CS255: Cryptography and Computer Security **Winter 1998**

Assignment #1

Due: Friday, January 23rd, 1998.

1. Consider the following encryption scheme for n-letter messages. The secret key π corresponds to a permutation on n locations. Given a message $M = m_1 m_2 \cdots m_n$, one computes its encryption $E_\pi(M)$ by
$$E_\pi(M) = m_{\pi(1)}, \ldots, m_{\pi(n)}.$$
For example, suppose that $\pi = \begin{pmatrix} 1 & 2 & 3 \\ 3 & 1 & 2 \end{pmatrix}$. That is $\pi(1) = 3$, $\pi(2) = 1$ and $\pi(3) = 2$. Then the encryption $E_\pi(\text{“Dan”})$ would be “nDa”.

   A. How does the security of this scheme vary according to the message size?

   B. What information is leaked by a single encryption using this scheme? That is, given $E_\pi(M)$, what can we determine about M.

   Now let's try to break this scheme completely given multiple encryptions $E_\pi(M_1), \ldots, E_\pi(M_k)$ for a fixed key π. We assume that we don't know $M_1, \ldots, M_k$ *a priori* but that they are in english.

   C. For a given i and j, how might we determine whether $\pi^{-1}(i) = \pi^{-1}(j) + 1$, given enough messages? That is, can we determine if letters i and j of the ciphertext correspond to consecutive letters of the plaintext.

   D. Using the answer to Part 3, show how to reconstruct π.

   E. Suppose that $M_1, \ldots, M_k$ were not english but instead were just random strings. Could we find π given the ciphertext only?

2. Data compression is often used in data storage or transmission. Suppose you want to use data compression in conjunction with encryption. Does it make more sense to

   A. Compress the data and then encrypt the result, or

   B. Encrypt the data and then compress the result.

   Justify your answer. Try to give at least two reasons.

3. Before DESX was invented, the researchers at RSA Labs came up with DESV and DESW, defined by
$$\begin{aligned} DESV_{kk_1}(M) &= DES_k(M) \oplus k_1 \text{ and} \\ DESW_{kk_1}(M) &= DES_k(M \oplus k_1) \end{aligned}$$
As with DESX, $|k| = 56$ and $|k_1| = 64$. Show that both these proposals do not increase the work needed to break the cryptosystem using brute-force key search. That is, show how to break these schemes using on the order of 2^{56} DES encryptions/decryptions. You may assume that you have a moderate number of plaintext-ciphertext pairs, $C_i = DES\{V/W\}_{kk_1}(M_i)$.

4. Given a cryptosystem E_k, define the randomized cryptosystem F_k by

$$F_k(M) = (E_k(R), R \oplus M),$$

where R is a random bit string of the same size as the message. That is, the output of $F_k(M)$ is the encryption of a random one-time pad along with the original message XORed with the random pad. A new independent random pad R is chosen for every encryption.

We consider two attack models. The goal of both models is to reconstruct the actual secret key k.[1]

- In the key-reconstruction chosen plaintext attack (KR-CPA), the adversary is allowed to generate strings $M_1, M_2, \ldots$ and for each M_i learn a corresponding ciphertext.
- In the key-reconstruction random plaintext attack (KR-RPA), the adversary receives random plaintext/ciphertext pairs.

Note that for the case of F_k the opponent has no control over the random pad R used in the creation of the given plaintext/ciphertext pairs.

Prove that if E_k is secure against KR-RPA attacks then F_k is secure against $\mathsf{KR-CPA}$ attacks.

[Hint: It is easiest to show the contrapositive. Given an algorithm A that executes a successful $\mathsf{KR-CPA}$ attack against F_k, exhibit an algorithm B (using A as a "subroutine") that executes a successful $\mathsf{KR-RPA}$ attack against E_k.]

[1]This is a very strong goal - one might be able to decrypt messages without ever learning k.