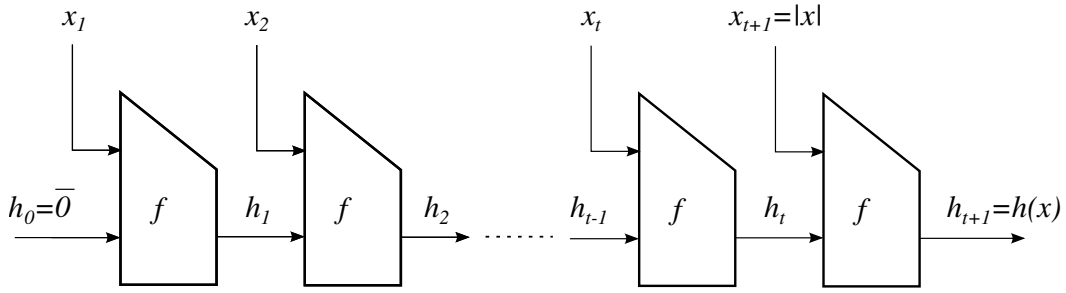

Figure 1: A graphical illustration of the Merkle-Damgård Construction from task 5.3.

be a padding function sucht that $|\hat{x}| = mt$ for some $t \geq 1$ and write $\hat{x} = \hat{x}_1 \parallel \cdots \parallel \hat{x}_t$ with $\hat{x}_k \in \{0,1\}^m$. Then set

$$h_1 = f((0,\ldots,0) \parallel \hat{x}_1)$$

and iteratively for $2 \leq k \leq t$,

$$h_k = f(h_{k-1} \parallel 1 \parallel \hat{x}_k).$$

Now define the hash $h(x)$ of x to be the output h_t of the last compression stage, $h(x) = h_t$. Such constructions of a hash function from a compression function are called Merkle-Damgård constructions.

a) Assume that $m \geq 2$ and consider the straightforward padding

$$\hat{x} = x \parallel (0,\ldots,0)$$

which just appends zero or more 0's to fill the last block. Find a collision of h with this padding. So give two different message $x \neq y$ such that $h(x) = h(y)$.

b) Now let the padding be given by

$$\hat{x} = x \parallel (0,\ldots,0) \parallel \langle d \rangle$$

where we append $d \geq 0$ zeros such that $|x| + d$ is a multiple of m, and the number d of appended zeros written as an m-bit binary string. Show that if you have an algorithm that wins the collision-finding game for h, you can use it in order to win the collision-finding game for f.

Discussion of solutions:

26/27.3.2018 (Task 5.1 and Tasks 5.2a, 5.2b)

9/10.4.2018 (Task 5.2c and Task 5.3)

The Monday and Tuesday sessions of each week cover the same material.

Midterm: April 11, 2018, during the first lecture.

The solutions for this exercise sheet will be made available one week before the midterm.