CNT5412 – Network Security **Spring 2025**

Homework 3: Deadline Friday 4/11

Instructor: Viet Tung Hoang

Recall that your solution must be typed via Latex.

1. (**Password hashing**) (60 points) Let H be a good cryptographic hash function, such as HMAC-SHA-256. Let $H^k(M)$ denote the string obtained by iteratively hashing M for k times with H. Assume that we have N users whose passwords are from a dictionary of size D. Explain why the following password-hashing algorithms are bad and give the Big-Theta of the cost to recover all N passwords if such algorithms are used.

   a) (15 points) On password P and salt S, output $H^k(P)$ with $k = 10,000$.

   b) (15 points) On password P and salt S, output $H^k(S\|P)\|S$ with $k = 10$.

   c) (30 points) On password P and salt S, output $(H^k(S) \oplus H(P))\|S$ with $k = 10,000$.

2. (**Breaking IPSec encryption**) (65 points) Recall that IPSec works as follows. If Alice wants to send an IP packet P to Bob, she would send P to her gateway G_1. The latter will first pad P and append an additional Next-Header (NH) byte to obtain a string M, and then encrypt M to obtain a ciphertext C. It then creates an IPSec header H and sends a new IP packet P^* to G_2 in which the payload is $H\|C$. When G_2 receives P^*, it decrypts C, checks if the NH byte in the decrypted message M is `0x04`, and unpads M to recover P. If the checking of the NH byte fails or the padding is incorrect then G_2 drops the packet and sends an encrypted ICMP error message to G_1; otherwise it sends P to Bob.

   IPSec allows several choices of the encryption scheme. One such choice is CBC with the following padding mechanism. Assume that the underlying blockcipher has 16B block length (like AES). You need to pad just enough bytes to the next multiple of 16 bytes *minus* one byte: recall that we need to add the NH byte to the padded string, and then run CBC on top of that. Moreover, in the padding, append either `0x00` or `0x0101` or `0x020202` and so on. For example, if you have a 3B message then you must pad 12 additional bytes to it, each byte is `0x0b`.

   In this exercise, you'll break IPSec encryption with the CBC choice above.

   a) (10 points) Write a careful fragment of pseudocode for an algorithm Decrypt that decrypts an IPSec ciphertext C. (It should checks the NH byte and the padding format.) Let $\text{Decrypt}_K(C)$ return the distinguished symbol $\perp$ if it is provided an invalid ciphertext; otherwise, it returns a byte string P.

   b) (35 points) Suppose that the adversary is given an oracle Valid that, given an IPSec ciphertext C, returns a single bit: the bit "1" if C is valid–meaning $\text{Decrypt}_K(C) \in \{0,1\}^*$—and the bit "0" if it is not—meaning $\text{Decrypt}_K(C) = \perp$. (This oracle can be realized by exploiting Bob's gateway.) Show how to use the oracle to decrypt a block $Y = E_K(X)$ for an arbitrary 16-byte X. Note that Y is **not** even a CBC ciphertext and thus you can't query it directly to Valid.

   Hint: all your queries to the Valid oracle will be 32 bytes (namely 2 blocks). I don't mind if you make several thousand of them.

   c) (20 points) Show how to decrypt any IPSec ciphertext C given a Valid oracle.

   Note: If you can't solve part (b), you can assume that there is an adversary A that does the job for part (b), and show how to use it for part (c).