CO 487/687: Assignment #2 Due date: February 8 (Wednesday)

NOTE: Crowdmark links for Assignment #2 will be emailed to you before 11:00 am on January 27. If you do not receive the link, please send an email to ajmeneze@uwaterloo.ca.

1. **Chosen-plaintext attack on a Feistel cipher** (2+3+5+5 marks)
   Notation: In this question, bitstrings of length 4 are represented in hexadecimal notation. That is 0000 ↔ `0`, 0001 ↔ `1`, 0010 ↔ `2`, 0011 ↔ `3`, 0100 ↔ `4`, 0101 ↔ `5`, 0110 ↔ `6`, 0111 ↔ `7`, 1000 ↔ `8`, 1001 ↔ `9`, 1010 ↔ `a`, 1011 ↔ `b`, 1100 ↔ `c`, 1101 ↔ `d`, 1110 ↔ `e`, 1111 ↔ `f`.

   Consider the Feistel cipher F defined as follows.

   (A) $n = 24$.

   (B) $h = 3$.

   (C) The key is a randomly chosen function $S_k : \{0,1\}^3 \to \{0,1\}^3$.

   (D) Each subkey is S_k itself.

   (E) The component function $f : \{0,1\}^{24} \to \{0,1\}^{24}$ that is used in each round of encryption is the following. (This is similar to the component function for the NDS cipher presented in class. You might find it useful to draw a diagram to illustrate this function, like the NDS diagram handed out in class.) To calculate $f(z)$ where $z \in \{0,1\}^{24}$, do:

   (i) Divide z into 3 bytes: $z = (z^1, z^2, z^3)$.

   (ii) Divide each byte z^j into two nibbles: $z^j = (n_1^j, n_2^j)$ (for $1 \le j \le 3$).

   (iii) Apply $S_0 : \{0,1\}^4 \to \{0,1\}^4$ to each n_1^j to get p_1^j (for $1 \le j \le 3$). S_0 is defined as follows:

x	`0`	`1`	`2`	`3`	`4`	`5`	`6`	`7`	`8`	`9`	`a`	`b`	`c`	`d`	`e`	`f`
$S_0(x)$	`1`	`9`	`2`	`9`	`a`	`7`	`f`	`e`	`6`	`b`	`c`	`8`	`4`	`3`	`2`	`0`

   (iv) Apply $S_1 : \{0,1\}^4 \to \{0,1\}^4$ to each n_2^j to get p_2^j (for $1 \le j \le 3$). S_1 is defined as follows:

x	`0`	`1`	`2`	`3`	`4`	`5`	`6`	`7`	`8`	`9`	`a`	`b`	`c`	`d`	`e`	`f`
$S_1(x)$	`6`	`b`	`b`	`a`	`a`	`c`	`c`	`d`	`d`	`1`	`1`	`2`	`2`	`4`	`5`	`7`

   (v) Let z^* be the 3-bit string obtained by taking the first bit of each byte of z, and compute the 3-bit string $t = S_k(z^*)$.

   (vi) For each $j \in \{1,2,3\}$ do the following:
   If the jth bit of t is 1, then swap p_1^j and p_2^j.

   (vii) Apply the following permutation to the 6 transformed nibbles $p_1^1, p_2^1, p_1^2, p_2^2, p_1^3, p_2^3$:
   Swap p_1^1 and p_1^2, swap p_2^1 and p_2^3, swap p_2^2 and p_1^3.

   (viii) The resulting 24-bit string $(p_1^1, p_2^1, p_1^2, p_2^2, p_1^3, p_2^3)$ is the output $f(z)$.

   (To make sure you understand the description of this encryption function, you may wish to verify that if the secret key S_k is the following:

y	`0`	`1`	`2`	`3`	`4`	`5`	`6`	`7`
$S_k(y)$	`2`	`5`	`7`	`6`	`6`	`6`	`3`	`4`

   then the plaintext $m =$ `001122334455` gets encrypted to $c =$ `5e629957d73f`. The intermediate blocks are $m_0 =$ `001122`, $m_1 =$ `334455`, $m_2 =$ `ac8688`, $m_3 =$ `5e6299`, $m_4 =$ `57d73f`.)

(a) What is the size of the key space? (Equivalently, what is the value of the parameter ℓ?)

(b) Encrypt the plaintext $m = \texttt{001122334455}$ with the secret key

y	0	1	2	3	4	5	6	7
$S_k(y)$	2	2	3	3	4	4	5	5

Hand in the ciphertext (and clearly labelled intermediate calculations if you want part marks).

(c) Suppose now that the secret key is unknown. You have access to an encryption oracle on the course web site. Determine $S_k(0)$. Hand in a *brief* description of your attack, and the list of plaintext/ciphertext pairs you obtained from the encryption oracle.

(d) Decrypt the ciphertext $c = \texttt{0123456789ab}$. Here, the secret key is unknown (the same key as in (c)). You have access to an encryption oracle on the course web site.
(Hint: decryption is *very easy* with the encryption oracle!)

