C&O 487/687: Assignment #2

Due date: October 17, 2023 (Tuesday)

Note: Some questions use randomization to customize to you specifically. Please include your max-8-character UW user id (`g66xu`) at the beginning of your answer so we can look up your custom solution.

1. [10 marks] **Linear cryptanalysis**

   Please include your max-8-character UW user id (`g66xu`) at the beginning of your answer so we can look up your custom solution.

   This problem uses the simplified cipher described in Section 2 of "A Tutorial on Linear and Differential Cryptanalysis" by Howard M. Heys, available at
   `http://www.engr.mun.ca/~howard/Research/Papers/ldc_tutorial.html`
   We refer to this cipher as the "Heys cipher".

   For the purposes of this problem, each student has a fixed, unknown 80-bit key. You will be carrying out a known-plaintext attack against the Heys cipher using linear cryptanalysis, using a set of 20,000 distinct random plaintext-ciphertext pairs. You can download your plaintexts and ciphertexts (unique to you) at the following addresses:
   `https://www.math.uwaterloo.ca/~dstebila/as/as.cgi/download/co487_f23/a2q1plaintexts.txt`
   `https://www.math.uwaterloo.ca/~dstebila/as/as.cgi/download/co487_f23/a2q1ciphertexts.txt`

   The format of the files is that the nth line of the ciphertext file equals the encryption of the nth line of the plaintext file under your secret key.

   For the programming aspects of questions 1.a.i, 1.b, and 1.d.ii, you may work with a partner to do the programming and you may submit the same computer program source code, but you must submit your own write-up and explanations for the non-programming parts of those questions. Please indicate in your submission who you worked with.

   (a) [2 marks] Using the linear approximation $U_{4,6} \oplus U_{4,8} \oplus U_{4,14} \oplus U_{4,16} \oplus P_5 \oplus P_7 \oplus P_8 \cong 0$, Carol guesses that the target partial subkey $[K_{5,5}, K_{5,6}, K_{5,7}, K_{5,8}, K_{5,13}, K_{5,14}, K_{5,15}, K_{5,16}]$ has the value $[0, 1, 1, 1, 0, 1, 1, 0]$. Note that Carol's guess is not necessarily correct! Do one of the following:

      i. Determine the magnitude of the bias for this partial subkey value over your twenty thousand plaintext/ciphertext pairs, using a computer program, and provide the source code for your program, or:

      ii. Determine the magnitude of the bias for this partial subkey value over your first *ten* plaintext/ciphertext pairs, without using a computer program, and show your work.

   (b) [3 marks] Find the value of the partial subkey $[K_{5,5}, K_{5,6}, K_{5,7}, K_{5,8}, K_{5,13}, K_{5,14}, K_{5,15}, K_{5,16}]$ for your key, by calculating the target partial subkey which yields the largest magnitude of bias over your 20,000 plaintext/ciphertext pairs. You will almost certainly need a computer program for this task; provide the source listing for any computer code that you or your collaborators write.

   (c) [2 marks] By using Table 4 in the tutorial, compute the bias in each of the following individual S-box linear approximations:

$$S_{11} : X_1 \oplus X_4 \cong Y_1$$

$$S_{13} : X_1 \oplus X_4 \cong Y_1$$

$$S_{21} : X_1 \oplus X_3 \cong Y_2$$

$$S_{32} : X_1 \cong Y_1 \oplus Y_2 \oplus Y_3 \oplus Y_4$$

   Then, combine these to find a linear approximation of the first three rounds of the Heys cipher, and calculate the theoretical magnitude of its bias.

(d) [3 marks] Do *one* of the following:

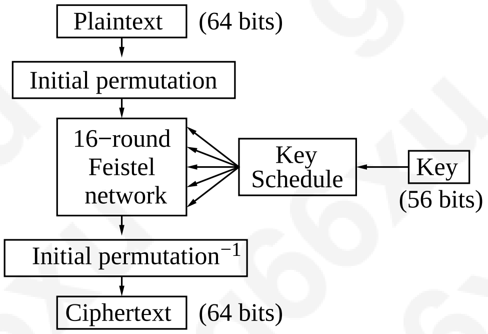
i. Using the linear approximation from part (c), determine the entire subkey K_5. You will almost certainly need a computer program for this task; provide the source listing for any computer code that you or your collaborators write, or:

ii. Using the (incorrect) guess $K_5 = [1100011111100110]$ for the fifth subkey, determine the magnitude of the bias of the approximation from part (c) for this subkey over your first ten plaintext/ciphertext pairs, without using a computer program, and show your work.

