CSC 474 Network Security

TA’s reference solution to HW Assignment #2

1. (10 points) Problem 2 on page 57 in textbook.

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

2. (10 points) Problem 3 on page 57 in textbook.

Solution:
Without using the public key technology, they will still need to know each other’s key to do the verification, which means they can still forge each other’s messages.

3. (10 points) Problem 4 on page 57 in textbook.

Solution:
If it is easy to find two messages with the same message digest, this hash function will not be subject to strong collision resistance. You can forge a signature on any message with the same hash as one that has been legitimately signed.

4. (10 points) Problem 5 on page 58 in textbook.

Solution:
Alice doesn’t 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-connection challenge. She can abort the second connection.

5. (10 points) Give three examples of ancient ciphers. For each of them, you need to explain (a) what is the key, (b) the encryption procedure, and (c) the decryption procedure.

Solution:
Caesar cipher: Caesar cipher is one of the simplest and most widely known encryption techniques. It is a type of substitution cipher in which each letter in the plaintext is replaced by a letter some fixed number of positions down the alphabet. For example, given that the key is 3, the plaintext ‘a’ will be encrypted the letter ‘d’ after shifting the alphabet as the key. The decryption can be done the reverse way to encryption.

Vigenere cipher: Vigenere cipher is a method of encrypting alphabetic text by using a series of different Caesar ciphers based on the letters of keyword. It is a simple form of ployalphabetic substitution. Each row starts with a key letter. The remainder of the row holds the letters A to Z in shifted order. For instance, if the keyword is LEMON and the plaintext to be encrypted is ATTACK, the ciphertext will be LXFOPV (i.e., Plaintext: ATTACK, KEY: LEMONL, ciphertext: LXFOPV).
Permutation cipher: Permutation cipher is a transposition cipher in which the key is a permutation. To apply a cipher, a random permutation of size $e$ is generated. The plaintext is then broken into segments of size $e$ and the letters within that segment are permuted according to this key. For example, if given permutation key is (1 2 3) and plaintext is PLAINTEXT, the ciphertext will be PIELNXATT.

<table>
<thead>
<tr>
<th>Encryption</th>
<th>Decryption</th>
</tr>
</thead>
<tbody>
<tr>
<td>Plaintext: PLAINTEXT</td>
<td>Ciphertext: PIELNXATT</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>P</td>
<td>L</td>
</tr>
<tr>
<td>I</td>
<td>N</td>
</tr>
<tr>
<td>E</td>
<td>X</td>
</tr>
<tr>
<td>Ciphertext: PIELNXATT</td>
<td>Plaintext: PLAINTEXT</td>
</tr>
</tbody>
</table>

| 1 | 2 | 3 |
| P | I | E |
| L | N | X |
| A | T | T |

Any example of ancient cipher schemes is acceptable to get full credit.

More information:
http://en.wikipedia.org/wiki/Classical_cipher
http://en.wikipedia.org/wiki/Permutation_cipher
http://en.wikipedia.org/wiki/Vigen%C3%A8re_cipher
http://en.wikipedia.org/wiki/Caesar_cipher

6. (10 points) Explain what is one-time pad. You should explain (a) what is the key, (b) the encryption procedure, and (c) the decryption procedure.

**Solution:**
One-time pad (OTP) is a type of encryption, which has been proven to be impossible to crack if used correctly. Each bit or character from the plaintext is encrypted by a modular addition with a bit or character from a secret random key (or pad) of the same length as the plaintext, resulting in a ciphertext. The key of OTP is a randomly generated bit string. It is only used once and then discarded. In order to encrypt a plaintext, it performs XOR operation with the plaintext and one-time key. To decrypt the ciphertext, it performs XOR operation with the ciphertext and the same key again.

http://en.wikipedia.org/wiki/One-time_pad

7. (10 points) Explain what is a "product cipher".

