

Homework 2 of CS549 Cryptography and Network Security

ASSIGNED: FEB 26TH, 2012

DUE DATE: 11:59PM, APRIL 13TH, 2012

EXTENSION DATE (WITH 10% DEDUCTION ON GRADING): 11:59PM, APRIL 20TH, 2012

SPRING 2012, CS DEPARTMENT, IIT

Please type your answer. You can have to upload the PDF file of your solution (for non-programming part) to IIT blackboard system. In addition, you could also print it and give TA the hardcopy.

Keep in mind that you have to upload the electronic file in PDF or Postscript format to the blackboard. Put your name on your solution and name your file as "lastname-HW2.pdf", where lastname is your last name.

Notice: We do not accept MS WORD file and an electronic submission is mandatory.

- (10 points) Assume that there is a block cipher, named XXX, that always encrypts block of $b = 72$ bits using key of $k = 60$ bits. Assume that we know that XXX will be broken even using simple brute-force attacking by guessing the encryption key. To enhance the security level, assume that someone proposes to use 2XXX by use of 2 encryptions of XXX with 2 different independently and randomly chosen keys. Prove that 2XXX does not provide a much stronger security than XXX. Prove in detail why this statement is true (you have to analyze in detail a method attacking 2XXX using time that is not much longer than the brute force attacking on XXX).
- (10 points) Continue from preceding question. Assume now that someone wants to use 3XXX to enhance the security by using 3 rounds of encryptions of XXX with 3 different and independent keys. Design a method that can attack 3XXX whose time complexity is with order of $2^{2k} = 2^{120}$ instead of naive complexity of 2^{3k} . In your method, how many pairs of plaintexts and ciphertexts do you need? What is the space complexity your method will need? Given this many pairs of plaintexts and ciphertexts, what is the probability that you will find the correct encryption key?
- (10 points) This problem is about Hill cipher system. Assume that in a Hill cipher system, the input alphabet is $\{0, 1, 2, \dots, 24, 25\}$. Assume that each time it encrypts m characters. The key K is then a $m \times m$ matrix over Z_{26} . For any input $x = (x_1, x_2, \dots, x_m)^T$, we compute the encryption as $y = K \cdot x \pmod{26}$.

Suppose that Oscar has learned that a plaintext

1, 2, 3, 4, 5, 6, 0, 3, 7

is encrypted by Alice as

16, 7, 3, 11, 6, 25, 23, 2, 24

and also Oscar knows that $m = 3$.

- What is the key K used by Alice?
 - If Oscar intercepts another ciphertext (2, 5, 7) encrypted by Alice using the same key, what is the plaintext corresponding to this ciphertext?
- (10 points) Prove that RSA algorithm works. In other words, prove that $(M^e)^d = M \pmod{n}$ for any message $M \in [0, n - 1]$. Here $n = p \cdot q$, $e \cdot d = 1 \pmod{\phi(n)}$ and p, q are large prime numbers.
 - (10 points) This question is about RSA again. Assume that Bob uses RSA and selects two "large" prime numbers $p = 101$ and $q = 73$.
How many possible public keys Bob from which Bob can choose?
Assume also that Bob uses a public encryption key $e = 91$. Alice sends Bob a message $M = 2008$. What will be the ciphertext received by Bob? Show the detailed procedure that Bob decrypts the received ciphertext.

6. (10 points) This question is about the Knapsack encryption system. Consider a super-increasing set $s = (3, 5, 10, 21, 43)$ and a "large" prime number $p = 97$ that is larger than the summation of all numbers in s . Let $x = 13$ be the secret key selected by a user Alice. What is the public key Alice should publish? If a user Bob wants to send a message with 10 bits as 1001011001, what is the encrypted message Bob should send to Alice? How Alice decrypt the ciphertext received from Bob?
7. (10 points) A common way to speed up the RSA decryption is to use the Chinese Remainder Theorem. Suppose that we want to decrypt of a ciphertext y and the decryption key is d and the modular $n = p \cdot q$, where p and q are two large prime numbers used by RSA. Let $d_p = d \bmod (p-1)$ and $d_q = d \bmod (q-1)$; and let $M_p = q^{-1} \bmod p$ and $M_q = p^{-1} \bmod q$.
Let $x_p = y^{d_p} \bmod p$ and $x_q = y^{d_q} \bmod q$. Let $x = M_p \cdot q \cdot x_p + M_q \cdot p \cdot x_q \bmod n$.
Prove that the computed x is indeed the original plaintext, i.e., $x = y^d \bmod n$.
Given $p = 1511$ and $q = 2003$ and $y = 152702$, use the above method to decrypt the ciphertext y when the decryption key is $d = 153$.
8. (15 points) Suppose Bob uses the DSA signature scheme with prime numbers $q = 101$, $p = 7879$, and the primitive root $g = 170$. And Bob selects his private key as $x = 75$, and publishes his public key as $y = g^x \bmod p = 4567$.
- Determine Bob's signature on a message with hashed value $h(M) = 5001$ when using a random number $k = 49$.
 - Assume Alice wants to verify this signature, show the procedure Alice used to verify the signature.
9. (15 points) Elgamal signature scheme works as follows. Let p be a large prime number and $g \in Z_p^*$ be a primitive element. Suppose Bob is using the ElGamal Signature Scheme, Bob picks a random integer $0 < a < p$ and publishes $\beta = g^a \bmod p$, but keep a secret. Bob signs a message m as follows
- Bob picks a random integer $k < p$, and computes (γ, δ) , where $\gamma = g^k \bmod p$,
 - Bob computes the hash value, say $x = h(m)$, of the message m , and computes

$$\delta = k^{-1}(x - a\gamma) \bmod (p-1).$$

The pair (γ, δ) is the signature of the message m .

When Alice receives the signature (γ, δ) for the message m , Alice also computes the hash value, say $x = h(m)$, of the message m . Alice then verifies the signature by checking if

$$\beta^\gamma \gamma^\delta = g^x \bmod p.$$

- Prove that Alice accepts the signature if the signature is indeed created by Bob.
 - Suppose Bob signs two messages x_1 and x_2 with signatures (γ, δ_1) and (γ, δ_2) , respectively. (The same value for γ occurs in both signatures.) Suppose also that $\gcd(\delta_1 - \delta_2, p-1) = 1$. Describe how the signature scheme can then be broken. In other words, how to find number a ?
10. (25 points) Assume that you are given an integer n , which is the production of two large prime numbers. You are also given two integers e and d such that $e \cdot d = 1 \bmod \phi(n)$. Notice that you do NOT know the value of $\phi(n)$. Implement a program to factorize n with high probability, using the information n , e and d . Your code should be able to deal with large integers as in HW1.
You have to submit the code of your program.