Computer Security Principles – Lab #4

Mohammed Alom – R00144214

Task 1.1

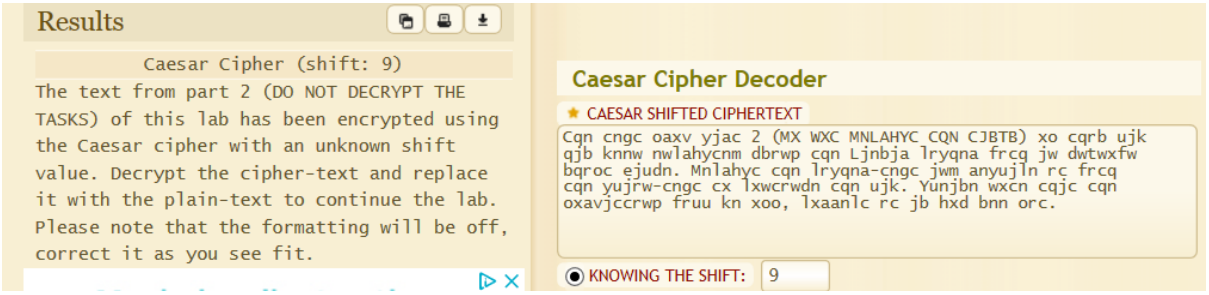
Task 1.2 has been encrypted using the Caesar cipher with a shift value of 9. Decrypt the cipher-text and replace it with the plain-text to continue the lab.

Task 1.2

Cqn cngc oaxv yjac 2 (MX WXC MNLAHYC CQN CJBTB) xo cqrb ujk qjb knnw nwlahycnm dbrwp cqn Ljnbja lryqna frcq jw dwtwxfw bqroc ejudn. Mnlahyc cqn lryqna-cngc jwm anyujln rc frcq cqn yujrw-cngc cx lxwcrwdn cqn ujk. Yunjbn wxcn cqjc cqn oxavjccrwp fruu kn xoo, lxaanlc rc jb hxd bnn orc.

Answer –1.2

The text from part 2 (DO NOT DECRYPT THE TASKS) of this lab has been encrypted using the Caesar cipher with an unknown shift value. Decrypt the cipher-text and replace it with the plain-text to continue the lab. Please note that the formatting will be off, correct it as you see fit.



Part 2 – Substitution and Frequency Analysis

When a shift value is used, the cipher alphabet is automatically generated, as seen with the Caesar cipher. This means that the key is just a number, making remembering the key and passing the key around quite easy. However, this also means that it can be quite easy to retrieve the plain-text by simply guessing the key. An arbitrary cipher alphabet (also known as a substitution alphabet) could be defined instead, with each plain-text letter then being substituted by the corresponding cipher-text letter. Example: A B C D E F G H I J K LM N O P Q R S T U V W X Y Z C T K J F Z Y G O S L V A Q X B E N R M H I D T P W With the above arbitrary cipher alphabet mapped to the plain-text alphabet Bob can encrypt the message

“COMPUTER SECURITY” as “KXABHMFN RFKHNOMP”. Alice would have to know what the substitution alphabet is in order to decrypt the message. This essentially makes the substitution alphabet the key. Applying bruteforce against this could eventually return the correct plain-text. However this would require testing all possible combinations of the substitution alphabet of which there are 4.032914611*10^26 possible combinations, assuming that only the 26 letter English alphabet is used. A smarter way to do this would be to apply statistical analysis to the cipher-text, specifically frequency analysis. Frequency analysis is the idea of documenting the frequency of letters, letter pairs, letter groups, numbers, and symbols in the cipher-text, and using this information to derive the plain-text. This is made possible because of language characteristics, word structure, and sentence structure. In an arbitrary sentence, letters will appear with varying frequencies. By documenting these frequencies and cross-referencing them with the general letter frequencies in a language, the plain-text can be deciphered. The longer the cipher-text the more successful this analysis should be. In the English Language, the following is the order of the most commonly used letters, going from left to right with “e” as the most common and “z” as the least common: e t a o i n s r h l d c u m f p g w y b v k x j q z This frequency can actually change depending on the context of the document analysed. http://letterfrequency.org/ has a list of the varying frequencies. The most common double letter frequencies are: ss ee tt ff ll mm oo Common sense is also a valuable tool when used alongside frequency analysis. For example, after discovering that the first 3 letters of a 4 letter word are “thi-”, it could be guessed that the 4th letter is “s” making the word “this”. Similarly, if cipher-text was to start “SDPOP VOP” then it would be possible to guess that the plain-text may be “THERE ARE”. We can therefore make the assumption that T is S, D is H, E is P, and so on. Taking these assumptions and applying the relevant substitutions to the rest of the cipher-text will quickly show whether or not the assumption was correct. By pairing the known frequencies with common sense, it becomes possible to crack weak substitution ciphers. You’ll need to decrypt Task 2.1 the same way this text was decrypted. Tools The following site has a tool which can be used to encrypt and decrypt and substitution cipher and perform frequency analysis and automatically crack a cipher: Cipher: http://www.dcode.fr/monoalphabetic-substitution Frequency Analysis: http://www.dcode.fr/frequency-analysis

Task 2.1

Task 2.2 has been encrypted using a substitution cipher. The key is “VUTWZHLSKDYMBQIXFCGJERNOAP”. Decrypt the cipher-text and replace it with the plain-text to continue the lab.

Task 2.2

TASK 2.3 HAS BEEN ENCRYPTED WITH A SUBSTITUTION CIPHER. THE KEY IS NOT KNOWN. DECRYPT THE CIPHER-TEXT MANUALLY WITH THE AID OF FREQUENCY ANALYSIS AND REPLACE IT WITH THE PLAIN-TEXT TO CONTINUE THE LAB. YOU WILL NEED TO USE THE MANUAL DECRYPTION METHOD ON THE TOOL PROVIDED IN THE ABOVE LINK. HINT: THERE IS A

URL IN THE TEXT WHICH POINTS TO THE SAME SITE IN THE ABOVE LINKS. THERE IS ALSO CIPHER-TEXT BETWEEN QUOTATIONS.

