CS255: Cryptography and Computer Security **Winter 2001**

Assignment #1

Due: Friday, February 1st, 2002.

Problem 1 Let p be a 128-bit prime and let $\mathbb{Z}_p$ be the set of integers $\{0, \ldots, p-1\}$. Consider the following encryption scheme. The secret key is a pair of integers $a, b \in \mathbb{Z}_p$ where $a \neq 0$. An encryption of a message $M \in \mathbb{Z}_p$ is defined as:

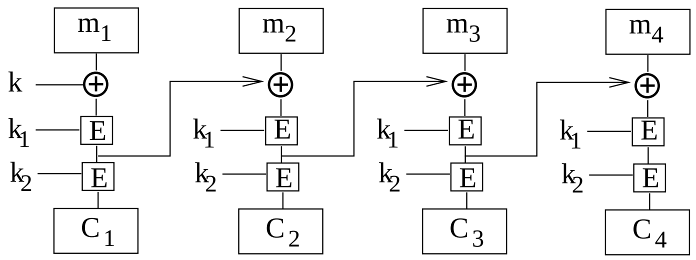
$$E_{a,b}[M] = aM + b \pmod{p}$$

a. Show that when E is used to encrypt a random message $M \in \mathbb{Z}_p$ the system has perfect secrecy in the sense of Shannon.

b. Show that if the system is used to encrypt messages $\langle M_1, M_2 \rangle$ then the system does not have perfect secrecy. Hence, although the system has perfect secrecy for one message it is not very useful as is.
Hint: consider the case $M_1 = M_2$.

c. Show that given two random plaintext/ciphertext pairs $C_i = E_{a,b}[M_i]$ for $i = 1, 2$ with $M_1 \neq M_2$ it is possible to recover the key a, b with high probability.

Problem 2 Let E, D be the encryption/decryption algorithms of a certain block cipher. Consider the following chaining method for double DES like encryption:



The secret key is a triple (k, k_1, k_2) where k is as long as E's block size (64 bits for DES) and k_1, k_2 are as long as E's key size (56 bits for DES). For example, when E is DES the total key size is 64+56+56 = 176 bits.

a. Describe the decryption circuit for this system.

b. Show that using two short chosen ciphertext decryption queries an attacker can recover the full key (k, k_1, k_2) in approximately the time it takes to run algorithm D 2^ℓ times (i.e. the attack running time should be $O(2^\ell \text{time}(D))$. Here ℓ is the block cipher's key-length (56 bits for DES). Your attack shows that this system can be broken much faster than exhaustive search.
Hint: Consider the two decryption queries $\langle C_1, C_2, C_3, C_4 \rangle$ and $\langle C_1', C_2, C_3', C_4 \rangle$ where $C_1, \ldots, C_4$ and C_1', C_3' are random ciphertext blocks.

Problem 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, $C_i = DES\{V/W\}_{kk_1}(M_i)$.

Problem 4 The movie industry (i.e. MPAA) wants to protect digital content distributed on DVD's. We study one possible approach. Suppose there are at most a total of n DVD players in the world (e.g. $n = 2^{32}$). We view these n players as the leaves of a binary tree of height $\log_2 n$. Each node v_i in this binary tree contains an AES key K_i. These keys are kept secret from consumers and are fixed for all time. At manufacturing time each DVD player is assigned a serial number $i \in [0, n-1]$. Consider the set S_i of $\log_2 n$ nodes along the path from the root to leaf number i in the binary tree. The manufacturer of the DVD player embeds in player number i the $\log_2 n$ keys associated with the nodes in S_i. In this way each DVD player ships with $\log_2 n$ keys embedded in it (these keys are supposedly inaccessible to consumers). A DVD movie M is encrypted as

$$DVD = \underbrace{E_{K_{root}}(K)}_{\text{header}} \Big\| \underbrace{E_K(M)}_{\text{body}}$$

where K is some random AES key called a content-key. Since all DVD players have the key K_{root} all players can decrypt the movie M. We refer to $E_{K_{root}}(K)$ as the header and $E_K(M)$ as the body. In what follows the DVD header may contain multiple ciphertexts where each ciphertext is the encryption of the content-key K under some key K_i in the binary tree.

a. Suppose the $\log_2 n$ keys embedded in DVD player number r are exposed by hackers and published on the Internet (say in a program like DeCSS). Show that when the movie industry is about to distribute a new DVD movie they can encrypt the contents of the DVD using a header of size $\log_2 n$ so that all DVD players can decrypt the movie except for player number r. In effect, the movie industry disables player number r.
Hint: the header will contain $\log_2 n$ ciphertexts where each ciphertext is the encryption of the content-key K under certain $\log_2 n$ keys from the binary tree.

b. Suppose the keys embedded in k DVD players $R = \{r_1, \ldots, r_k\}$ are exposed by hackers. Show that the movie industry can encrypt the contents of a new DVD using a header of size $O(k \log n)$ so that all players can decrypt the movie except for the players in R. You have just shown that all hacked players can be disabled without affecting other consumers.

Problem 5 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 ($\mathsf{KR\text{-}CPA}$), the adversary is allowed to generate q strings $M_1, M_2, \ldots, M_q$ and for each M_i learn a corresponding ciphertext.
- In the key-reconstruction random plaintext attack ($\mathsf{KR\text{-}RPA}$), the adversary is given q 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. Clearly a $\mathsf{KR\text{-}CPA}$ attack gives the attacker more power than a $\mathsf{KR\text{-}RPA}$ attack. Consequently, it is harder to build cryptosystems that are secure against $\mathsf{KR\text{-}CPA}$.

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

Hint: It is easiest to show the contrapositive. Given an algorithm A that executes a successful $\mathsf{KR} - \mathsf{CPA}$ attack against F_k, construct an algorithm B (using A as a "subroutine") that executes a successful $\mathsf{KR} - \mathsf{RPA}$ attack against E_k. First, define precisely what algorithm A takes as input, what queries it makes, and what it produces as output. Do the same for B. Then construct an algorithm B that runs A on a certain input and properly answers all of A's queries. Show that the output produced by A enables B to complete the $\mathsf{KR} - \mathsf{RPA}$ attack against E_k.

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