**Information Security Technologies COMP607**

**Assignment 1 (20%)**

Instructions:

1. Type (preferable) or write your answers neatly on A4 paper. The assignment must be done in English. It must be your own work. Show all workings. Do not copy material from anywhere without appropriate referencing of the source as it will be penalised.

2. As far as possible, take screenshots of your work and paste into your submission to show evidence of your work.

3. Begin each question on a fresh page, if possible.

4. Submit your answers (in pdf format), files, etc. in *https://canvas.aut.ac.nz/Assignments.*

5. Marks will be deducted for untidy work.

6. Date due: refer to Canvas

Notes: Some of the questions require you do the tasks using Linux commands. If you don't have Linux, you can use the server at *scopius.aut.ac.nz*. To do this, open a Windows Powershell and at the $ prompt, type ssh username@scopius.aut.ac.nz. Your username is your AUT login name and password is your day of birth, e.g. *01apr*

To copy the file from your directory in the Linux server to your local PC, you can use the WinSCP application.

Where required, to access files in *https://scopius.aut.ac.nz,* username/password: *student/student*

You may also need to use a scientific calculator capable of doing modulo math. You can use the genius math tool in the scopius server by typing at the $ prompt, genius

Questions/Tasks:

**1. (a) The following cipher text is obtained using a rail-fence method. Using brute force, determine the key and the plaintext message in English? (2 marks)**

**AAEHDSGNMBTTAOHTODESTRNOAIOIEGB**

-The decrypted message is:

A PHYSICIAN SHOULD NEVER SELF-DESTRUCT, FOR THIS ACTION CAN CAUSE HARM TO PATIENTS.

**(b) Encrypt the following plaintext using the Vignere cipher method, using modulo 26 addition (preferable) or the Vignere table. (3 marks)**

***KNOWLEDGE IS NEVER WASTED***

**Key: *EQUATORIAL***

**-** Plaintext: [10, 13, 4, 14, 7, 5, 22, 24, 9, 14, 22, 22, 23, 15, 18, 20, 25, 1, 18, 23, 20, 24, 21, 13, 23, 20, 16, 20, 17, 25, 20, 15, 18, 17, 23, 15, 24, 22, 17, 17, 17, 24, 19, 23, 18, 15, 25, 18, 16, 24, 18, 23, 23, 19, 22, 15, 23, 15, 17, 22, 23, 24, 19, 18, 19, 16, 21, 16, 21, 23, 15, 18, 22, 19, 18, 15, 17, 19, 23, 20, 18, 15, 21, 18, 22, 16, 15, 23, 19, 17, 15, 22, 15, 15, 22, 24, 22, 16, 21, 23, 24, 15, 23, 18, 24, 15, 17, 24, 23, 22, 18, 24, 24, 17, 15, 18, 16, 20, 17, 20, 19, 19, 23, 18, 24, 17, 21, 22, 21, 18, 21, 24, 24, 23, 19, 18, 23, 20, 23, 23, 20, 21, 20, 24, 24, 20, 21, 20, 22, 21, 21, 20, 22, 20, 24, 21, 23, 22, 24, 22, 23, 23, 23, 24, 23, 22, 24, 22, 21, 22, 21, 24, 24, 23, 24, 24, 22, 21, 21, 21, 24, 21, 22, 23, 22, 22, 23, 24, 23, 22, 21, 24, 24, 23, 24, 23, 23, 22, 22, 24, 24, 23, 22, 24, 23, 22, 23, 23, 22, 23, 24, 22, 22, 22, 24, 23, 24, 22, 23, 22, 24, 23, 22]

(c). Using a text editor, create a textfile called *secretLetter.txt* containing the text

*IF YOU ARE STANDING STRAIGHT, DON'T WORRY IF YOUR SHADOW IS CROOKED.*

Encrypt this file using the openssl toolset with your family nameas the encryption key, using each of the following methods:

1. AES 192 bit key, ECB mode, name the encrypted file s*ecretLetter\_aes\_ecb.enc*

My encryption key is Vo

A blue screen with white text

Description automatically generated

1. DES OFB mode, name the encrypted file *secretLetter\_des\_ofb.enc*

A blue screen with white text

Description automatically generated

Take screenshots of your work.

