601.445/601.645: Practical Cryptographic Systems September 13, 2021

Weekly Homework 1

Instructors: Matthew Green and Alishah Chator *Due: 11:59pm, September 22*

Name: ____

The assignment should be completed individually. You are permitted to use the Internet and any printed references.

Please submit the completed assignment via Gradescope.

Problem 1: For the following questions, let $\mathcal{K}$ be the set of possible keys for a cryptosystem, let $\mathcal{C}$ be the set of possible ciphertexts, and let $\mathcal{P}$ be the set of plaintexts. The notation $|\mathcal{C}|$ refers to the cardinality of the set $\mathcal{C}$, *i.e.,* the number of ciphertexts (as an example). Answer the following questions:

1. If the cryptosystem is a block cipher, explain why it is important that $|\mathcal{C}| = |\mathcal{P}|$ (or at least that $|\mathcal{C}| \geq |\mathcal{P}|$.)
2. Imagine that E is the encipherment mode of a block cipher, with $|\mathcal{P}| = 2^\ell$. Give an argument for why $T = E(k, M)$ might be a good Message Authentication Code for the ℓ-bit message M using key k.
3. Assume the cryptosystem is the CBC mode operation using a block cipher. You are given the encryption of an unknown single-block message, which is either M_0 or M_1 encrypted under some unknown key k. You want to know which of those two messages it contains. The relevant "target" ciphertext is:

   $$IV^*, C^*$$

   You also have access to an encryption machine that contains the unknown key k. You can use this machine to encrypt any single-block message M that you choose. It will pick IV and return the CBC encryption:

   $$IV, C$$

   Imagine that the machine has an error that will allow you to see which value of IV it will choose *before* you submit M to be encrypted. Can you use this machine to figure out which message the target ciphertext encrypts?

Problem 2: A hash function takes in messages from some domain, which is typically the set of messages of any size (although specialized hash functions can also have fixed-size inputs.) A *collision* in a hash function H is a pair of distinct inputs M_1, M_2 such that $M_1 \neq M_2$ and yet $H(M_1) = H(M_2)$.

1. Let $H : \{0,1\}^\ell \to \{0,1\}^k$ be a hash function with an input size of ℓ bits, and an output size of k bits. Let $\ell > k$. Do there exist collisions in H? Give a simple argument for why or why not.

2. Imagine that H is *collision-resistant*, in the sense that, on receiving H, no efficient attacker can find a pair (M_1, M_2) such that $H(M_1) = H(M_2)$. Show that this does not necessarily mean H is pre-image resistant. Hint: build an example hash function that is collision resistant, but not pre-image resistant (note: you can use another collision-resistant hash function H' as the ingredient for building your function.)

3. The Merkle-Damgard construction allows us to convert a fixed-input-size "compression function" $f : \{0,1\}^k \times \{0,1\}^k \to \{0,1\}^k$ into a variable-length-input hash function of the form $H : \{0,1\}^* \to \{0,1\}^k$. Sketch the construction.

4. Explain how length-extension attacks work in Merkle-Damgard.

5. Assume a block cipher with block size ℓ bits. Approximately how many messages can we expect to encrypt using CBC-mode encryption before a (random) initialization vector repeats, with probability 0.5?