Introduction to Modern Cryptography
Exercise Sheet #3

University of Amsterdam, Master of Logic, 2012
Lecturer: Christian Schaffner
TA: Maria Velema

13 November 2012
(to be handed in by Wednesday, 21 November 2012, 9:00)

1. Show that one has to be very careful with modifications of CBC-MAC, small modifications can be disastrous. Exercises 4.9 and 4.8 of [KL].

2. CCA-Security: Exercise 3.22 from [KL].

3. Insecurity of Encrypt-and-Authenticate: Exercise 4.19 of [KL].

4. **Different security goals should always use independent keys!** We derive an example what can go wrong if the same key is used in the Encrypt-then-Authenticate approach (which yields CCA-security if independent keys are used!).
   Let F be a strong pseudorandom permutation according to Definition 3.28 in [KL]. Let the key $k \leftarrow \{0,1\}^n$ be picked uniformly at random by Gen. Define $\mathsf{Enc}_k(m) = F_k(m\|r)$ for $m \in \{0,1\}^{n/2}$ and a random $r \leftarrow \{0,1\}^{n/2}$, and define $\mathsf{Mac}_k(c) = F_k^{-1}(c)$.
   (a) Define the corresponding decryption function $\mathsf{Dec}_k(\cdot)$ and prove that this encryption scheme $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ is CPA-secure.
   (b) Prove that the authentication code is a secure MAC.
   (c) Conclude that the combination of the two schemes in the Encrypt-then-Authenticate approach *using the same key* k is completely insecure.

5. **One-time MAC:** Let us consider the following message authentication code:

   $\mathsf{Gen}(1^n)$**:** Let $p = \text{NextPrime}(2^n)$; pick $a \leftarrow \mathbb{Z}_p^*$, $b \leftarrow \mathbb{Z}_p$ (so $a \in \{1, 2, \ldots, p-1\}$, $b \in \{0, 1, 2, \ldots, p-1\}$.) Output p, a, b.

   $\mathsf{Mac}_{p,a,b}(m)$**:** Output $[am + b \mod p]$.

   $\mathsf{Vrfy}_{p,a,b}(m, t)$**:** Output 1 if $\mathsf{Mac}_{p,a,b}(m) = t$, output 0 otherwise.

   Note that this MAC handles messages $m \in \mathbb{Z}_p$ (only).

   Show that the above MAC is secure against any adversary making at most one query (see Definition 4.2 in [KL]). In particular, show that this MAC is secure even if the adversary is *not* restricted to run in polynomial time.

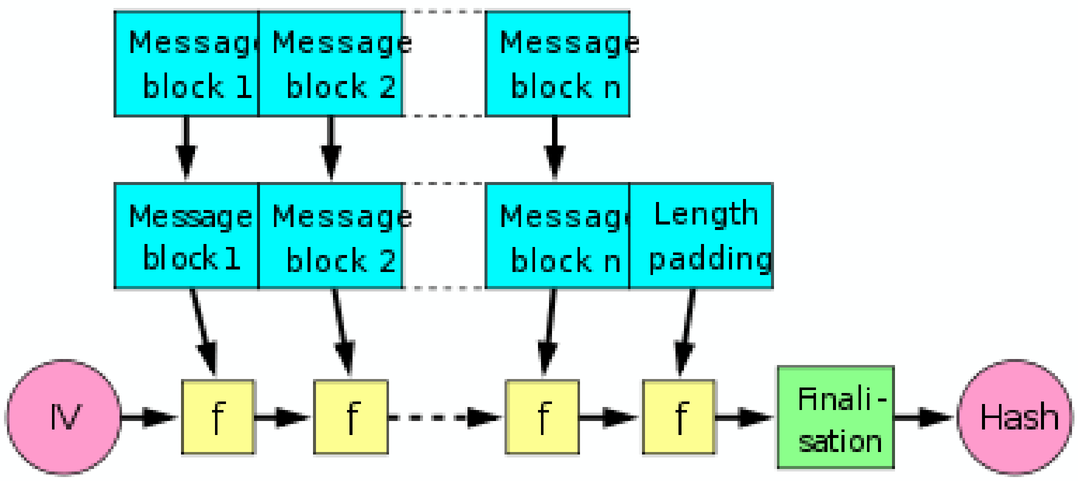
more on the back side

6. **Pre-image resistance of hash functions:** Exercise 4.10 of [KL].

7. **Double-hash:** Exercise 4.12 in [KL]. **Hint:** Yes.

8. **Another exercise in formal reduction proofs:** Exercise 4.13 in [KL]. **Tip:** You are *not* required to reprove statements that are already derived in the proof of Theorem 4.14 in the book. You *are* asked to write down (as precisely as you can) the formal reduction, for example, specify exactly what the adversary against h does.

9. **A dangerous idea:** Exercise 4.17 of [KL]. **Hint:** Use $\mathsf{Mac}_k(m)$ to construct a valid tag on a particular longer message $\mathsf{Mac}_k(m')$. Note that Merkle-Damgård appends the length of the message to the end of the (padded) input string, you'll need to figure out how to get around that.



The Merkle-Damgård construction
Image credit: `David Göthberg, wikimedia.org` .