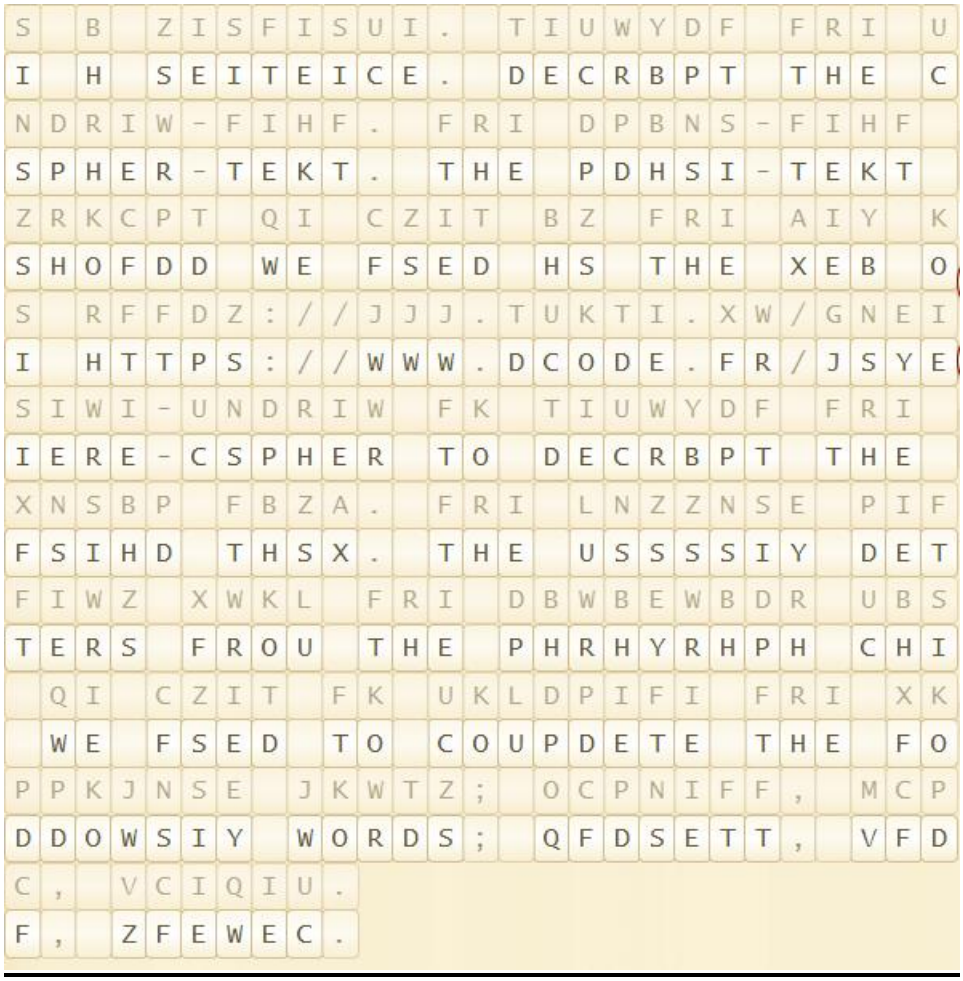
Task 2.3

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
X	H	F	P	Y	T	J	K	E	W	O	U	V	S	Q	D	W	H	I	D	C	Z	R	F	B	S

⇒ Alphabet Encrypt XHFPYTJKEWOUVSQDWHIDCZRFBS

⇒ Alphabet Decrypt YUPTICXBRSGHKDOWNZFLMJQAEV

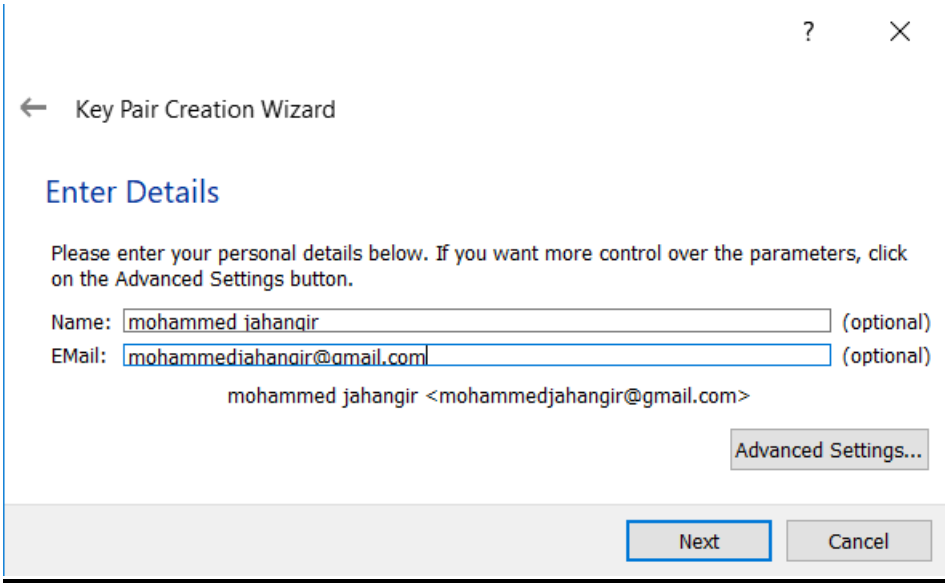
F	R	I		X	K	P	P	K	J	N	S	E		U	N	D	R	I	W	-	F	I	H	F		J
T	H	E		F	O	D	D	O	W	S	I	Y		C	S	P	H	E	R	-	T	E	K	T		W
N	P	P		S	I	I	T		L	B	S	C	B	P		B	S	B	P	Y	Z	N	Z		B	S
S	D	D		I	E	E	D		U	H	I	F	H	D		H	I	H	D	B	S	S	S		H	I
T		U	K	L	L	K	S		Z	I	S	Z	I		F	K		T	I	U	N	D	R	I	W	
D		C	O	U	U	O	I		S	E	I	S	E		T	O		D	E	C	S	P	H	E	R	
"	C	N	M		F	T	Z	Z		Y	M	Z	Z		E	H	M	A		C	N	M		J	T	Z
"	F	S	V		T	D	S	S		B	V	S	S		Y	K	V	X		F	S	V		W	D	S
Z	"	.		W	I	L	I	L	Q	I	W	,		F	R	I	W	I		N	Z		B		X	N
S	"	.		R	E	U	E	U	W	E	R	,		T	H	E	R	E		S	S		H		F	S
S	N	F	I		S	C	L	Q	I	W		K	X		F	R	W	I	I		P	I	F	F	I	W
I	S	T	E		I	F	U	W	E	R		O	F		T	H	R	E	E		D	E	T	T	E	R
	J	K	W	T	Z	,		J	N	F	R		Z	K	L	I		Q	I	N	S	E		L	K	W
	W	O	R	D	S	,		W	S	T	H		S	O	U	E		W	E	S	I	Y		U	O	R
I		U	K	L	L	K	S		F	R	B	S		K	F	R	I	W	Z	.		F	R	I		G
E		C	O	U	U	O	I		T	H	H	I		O	T	H	E	R	S	.		T	H	E		J
I	W	Y		Z	B	L	I		U	B	S		Q	I		Z	B	N	T		X	K	W		J	K
E	R	B		S	H	U	E		C	H	I		W	E		S	H	S	D		F	O	R		W	O
W	T	Z		J	R	N	U	R		I	S	T		N	S		T	K	C	Q	P	I		P	I	F
R	D	S		W	H	S	C	H		E	I	D		S	I		D	O	F	W	D	E		D	E	T
F	I	W	Z		B	S	T		U	B	S		X	N	F		F	K	E	I	F	R	I	W		N
T	E	R	S		H	I	D		C	H	I		F	S	T		T	O	Y	E	T	H	E	R		S


S B ZISFISUI. TIUWYDF FRI U
I H SEITEICE. DECRBPT THE C
NDRIW-FIHF. FRI DPBNS-FIHF
SPHER-TEKT. THE PDHSI-TEKT
ZRKCPT QI CZIT BZ FRI AIY K
SHOFDD WE FSED HS THE XEB O
S RFFDZ:// JJJ.TUKTI.XW/GNEI
I HTTPS://WWW.DCODE.FR/JSYE
SIWI-UNDRIW FK TIUWYDF FRI
IERE-CSPHER TO DECRBPT THE
XNSBP FBZA. FRI LNZZNSE PIF
FSIHD THSX. THE USSSSIY DET
FIWZ XWKL FRI DBWBEWBDR UBS
TERS FROU THE PHRHYRHPH CHI
Q I CZIT FK UKLDPIFI FRI XK
WE FSED TO COUPDETE THE FO
PPKJNSE JKWTZ; OCPNIFF, MCP
DDOWSIY WORDS; QFDSETT, VFD
C, VCIQIU.
F, ZFEWEC.