**Solution:**
A product cipher combines two or more transformations in a manner intending that the resulting cipher is more secure than the individual components to make it resistant to cryptanalysis. The product cipher combines a sequence of simple transformations such as substitution, permutation, and modular arithmetic.

http://en.wikipedia.org/wiki/Product_cipher
8. (10 points) Explain what is a stream cipher, and what is a block cipher.

Solution:
Stream cipher: A stream cipher is a symmetric key cipher where plaintext bits are combined with a pseudorandom cipher bit stream (keystream), typically by XOR operation. In a stream cipher the plaintext digits are encrypted one at a time, and the transformation of successive digits varies during the encryption.


Block cipher: A block cipher is a symmetric key cipher operating on fixed-length groups of bits, called blocks, with an unvarying transformation. A block cipher encryption algorithm might take (for example) a 128-bit block of plaintext as input, and output a corresponding 128-bit block of ciphertext. The exact transformation is controlled using a second input — the secret key. Decryption is similar: the decryption algorithm takes, in this example, a 128-bit block of ciphertext together with the secret key, and yields the original 128-bit block of plaintext.

http://en.wikipedia.org/wiki/Block_cipher

9. (20 points) Alice is establishing an account with Bob, a discount on-line broker. She wants her trading to be private. Since both Alice and Bob have heard that the one-time pad (a.k.a. the Vernam cipher) is a very secure cryptosystem, they generate a 96-bit-long random pad K for Alice to use in the future for encrypting all her buy/sell orders to Bob. They agree on the following format for each order: First, Alice writes down a single character, either `B' for `Buy' or `S' for `Sell.' Then she puts a single space, followed by a five-digit decimal number for the number of shares she wants to buy or sell (if she doesn't need to use all five digits, she puts zeros in the front). Finally, she puts another space followed by the four-letter ticker symbol of the stock she wants to buy or sell (if not all four letters are needed, she puts spaces in the front). Thus, for example, ``B 00100 MSFT'' means "Buy 100 shares of Microsoft" and "S 25000 AOL" means "Sell 25,000 shares of AOL Time Warner." Whenever Alice wants to send an order to Bob, she puts her order in the above format, converts the resulting 12-character string to ASCII to get 12 bytes (=96 bits), and encrypts it using the Vernam cipher with the agreed-upon key K.

Describe at least one attack against this scheme. You may have certain assumptions (e.g., you may get a copy of a plaintext message). In your submission, you need to clearly state your assumption(s), describe the procedure of your attack, and give an example illustrating your idea. (Hints: there are four types of cryptanalysis techniques: plaintext-only, known plaintext, chosen plaintext, and chosen ciphertext.)

Solution:
One possible solution is as follows:
Let define the 96-random pad as \( k_{\{0\}} \), on which Bob and Alice agreed upon before the system initialization. Let \( F: \{0,1\}^{96} \rightarrow \{0,1\}^{96} \) is a public one-way function (e.g. derived from SHA-1). Further, encryption and decryption are performed with XOR operation. Last, \( (P_{\{j\}} - C_{\{j\}}) \) corresponds j'th (plaintext-ciphertext) pair associated with j'th transaction.

Assume that Alice and Bob performs their transaction as follows:
- a) For each transaction j, Alice updates the one time key as \( k_{\{j+1\}} = F(k_{\{j\}}) \), and deletes \( k_{\{j\}} \).
b) Alice encrypts $P_{j+1}$ as $C_{j+1} = P_{j+1} \ XOR \ k_{j+1}$.

c) Bob can decrypt $C_{j+1}$ by following the similar procedure that Alice used. In this case, assume that the adversary obtained the plaintext associated with transaction $x$. The adversary then can obtain the secret key of this transaction as:

$$k_{x} = P_{x} \ XOR \ C_{x},$$

where $C_{x}$ can be observed over communication line simply.

After this point, for each transaction $(j \geq x)$, the adversary can compute the corresponding key and decrypt the corresponding ciphertext. Therefore, the scheme is broken starting from transaction $x$.

Note: In this question, answers mentioning simple XOR operation without making *explicit and clear assumption list* will not be given full credit.