Homework 1

Due: 7:00PM, Thursday, 10/8/2015

Name: ID:

1. Random J. Protocol-Designer has been told to design a scheme to prevent messages from being modified by an intruder. Random J. decides to append to each message a hash of that message. Why doesn't this solve the problem?

   Anyone who knows which hash function is used can forge a message.

2. Suppose Alice, Bob, and Carol want to use secret key technology to authenticate each other. If they all used the same secret key K, then Bob could impersonate Carol to Alice (actually any of the three can impersonate the other to the third). Suppose instead that each had their own secret key, so Alice uses KA, Bob uses KB, and Carol uses KC. This means that each one, to prove his or her identity, responds to a challenge with a function of his or her secret key and the challenge. Is this more secure than having them all use the same secret key K?

   They still need to know each other's keys to do the verification, which means they can still impersonate the other to the third.

3. If you want nonrepudiation, would it be easier to use public or secret user keys?

   No. Public key or secret key are shared by more than one user, so it cannot achieve nonrepudiation.

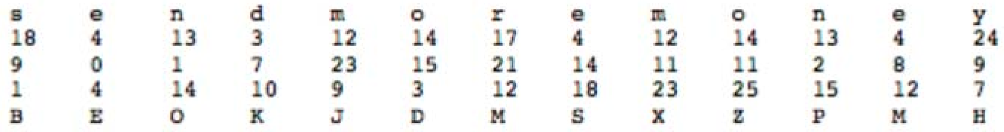
4. Assume a cryptographic algorithm that is linear in the length of the key to perform "good guy operations", e.g., encryption, decryption, key generation, integrity check generation, integrity check verification; and that it is exponential in the length of the key to perform "bad guy operations", e.g., brute force breaking. Suppose advances in computation make computers an order of magnitude faster. Does this work to the advantage of the good guys, the bad guys, or neither?

   If the good guys keep the same key size, then it is an advantage for the bad guys. But if the good guys double the key size, doubling their work while still taking the same amount of time as it used to, then it is much worse for the bad guys, since their work squares.

For example, suppose the good guys were using an n-bit key. With a computer twice as fast, they can use a 2n-bit key with the same performance, since doubling the length of the key just doubles their work. However, the bad guys have 2^n times as much work to do with a key twice as long, so it works to the advantage of the good guys.

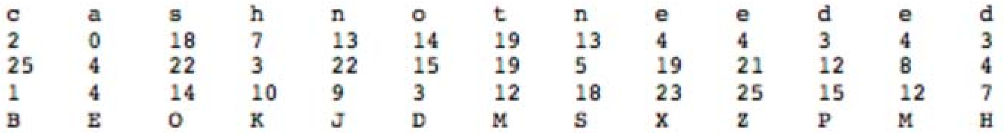
5. This problem explores the use of a one-time pad version of the Vigenere cipher. In this scheme, the key is a stream of random numbers between 0 and 26. For example, if the key is 3 19 5…, then the first letter of plaintext is encrypted with a shift of 3 letters, the second with a shift of 19 letters, and so on.
   a. Encrypt the plaintext sendmoremoney with the key stream 9 0 1 7 23 15 21 14 11 11 2 8 9.
   b. Using the ciphertext produced in part a, find a key so that the ciphertext decrypts to the plaintext cashnotneeded.

a.



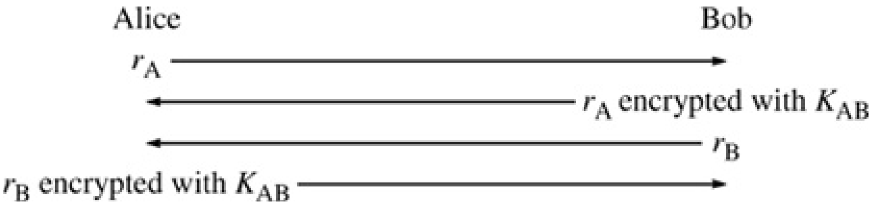
s	e	n	d	m	o	r	e	m	o	n	e	y
18	4	13	3	12	14	17	4	12	14	13	4	24
9	0	1	7	23	15	21	14	11	11	2	8	9
1	4	14	10	9	3	12	18	23	25	15	12	7
B	E	O	K	J	D	M	S	X	Z	P	M	H

b.



c	a	s	h	n	o	t	n	e	e	d	e	d
2	0	18	7	13	14	19	13	4	4	3	4	3
25	4	22	3	22	15	19	5	19	21	12	8	4
1	4	14	10	9	3	12	18	23	25	15	12	7
B	E	O	K	J	D	M	S	X	Z	P	M	H

6. How Alice and Bob authenticate each other by using secret key cryptography is shown as follows. What is wrong with this scheme if Alice can open two connections to Bob? (Hint: Alice can be authenticated without knowing the secret key K_{AB})



Alice does not need to know K_{AB}; when Bob challenges with r_B, Alice just opens a second connection to Bob and challenges him with r_B, then uses his response to respond to his first-construction challenge. She can abort the second connection.

7. What is the time and memory requirements as compared to single DES for both the good guys and the bad guys, and explain the technique for doing brute force against double DES in each of the following cases?
   a. the same key is used twice (encrypt with K, then encrypt with K again)
   b. same key is used twice (first use K in encrypt mode, then use it in decrypt mode)

   a. The time requirements for both the good guys and the bad guys double since they need to perform encryption twice with double DES. The memory requirements for them are the same as single DES.
   b. The time and memory requirements for the good guys double, since they need to perform one encryption and one decryption with double DES and store both the encryption and decryption algorithms. The time and memory requirements for the bad guys reduce to O(1), since now the ciphertext is identical to the plaintext.

8. How many DES keys, on the average, encrypt a particular plaintext block to a particular ciphertext block?

   There are 2^{56} possible keys and 2^{64} possible ciphertext blocks for a particular plaintext block. So only about $2^{56}/2^{64}=1/256$ of the possible ciphertext blocks can be obtained by with a DES key.

9. Can we use the ciphertext of the last block of CTR as the MAC of a message for integrity protection? Why?

   We cannot use the ciphertext of the last block of CTR as the MAC of a message for integrity protection. Because the last ciphertext block of CTR is computed independently to the previous blocks, it will not be affected by precedent plaintext or ciphertext blocks. The last ciphertext block c(n) is computed using IV+(n-1), secret key, and m(n). In other words, the last ciphertext block does not change even if an intermediate plaintext or ciphertext is modified, because there is no chain in encrypting each block to get the next one.