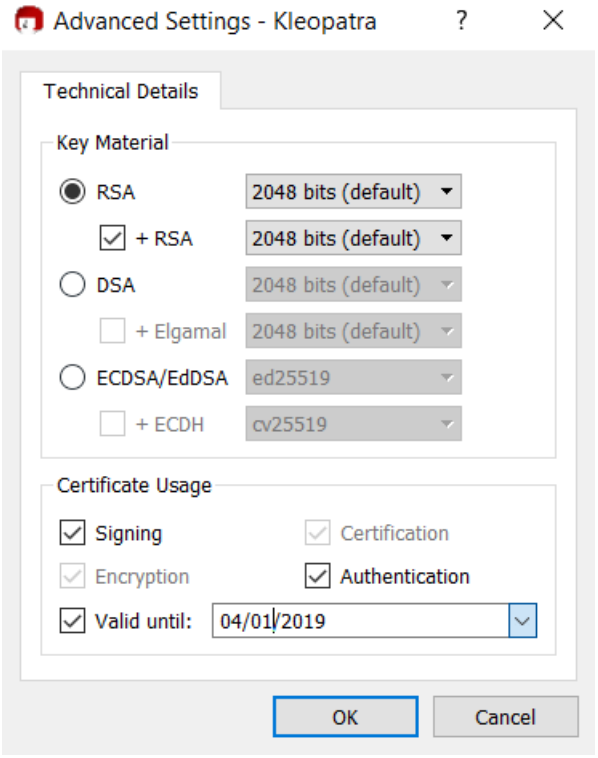
Part 3: Public key cryptography.

