Year and Semester	2018 FALL
Course Number	CS-336
Course Title	Intro. to Information Assurance
Work Number	HA-03
Work Name	Selected Problems from Textbook: Chapter 02 (Stallings and Brown, 2015)
Work Version	Version 1
Long Date	Sunday, 9 September 2018
Author(s) Name(s)	Zane Durkin

Abstract

In this article I will be going over a few problems from the Textbook [1]. I'll be solving 4 problems from the end of Chapter 02, and I'll being going through review questions 2.1 to 2.13.

1 Problems from the end of Chapter 02

1.1 Problem 2.1 [1, p. 68]

Although this process would work to verify that both participating parties have the same, or different, secrete key, this process creates a major vulnerability for the security of the secret key. Since the random key, and the XOR of the random key and secret key are being sent across a channel unencrypted, it is possible to determine what the secret key is by preforming an XOR of the random string and the XOR of the random and secret string.

1.2 Problem 2.2 [1, p. 68]

(a) For the encryption of the message using double transposition I will be ignoring spaces and other punctuation on the key works and message, and I will count douplicate letters in the keywords from left to right. The plaintext string is written horizontally across a matrix, with each column corisponding to a letter in the key word.

2	10	13	8	12	7	4	11	1	9	5	6	3
c	r	y	p	t	o	g	r	a	p	h	i	c
b	e	a	t	t	h	e	t	h	i	r	d	p
i	l	l	a	r	f	r	o	m	t	h	e	l
e	f	t	o	u	t	s	i	d	e	t	h	e
l	y	c	e	u	m	t	h	e	a	t	r	e
t	o	n	i	g	h	t	a	t	s	e	v	e
n	i	f	y	o	u	a	r	e	d	i	s	t
r	u	s	t	f	u	l	b	r	i	n	g	t
w	o	f	r	i	e	n	d	s				

Next is to take the columns and align them horizontally to make a new string of text. This new string will be aligned in the same fasion of the previous set, but with a new keyword to number the columns.

6	2	11	14	7	8	5	10	3	1	13	9	4	12	15
n	e	t	w	o	r	k	s	e	c	u	r	i	t	y
i	r	e	a	l	t	f	m	p	r	f	i	t	i	b
e	d	t	e	d	s	t	t	v	i	o	u	r	t	i
u	d	r	n	h	w	n	h	r	o	f	t	e	t	h
a	h	t	e	e	e	u	s	a	c	l	t	r	e	y
u	s	t	n	e	i	b	f	d	l	h	t	t	o	a
n	a	s	y	i	g	e	s	h	o	f	r	e	e	t
m	e	i	l	r	l	g	t	u	o					

Now we can take the text horizontally and send it as our encrypted message. Our encrypted text is:
irealtfmprfitibedtedsttviourtiudrnhwnhroftethahteeeusacltreyustneibfdlhttoanasyigeshofreetmeilrlgtuo

(b) Next, to decypt the text, we will follow the procedure in reverse order, as shown.
starting with the encrypted text, we will use the second keyword to align the columns.

6	2	11	14	7	8	5	10	3	1	13	9	4	12	15
n	e	t	w	o	r	k	s	e	c	u	r	i	t	y
i	r	e	a	l	t	f	m	p	r	f	i	t	i	b
e	d	t	e	d	s	t	t	v	i	o	u	r	t	i
u	d	r	n	h	w	n	h	r	o	f	t	e	t	h
a	h	t	e	e	e	u	s	a	c	l	t	r	e	y
u	s	t	n	e	i	b	f	d	l	h	t	t	o	a
n	a	s	y	i	g	e	s	h	o	f	r	e	e	t
m	e	i	l	r	l	g	t	u	o					

Now we can use the first keyword to finish the decryption

2	10	13	8	12	7	4	11	1	9	5	6	3
c	r	y	p	t	o	g	r	a	p	h	i	c
b	e	a	t	t	h	e	t	h	i	r	d	p
i	l	l	a	r	f	r	o	m	t	h	e	l
e	f	t	o	u	t	s	i	d	e	t	h	e
l	y	c	e	u	m	t	h	e	a	t	r	e
t	o	n	i	g	h	t	a	t	s	e	v	e
n	i	f	y	o	u	a	r	e	d	i	s	t
r	u	s	t	f	u	l	b	r	i	n	g	t
w	o	f	r	i	e	n	d	s				

(c) This technique would be useful when you need to encrypt a message to send over a non-secure network, and when both parties have already securely shared the keywords prior to the encryption of the message.

1.3 Problem 2.6 [1, p. 69]

Given a hash function that maps an arbitrary bit length message into an n-bit hash value, it is false that for every x, x' where x $\neq$ x' there will always be H(x) $\neq$ H(x'). This is because an arbitratry bit length message that is larger than the fixed n-bit hash output would require that there is some collisons in hash values. The hashing function must loss some of the data of the longer message inorder to match the fixed length hash output, this loss in data would allow two different messages to have the same hash output after the extra data has been lost.

