Last Name:

Examination 2
CEN 5540 Computer and Network Security
15 June 2000

Instructions:

- **Failure to follow instructions may result in loss of points.**
- Do not start the examination until instructed to do so.
- This test is closed book, but you may use one 8.5” by 11” sheet of notes you have prepared.
- **Answer two (2) questions below.**
- Start the answer to each question on a new page (i.e., do **not** put the answer to more than one question on the same page).
- Show your work.
- Assemble your answers in numerical order of the questions when you submit them.
- **Leave a 1” square in the upper left corner for a staple.**
- Be sure to include your name on your answer sheets.
- You have 60 minutes to complete this examination.
- **Read and sign the following statement.** You may write this on your exam and sign it there if you wish to take the exam questions home with you today. Do not discuss this exam with anyone in this course who has not yet taken this exam.

On my honor, I have neither given nor received unauthorized aid on this examination, and I will not discuss the contents of this examination with any student who has not yet taken this examination.

Signed:

1. (a) (4) What advantages does taking the Index of Coincidence (or variance) of some ciphertext have over applying the Kasiski attack? What advantages does the Kasiski attack have over taking the IC? Give two examples, one showing an advantage of IC over KA, and the other showing an advantage of KA over IC.

   (b) (2) Define a "weak key" for a transposition cipher to be one that causes the period to less than the block size. Characterize the transpositions whose keys (i.e., permutations) are weak in this sense.

   (c) (4) Most transposition ciphers are block ciphers: a permutation on locations within each block is defined as the key and the same permutation is applied to each block of text. What is necessary to create a "non-local" transposition, a transposition cipher that does not confine all of its relocations to the same block (i.e., ciphertext block c_i may contain symbols from plaintext blocks other than p_i)? How does its key length compare to those for a local transposition cipher (parametrize by block length B)?

   (d) (5) Give a complete example of a "non-local" transposition, including how the first and last block of plaintext are handled, along with decryption.

2. (a) (5) How many iterations of the DES round function are required before every output bit is affected by the first (leftmost) input bit after the initial permutation (i.e., input to round one)? Prove it, assuming that every input bit to an S-box affects every output bit of the S-box. The P-box permutation is given below in Table 1. Show which bits of are affected by the first input bit after each round, and argue why that is so.

   (b) (2) What happens if the P-box is omitted from the round function?

   (c) (5) Show that DES has the complementation property: if X' is the bit-wise complement of X, then for DES, $E(P', K') = C'$ if $E(P, K) = C$.

   (d) (3) Show that the Feistel structure for encryption also may be used for decryption (assuming the same round function) if the input halves are swapped before decryption and the output halves are swapped after decryption.

Table 1: P-Box permutation for DES round

Bit	Goes To Position							
1-8	9	17	23	31	13	28	2	18
9-16	24	16	30	6	26	20	10	1
17-24	8	14	25	3	4	29	11	19
25-32	32	12	22	7	5	27	15	21

3. (a) (5) Recall that public key cryptosystems (PKCs) have two keys to the single key of symmetric cryptosystems. Multi-key cryptosystems (MKCs) have more than two keys, $K_1, K_2, ..., K_n$, and only by performing the "encryption" operation with all n keys may the plaintext be recovered,

$$E(E(...E(E(P, K_n), K_{n-1})), ..., K_2), K_1) \;=\; P$$

What uses might such a system have? List three distinct uses and explain how a MKC would be employed. For each use, what other properties must the MKC have?

(b) (5) For the uses that you provide in the previous part, either show how a standard (two-key) PKC could be used instead, or argue that it could not be used for that purpose. If a standard PKC could be used, the compare the PKC approach and the MKC approaches for each application.

(c) (5) Develop an MKC. That is, give methods for encryption and decryption, and specify what properties the keys must have explicitly.

4. (a) (5) Karn gave an encryption algorithm based on one-way hash functions as follows. The plaintext is split into two pieces, P_l and P_r, and the key is also split into two pieces, K_l and K_r:

$$P \;=\; P_l|P_r \qquad\qquad K \;=\; K_l|K_r$$

The ciphertext is produced as two halves, C_l and C_r as follows:

$$C_r \;=\; P_r \oplus H(P_l, K_l) \qquad\qquad C_l \;=\; P_l \oplus H(C_r, K_r) \qquad\qquad C \;=\; C_l|C_r$$

Show how to decrypt the ciphertext given the key and the hash function. What relationships between the hash function, the key size and the plaintext size must exist?

(b) (5) Show how Karn's algorithm is insecure if two ciphertexts encrypted with the same key are obtained that correspond to two plaintext messages that have a common first half (i.e., $P_l \;=\; P_l'$ for plaintexts P and P'), and the first plaintext message P is known. Suggest a (simple) fix to this problem and argue that it works.

(c) (5) Lamport described an authentication method based on one-way hashes that works as follows. Alice generates a random number (or more likely, a hash of some passphrase) r_a, then picks a desired lifetime, N,for this number then repeatedly hashes r_a N times, computing $H^N(r_a) \;=\; H(H(...H(r_a)...))\ N\ times$ (e.g., $H^2(r_a) \;=\; H(H(r_a))$). Alice then stores this final hashed value and N with the host to whom she wishes to authenticate herself later. The host maintains a world-readable, but integrity-protected table associating users with their hashes and indexes. When Alice wants to authenticate herself to the host, she provides her name, and the host looks up her entry in the table and sends $N-1$ to her. She then responds with $X \;=\; H^{N-1}(r_a)$, which the host verifies by taking its hash and comparing to the stored value, $H^N(r_a) \;=\; H(X)$. If the values match, the host accepts Alice as authenticated, then replaces N with $N-1$ and $H^{N-1}(r_a)$ with X in the table. What weaknesses exist (if any) in this protocol on a single server? What about if there are multiple servers in a network and Alice wants to use the same r_a (passphrase) for all of them? Suggest an approach that will allow her to do this and still maintain security.