Task 3.1 Generating a public/private key pair

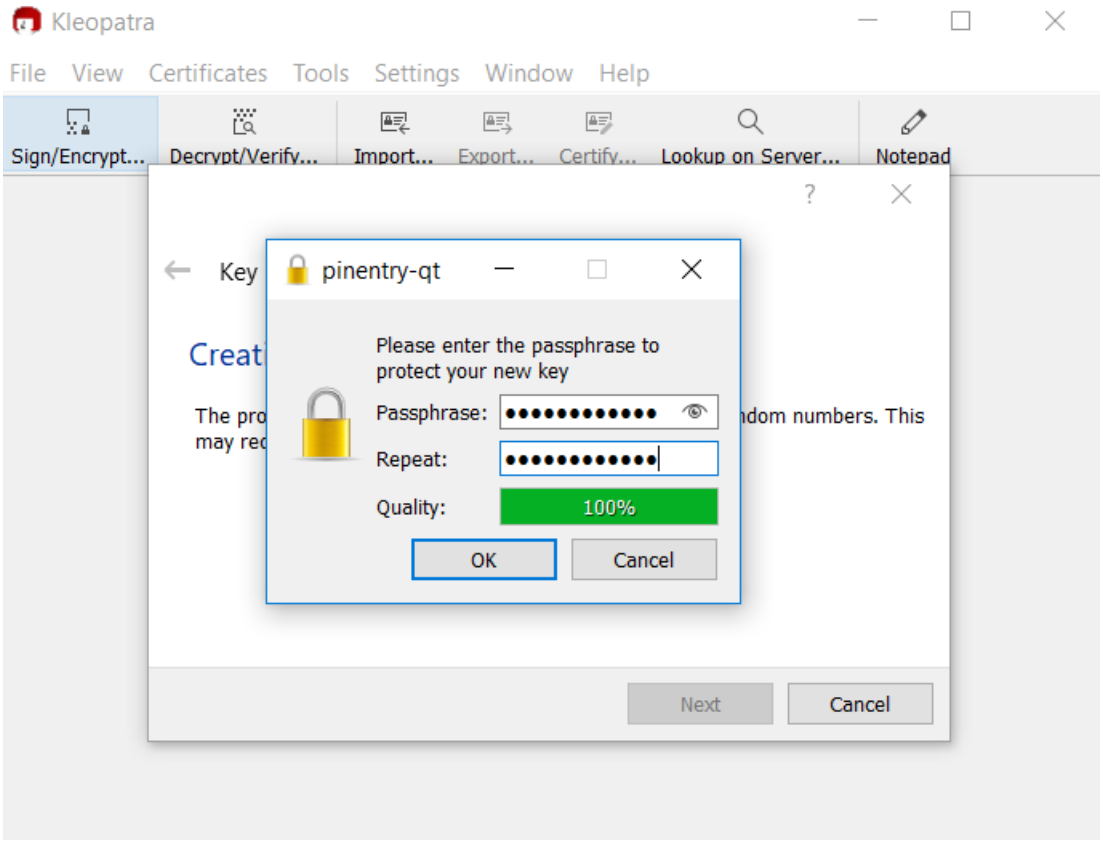
Entering Name and Password



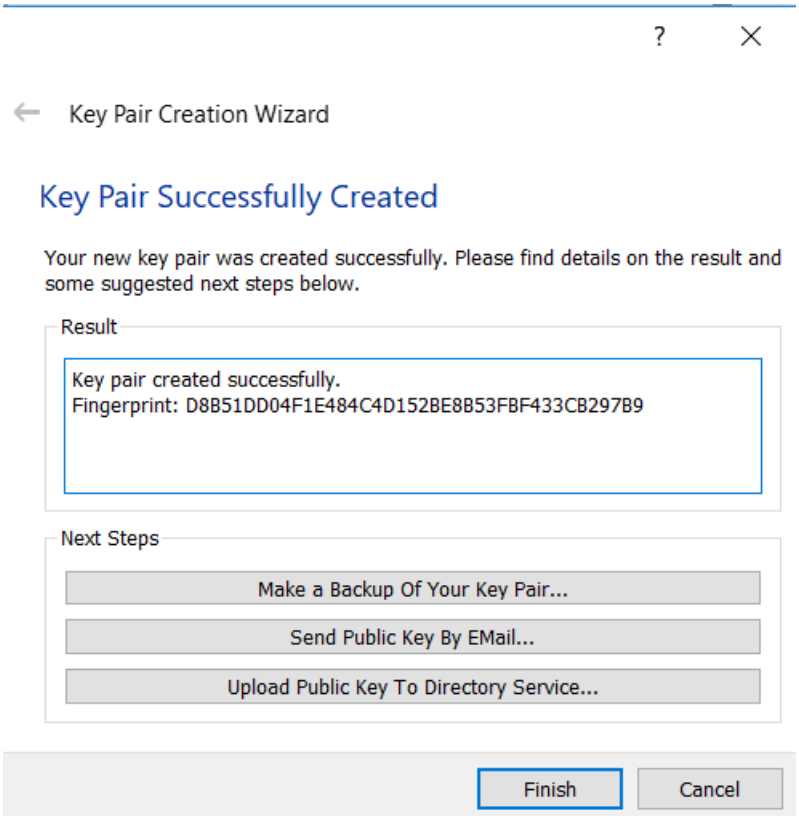
Advance Setting



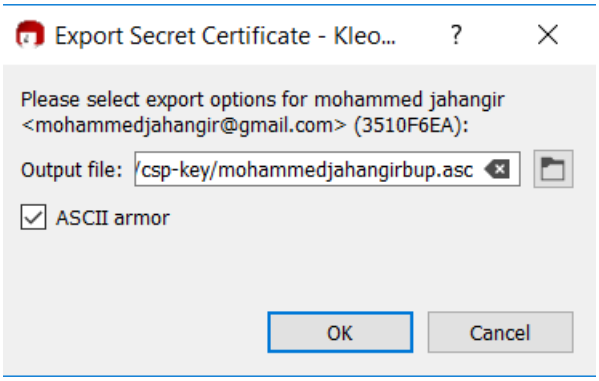
Setting the passphrase



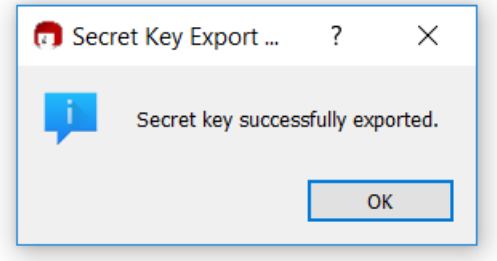
Key is created and its screeshot



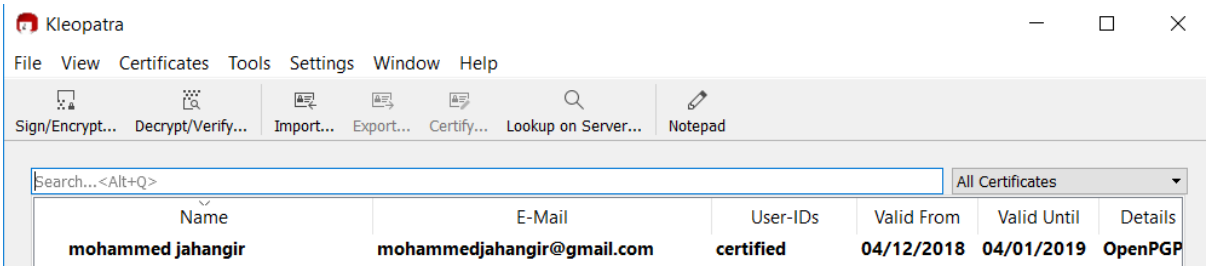
Saving Backup the key



