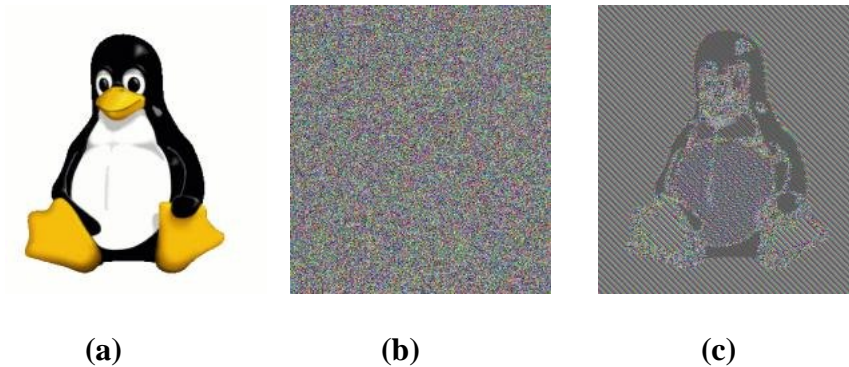


## HW #2: Private Key Cryptography

CS 392/681: Computer Security  
Fall 2006

**[100pts] DUE 09/23/2006 (noon)**

### Problem 1 [10pts]



A pixel map (consisting of a matrix corresponding to the pixel values) of image in Figure (a) was encrypted with DES in two different modes, ECB and CBC, to obtain the images in Figure (b) and Figure (c), “not” respectively. Which mode does Figure (b) correspond to? Which mode does Figure (c) correspond to? Explain why. Which image represents a “good” encryption and why?

### Problem 2 [20pts]

In the class, we studied the Caesar’s cipher. A similar cipher is called affine cipher which works as follows. Encryption: each plaintext letter  $P$  is encrypted to obtain the ciphertext letter  $C$  such that  $C = aP + b \pmod{26}$ , (where  $a, b$  are numbers between  $0, 1, \dots, 25$ , and represent the secret key). Decryption: each ciphertext letter  $C$  is decrypted to obtain the plaintext letter such that  $P = (C - b)a^{-1} \pmod{26}$ .

I need to send a message to the class

“HELLOCLASSALLOFYOUWILLGETANAYOUDONOTNEEDTODOTHEHOMEWORK” (**I don’t mean it!**), and want to send it encrypted using the affine cipher so that the department chair does not learn the message ☺. The chair intercepts the 1<sup>st</sup> and 6<sup>th</sup> letters of the cipher text, ‘N’ and ‘O’ respectively, and somehow learns the corresponding plaintext letters, i.e., ‘H’ and ‘C’ respectively. Can he decrypt the message (and take a disciplinary action against me ☺)? If so, explain how? What is the secret key? What is the original ciphertext that I sent out?

### Problem 3 [25pts]

A 64-bit long message “10110100 01010101 01001010 10101001 10101001 10101001 01010101 11111010” (ignore the “spaces”) was encrypted with DES in ECB mode and following ciphertext was obtained “10010101 01101010 10111101 10001001 10101111 01010101 10101010 10001001”

Can you figure out the ciphertext when the message “01001011 10101010 10110101 01010110 01010110 01010110 10101010 00000101” is encrypted with DES ECB? If so, explain the details how.

[Hint: You don't need to write a program; you only need to know how DES works (which was explained in the class)]

### Problem 4 [25pts]

The reading assignment in the last lecture was to understand how AES (Rjindael) works. In this homework assignment, you will get an idea as to how much processing time AES takes to perform key generation, encryption and decryption.

Download the AES code from <http://fp.gladman.plus.com/AES/aes.zip>. Get familiarized with AES key generation, encryption and decryption functions and the test code in the package.

Choose a key and block size of 128 bits. Choose any plaintext M 128\*5 bit long.

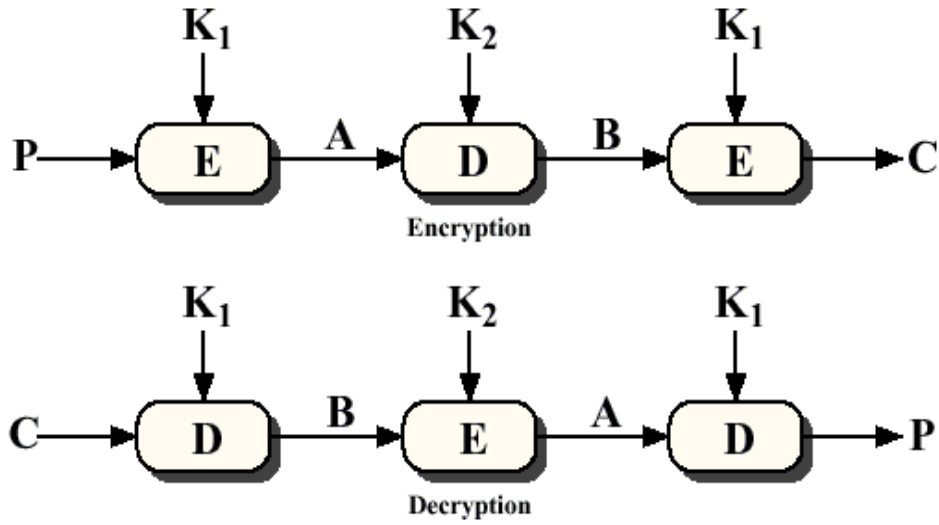
1. execute the key generation function to get a key K
  - o compute the execution time
2. execute the encryption function to encrypt M using K, and obtain the ciphertext C;
  - o compute the execution time
3. execute the decryption function to decrypt C using K, and to obtain M back.
  - o compute the execution time

Repeat each of the above 100 times and give the average execution time for each of the three functions. List the type and speed of the processor, and the memory (RAM) of the machine you execute the code on.

[I hope you know how to measure execution time! If not, figure it out yourself.]

### Problem 5 [20pts]

In the class, we studied the meet-in-the-middle attack on double-DES, which is a *known-plaintext* attack. Let us now look at the Triple-DES with two keys (we briefly studied this too), which works as follows:



Is the above Triple-DES susceptible to the meet-in-the-middle attack? Why or why not?  
 Is it secure against a *chosen-plaintext* attack? If so, explain why? If not, explain the attack?  
 (note that an attack is considered an “attack” only if it requires encryption and decryption operations fewer than  $2^{112}$ . Also, assume you have infinite memory at your disposal)