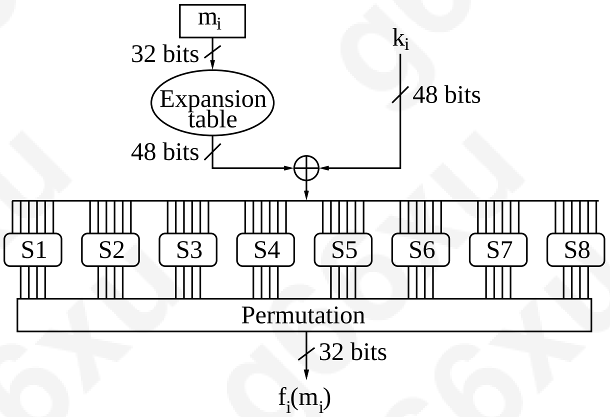
(e) [2 bonus marks] A small amount of extra credit is available if you can determine any additional key bits.

2. [6 marks] **DES complementarity property.**

This question considers a certain property of Data Encryption Standard (DES) called *complementarity*. A diagram showing an overview of DES is provided for convenience. Part (a) is meant to assist in part (b).



(a) [3 marks] Consider the structure of the component functions shown below:



Let $f(k_i, m)$ represent the output of a component function on input m, with key k_i. Show that for any choice of the S-boxes or permutation in the component function f_i, it holds that

$$f_i(k_i, m) = f_i(\overline{k_i}, \overline{m}).$$

for any message m and key k_i, where $\overline{m}$ and $\overline{k_i}$ represent the bitwise complements of m and k_i respectively.

(b) [3 marks] Now consider the entire Feistel network. Let $F(k, L_0, R_0)$ represent the 16-round Feistel network used in DES, with initial input (L_0, R_0) and key k. Show that

$$F(k, L_0, R_0) = \overline{F(\overline{k}, \overline{L_0}, \overline{R_0})}.$$

Conclude that $DES_k(m) = \overline{DES_{\overline{k}}(\overline{m})}$ for any key k and plaintext m.

3. [10 marks] **Quadruple-DES.**

   Consider quadruple DES, which is defined as follows. The key is $k = (k_1, k_2, k_3, k_4)$ where k_1, k_2, k_3, k_4 independently chosen DES keys. Encryption is $c = E_{k_4}(E_{k_3}(E_{k_2}(E_{k_1}(m))))$ where $E_k(m)$ is DES encryption with key k and message m.

   (a) [1 marks] What is the decryption function for quadruple DES?

   (b) [1 marks] What is the cost of an exhaustive key search attack on quadruple DES?

   (c) [3 marks] Describe how to use the meet-in-the-middle attack technique to perform key recovery in less time than an exhaustive key search. What is the runtime of your attack? How much data does it have to store?

   (d) [2 marks] Describe a time-space tradeoff to your attack in which you can decrease the amount of storage at the cost of increasing runtime. Quantify the trade-off.

   (e) [3 marks] Suppose now that the goal of an attack is not key recovery, but just to detect some abnormality. In particular, consider the following task. There are two possible worlds, and the adversary will be running in one of them, but they are not told which of the two worlds they are running in. The adversary's goal is determine which of the two worlds they are running in.

   - World A: A secret key $k = (k_1, k_2, k_3, k_4)$ is chosen uniformly at random. ℓ single-block messages $m_1, \ldots, m_\ell$ are chosen uniformly at random, subject to the constraint that all the chosen messages are distinct. Each message m_i is encrypted under k using quadruple DES to obtain ciphertext c_i. The adversary is given the ℓ ciphertexts $c_1, \ldots, c_\ell$.
   - World B: A secret key $k = (k_1, k_2, k_3, k_4)$ is chosen uniformly at random. ℓ single-block messages $m_1, \ldots, m_\ell$ are chosen uniformly at random, with no further constraints. Each message m_i is encrypted under k using quadruple DES to obtain ciphertext c_i. The adversary is given the ℓ ciphertexts $c_1, \ldots, c_\ell$.

   Describe an attack that allows an adversary to determine whether they are in world A or world B with probability substantially better than just randomly guessing (i.e., substantially better than 1/2, e.g. 3/4). How big does ℓ need to be for your attack to work with high probability? What is the runtime of your attack? It should be substantially faster than an exhaustive key search or even the meet-in-the-middle attack you described in parts (c) and (d).