Exported secret key

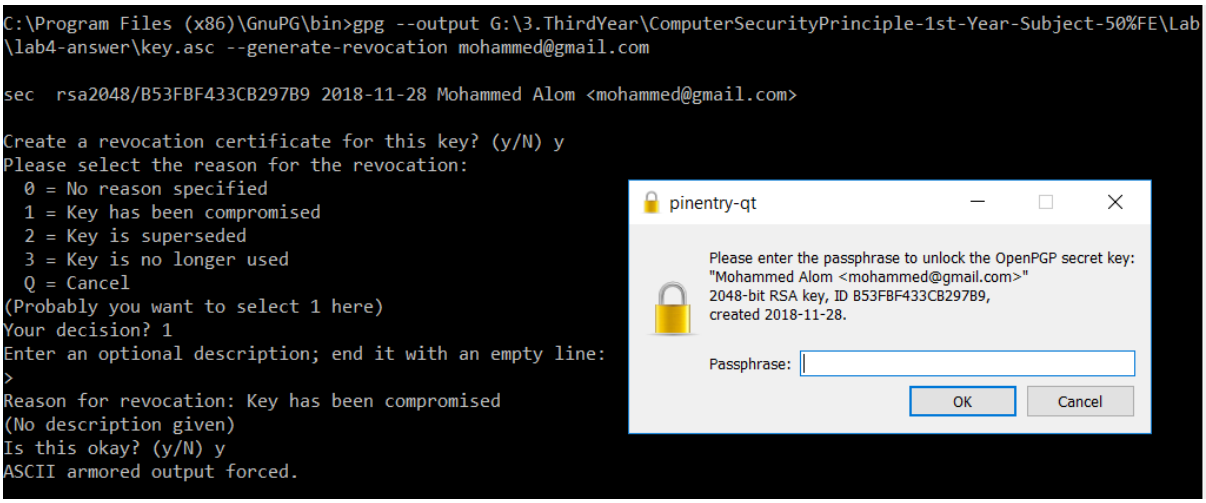


Key is visible now

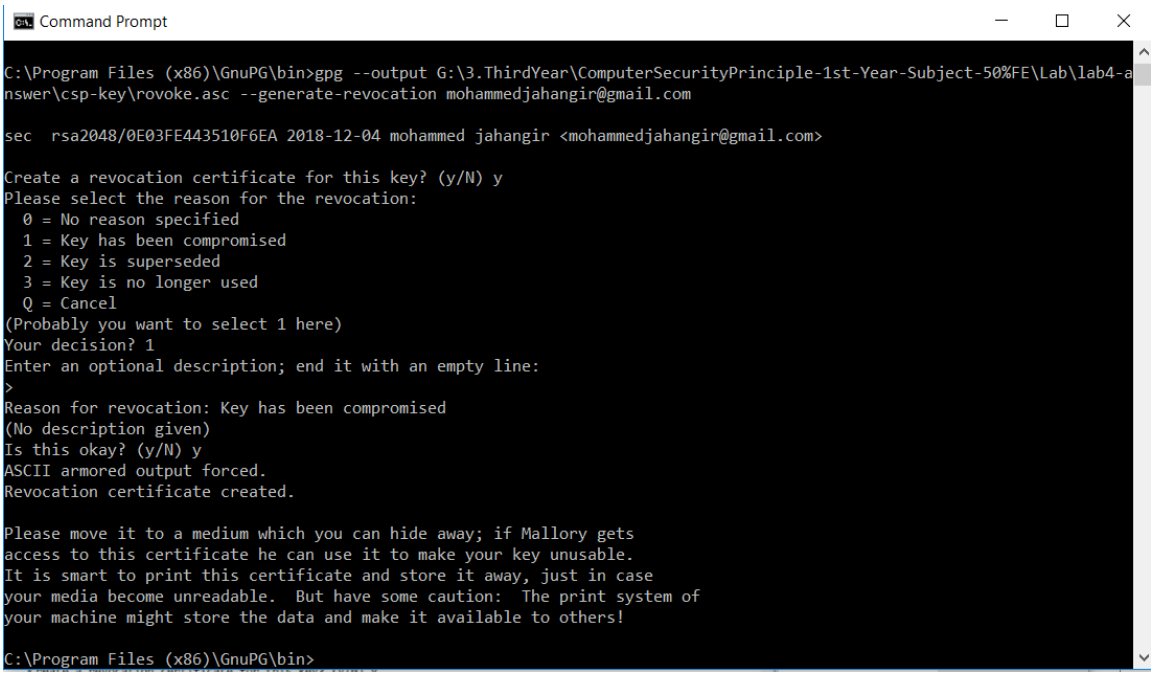


Task 3.2 Generating a revocation key in case of private-key theft

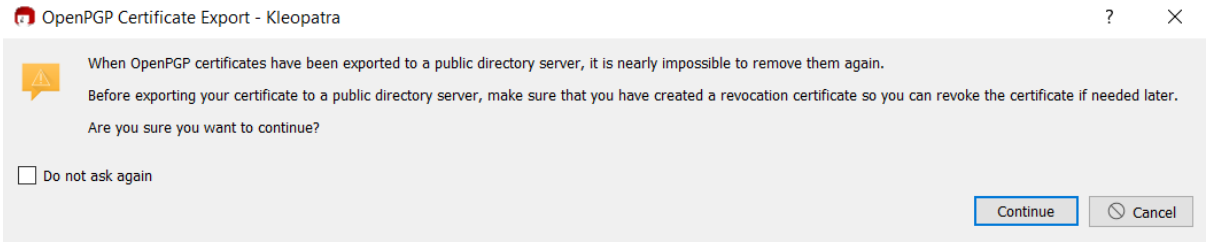
Generating the key



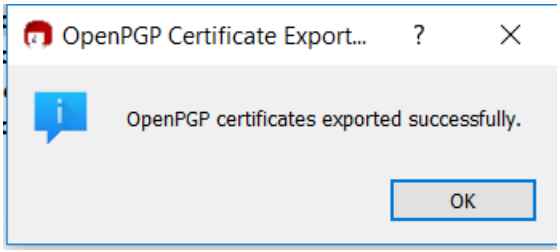
Storing the key..



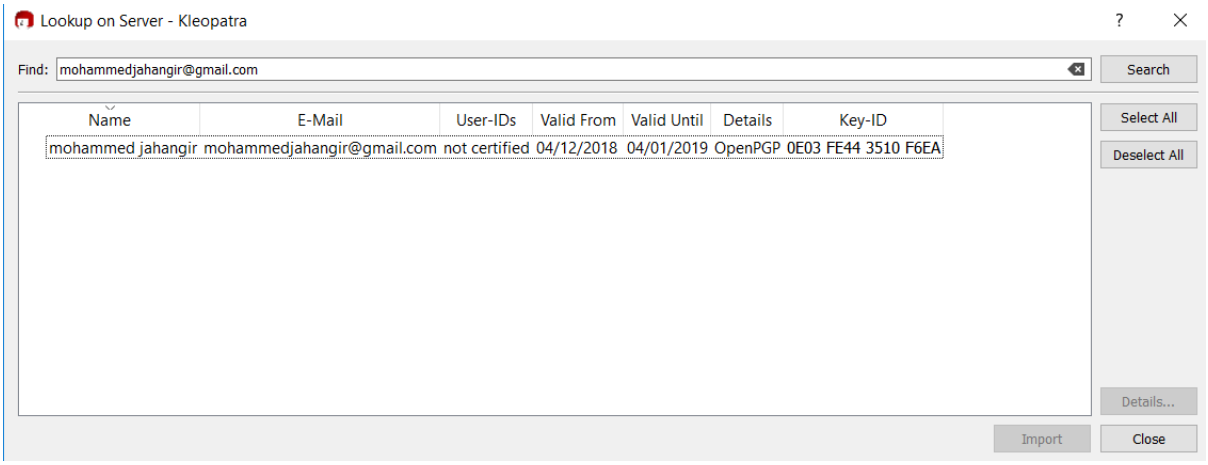
Task 3.3 Exporting the key to the key-server



