Introduction to Cryptography, 2021 Spring **Due: 2021/5/28 (Friday)**

Homework 5

Instructor: Prof. Wen-Guey Tseng

Part 1: Written Problems

1) Now consider the opposite problem: using an encryption algorithm to construct a one-way hash function. Consider using RSA with a known key. Then process a message consisting of a sequence of blocks as follows: Encrypt the first block, XOR the result with the second block and encrypt again, etc. Show that this scheme is not secure by solving the following problem. Given a two-block message B1, B2, and its hash

$$\mathrm{RSAH}(\mathrm{B1}, \mathrm{B2}) = \mathrm{RSA}(\mathrm{RSA}(\mathrm{B1}) \oplus \mathrm{B2})$$

Given an arbitrary block C1, choose C2 so that $\mathrm{RSAH}(\mathrm{C1}, \mathrm{C2}) = \mathrm{RSAH}(\mathrm{B1}, \mathrm{B2})$.
Thus, the hash function does not satisfy weak collision resistance.

2) DSA specifies that if the signature generation process results in a value of s = 0, a new value of k should be generated and the signature should be recalculated. Why?

3) Compute the signature of M=”Hello!” using the specified methods, where H(W)=last 4 bits of SHA256(W) for a binary string W. Also, compute the corresponding public keys and verify correctness of the signatures.
   a) RSA: n=323=17x19, private key=(323, 7^{-1} mod 288).
   b) ElGamal: q=103, α=11, private key X_A=35.
   c) Schnorr: p=103, q=17, a=72, private key = (103, 17, 72, 10)
   d) DSA: p=103, q=17, g=72, private key = (103, 17, 72, 7)

4) Use the DFT method to factor M=77 by choosing a=8, m=7, n=12. Use a tool, such as Matlab, to compute DFT. You need to show all steps of computation.

Part 2: Programming Problem

This programming problem is to simulate the bitcoin mining. Note that this is not the real bitcoin mining. It only verifies the difficulty of finding hash values with many leading zeros. Use Crypto++ for computing sha256.

I. Do sha2-256 (that is, sha256) on the following text, where the first row is the test sample:

Message (in ASCII)	Message digest (in Hex)
“Hello!”	334d016f755cd6dc58c53a86e183882f 8ec14f52fb05345887c8a5edd42c87b7
“Bitcoin is a cryptocurrency, a form of electronic cash.”	?

II. Mine cryptocurrency:

A. Build the blockchain in the following table until the requirement of leading zeros (in Hex) cannot be met. We start with the hash value Sha256("Bitcoin") = B4056DF6691F8DC72E56302DDAD345D65FEAD3EAD9299609A826E2344EB63AA4

# of leading zeros	Preimage = Previous hash (in Hex)+ Nonce (32 bits, in Hex)	Hash value (in Hex), with the specified leading zeros (in Hex)
0	B4056DF6691F8DC72E56302DDAD345D65FEAD3EAD9299609A826E2344EB63AA4 00000000	2767667C2AF3BE01EFAC4FB387EC27C10B9D3BEE9C5D48CFF4CFB9F523560B24
1	2767667C2AF3BE01EFAC4FB387EC27C10B9D3BEE9C5D48CFF4CFB9F523560B24 0000000A	0DE32E85C2AC9D96659D42C8A3EA3D2C05FDE384B468E6EFE062B6E21288CBCA
2	?	?
3	?	?
…	?	?

B. Submission: you need to upload two files: hashchain.cpp and out.txt, where out.txt contains the chain in the form (the number of leading zeros, previous hash, nonce, current hash, …):

```
0
B4056DF6691F8DC72E56302DDAD345D65FEAD3EAD9299609A826E2344EB63AA4
00000000
2767667C2AF3BE01EFAC4FB387EC27C10B9D3BEE9C5D48CFF4CFB9F523560B24
1
2767667C2AF3BE01EFAC4FB387EC27C10B9D3BEE9C5D48CFF4CFB9F523560B24
0000000A
0DE32E85C2AC9D96659D42C8A3EA3D2C05FDE384B468E6EFE062B6E21288CBCA
…
```

C. Grading: the more leading zeros your hash values have, the higher your grade is.

D. There is no on-site test due to the recent Covid-19 breakout.