1.4 Problem 2.3 [1, p. 69]

(a) the decryption equation would be an inverse of the encryption equation, as shown below
$P = (C \ominus K_1) \oplus K_0$
where P = plaintext, C = ciphertext, K = secret key, K_1 = rightmost 64 bits of K, K_0 = leftmost 64 bits of K, $\oplus$ = bitwise exlusive or, and $\ominus$ = subtraction mod 2^{64}

2 Review questions from Chapter 02

2.1 Problem 2.1 [1, p. 68]

The essential ingredients of a symmetric cipher are:

- Plain Text
- The Encryption Algorithm
- The Secret Key
- The Cipher Text
- The Decryption Algorithm

2.2 Problem 2.2 [1, p. 68]

Only a single key is required for two people to communicate via a symmetric cipher.

2.3 Problem 2.3 [1, p. 68]

The two principal requirements for the secure use of a symmetric encryption are

- The secret key must be shared privately between only the two members before communication.
- The algorithm must be strong enough to not be reversed without the key

2.4 Problem 2.4 [1, p. 68]

The Three approaches to message authentication are:

- The message is sent with a message authentication tag to all machines that need the data. There is one machine on the network that is responsible for authenticating the message using the attached message authentication tag. If there is a problem with the message, the authenticating device will broadcast to all machines that the message was not authenticated.
- For systems that have a heavy load and cannot authenticate each message, they can authenticate on a selective basis. So messages are selected at random to be authenticated.
- Authentication with the program in plaintext. To reduce the resources required to decrypt the program every time. The message could contain an additional authentication tag to be used fro integrity checking whenever needed.

2.5 Problem 2.5 [1, p. 68]

A Message Authentication code is a one way hashing of a message using a secret key that has been shared before hand in a secure fashion. The MAC is attached with the message, and authenticity can be assured by the client by re-hashing the message with the same secret key and then comparing with the MAC.

2.6 Problem 2.6 [1, p. 68]

Figure 2.3 is a diagram of the use of a Message Authentication Code. The figure shows that the message is combined with a key 'K' to produce a MAC. The MAC is then attached to the end of the message and transmitted to the receiver. The receiver then spates the message from the MAC and re-generated the MAC with the message and key. After generating the new MAC, it is compared with the given MAC to verify the message's authenticity.

2.7 Problem 2.7 [1, p. 68]

In order for a hash function to be useful for message authentication it must have the following functions:

- It can be applied to any block size of data.
- It produces a fixed length output
- H(x) is relatively easy to compute for any x.
- It is computationally in feasible to find a value x that will match a given h in H(x)=h
- When H(x)=H(y) is computationally in feasible to not have y=x. (second preimage resistant)
- For any value of x, it is computationally in feasible to find a value y that will match H(x)=H(y). (collision resistant)

2.8 Problem 2.8 [1, p. 68]

The principle ingredients of public-key cryptography are

- Plain text
- Encryption Algorithm
- Public and Private key
- Cipher text
- Decryption Algorithm

2.9 Problem 2.9 [1, p. 68]

Three uses of a public key cryptosystem are:

- Digital Signature. A public key cryptosystem can be used to sign a message to verify that the message came from the person who has the private key.
- Symmetric key Distribution. Public key cryptosystems can be used to confidentially distribute symmetric keys by encrypting the message with the public key, so only the owner of the private key can view the message.
- Encryption of Secret keys. The use of the Private key to encrypt a message can be done. The encrypted message can then be decrypted by the public key pair.

2.10 Problem 2.10 [1, p. 68]

The difference between a private key and a secret key is that a private key is used in asymetric encryptions and is often the counter part to a public key. While a secret key is on a key that is only known by participating parties in a symmetric encryption.

2.11 Problem 2.11 [1, p. 68]

A digital signature is the use of a private key to encrypt a hash of a message to prove that the message was created by the owner of the private key. This is useful for verifying that a message has not been forged by a third party, given that the private key has been properly secured.

2.12 Problem 2.12 [1, p. 68]

A public key certificate is a list of digital signatures that creates a chain of trust. The certificate verifies that a public key is owned by the proper party, because the public key is signed by a third party who has already verified the owner ship of the public key. The certificate is ultimatley trusted if you trust the Certificate authority at the end of the chain, or anyone prior on the chain.

2.13 Problem 2.13 [1, p. 68]

Public key encryption is useful for distributing a secret key because it gives a way for a secret key to be shared without a prior exchange between the two parties. The parties can encrypt a secret key using the other's public key (after verifing the public keys using a certificate authority). This ensures that secret key is kept confidencital while it is being exchanged.

References

[1] Stallings, W., and Brown, L. *Computer Security Principles and Practice*, 3 ed. Pearson Education, One Lake Street, Upper Saddle River, New Jersy 07458, 7 2014.