Submit screenshots and both files. (5 marks)

2. Consider a cryptosystem where the user enters a key in the form of a password.

1. **Assume a password consists of 10 Latin characters, where each one is encoded using the ASCII scheme (7 bits per character). What is the size of the key space? (2 marks)**

- The size of key space is 2^7\*2^7\*2^7\*2^7\*2^7\*2^7\*2^7\*2^7 = 2^56

**b. What is the corresponding key length in bits? (2 marks)**

**-** The corresponding key length in bits its : 7 \* 8 = 56 bits (7=bits, 8=8-letter long)

**c. Assume that most users use only 26 lowercase letters from the alphabet instead of the full 7 bits of the ASCII encoding. What is the corresponding key length in bits in this case? (2 marks)**

**-** In order to be able to build a key for a password with 8 letters, where each letter is one of the 26 lowercase letters, the key length in bits is: 5 × 8 = 40 𝑏𝑖𝑡𝑠

**d. At least how many characters are required for a password in order to generate a key length of 128 bits in case of letters consisting of (4 marks)**

**(i). 7- bit characters?**

**-** We have password with n-letter long:

2^7 × 2^7 × . . .× 2^7 (𝑛 𝑡𝑖𝑚𝑒𝑠) = 2^7𝑛

total different options available for the key space is 2^128  
=> 2^7𝑛=2^128 <=> 7𝑛=128

<=> 𝑛 = 128/7 ≈ 18.286 characters < 19 characters

**(ii). 26 lowercase letters from the alphabet?**

- Total different options available for the key space is 2^5𝑛

=> 2^5𝑛 = 2^128 <=> 5𝑛 = 128 <=> 𝑛=128/5 = 25.6 letters < 26 letters

3. The following ASCII bits (8 bits per character) is obtained using a stream cipher to encrypt an English plaintext message. (The spaces are only inserted for readability). The encryption key is a single alphabet character (8 bits in ASCII). Using frequency analysis, or otherwise, obtain the plaintext. State and explain the weakness in the way this cipher is used. (10 marks)

00010111 00011111 00011111 00001110 00011011 00001110 00011111 00010110 00011111 00001100 00011111 00010100 00011011 00010111

We can know that the encryption key is 8-bit long, so every 8 bits in the cipher is the result of performing a XOR operation between every 8 bits in the with the encryption key.

The most common block of 8 bits in the cipher corresponds to the most common 8 bits in ASCII of the most frequent letter in the plaintext. From the cipher, the most common block of 8 bits is he 00011111 (appeared 5 times in total) and we can transfer it to an English letter, the letter e.

The 8-bit ASCII representation of letter e is: 01100101

Now we can assume that the encryption key is 01111010, we can try to find the plaintext by reversing the XOR operation.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Ciper | 00010111 | 00011111 | 00011111 | 00001110 | 00011011 | 00001110 | 00011111 | 00010110 | 00011111 | 00001100 | 00011111 | 00010100 | 00011011 | 00010111 |
| Encryption key | 01111010 | 01111010 | 01111010 | 01111010 | 01111010 | 01111010 | 01111010 | 01111010 | 01111010 | 01111010 | 01111010 | 01111010 | 01111010 | 01111010 |
| Plaintext | 01101101 | 01100101 | 01100101 | 01110100 | 01100001 | 01110100 | 01100101 | 01101100 | 01100101 | 01110110 | 01100101 | 01101110 | 01100001 | 01101101 |
| Result | m | e | e | t | a | t | e | l | e | v | e | n | a | m |

So the plaintext is: meet at eleven am.

4. Use md5 to check for file integrity.

Using your browser, go to [https://scopius.aut.ac.nz](https://156.62.140.124/). Click on Information Securities Technologies, Software directory. Use *student* for both username and password.

There are two *putty-0.70* files, one of which is corrupted. The md5 sum of the good file is also available on the website.

Download both versions of the putty-0.70 files and identify the good copy.

(a) Run the md5sum on the downloaded file and save the output to a file called putty.md5 show the screenshot of the content of this md5 file.

