Introduction to Cryptography and Software Security
Problem Set 2: MACs and Hash Functions

Due date: 2/5/2019

Instructions.

- You are allowed to rely on any statement that we proved in class, unless you are explicitly asked to prove it, and as long as you state it clearly and accurately.
- Justify your answers with formal proofs.
- **On-line submissions only! Hard copies will not be accepted.** Please make sure that scanned submissions are readable (unreadable submissions will not be accepted).

Problem 1 (20%). Let F be a pseudorandom function such that for any $n \in \mathbb{N}$ and $k \in \{0,1\}^n$ it holds that $F_k : \{0,1\}^n \to \{0,1\}^n$. Show that each of the following MAC schemes is insecure. In each case, the scheme's key-generation algorithm, Gen, outputs a uniform key $k \in \{0,1\}^n$, and we let $\langle i \rangle$ denote an $n/2$-bit encoding of the integer i.

1. To authenticate a message $m = m_1 || \cdots || m_d$, where $m_i \in \{0,1\}^n$ for every $i \in \{1,\dots,d\}$, output $t = F_k(m_1) \oplus \cdots \oplus F_k(m_d)$.
2. To authenticate a message $m = m_1 || \cdots || m_d$, where $m_i \in \{0,1\}^{n/2}$ for every $i \in \{1,\dots,d\}$, output $t = F_k(\langle 1 \rangle || m_1) || \cdots || F_k(\langle d \rangle || m_d)$.
3. To authenticate a message $m \in \{0,1\}^n$, sample a uniform $r \leftarrow \{0,1\}^n$, and output (r,t), where $t = F_k(r) \oplus F_k(m)$.

Problem 2 (15%). Let $\Pi_1 = (\mathsf{Gen}_1, \mathsf{Mac}_1, \mathsf{Vrfy}_1)$ and $\Pi_2 = (\mathsf{Gen}_2, \mathsf{Mac}_2, \mathsf{Vrfy}_2)$ be two MAC schemes, for which it is known that at least one is secure (both schemes are guaranteed to be *correct*). The problem is that you do not know which one is secure, and which one may not be.
Show how to combine them into one MAC scheme that is guaranteed to be secure as long as at least one of them is secure. Prove the security of your proposal.

Problem 3 (15%). Let $\mathcal{H} = (\mathsf{Gen}, H)$ be a collision-resistant hash family such that for any security parameter $n \in \mathbb{N}$ and for any key $s \in \{0,1\}^n$ that is produced by $\mathsf{Gen}(1^n)$ it holds that $H_s : \{0,1\}^* \to \{0,1\}^n$.

1. For every $s \in \{0,1\}^n$, $x \in \{0,1\}^*$ and $b \in \{0,1\}$, let $\widehat{H}_s(x||b) = H_s(x)||b$. Is $\widehat{\mathcal{H}} = \left(\mathsf{Gen}, \widehat{H}\right)$ necessarily a collision-resistant hash family? Prove your answer.
2. For every $s \in \{0,1\}^n$ and $x \in \{0,1\}^*$ let $W_s(x)$ denote the leftmost $n-1$ bits of $H_s(x)$. Is $\mathcal{W} = (\mathsf{Gen}, W)$ necessarily a collision-resistant hash family? Prove your answer.

Problem 4 (20%). Let F be a pseudorandom function such that for any $n \in \mathbb{N}$ and key $k \in \{0,1\}^n$ it holds that $F_k : \{0,1\}^n \rightarrow \{0,1\}^n$. Let $\Pi = (\mathsf{Gen}, \mathsf{Mac}, \mathsf{Vrfy})$ be the following MAC scheme:

- The algorithm Gen on input 1^n uniformly and independently samples $k_1, k_2 \leftarrow \{0,1\}^n$, and then outputs $k = (k_1, k_2)$.
- The algorithm Mac on input a key $k = (k_1, k_2) \in \{0,1\}^{2n}$ and a message $m \in \{0,1\}^n$ outputs $t = F_{k_1}(m) || F_{k_2}(F_{k_1}(m))$.
- The algorithm Vrfy on input a key $k = (k_1, k_2) \in \{0,1\}^{2n}$, a message $m \in \{0,1\}^n$ and a tag $t \in \{0,1\}^{2n}$, outputs 1 if $t = \mathsf{Mac}_{k_1,k_2}(m)$ and outputs 0 otherwise.

Is Π necessarily a secure MAC scheme? Prove your answer.

Problem 5 (15%). We say that $\Pi = (\mathsf{Gen}, \mathsf{Mac}, \mathsf{Vrfy})$ is a *leftmost-bit leakage-resilient* MAC scheme if for any probabilistic polynomial-time algorithm A and for any polynomial $p(\cdot)$ it holds that

$$\Pr\left[\mathsf{MacForgeBitLeakage}_{\Pi,A}(n) = 1\right] \leq \frac{1}{p(n)}$$

for all sufficiently large $n \in \mathbb{N}$, where the experiment $\mathsf{MacForgeBitLeakage}_{\Pi,A}(n)$ is obtained from the experiment $\mathsf{MacForge}_{\Pi,A}(n)$ as follows: Whereas in the experiment $\mathsf{MacForge}_{\Pi,A}(n)$ the input of the algorithm A is just the security parameter 1^n, in the experiment $\mathsf{MacForgeBitLeakage}_{\Pi,A}(n)$ the input of the algorithm A is the security parameter 1^n and the leftmost bit k_1 of the key k that is sampled at the beginning of the experiment.

1. Prove that any secure MAC scheme is also a leftmost-bit leakage-resilient MAC scheme.

2. Extend the notion of a leftmost-bit leakage-resilient MAC scheme to that of a *left-half* (resp. *right-half*) leakage-resilient MAC scheme, where the adversary is given as input the left half k_L (resp. right half k_R) of the key.

3. Assuming the existence of any secure MAC scheme Π, construct a secure MAC scheme Π' that is not left-half leakage resilient.

Problem 6 (15%). Let $\Pi = (\mathsf{Gen}, \mathsf{Mac}, \mathsf{Vrfy})$ be a secure MAC scheme that uses n-bit keys for n-bit messages for any $n \in \mathbb{N}$, and recall the notion of a *left-half* (resp. *right-half*) *leakage-resilient* MAC scheme that you have defined in Problem 5. Consider the following scheme $\Pi' = \left(\mathsf{Gen}', \mathsf{Mac}', \mathsf{Vrfy}'\right)$:

- The algorithm Gen' on input 1^n samples $k_L \leftarrow \mathsf{Gen}(1^n)$ and $k_R \leftarrow \{0,1\}^n$ independently, and outputs the key (k_L, k_R).
- The algorithm Mac' on input a key (k_L, k_R) and a message $m \in \{0,1\}^n$ outputs $t = \mathsf{Mac}_{k_L}(m \oplus k_R)$.
- The algorithm Vrfy' on input a key (k_L, k_R), a message m and a tag t, outputs 1 if $\mathsf{Vrfy}_{k_L}(m \oplus k_R, t) = 1$ and outputs 0 otherwise.

1. Prove that Π' is a right-half leakage-resilient MAC scheme.

2. Prove that Π' is not necessarily a left-half leakage-resilient MAC scheme.
   [That is, assuming the existence of any secure MAC scheme, construct a secure MAC scheme Π for which Π' is not left-half leakage resilient.]