دانشکده‌ی علوم ریاضی

مقدمه‌ای بر رمزنگاری	تحویل اصلی: ۱۵ آذر ۱۳۹۹

تمرین شماره ۳

مدرّس: دکتر شهرام خزائی	تحویل نهایی: ۲۲ آذر ۱۳۹۹

- Upload your answers on courseware with the name: StudentNumber.pdf
- Upload a PDF file. Image and zip formats are not accepted.
- Similar answers will not be graded.
- NO answers will be accepted via e-mail.
- You can't upload files bigger than 2 Mb, so you'd better type.
- Deadline time is always at 23:55 and will not be extended.
- You should submit your answers before soft deadline.
- You will lose 5 percent for each day delay if you submit within a week after soft deadline.
- You can not submit any time after hard deadline.
- All problem sets include 100 points.
- For any question contact Mahtab Alghassi via `mahtab.alghassi@gmail.com`.

Problem 1

Message Authentication Code:

a. (5 Points) Let (S, V) be a secure MAC defined over (K, M, T) where $T = \{0,1\}^n$. Define a new MAC (S', V') as follows: $S'(k, m)$ is the same as $S(k, m)$, except that the last eight bits of the output tag t are truncated. That is, S' outputs tags in $\{0,1\}^{n-8}$. Algorithm $V'(k, m, t')$ accepts if there is some $b \in \{0,1\}^8$ for which $V(k, m, t'||b)$ accepts. Is (S', V') a secure MAC? Give an attack or argue security.

b. (10 Points) Prove that the following modification of basic CBC-MAC gives a secure MAC for arbitrary-length messages (for simplicity, assume all messages have length a multiple of the block length).
$MAC_k(m)$ first computes $k_L = F_k(L)$, where L is the length of m. The tag is then computed using basic CBC-MAC with key k_L. Verification is done in the natural way.

c. (5 points) Recall that in CBC-MAC the IV is fixed. Suppose we chose a random IV for every message being signed and include the IV in the MAC, i.e. $S(k, m) := (r, CBCr(k, m))$, where $CBC_r(k, m)$ refers to the raw CBC function using r as the IV. Describe an existential forgery on the resulting MAC.

Problem 2

(20 Points) **Multicast MACs.** Suppose user A wants to broadcast a message to n recipients $B_1, ..., B_n$. Privacy is not important but integrity is. In other words, each of $B_1, ..., B_n$ should be assured that the message he is receiving were sent by A. User A decides to use a MAC.

a. (5 point) Suppose user A and $B_1, ..., B_n$ all share a secret key k. User A computes the MAC tag for every message she sends using k, and every user B_i verifies the tag using k. Using at most two sentences explain why this scheme is insecure, namely, show that user B_1 is not assured that messages he is receiving are from A.

b. (5 point) Suppose user A has a set $S = \{k_1, ..., k_L\}$ of L secret keys. Each user B_i has some subset $S_i \subseteq S$ of the keys. When A transmits a message she appends L MAC tags to it by MACing the message with each of her L keys. When user B_i receives a message he accepts it as valid only if all tags corresponding to keys in S_i are valid. Let us assume that the users $B_1, ..., B_n$ do not collude with each other. What property should the sets $S_1, ..., S_n$ satisfy so that the attack from part (a) does not apply?

c. (5 point) Show that when $n = 10$ (i.e. ten recipients) it suffices to take $L = 5$ in part (b). Describe the sets $S_1, ..., S_{10} \subseteq k_1, ..., k_5$ you would use.

d. (5 pint) Show that the scheme from part (c) is completely insecure if two users are allowed to collude.

Problem 3

a. (10 points) . Let $H : M \to T$ be a collision resistant hash where $M = \{0,1\}^L$ and $T = \{0,1\}^n$. For each of the following, explain why it is collision resistant, or describe an efficient way to find collisions:

- for a fixed $0^L \neq \Delta \in M$ define $H_1(m) := H(m) \oplus H(m \oplus \Delta)$.
- for a fixed $0^n \neq \Delta \in T$ define $H_2(m) := H(m) \oplus \Delta$.

b. (5 points) Suppose $H : X \to Y$ is a collision resistant hash function, where $Y \subseteq X$. Is the function $H^2(x) = H(H(x))$ collision resistant? Give an attack on H^2 , or prove that H^2 is collision resistant by showing that an attack on H^2 gives an attack H.

c. (5 points) Let $H : M \to \{0,1\}^{128}$ be a collision resistant hash function known to the adversary. Does the function $f(k, m) = H(m) \oplus k$ give a secure MAC? If so explain why. If not, describe an attack

Problem 4

(20 Points) prove or disaprove:

a. (5 points) if (Gen, h) is preimage resistant, then so is the hash function (Gen, H) obtained by applying the Merkle–Damgard transform to (Gen, h).

b. (5 points) if (Gen, h) is second preimage resistant, then so is the hash function (Gen, H) obtained by applying the Merkle–Damgard transform to (Gen, h).

c*. (10 points, **optional**) Show how to find a collision in the Merkle tree construction if t is not fixed. Specifically, show how to find two sets of inputs $x_1, ..., x_t$ and $x'_1, ..., x'_{2t}$ such that $\mathcal{MT}_t(x_1, ..., x_t) = \mathcal{MT}_{2t}(x'_1, ..., x'_{2t})$[1].

[1] 5.6.2 of Katz-Lindell book

Problem 5

Carter-Wegman MAC. An important family of MACs is called Carter-Wegman MACs.

a. (2 Points) A one-time MAC is a MAC that is secure as long as the MAC key is only used to authenticate at most one message. Write out the security definition for a one-time MAC by suitably adapting the security definition for a (many time) MAC.

b. (3 Points) Let p be a prime so that $1/p$ is negligible. Here is a simple candidate one-time MAC with message space $\mathcal{M} := (\mathbb{Z}_p)^{\leq L}$, for some $L \leq p$, and key space $\mathcal{K} := \mathbb{Z}_p^2$:

$$S((k,k'), m = (m_1, ..., m_n)) = \{output \leftarrow k' + \sum_{i=1}^{n} m_i.k^i \in \mathbb{Z}_p\}$$

Verification $V((k,k'), m, t)$ works by checking that $S((k,k'), m) = t$. Show that this MAC is insecure as a one-time MAC.

Hint: use the fact that the MAC can be used to sign messages of varying lengths.

c. (5 Points) We can fix the problem from part (b) by defining

$$S'((k,k'), m = (m_1, ..., m_n)) = \{output \leftarrow k' + k^{n+1} + \sum_{i=1}^{n} m_i.k^i \in \mathbb{Z}_p\}$$

Verification V' works as before by recalculating $S'((k,k'), m)$. This MAC can be shown to be one-time secure whenever L/p is negligible. Instead, show that this MAC is not two-time secure.

Note: this one-time MAC is blindingly fast, requiring only one addition and one multiplication per message b.

d*. (10 Points, **optional**) We can convert (S', V') into a many-time MAC using a secure PRF. Let F be a secure PRF defined $(\mathcal{K}, \mathbb{Z}_p, \mathbb{Z}_p)$. Define the Carter-Wegman MAC as

$$S''((k,k'), m) := \{r \leftarrow \mathbb{Z}_p, t \leftarrow F(k', r) + k^{n+1} + \sum_{i=1}^{n} m_i.k^i, output(r,t)\}$$

.

Note that the PRF (typically AES) is only applied to the single block r. As a result, this MAC can be faster than CBC-MAC. Explain how the verification algorithm V'' works.