**Information Security Technologies COMP607**  
**Assignment 1 (20%)**

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 name as the encryption key,  
using each of the following methods:

IST2024\_assig1  
(i) AES 192 bit key, ECB mode, name the encrypted file secretLetter\_aes\_ecb.enc  
(ii) DES OFB mode, name the encrypted file secretLetter\_des\_ofb.enc  
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.**  
**a. 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 corresponing key lenght 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 lettes, 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.

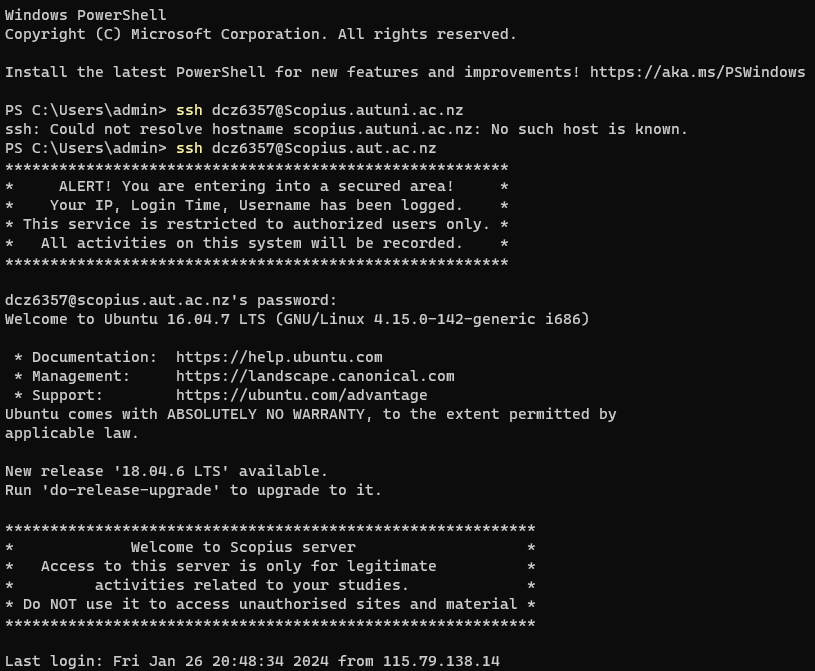
The most common block of 8 bits in the cipher corresponds to the most common 8 bits in ASCII of the

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 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 |

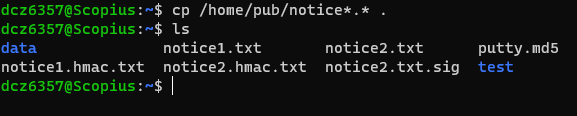
4. Use md5 to check for file integrity.  
Using your browser, go to <https://scopius.aut.ac.nz>. Click on Information  
Securities Technologies, Software directory. Use student for both  
username and password.  
There are two putty-0.70 files, one of which is corrupted. The md5 sum of the good file 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 notice1.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.  
Which notice is the authentic version? (10 marks)

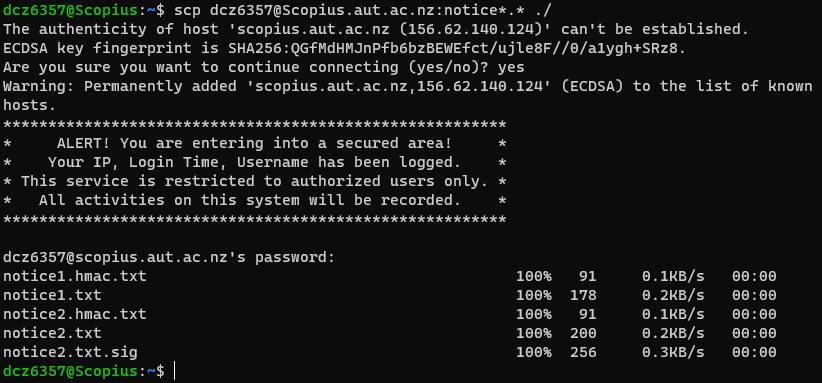
First, login to Scopius server:



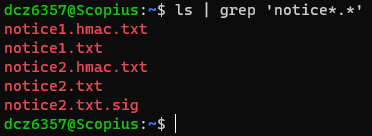
Copy all the files into folder of my account :



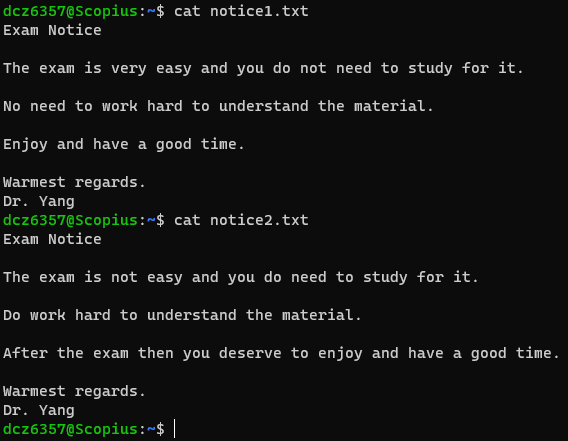
Use scp command to copy from server to my computer :



Check if the content inside my computer :