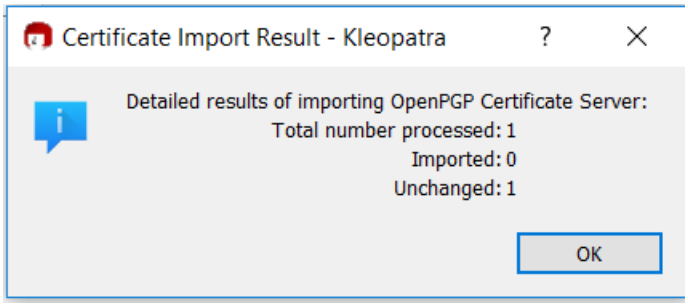
Publish to the server successfully



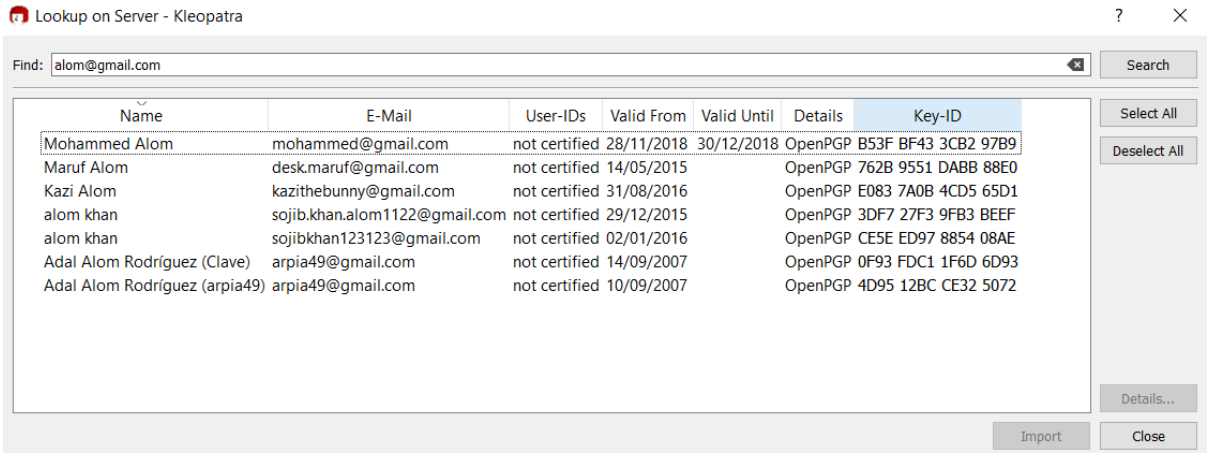
Account is showing in the server



Imported the public key

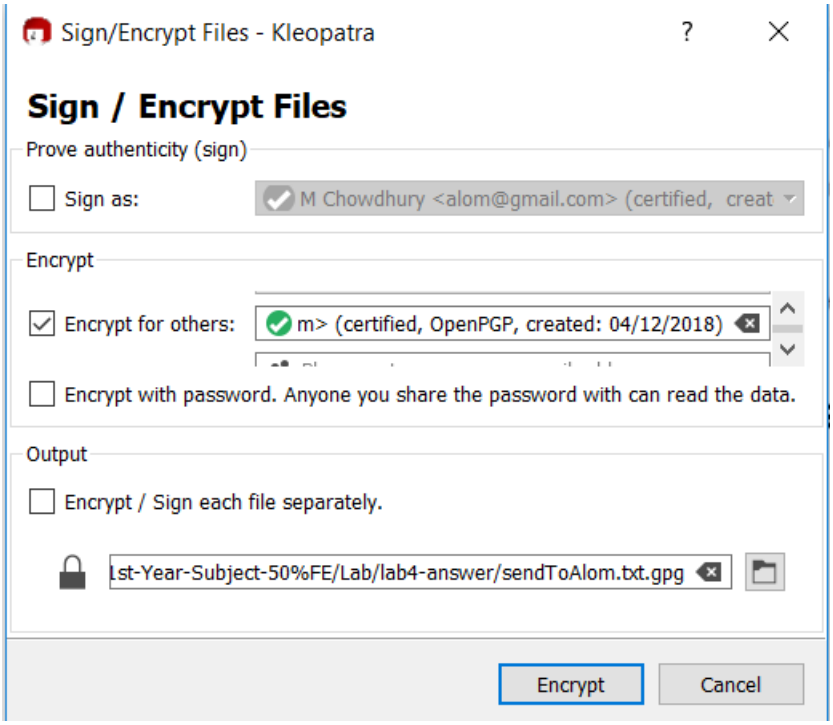


Looking the person on the server

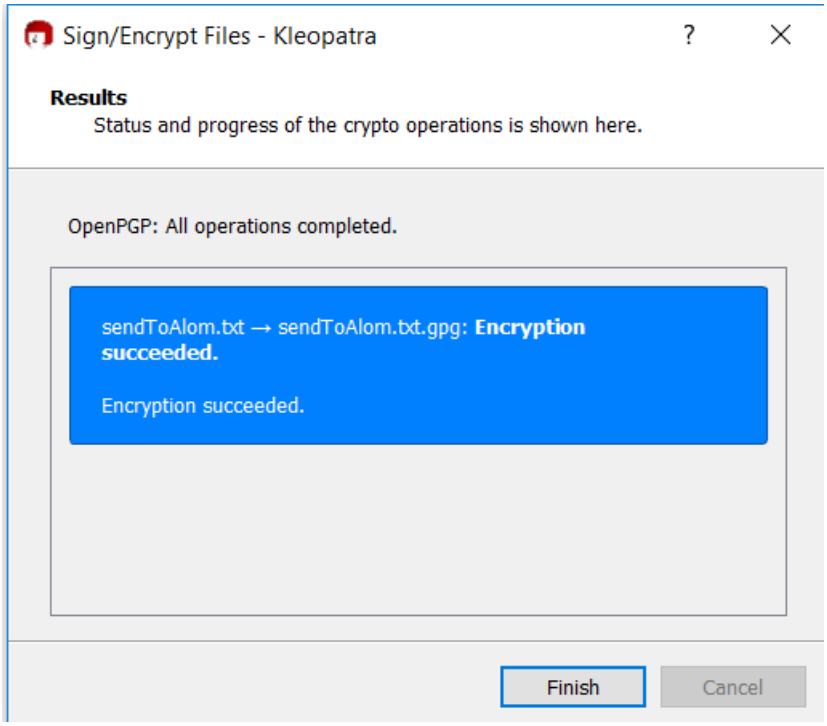


Task 3.4 Encrypting a file for sending to a third party

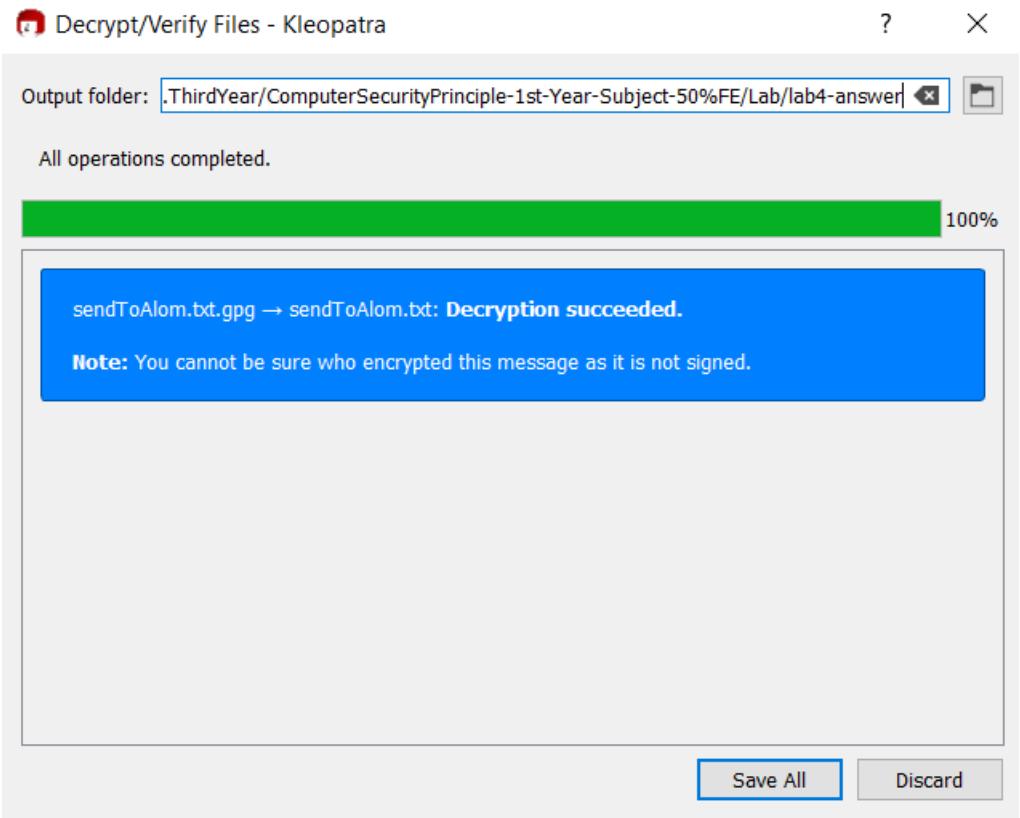
Encrypting the file



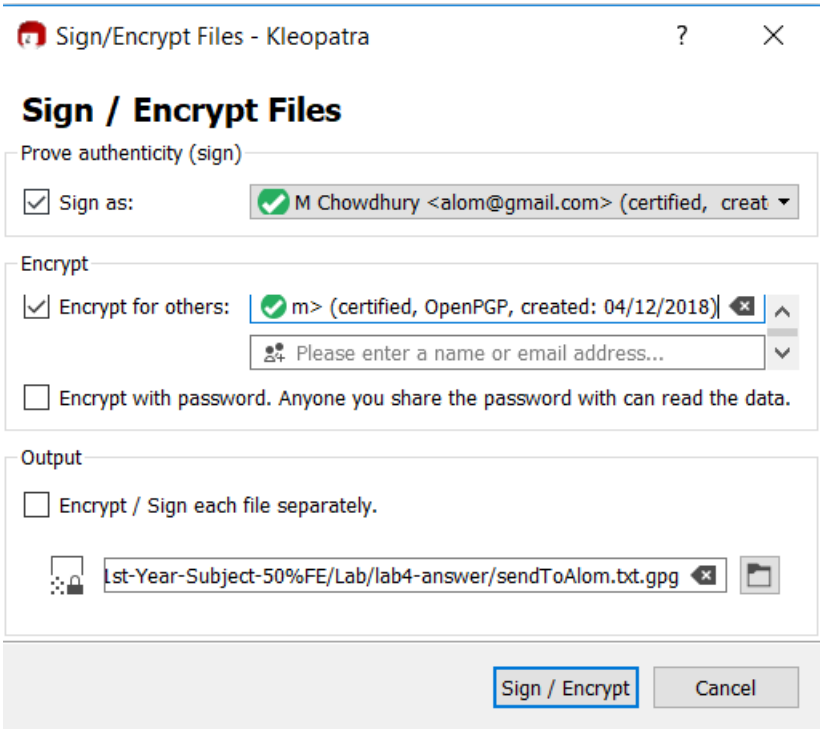
Encrypted was successful

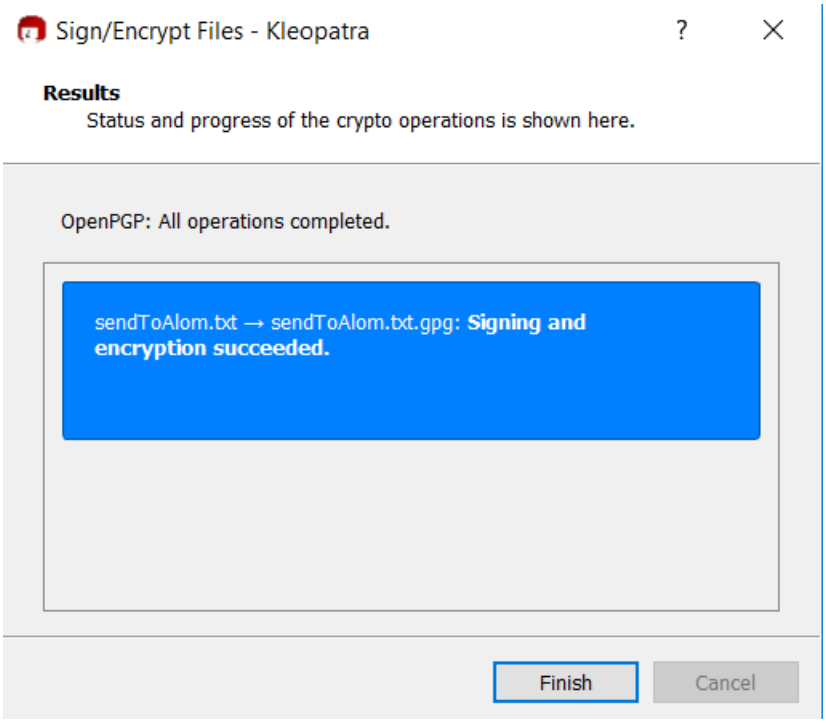


Task 3.5 Decrypting a message using my private key

