Introduction to Cryptography – LMAT2450
Final Examination

January 25, 2018

Instructions

1. You can use the slides presented during the class, and all your personal notes. No book or other printed/photocopied material is allowed.
2. The duration of the exam is 3 hours. Answer the questions on ***separate*** sheets of paper.
3. You have the possibility to present your answers to the examiners.

Question 1 Let F be a length-preserving PRF (that is, if $|k| = n$ then $F_k : \{0,1\}^n \mapsto \{0,1\}^n$), and let us remind the following variable-length MAC scheme $\Pi^1 := \langle \mathsf{Gen}^1, \mathsf{Mac}^1, \mathsf{Vrfy}^1 \rangle$, which we discussed during Lecture 4.

- $\mathsf{Gen}^1(1^n)$ returns a uniformly random $k \leftarrow \{0,1\}^n$.
- Mac^1: on input $k \in \{0,1\}^n$ and m of length $l < 2^{\frac{n}{4}}$.
  - Parse m into blocks $m_1, \ldots, m_d$ of length $\frac{n}{4}$ each (pad with 0's if necessary).
  - Choose a random $r \leftarrow \{0,1\}^{\frac{n}{4}}$.
  - Compute $t_i := F_k(r||l||i||m_i)$ for $1 \leq i \leq d$, with $|r| = |l| = |i| = \frac{n}{4}$.
  - Output $t := \langle r, t_1, \ldots, t_d \rangle$.
- Vrfy^1: on input $k, m, t = \langle r, t_1, \ldots, t_{d'} \rangle$,
  - Parse m into blocks $m_1, \ldots, m_d$ of length $\frac{n}{4}$ each.
  - Output 1 iff $d = d'$ and, $\forall \quad 1 \leq i \leq d$, $t_i = F_k(r||l||i||m_i)$.

Let us now consider the following variant of Π^1. Let $\Pi^2 := \langle \mathsf{Gen}^2, \mathsf{Mac}^2, \mathsf{Vrfy}^2 \rangle$, of which we define the first two components as follows.

- $\mathsf{Gen}^2(1^n)$: returns a uniformly random $k \leftarrow \{0,1\}^n$ and $k' \leftarrow \{0,1\}^n$.
- Mac^2: on input $k, k' \in \{0,1\}^n$ and m of length $l < 2^{\frac{n}{4}}$.
  - Run $\mathsf{Mac}^1_k(m)$ and obtain $t = \langle r, t_1, \ldots, t_d \rangle$
  - Return the tag $t, F_{k'}(r)$.
- Vrfy^2: on input $k, k' \in \{0,1\}^n$, m of length $l < 2^{\frac{n}{4}}$ and $t = \langle r, t_1, \ldots, t_d, fr \rangle$, output 1 iff $\mathsf{Vrfy}^1_k(m, \langle r, t_1, \ldots, t_d \rangle) = 1$ and $fr = F_{k'}(r)$.

Prove that, if Π^1 is EUF-CMA secure, then so is Π^2.

Solution indication: Suppose that A can break Π^2 with probability ϵ (as a function of the security parameter). We build B as an attacker against Π^1 as follows:

1. B starts an instance of A and selects a random $k' \leftarrow \{0,1\}^n$.
2. When A asks for a MAC of a message m, B asks the Π^1 challenger for a MAC t on the same m and returns the corresponding tag to A, to which it concatenates $F_{k'}(r)$, where r is the first component of t.
3. When A produces a forgery (m^*, t^*), B produces a forgery (m^*, τ^*), in which τ^* is t from which the last component is removed.

We observe that, if A is PPT, then so is B and that, if A produces a valid forgery, then so does B. As a result, and given that Π^1 is secure, the probability ϵ must be a negligible function, which shows that Π^2 is also secure.

Question 2 Suppose that we need to hash a message m of 1 gigabyte ($\approx 2^{33}$ bits) and that we have $n = 16$ processors available, which can all process m in parallel. To this purpose we have a hash function H^s, which is a Merkle-Damgård-based hash function with an output of 256 bits (think of SHA256). We propose to define a new hash function PH^s (for "parallel hash") which proceeds as follows:
a. Split m into n pieces $m_1, \ldots, m_n$ of equal length.
b. Compute the hashes $h_i = H^s(m_i)$, for $i \in 1 \ldots n$.
c. Return $PH^s(m) := H^s(h_1 \| \cdots \| h_n)$.

We ask you to:

1. Explain the benefits of PH^s over H^s in terms of efficiency for hashing the message m above. In your evaluation, you can ignore factors that are less than 1%.

2. Show that PH^s is collision resistant if H^s is collision resistant. You can assume that they share the same key generation algorithm.