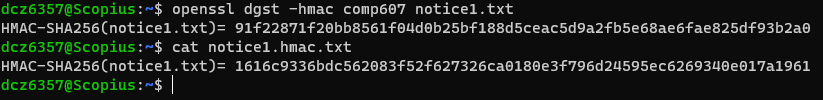


**(a) Display both files. Paste the screenshots.**

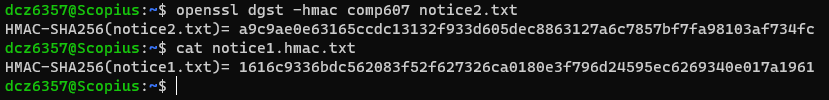


**(b) Use openssl tools to determine their HMACs and determine which is the authentic one. First, generate the HMAC for each one using the shared key, comp607. Compare with the given HMACs, notice1.hmac.txt and notice2.hmac.txt**

Generate HMAC for notice1.txt :

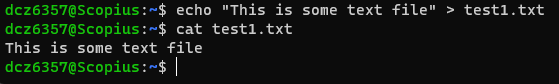


Generate HMAC for notice2.txt :

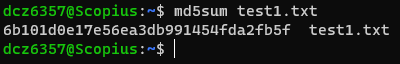


**6. In your Linux working directory, create a text file using one of the following:**  
**Using a text editor such as pico enter or copy and paste some text into it, and and save.**

Create file test1.txt :

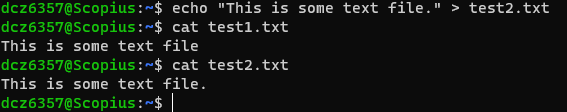


**(a) Obtain the md5 hash of your file: (To get help: $ md5sum -h)**  
**$ md5sum test1.txt (2 marks)**

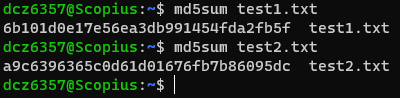


**(b) Make a small change in your file such as adding a space, dot, etc.**  
**Save the file a with a new name, e.g. "test2.txt" and obtain the new md5 hash.**  
**Compare them. Are they same, different, or very different?**

**Do this visually:**  
**$ cat test1.txt**  
**$ cat test2.txt (4 marks)**

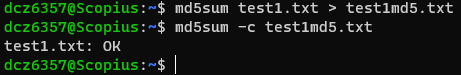


Obtain md5sum for test2.txt and compare to test1.txt :



=> Althought the content are barely the same, md5sum from both file are completely different

**(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 (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 = 29, 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)

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)

=> Using Bob’s public key, Alice can now encrypt her message M = 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 = 11, q =23, M = 109. 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)

n = p\*q = 11\*23 = 253

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**

Both agree to use the public parameters mentioned in the question: 𝑔 = 2 and 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

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

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

**(i) using the above numbers for M, a, b**

- Step 1: Bob defines 𝐺 = 2, 𝑝 = 443, secret key 𝑏 = 231. Then Bob computes his public key  
𝐵 = 𝐺𝑏 (𝑚𝑜𝑑 𝑝) = 2231 (𝑚𝑜𝑑 443) = 305  
Bob then sends his public key 𝑘𝐵𝑝𝑢𝑏 = 〈𝐺, 𝑝, 𝐵〉 = 〈2, 443, 305〉  
- Step 2: Alice selects her private key 𝑎 = 198. Then Alice computes her public key  
𝐴 = 𝐺𝑎 (𝑚𝑜𝑑 𝑝) = 2198 (𝑚𝑜𝑑 443) = 144  
Then Alice computes the encryption key 𝐾𝑠 = 𝐵𝑎 ≡ 𝐺𝑏𝑎 (𝑚𝑜𝑑 𝑝) = 305198 (𝑚𝑜𝑑 443) = 321  
- Step 3: Alice encrypts the message 𝑀 = 215, obtaining the cipher  
𝐶 = 𝑀 × 𝐾  
𝑠 (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 = 𝐴𝑏 (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   
  
**(ii) using your own choice of numbers for M, a, b (10 marks)**

Give M=539, a=777, b=330

- Step 1: Bob defines 𝐺 = 5, 𝑝 = 747, secret key 𝑏 = 330. Then Bob computes his public key  
𝐵 = 𝐺𝑏 (𝑚𝑜𝑑 𝑝) = 5330 (𝑚𝑜𝑑 747) = 523  
Bob then sends his public key 𝑘𝐵𝑝𝑢𝑏 = 〈𝐺, 𝑝, 𝐵〉 = 〈5, 747, 523〉  
- Step 2: Alice selects her private key 𝑎 = 777. Then Alice computes her public key  
𝐴 = 𝐺𝑎 (𝑚𝑜𝑑 𝑝) = 5777 (𝑚𝑜𝑑 747) = 737  
Then Alice computes the encryption key 𝐾𝑠 = 𝐵𝑎 ≡ 𝐺𝑏𝑎 (𝑚𝑜𝑑 𝑝) = 523777 (𝑚𝑜𝑑 747) = 100  
- Step 3: Alice encrypts the message 𝑀 = 5390, obtaining the cipher  
𝐶 = 𝑀 × 𝐾  
𝑠 (𝑚𝑜𝑑 𝑝) = 539 × 100 (𝑚𝑜𝑑 747) = 116  
Alice then sends her public key 𝐴 = 737 and the cipher text 𝐶 = 116 to Bob  
- Step 4: Bob derives the encryption key by the calculation: 𝐾𝑠 = 𝐴𝑏 (𝑚𝑜𝑑 𝑝) = 737330 (𝑚𝑜𝑑 747) = 100  
Bob then decrypts to cipher to receive the message: 𝐶 × 𝐾𝑠-1 (𝑚𝑜𝑑 𝑝) = 116 × 127 (𝑚𝑜𝑑 747) = 539 = 𝑀