University of Maryland
CMSC456 — Introduction to Cryptography
Professor Jonathan Katz

Problem Set 3

Due at the *beginning* of class on Oct. 21

1. Let $F : \{0,1\}^k \times \{0,1\}^k \rightarrow \{0,1\}^k$ be a (t,ϵ)-secure PRF for t very large and ϵ very small. Consider the following message authentication code which is defined for messages of length $2k-2$ (so the adversary can only ask for the MAC of messages of this length, and must try to forge a MAC for a new message of this length): The key $sk \in \{0,1\}^k$ is chosen randomly. To authenticate a message M, the sender parses M as $m_0||m_1$ where $|m_0| = |m_1| = k-1$. Authentication tags are computed as follows:

$$\mathsf{Mac}(m_0 \circ m_1) = F_{sk}(0 \circ m_0)||F_{sk}(1 \circ m_1),$$

   Show that this scheme is insecure.

2. Let $F : \{0,1\}^k \times \{0,1\}^m \rightarrow \{0,1\}^m$ be a (t,ϵ)-secure PRF for t very large and ϵ very small. Consider the following MAC: The sender and receiver share a random $sk \in \{0,1\}^k$ and fix ℓ which is assumed to be known to the adversary. Let $\langle i \rangle$ denote the ℓ-bit representation of integer i. To authenticate message M, the sender parses M as a sequence of $(m-\ell)$-bit blocks $M_1, \ldots, M_n$ (assume that message lengths are always a multiple of $(m-\ell)$, and that $n < 2^\ell$), chooses a random $r \in \{0,1\}^m$ and computes:

$$T = F_{sk}(r) \oplus F_{sk}(\langle 1 \rangle \circ M_1) \oplus \cdots \oplus F_{sk}(\langle t \rangle \circ M_n).$$

   The sender sends both T and r as the message authentication code for M.

   Suggest an attack on the above scheme which is better than brute-force search for the key sk, and describe the complexity of your attack and its probability of success in forging a valid tag on a new message. (I am aware of at least two different attacks, but any correct attack you give which is better than brute-force key search is fine.)

3. Show that the CBC-MAC discussed in class is *insecure* when it is used to authenticate variable-length messages (note: this means that the adversary can request tags for messages of any length, and can also output a valid tag for a message of any length). You should assume that all messages, however, have length which is a multiple of the block-length of the block cipher being used.

4. Assume we want to achieve both secrecy and integrity in the private-key setting. Let $(\mathcal{E}, \mathcal{D})$ denote a private-key encryption scheme which is secure against chosen-plaintext attacks, and let $(\mathsf{Mac}, \mathsf{Vrfy})$ denote a secure message authentication code. Assume the sender and receiver have shared random keys s_1, s_2.

   (a) One approach is to separately encrypt and authenticate the message. Thus, to send M the sender would compute $C \leftarrow \mathcal{E}_{s_1}(M)$ and $T \leftarrow \mathsf{Mac}_{s_2}(M)$, and then send $C||T$ to the receiver. Show that, in general, this does not provide *secrecy*. (Hint: construct a secure MAC which leaks information about $M\ldots$)

(b) Another approach is to encrypt the message and then authenticate the resulting ciphertext. Thus, to send M the sender would compute $C \leftarrow \mathcal{E}_{s_1}(M)$ followed by $T \leftarrow \mathsf{Mac}_{s_2}(C)$, and then send $C||T$ to the receiver. Do you think this approach provides both secrecy and integrity? (You do not need to provide a complete proof of security/insecurity in each case, but you may want to sketch the proofs for yourself to make sure you get the correct answer!)

(c) A variant of the above scheme decreases the length of the shared key by using the *same* key s for both encryption and authentication (thus, the sender would compute $C \leftarrow \mathcal{E}_s(M)$ followed by $T \leftarrow \mathsf{Mac}_s(C)$, and then send $C||T$ to the receiver). Show that, in general, this definitely does *not* achieve either secrecy or integrity. (Hint: more clever solutions may be possible, but one option is to view the key s as being made up of two parts with one used for encryption and the other used for authentication. But now the encryption/authentication schemes may leak side information...)