Cryptography & Blockchain – 5IF, INSA Lyon

DES

Attempt the following exercises. Describe your analysis results in a report. Upload your report and source code to the course Moodle page.

1. Design your own symmetric-key encryption and decryption algorithms based on the Feistel cipher and implement them. You may use any programming language e.g. Python, C++, Java. Note: You may select any trivial round function F and subkey generation (key schedule) algorithm for this exercise. For example, you could use a simple XOR operation as the round function and you could use the same subkey for each of the rounds.

2. Generate a random secret key. Use your implementation of the Feistel cipher to encrypt several plaintext messages. Decrypt the ciphertext to verify the correctness of the cryptosystem.

3. Implement the DES 56-bit symmetric-key encryption and decryption algorithms. Implement the round function F and subkey generation algorithm as used by DES. The round function involves an expansion operation, substitution (using S-boxes), and permutation (using P-boxes). The subkey generation algorithm involves circular left shift operations and a contraction permutation operation (called Permuted Choice 2). You may look at the following or any other guide for a description of DES:
https://www.geeksforgeeks.org/computer-networks/data-encryption-standard-des-set-1/

4. Analyze the performance of your DES implementation for varying message sizes. Observe performance in terms of the processing time required for encryption. You may use the following message sizes: 2^{10} B, 2^{11} B, 2^{12} B, …, 2^{26} B.

Additional exercises:

5. Use an existing library implementation of DES to encrypt plaintext messages. For example, in Python, you will find a DES implementation in the Crypto.Cipher package. Compare its performance with the performance of your own implementation.

6. Use a fairly large English text as the plaintext. Encrypt the entire text to ciphertext using a simple substitution cipher e.g. Caesar or Playfair. Run an English language alphabet frequency analysis attack on the ciphertext. Analyze the results. Then, encrypt the same plaintext with your DES implementation. Rerun the English language alphabet frequency analysis attack on the ciphertext. Analyze the results.