Solution indication:
1. PH can run $\approx n$ times faster than H. Indeed, if we assign the computation of h_i to processor i, we see that all the h_i will be available in a time that corresponds to the hashing of a message of size $|m|/n$, that is, 62.5 Mo. The final hash is computed on a message of 512 bytes, which is negligible in front of the first part of the computation.
2. Let us assume that A can produce two distinct messages m and m' such that $PH^s(m_1 \| \cdots \| m_n) = PH^s(m'_1 \| \cdots \| m'_n)$, and assume that $m_i \neq m'_i$. Then we either have that $h_i = h'_i$, which means that $H^s(m_i) = H^s(m'_i)$, or we have that $h_i \neq h'_i$, which means that $H^s(h_1 \| \cdots \| h_i \| \cdots \| h_n) = H^s(h'_1 \| \cdots \| h'_i \| \cdots \| h'_n)$. In both cases we have a collision on H^s.

Question 3 Let $\mathbb{G}$ be a finite cyclic group of prime order q and $(pk, sk) \in \{\mathbb{G}, \mathbb{Z}_q\}$ be a public/secret key pair for the ElGamal encryption scheme. Let x be an element of $\mathbb{G}$, chosen at random.

How many different elements $y \in \mathbb{G}$ are there for which $\langle x, y \rangle$ is the encryption of some plaintext?

Solution indication: There are q such values: any group element y will do. Indeed, x defines a unique r such that $x = g^r$, which in turn defines $y' = pk^r$. As a result, (x, y) is an encryption of the message y/y'.

Question 4 Let $\Pi = \langle \mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec} \rangle$ be an authenticated encryption scheme such that Enc encrypts messages of n bits. Do the following systems provide authenticated encryption? For those that do, briefly explain why. For those that do not, present an attack that breaks one of the security properties of an authenticated encryption scheme.

1. $\Pi' = \langle \mathsf{Gen}, \mathsf{Enc}', \mathsf{Dec}' \rangle$ with $\mathsf{Enc}'_k(m) = (\mathsf{Enc}_k(m), \mathsf{Enc}_k(m \oplus (0^{n-1} \| 1)))$ and $\mathsf{Dec}'_k(c_1, c_2) = \mathsf{Dec}_k(c_1)$ if $\mathsf{Dec}_k(c_1) \oplus \mathsf{Dec}_k(c_2) = 0^{n-1} \| 1$ and $\perp$ otherwise.

   Solution indication: It is not an authenticated encryption scheme:

- In order to break the CCA security, run a test query in which you ask for an encryption of messages m_0 or m_1 that differ by a bit that is not their LSB, and obtain an encryption (c_1, c_2) of m_b. Then ask for the decryption of (c_2, c_1). This is a valid ciphertext, different of the previous one, and its decryption is $m_b \oplus (0^{n-1}\|1)$, which makes it immediate to get b.
- In order to break the unforgeability, ask for an encryption of m, get (c_1, c_2), and output the ciphertext (c_2, c_1). This is a valid ciphertext that decrypts to $m \oplus (0^{n-1}\|1)$.

2. $\Pi' = \langle \mathsf{Gen}, \mathsf{Enc}', \mathsf{Dec}' \rangle$ with $\mathsf{Enc}'_k(m) = (\mathsf{Enc}_k(m), \mathsf{Mac}_k(m))$ and $\mathsf{Dec}'_k(c_1, c_2) = \mathsf{Dec}_k(c_1)$ if $\mathsf{Vrfy}_k(\mathsf{Dec}_k(c_1), c_2) = 1$ and $\perp$ otherwise. Here, Mac and Vrfy are deterministic algorithms that are part of a secure MAC scheme that is compatible with Gen.

   Solution indication:

   - In order to break the CCA security, ask for the encryption of two different messages m_0 and m_1, then make a test query on (m_0, m_1). Since Mac is a deterministic fonction, it is enough to compare the c_2 components of the corresponding ciphertext with those received before to determine which message has been encrypted.
   - Breaking the unforgeability of this scheme would immediately result in breaking the unforgeability of the MAC and of the original authenticated encyption scheme.

Question 5 Consider the following interactive protocol between a prover P and a verifier V. The common input of P and V is a public verification key vk for a EUF-CMA secure signature scheme, and P is expected to know the corresponding signing key sk.

In order to obtain a proof of the knowledge of sk, V selects a random message m and sends it to P. P answers with $\sigma = \mathsf{Sign}_{sk}(m)$, and V accepts the proof if $\mathsf{Vrfy}_{vk}(m, \sigma) = 1$.

1. Is this protocol complete?

   Solution indication: Yes: Given that P knows sk, he can always compute the requested signature, and V will accept if the signature scheme is correct.

2. Is this protocol zero-knowledge?

   Solution indication: No. Any valid transcript produced by a simulator for this protocol is a forgery for the signature scheme.