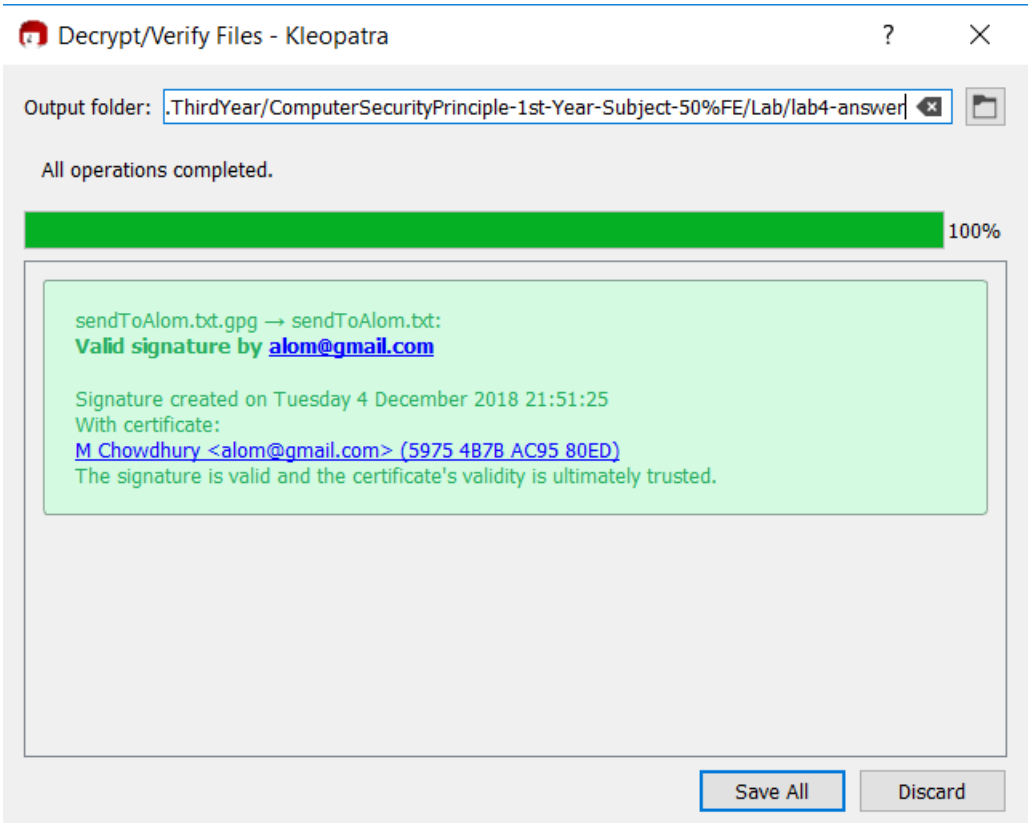


Task 3.6 Signing a message





Task 3.8 Verifying a message



Task 3.9 Communicating securely with your lab partner.

There was no lab partner was available that time to check that functionality.

Part 4 Verify Software using a Detached Signature File

Downloading the Putty 32 bit version putty.exe and signature file

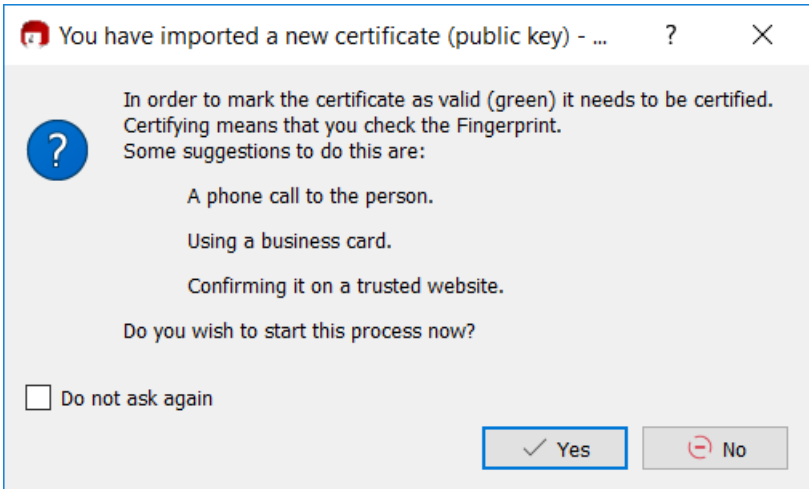
The installer packages above will provide all of these (except PuTTYtel), but you can download them one by one if you prefer.

