CPSC 329 W17 - Assignment 3

University of Calgary
Due Sunday April 9, 11:59 pm.
Individual assignment, no group work.

Part 1: Apple vs. FBI (10 points)

Apple vs. the FBI dispute has been in the news since February 2016. See for example

- https://en.wikipedia.org/wiki/FBI%E2%80%93Apple_encryption_dispute
- https://www.fbi.gov/news/pressrel/press-releases/fbi-director-comments-on-san-bernardino-matter
- http://www.apple.com/customer-letter/

An informative panel discussion on this topic is in a video below. The relevant discussion starts at the 10 minute mark:

- https://www.youtube.com/watch?v=k76qLOrna1w

Q1.1 Briefly outline the case, in particular what is requested by the FBI, and Apple's position.

Q1.2 Give at least one reason in support of FBI's position, and one reason in support of Apple's position. Your reasons must be clearly argued and justified. For example, if your reason is "user privacy" you need to say how it could/would be threatened.

Part 2: XOR cryptography (20 points)

Consider the binary one-time-pad with the binary key

K = 11010 10010 00100 01011 10100 10111 00101 10010

Q2.1 First, use MyCode below to generate the binary string M corresponding to your first name (consider up to 8 characters). Then, encrypt this string M to obtain ciphertext $C_1 = M \oplus K$, where "$\oplus$" is bitwise XOR operation.

Q2.2 Demonstrate that $C_1 \oplus K$ correctly recovers (decrypts) the message.

Q2.3 Replace the first 5 bits of the ciphertext C_1 with 10101, and call it the new ciphertext C_2. Find the plaintext M_2 corresponding to C_2.

How is this plaintext different from the original M? Is this expected? Justify your answer.

Q2.4 Can the security of the encryption system be improved by using double encryption with two randomly chosen keys? By double encryption we mean using two keys K_1 and K_2, each 20-bit long, to find ciphertext $C = (M \oplus K_1) \oplus K_2$. Justify your answer.

MyCode: a code for converting alphanumeric characters to and from binary. For example, the string "TEST" would be encoded to "10011 00100 10010 10011".

```
A=00000  B=00001  C=00010  D=00011  E=00100  F=00101  G=00110  H=00111
I=01000  J=01001  K=01010  L=01011  M=01100  N=01101  O=01110  P=01111
Q=10000  R=10001  S=10010  T=10011  U=10100  V=10101  W=10110  X=10111
Y=11000  Z=11001  1=11010  2=11011  3=11100  4=11101  5=11110  6=11111
```

Part 3: Simple Feistel algorithm (25 points)

Consider the SimpleFeistel algorithm below, which is a two round Feistel network. Let $S = \{0 \ldots 63\}$ denote a set of integers between (and including) 0 and 63. Note that each number in S has a 6-bit binary representation. The variables and operations used in the Simple Feistel are as follows:

- plaintext = (X_1X_2), ciphertext = (Z_1Z_2) and the key = (K_1K_2) are given by X_1, X_2, Z_1, Z_2, K_1 and K_2, all in S.
- "$\oplus$" is a bitwise XOR.
- $F(x,K) = (K + x)^2 \bmod 64$

To encrypt a plaintext block (X_1X_2) using the key $K = (K_1K_2)$, first $F(X_2,K_1)$ is computed and XOR-ed with X_1. The result is then swapped with X_2, to form the intermediate result (Y_1Y_2), which is the output of the first round. The same steps are repeated for the second round, replacing (X_1X_2) with (Y_1Y_2), and K_1 with K_2.

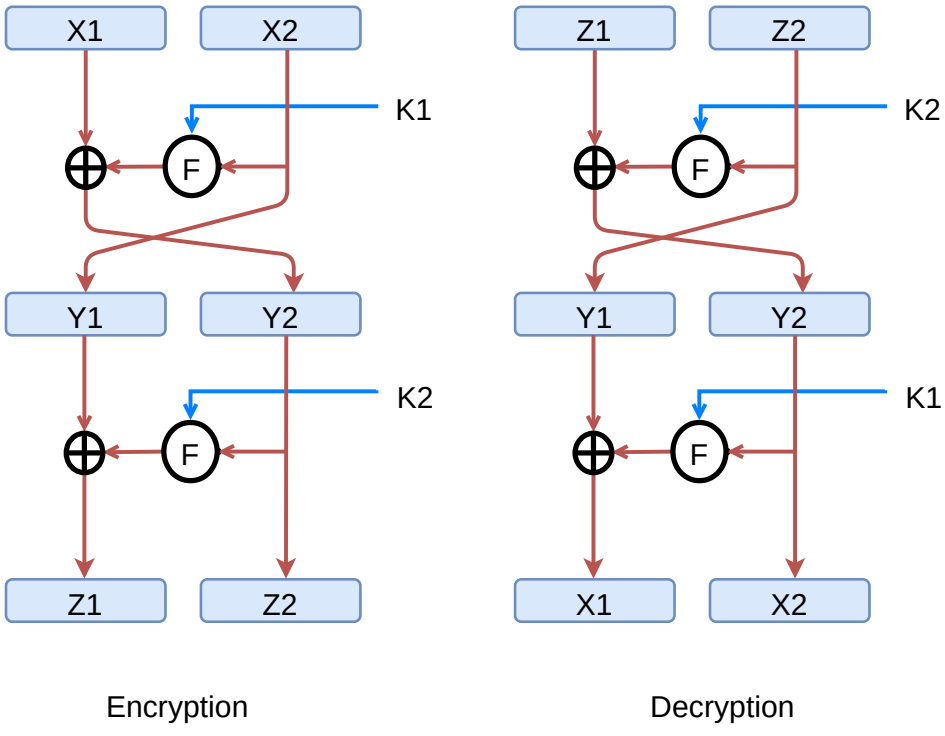

Figure 1: SimpleFeistel encryption and decryption algorithms.

