ECE 471/571: Sample Exam Problems Set

Instructor: Ming Li

Electrical and Computer Engineering, University of Arizona

Problem 1: (*True or False*)
These questions require YES or NO answers with explanations.
(1). In theory, if the key is truly random, never reused, and kept secret DES and AES are both provably secure against known plaintext attacks.

(2). A Feistel cipher structure lets you use the same hardware or software for decryption as for encryption.

(3). Recall the modes of encryption. For the ECB and counter modes, if we change one bit in one ciphertext block, then only one block in the decrypted plaintext will be garbled.

(4). The textbook RSA cryptosystem is secure against chosen ciphertext attacks.

(5). It is impossible for both public key encryption and digital signature schemes to achieve perfect secrecy.

Problem 2: Symmetric Key Encryption

a) What is a one-time pad?

b) Any good random number generator can be used as a secret-key encryption algorithm. Explain how?

Problem 3: Suppose Alice and Bob know each other's public key. Alice sends a message to Bob. How can she encrypt the message so that, when Bob receives it, he is sure about all of the following:

- Nobody else can view the content (confidentiality),
- The message is from Alice and no one has modified it (authentication, integrity).
- Nobody else (Eve) could trick Bob into thinking that Eve also generated the same message.

Problem 4: (Graduate students only) What is a practical method for finding a triple of keys that maps a given plaintext to a given ciphertext using EDE (where E stands for DES encryption, and D is for DES decryption)?

Problem 4: (Undergraduate students only) Suppose that the F function of DES mapped every input R to (a) a 32-bit string of ones and, (b) to a bitwise complement of R denoted as R', regardless of what the key k is. Find the ciphertext y for a given plaintext x.

<u>Problem 5</u>:
Suppose you are designing a processor that would compute with encrypted data. For example, given two encrypted data values $E_K(x)$ and $E_K(y)$, the processor would compute $E_K(x) + E_K(y)$, where "+" is an encrypted addition operator that performs addition on encrypted numbers. And the decryption: $D_K(E_K(x) + E_K(y))$ will be the same as $x + y$. None of the encryption algorithms we have learnt so far has the property that $E_K(x) + E_K(y) = E_K(x + y)$, although the encrypted addition operator does not necessarily have to be arithmetic addition. For DES and AES, is there any relationship between $E_K(x), E_K(y), E_K(x+y)$? Is this property desirable? How about for RSA?

Problem 6: (Undergraduate students only)
We consider the RSA encryption.
(1) 3 and 65537 are commonly used as the public key. Can they be used as the private key instead? Why or why not?

(2) To illustrate the RSA system, we use primes $p = 23$ and $q = 17$. As public encryption key we use $e = 3$. Show that the private key $d = 235$.

(3) Suppose Bob has an RSA Cryptosystem with a large modulus n for which the factorization cannot be found, e.g., n is 1024 bits long and Alice sends a message to Bob by representing each alphabetic character as an integer between 0 and 25 (i.e., $A \rightarrow 0$; $B \rightarrow 1$; ... $Z \rightarrow 25$) and then encrypting each letter as a separate plaintext character. Describe how Oscar can easily decrypt a message which is encrypted in this way.