2. **Triple-DES** (3+7 marks)
Recall that DES is a block cipher with key space $K = \{0,1\}^{56}$, plaintext space $M = \{0,1\}^{64}$, and ciphertext space $C = \{0,1\}^{64}$.
Recall that Triple-DES has key space $K = \{0,1\}^{168}$. A plaintext $m \in \{0,1\}^{64}$ is encrypted under key $k = (k_1, k_2, k_3)$ (where $k_1, k_2, k_3 \in_R \{0,1\}^{56}$) as follows:

$$E_k(m) = \mathrm{DES}_{k_3}(\mathrm{DES}_{k_2}(\mathrm{DES}_{k_1}(m))).$$

(a) For Triple-DES, how many known plaintext/ciphertext pairs are needed in order to uniquely determine the secret key with high probability? (Justify your answer.)

(b) Describe a known-plaintext attack on Triple-DES that is significantly faster than exhaustive key search. Estimate the time and space requirements of your attack.

3. **CBC mode of encryption** (3+7 marks)
Recall that AES is a block cipher with plaintext space, ciphertext space, and key space all equal to $\{0,1\}^{128}$. Plaintext messages m that are longer than 128 bits are broken into blocks $m = (m_1, m_2, \ldots, m_t)$ where each block m_i is 128 bits long. In the CBC mode of operation, m is encrypted by first selecting $c_0 \in_R \{0,1\}^{128}$ and then computing $c_i = \mathrm{AES}_k(m_i \oplus c_{i-1})$ for $1 \le i \le t$; the ciphertext is $c = (c_0, c_1, c_2, \ldots, c_t)$. (Note that a new c_0 is selected each time a message is encrypted.)

(a) Suppose that a single bit of the ciphertext block c_2 is inadvertently flipped during transmission of c. What affect does this error have on the plaintext blocks that are recovered by the decryption algorithm?

(b) Show that CBC encryption is *not* semantically secure against chosen-ciphertext attack. In this attack, the adversary is given a challenge ciphertext c. Her goal is to determine some information about the corresponding plaintext m, other than its length. The adversary also has access to a *decryption oracle* to which it can present any ciphertext for decryption other than c itself.

4. **Hash functions from block ciphers** (10 marks)
Recall that AES is a block cipher with message space $\{0,1\}^{128}$ and key space $\{0,1\}^{128}$. Let x and y denote bitstrings of length 128. Which of the following hash functions $H_i : \{0,1\}^{256} \to \{0,1\}^{128}$, if any, are preimage resistant? (Explain)

(i) $H_1(x, y) = \mathrm{AES}_x(y) \oplus y$.

(ii) $H_2(x, y) = \text{AES}_y(y) \oplus x$.

(iii) $H_3(x, y) = \text{AES}_y(x) \oplus y$.

5. **Review of elementary number theory** (0 marks)
In preparation for the upcoming lectures on public-key cryptography, please review the following notions from Math 135 (or ECE 103): divisibility, greatest common divisors, prime numbers, Fermat's little theorem, linear congruences, the integers modulo n, the Chinese Remainder Theorem. I will also assume in class that you can find greatest common divisors using the Euclidean algorithm, can find inverses of integers modulo n using the extended Euclidean algorithm, and can solve linear systems of congruences.

   Exercises (not to be handed in):

   (a) Use the Euclidean algorithm to compute $\gcd(12684, 16044)$.

   (b) Use the extended Euclidean algorithm to find integers x and y such that $1234x + 4321y = 1$.

   (c) Use the extended Euclidean algorithm to compute $47^{-1} \bmod 167$. (That is find the integer x, such that $47x \equiv 1 \pmod{167}$ and $0 \leq x \leq 166$.)

   (d) Solve the congruence $47x \equiv 17 \pmod{167}$.

   (e) Solve the following simultaneous congrences:

$$\begin{aligned} x &\equiv 46 \pmod{51} \\ x &\equiv 27 \pmod{52}. \end{aligned}$$

   (f) State Fermat's little theorem.

   (g) State the Chinese Remainder Theorem.

You should make an effort to solve all the problems on your own. You are also welcome to collaborate on assignments with other students presently enrolled in CO 487. However, *solutions must be written up by yourself.* If you do collaborate, please *acknowledge your collaborators* in the write-up for each problem. *If you obtain a solution with help from a book, research paper, a web site, or elsewhere, please acknowledge your source.* You are *not* permitted to solicit help from online bulletin boards, chat groups, newsgroups, or solutions from previous offerings of the course.

The assignment should be submitted via Crowdmark before 11:00 am on February 8. Late assignments will not be accepted except in *very* special circumstances (usually a documented illness of a serious nature). High workloads because of midterms and assignments in other courses will *not* qualify as a special circumstance.

Office hours:
Alfred Menezes (MC 5026): Monday 3:00-5:00 pm, Friday 1:00-3:00pm
Ted Eaton (MC 5481): Tuesday 10:30-11:30 am
Gabriel Gauthier-Shalom (MC 5113): Thursday 3:00-4:00 pm
Philip Lafrance (MC 5497): Tuesday 12:30-1:30 pm
Christopher Leonardi (MC 5494): Tuesday 2:00-3:00 pm
Cathy Wang (MC 6313): Thursday 2:00-3:00 pm