Answer the following questions:

Q3.1 Let $K_1 = 12$, $K_2 = 13$. Find the cipher text (Z_1Z_2) corresponding to $(X_1X_2) = (56,23)$. Write all steps of the algorithm.

Q3.2 Use the decryption algorithm on the ciphertext to show that it in fact recovers (decrypts) the plaintext.

Q3.3 Suppose the least significant bit of Z_1 is flipped to obtain Z'_1. Find the plaintext $(X'_1X'_2)$ corresponding to (Z'_1Z_2).

Q3.4 How many bits in the plaintext $(X'_1X'_2)$ corresponding to (Z'_1Z_2) remain the same as the original plaintext? Compare this with the results you obtained for Q2.3. Discuss if this is a desirable/undesirable property for encryption systems.

Q3.5 When we discussed the CTR mode of operation in lectures, we talked about how any block cipher can be used to create a keystream for implementing a stream cipher. Use the SimpleFeistel above to generate a sequence of 3 pseudo-random numbers (the first 3 numbers of a keystream). Assume the initial value of the counter is 123, and the key is the same as for Q3.1.

Part 4: Cryptographic hashing (45 points)

Consider the following one-way compression function f:

$$f(y_1,y_2) = (y_1 * y_1 + y_2 * y_2 + 11) \bmod 15$$

where y_1 and y_2 are 4-bit positive integers. This compression function f() can be used to construct a hash function h() using the Merkle-Damgard method as discussed in lectures. For this question assume the initialization vector IV=1111. Also assume that padding is performed in two steps:

1. if needed, pad the last block of the message with 0's, and
2. append an extra block representing the length of the message in binary (this will limit the size of the message to at most 15 bits).

Q4.1 Find the hash value of the message M=1010100101.

Q4.2 Flip the fourth bit of the message M and compute the hash value. How many bits in the hash value change as a result? Show the steps of deriving your results.

Q4.3 The hash function h() is used to construct *tinyHMAC* for the above message space (messages are 10 bit long), using HMAC structure and assuming *opad* = 0110 and *ipad* = 1001.

Draw the block diagram that describes how MAC generation will work. The diagram needs to include input and output of each block, and any additional parameters that are needed for the computation.

Q4.4 Using the above tinyHMAC, calculate the tag of the above message M when k = 1010.

Q4.5 Anette and Bernard are using the above tinyHMAC for protecting integrity of communication. To reduce the success chance of the attacker, they are considering two modifications:

(a) Use two keys k_1 and k_2. The MAC of a message m will be tinyHMAC(k_1,m) $\oplus$ tinyHMAC(k_2,m) .

(b) Use two keys k_1 and k_2. The MAC of a message m will be $tinyHMAC(k_1,m) \mid tinyHMAC(k_2,m)$.

Evaluate these two designs in terms of improved security and efficiency. Security is measured by the success chance of attacker in forging a message. Efficiency is measured by the number of extra bits that must be sent for each message.

Q4.6 Suppose we want to make the hash pre-image resistant to 40 bit security (success chance is 2^{-40}). What is the smallest size of the hash output? Justify your answer.

Submission details:

1. Submit a single PDF file for the entire assignment. Use a text editor or a word processor to write your assignment, then convert to PDF. This is to ensure that handwriting will not impede marking and evaluation of your work.
2. Write your name, student ID and tutorial section on the first line of the submitted assignment and use the following naming convention to name your file.

   `A1-<surname>-<student id>-<tutorial section>.pdf`

3. Submit your PDF file to D2L submission system. Submit on time. Late submissions will not be possible!

General information about all assignments:

1. Due time: All assignments are due at 23:59 on the due date listed on the assignment. Late assignments or components of assignments will not be accepted for marking without approval for an extension beforehand. What you have submitted in D2L as of the due date is what will be marked.
2. Extensions may be granted for reasonable cases, but only by the course instructor, and only with the receipt of the appropriate documentation (e.g. a doctor's note). Typical examples of reasonable cases for an extension include: illness or a death in the family. Cases where extensions will not be granted include situations that are typical of student life, such as having multiple due dates, work commitments, etc. Forgetting to hand in your assignment on time is not a valid reason for getting an extension.
3. After you submit your work to D2L, make sure that you check the content of your submission. It's your responsibility to do this, so make sure that you submit your assignment with enough time before it is due so that you can double-check your upload, and possibly re-upload the assignment.
4. All assignments should include contact information, including full name, student ID and tutorial section, at the very top of each file submitted.
5. Assignments must reflect individual work. Group work is not allowed in this class nor can you copy the work of others. For further information on plagiarism, cheating and other academic misconduct, check the information at this link: http://www.ucalgary.ca/pubs/calendar/current/k-5.html.
6. You can and should submit many times before the due date. D2L will simply overwrite previous submissions with newer ones. It's better to submit incomplete work for a chance of getting partial marks, than not to submit anything.
7. Only one file can be submitted per assignment. If you need to submit multiple files, you can put

them into a single container. The container types supported will be ZIP and TAR. No other formats will be accepted.

8. Assignments will be marked by your TAs. If you have questions about assignment marking, contact your TA first. If you still have questions after you have talked to your TA then you can contact your instructor.