$ md5sum puttyputty-0.70-installer.msi

(6 marks)

(b) Research how to use md5 to check the integrity of files. Check which file is the good copy. Show screen shots of your work. (4 marks)

5. There are two files in Scopius server in *home/pub* folder (and also in *https://scopius.aut.ac.nz*) called n*otice1.txt* and *notice2.txt* and their related HMACs. One of the notices is fake. You know that the authentic copy was made by the person who has the secret key ***comp607*** shared with you. You are required to determine which copy is authentic.

Change to your working directory and copy theses files into it.

$ cp /home/pub/notice\*.\* .

1. Display both files. Paste the screenshots.

A blue screen with white text

Description automatically generated

(b) Use openssl tools to determine their HMACs and determine which is the authentic one. First, generate the HMAC for each one using the shared key, ***comp607***

$ openssl dgst -hmac shared*key* notice1.txt

Compare with the given HMACs, *notice1.hmac.txt* and *notice2.hmac.txt*

$ cat notice1.hmac.txt

Which notice is the authentic version? (10 marks)

A computer screen shot of a blue screen

Description automatically generated

Compare the created HMACs with the given HMACs, the notice2.txt file is the authentic

6. In your Linux working directory, create a text file using one of the following:

Using a text editor such as pico enter or copy and paste some text into it, and and save.

$ pico test1.txt

Alternatively you can do as follows:

$ echo “This is some text file” > test1.txt

View the file:

$ cat test1.txt

A blue screen with white text

Description automatically generated

(a) Obtain the md5 hash of your file: (To get help: $ md5sum -h)

$ md5sum test1.txt

 (2 marks)

(b) Make a small change in your file such as adding a space, dot, etc.

Save the file a with a new name, e.g. "test2.txt" and obtain the new md5 hash. Compare them. Are they same, different, or very different? Do this visually:

$ cat test1.txt

$ cat test2.txt

A blue screen with white text

Description automatically generated (4 marks)

(c) Make an MD5 integrity check hash for the file test1.txt

$ md5sum test1.txt > test1md5.txt

Verify the integrity of test1.txt:

$ md5sum -c test1md5.txt

A blue screen with white text

Description automatically generated (4 marks)

7. RSA algorithm. *Alice* wish to send a message *M = 513* to *Bob* by encrypting it with *Bob's* public key. Use the following parameters to obtain the RSA keys for *Bob*. Assume that *Alice* is able to obtain *Bob’s* public key securely.

a). Use *p* = 23, *q* = 2*9,*  obtain suitable values for *Bob’s* private key *d* and public key *e* exponents. What is *Bob’s* public key?

What is ciphertext that *Alice* obtains by encrypting *M* using *Bob’s* public key?

Show how *Bob* can decrypt this ciphertext correctly using his private key. (5 marks)

We have: 𝑛 = 𝑝 × 𝑞 = 23 × 29 = 667

Totient function : 𝜙(𝑛) = (𝑝 − 1)(𝑞 − 1) = 22 × 28 = 616

Choose e such that 1 < e < ϕ(n) and e is coprime with ϕ(n) and n => choose e =7

Determine d as d ≡ e⁻¹ (mod ϕ(n)) => d=263 => Bob’s public key is (e, n) = (7, 667)

By using the Bob public key, alice now can encrypt her M message=513

Ciphertext c is computed as c ≡ M^e (mod n) = 513^7 mod 667 = 166

So Bob can decrypt this ciphertext correctly using his private key d buy calculating M ≡ c^d (mod n) = 166^263 mod 667 = 513

b). Use *p* = 1*1*, *q* =2*3*, *M* = 10*9*. Calculate *Bob’s* private and public keys exponents *d* and *e*. The public key exponent *e* should be a small number.

Bob 'signs' the message *M* by encrypting it with his private key. What is the ‘signature’ *s*, obtained by *Bob*?