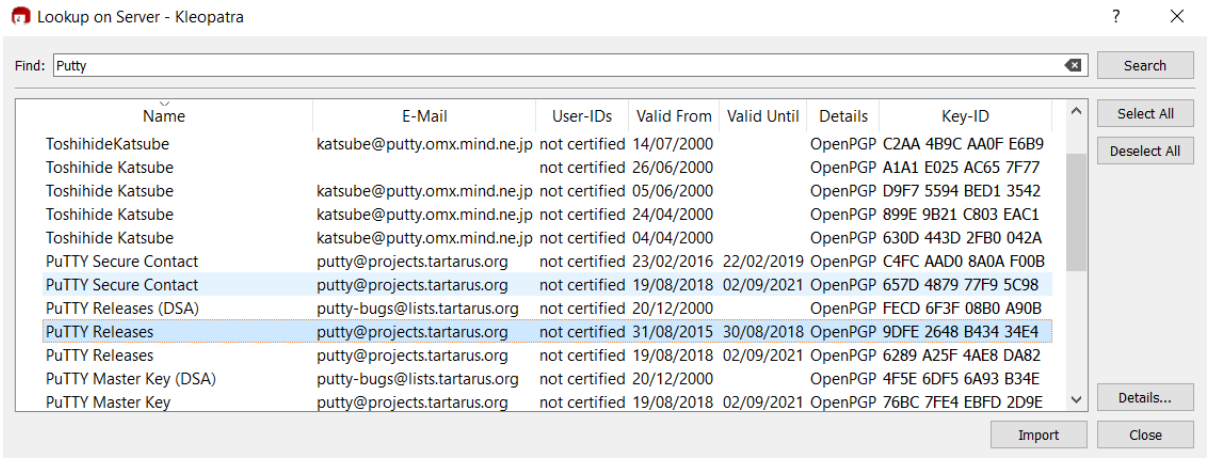
(Not sure whether you want the 32-bit or the 64-bit version? Read the FAQ entry.)

`putty.exe` (the SSH and Telnet client itself)

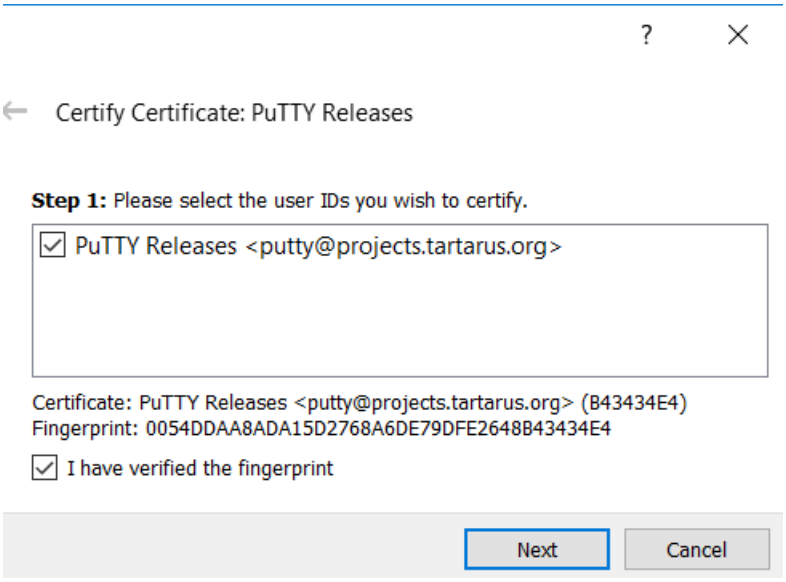
32-bit:	`putty.exe`	(or by FTP)	(signature)
64-bit:	`putty.exe`	(or by FTP)	(signature)

Importing the putty released key

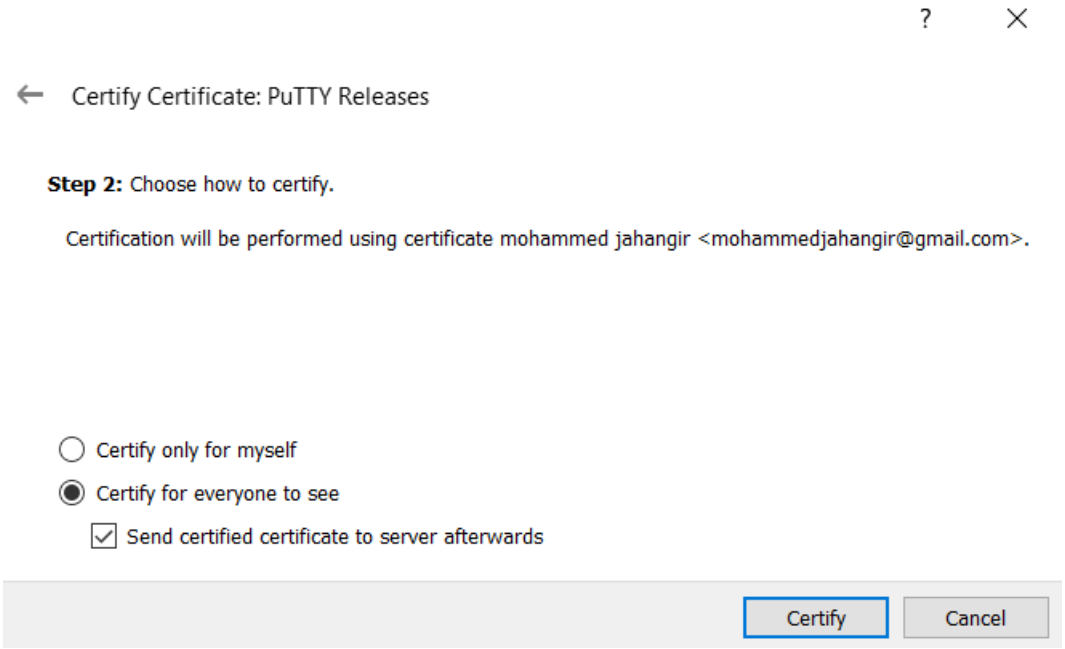




Certifying the putty key



Sending putty key to the server



Next click File → Decrypt/Verify Files → select putty.exe.gpg → click “file is a detached signature” → enter putty.exe as Signed Data

****

I was trying to do the above step but it did not work and I showed to lecturer Dylan about that.

I uninstall and re-install the application but its giving the same issue. I have attached the screenshot to verify that its not working for some reason.