650.445/600.454: Practical Cryptographic Systems November 12, 2018

Written HW 3

Instructor: Matthew Green *Due: 11:59 pm November 26, 2018*

Name: _____

The assignment must be completed individually. You are permitted to use the Internet and any printed references. In particular, you may find the Handbook of Applied cryptography (available online) useful for several of these questions.

Please submit the completed assignment via Blackboard.

Problem 1: Short answers. (You don't need to justify your answer, but you might want to for partial credit.)

1. Explain (from the Rogaway) paper what an AEAD scheme is. What are its advantages?

2. The Rabin encryption system is like RSA, except that it uses public exponent $e = 2$ and a different decryption process. Explain why the Rabin scheme is thought to be provably secure based on factoring. Intuitively explain the reasoning from the Koblitz paper.

3. Why is the OAEP encryption scheme's proof thought to be flawed?

4. Explain what the *remote attestation* function of Intel SGX is, and how the Foreshadow attack breaks it.

Problem 2: Longer answers

1. **Ideal cipher model and CBC-MAC.** The "simple" CBC-MAC scheme is a MAC based on CBC-mode encryption. Unlike CBC encryption, this mode uses a fixed Initialization Vector (*e.g.,* the 0 IV), and outputs the *final* block of CBC ciphertext as the "MAC". Let (E, D) be a block cipher with block size ℓ bits. Given a message $M = (M_1, \ldots, M_N)$, where each M_i is the length of a single cipher block (ℓ bits) and a randomly-generated κ-bit key k, the MAC algorithm $\mathsf{CBC\text{-}MAC}(k, M)$ works as follows:

   (a) Set $C_0 \leftarrow 0^\ell$.
   (b) For $i = 1$ to N: set $C_i \leftarrow E_k(C_{i-1} \oplus M_i)$.
   (c) Output C_N.

   Part 1: Proving security for fixed-length messages. Assume, for the moment, that this MAC scheme will be used to MAC only *fixed* size messages. That is, every message passed to the $\mathsf{CBC\text{-}MAC}$ algorithm is of identical length N blocks long, for

some modest N. Sketch a security argument *in the ideal cipher model* that the CBC-MAC scheme provides EU-CMA security.

Hint: Recall that the EU-CMA game involves an interaction between an attacker and a challenger. The challenger selects a random key k. The attacker can submit any number of messages $\hat{M}_i$ to the challenger, and receives $T = \mathsf{CBC\text{-}MAC}(k, \hat{M}_i)$ in response to each message (each message $\hat{M}_i$ is N blocks long). At the conclusion of te game, the attacker must output a pair (M^*, T^*) such that $T^* = \mathsf{CBC\text{-}MAC}(k, M^*)$ and M^* is not one of the messages previously MACed by the challenger.

Second hint: Your proof sketch should show the following things. First, describe how the challenger answers each MAC query by calling the ideal cipher oracle. Then provide an argument for why no (bounded) attacker should be able to win the above game, except with some (small) probability. Remember that the attacker can call the ideal cipher oracle too!

Part 2: What happens to variable-length messages. The CBC-MAC construction is not secure when messages have *variable* length. That is, the size of the message N can be different between different calls to the algorithm. Explain precisely where your proof sketch above breaks down in this case.

Part 3: How can we fix this flaw? Propose a modification to the CBC-MAC algorithm that addresses the flaw with variable-length messages, and give an intuitive argument for how this makes your proof work again.

2. **Hash trees**. The Bitcoin protocol relies on Merkle hash trees. Let $H : \{0,1\}^* \rightarrow \{0,1\}^\ell$ be a collision-resistant hash function. Answer the following questions:
   (a) Describe how to compute a hash tree over eight messages $(M_1, \ldots, M_8)$. Draw a picture if it helps.
   (b) Explain how I prove that a message M_5 is in the above hash tree, without producing all of the eight messages.
   (c) Let D be the root hash of the eight-item hash tree computed above. Imagine that I am able to provide a proof that a message $M' \notin \{M_1, \ldots, M_8\}$ is in the hash tree rooted by D. Show how this might contradict the assumption that H is collision-resistant.
3. **The CRT optimization for RSA**. A common optimization for speeding up RSA uses the Chinese Remainder Theorem. In this optimization, the private exponent d is converted into two separate exponents $d_p = d \ mod \ (p-1)$ and $d_q = d \ mod \ (q-1)$ and a value $q_{inv} = q^{-1} \ mod \ p$ where p, q is the factorization of the modulus N. The decryptor, given a ciphertext C, computes $A = C^{d_p} \ mod \ p$ and $B = C^{d_q} \ mod \ q$, and uses the Chinese Remainder Theorem to combine A, B into a final result M such that $M \equiv C^d \ mod \ N$.
   (a) Describe how this calculation works. *Hint:* See *e.g.,* the HAC, or Katz-Lindell.
   (b) Explain exactly why this process is more efficient than simply computing standard RSA decryption $mod \ N$.