4. [5 marks] **Block cipher modes of operation.**

   This question considers the impact of making a mistake in implementing various block cipher modes of operation. Suppose that, in implementing the following modes of operation, the key had accidentally been made public, while the initialization vector (IV) had been kept private. Answer the following questions for each mode of operation: Would an attack be enabled if the IV had been kept private while the key had been made public? If yes, what attack model is used (i.e, known-plaintext attack, chosen-plaintext attack, chosen-ciphertext attack, etc.) Justify your answer either by explaining any attack you find or arguing why this mistake does not lead to an attack.

   a) CBC

   b) CFB

   c) OFB

   d) CTR

5. [9 marks] **Hash functions.**

   For bits b_0, b_1, define the operator $\odot$ as follows:

$$b_0 \odot b_1 = \begin{cases} 1 & \text{if } b_0 = b_1 \\ 0 & \text{otherwise} \end{cases}$$

For bit strings, define $\odot$ bitwise so that

$$(b_0 b_1 ... b_1) \odot (b'_0 b'_1 ... b'_n) = (b_0 \odot b'_0)(b_1 \odot b'_1)...(b_n \odot b'_n)$$

Let $f : \{0,1\}^m \rightarrow \{0,1\}^m$ be a preimage-resistant bijection.

For $x \in \{0,1\}^{2m}$ write $x = x' \| x''$ (in other words, split the $2m$-bit string x into two m-bit halves x' and x''). Define $H : \{0,1\}^{2m} \rightarrow \{0,1\}^m$ as

$$H(x) = f(x' \odot x'')$$

a) [7 marks] For sufficiently large values of m (e.g., $m = 256$), does H have each of the following desired properties for a hash function? If yes, justify your answer by a contrapositive argument, similar to Assignment 1 Question 6(c). If no, justify you answer by showing how to break the property.

   i) Collision resistance.

   ii) Second preimage resistance.

   iii) Preimage resistance.

b) [2 marks] Suppose the input space for H is $\{0,1\}^{256}$. If you sample inputs uniformly at random, what is the expected number of steps before you find a collision in H? It is okay to use an approximation here, as long as you state, and justify that approximation.

6. [5 marks] **One-way security of OFB mode.**

Let (E, D) be a symmetric key encryption scheme. Consider the following security game:

1. The challenger C chooses a message m^* uniformly at random from the message space.
2. The challenger C chooses a key k^* uniformly at random from the key space.
3. The adversary A is given $c^* = E(k^*, m^*)$.
4. The adversary A is given access to an encryption oracle, meaning they can provide any message m and will receive $E(k^*, m)$.
5. The adversary A eventually outputs a message m', and wins the game if $m^* = m'$.

A symmetric key encryption scheme (E, D) is said to be *one-way against chosen-plaintext attack* (OW-CPA) if no computationally bounded adversary is able to win the above game with non-negligible probability.

Now we consider the OFB mode of operation. Assume that the underlying block cipher acts as a random permutation. *Added Oct. 6: In other words, you can assume that the underlying block cipher is an ideal cipher. This is equivalent to assuming it acts as a random permutation, but careful here: we mean a random permutation on the entire plaintext/ciphertext space – a random permutation mapping 2^m inputs to 2^m outputs, not a simple permutation of the bits of the plaintext.*

(a) [2 marks] For this part, suppose that, instead of the IV being chosen at random, it is fixed. Would an adversary be able to exploit this to win the OW-CPA game against OFB with non-negligible probability? Justify your response.

(b) [3 marks] For this part, suppose that in every encryption, including the challenge ciphertext c^*, the IV can be chosen by the adversary, but subject to the restriction that the adversary can use any IV at most once. Note that for the challenge ciphertext c^*, the IV is chosen by the adversary *before* the challenger chooses m^*. Would an adversary be able to exploit this to win the OW-CPA game against OFB with non-negligible probability? Justify your response.

Academic integrity rules

You should make an effort to solve all the problems on your own. You are also welcome to collaborate on assignments with other students in this course. 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, paper, a website, or any other source, please acknowledge your source. You are not permitted to solicit help from other online bulletin boards, chat groups, newsgroups, or solutions from previous offerings of the course.*

Due date

The assignment is due via Crowdmark by 8:59:59pm on October 17, 2023. Late assignments will not be accepted.

Changelog

- Fri. Oct. 6: Clarification on phrase ”random permutation” in question 6