When *Alice* obtains the message *M* and the signature *s*, show how *Alice* would verify the signature. (5 marks)

We have: n = p\*q = 11\*23 = 253

Totient function : ϕ(n) = (p-1)\*(q-1) = 10\*22 = 220

Choose e such that 1 < e < ϕ(n) and e is coprime with ϕ(n) and n => choose e = 3

Determine d as d ≡ e⁻¹ (mod ϕ(n)) => d = 147

=> Bob’s public key is (e, n) = (3, 253) and private key is d = 147

8. Describe how *Alice* and *Bob* are able to exchange a secret key using the DH algorithm.

(a) Demonstrate the process by using generator *g=2*, and prime modulus *n=*  4787

Alice and Bob generate his/her private key and then compute the public keys  
+ Alice: Suppose Alice chooses her private key 𝑎 = 4500 < 𝑛 = 4787, Alice’s public key is 𝐴 =24500 (𝑚𝑜𝑑 4787) = 2666   
+ Bob: Suppose Bob chooses his private key 𝑏 = 2750 < 𝑛 = 4787, Bob’s public key is 𝐵 =22750 (𝑚𝑜𝑑 4787) = 2679

Alice and Bob exchange their public keys, Bob obtains 𝐴 = 2666 from Alice and Alice obtains 𝐵 = 2679 from Bob   
Alice and Bob calculates the shared secret key 𝐾𝑠  
+ Alice: 𝐾𝑠 = 𝐾𝐴𝑠 = 26794500 (𝑚𝑜𝑑 4787) = 199

+ Bob: 𝐾𝑠 = 𝐾𝐵𝑠 = 26662750 (𝑚𝑜𝑑 4787) = 199   
+ The shared secret key for both parties is 𝐾𝑠 = 199

1. Which party, Alice, Bob, both, or none, can determine the value of the shared key?(10 marks)

-The value of the shared key is obtained through a set of calculations, and is equal to the remainder of 𝐺 𝑏𝑎 when divided by 𝑝.

- With the permission of Alice and Bob before the exchange, which are 𝑝, the public prime modulus and 𝐺, the generator. The value of 𝐺 𝑏𝑎 depends on the value 𝑎 and 𝑏, which are the private keys of Alice and Bob and only known by each of them. Because of that, the value of the shared key can only be determined by both parties, Alice and Bob, as the private key of each is required for the calculations to obtain the shared key.

9. The DH algorithm can also be used for encryption as well using the ElGamal scheme. Demonstrate this encryption scheme using a numerical example as follows.

*Alice* wish to encrypt a secret message, *M = 215* to *Bob*. They have chosen the parammeters and private keys as follows:

*Bob*: private key *b = 231*, generator *G=2*, prime modulus *p = 443*.

*Alice*: private key *a = 198*

*D*emonstrate how the scheme works by showing what each party computes and sends to each other, showing clearly the cipher texts, and the decrypted messages.

1. using the above numbers for *M, a, b*

**- Step 1**: Bob defines 𝐺 = 2, 𝑝 = 443, secret key 𝑏 = 231. Then he computes his public key:

𝐵 = 𝐺 𝑏 (𝑚𝑜𝑑 𝑝) = 2 231 (𝑚𝑜𝑑 443) = 305

Bob then sends his public key:

k B public = 〈𝐺, 𝑝, 𝐵〉 = 〈2, 443, 305〉

**- Step 2**: Alice selects her private key 𝑎 = 198. Then Alice computes her public key:

𝐴 = 𝐺 𝑎 (𝑚𝑜𝑑 𝑝) = 2 198 (𝑚𝑜𝑑 443) = 144

After that she computes the encryption key:

𝐾𝑠 = 𝐵 𝑎 ≡ 𝐺 𝑏𝑎 (𝑚𝑜𝑑 𝑝) = 305198 (𝑚𝑜𝑑 443) = 321

**- Step 3**: Alice encrypts the message 𝑀 = 215, obtaining the cipher:

𝐶 = 𝑀 × 𝐾𝑠 (𝑚𝑜𝑑 𝑝) = 215 × 321 (𝑚𝑜𝑑 443) = 350

Alice then sends her public key: 𝐴 = 144 and the cipher text: 𝐶 = 350 to Bob

**- Step 4**: Bob derives the encryption key by the calculation:

𝐾𝑠 = 𝐴 𝑏 (𝑚𝑜𝑑 𝑝) = 144231 (𝑚𝑜𝑑 443) = 321

Bob then decrypts to cipher to receive the message:

𝐶 × 𝐾𝑠 −1 (𝑚𝑜𝑑 𝑝) = 350 × 374 (𝑚𝑜𝑑 443) = 215 = 𝑀

The message has successfully been decrypted.

The message is 𝑀 = 215, the cipher is 𝐶 = 350

(ii) using your own choice of numbers for *M, a, b* (10 marks)