Cairo university **Faculty of engineering**
Computer department **Semester** **4th year**

Security project 2 report

Name: Yahia Ali **Sec: 2** **B.N:33**

Code explanations:

There are 2 file of python code:
Sender: take input from the user, establish the connection, encrypt the message then send it
Receiver: connect to the server, read the configuration, receive the message, decrypt it then print the result

Sender must run and give him the full input before run the receiver

Before encryption first make sure that the message is dividable by the block size if not add empty string until it is.
All mode divide the message into (message size/ block size block) blocks and work on each block.
All needed variable are initialize before running the encryption/decryption.
I used PyCrypto library for apply DES algorithm.
If the message is bigger than 502 bytes (buffer size-10) the program will divide it into n message (n=Message/500) each of them will encrypt and decrypt independently.

Mode	Encryption	Decryption
ECB	Cipher block: encrypt the block using the DES algorithm (library). Concatenate the encrypted blocks to send it.	Cipher block: decrypt the block using the DES algorithm (library). Concatenate the decrypted blocks to print the message.
CBC	Cipher block: XOR the block with IV then encrypt it. Update the initialization vector IV with the cipher block after each block encryption. Concatenate the cipher blocks to send it.	Cipher block: decrypt the block Then XOR it with IV. Update IV with cipher block after each block encryption. Concatenate the decrypted blocks to print the message.
CFB	Cipher block: encrypt IV then XOR it with the MSBs of the plain block. Update IV with the concatenation of last bits of IV and the cipher block after each block encryption. Concatenate the cipher blocks to send it.	Cipher block: Encrypt IV then XOR it with the cipher block. Update IV with the concatenation of last bits of IV and the cipher block. Concatenate the decrypted blocks to print the message.
CTR	Cipher block: encrypt the counter value CV then XOR it with the block. Increment the counter value. Concatenate the cipher blocks to send it.	Cipher block: encrypt the CV then XOR it with the block. Increment the counter value. Concatenate the decrypted blocks to print the message.

Input for the sender file:

It will ask u

1. **Message you want to sent**
2. **The mode (ECB,CBC,CFB,CTR) [default value=”ECB”]**
3. **The key (8 bytes , no spaces allow) [default value=”ABCDEFGH”]**
4. **The hmac key (no spaces allow) [default value=”AAAAAAAA”]**
5. **Block size (must be dividable by 8) [default value=8]**

If there are invalid input the program will use the default value instead.

Why hmac?

In General HMAC faster than CMAC, because hash functions are usually faster than block ciphers.

Analysis Part:

	Mode	Block size	Message size	Encryption Time (second)	Decryption time (second)	Total (ms)
8B 76M	ECB	8	76	0.0149993896484375	0.011982917785644531	26.97
	CBC	8	76	0.008987903594970703	0.008997678756713867	17.97
	CFB	8	76	0.011010408401489258	0.009991645812988281	21.0
	CTR	8	76	0.010984659194946289	0.00800466537475586	18.98
16B 2M	ECB	16	2	0.001983165740966797	0.0020093917846679688	3.98
	CBC	16	2	0.008013486862182617	0.00799250602722168	16.0
	CFB	16	2	0.008978605270385742	0.007992744445800781	16.96
	CTR	16	2	0.0019888877868652344	0.0019986629486083984	3.97
8B 2m	ECB	8	2	0.007992029190063477	0.007987260818481445	15.97
	CBC	8	2	0.007987737655639648	0.007002115249633789	14.98
	CFB	8	2	0.004004716873168945	0.004998207092285156	8.99
	CTR	8	2	0.00797891616821289	0.008007287979125977	15.97
16B 76M	ECB	16	76	0.003983974456787109	0.003996849060058594	7.97
	CBC	16	76	0.00897669792175293	0.008012056350708008	16.98
	CFB	16	76	0.009990692138671875	0.009994268417358398	19.98
	CTR	16	76	0.007971525192260742	0.00800323486328125	15.97

Increasing the block size decrease the total time in most cases.

Block size haven’t large effect on CBC but it does for ECB and CTR.

In general there are no big difference time between encryption and decryption time

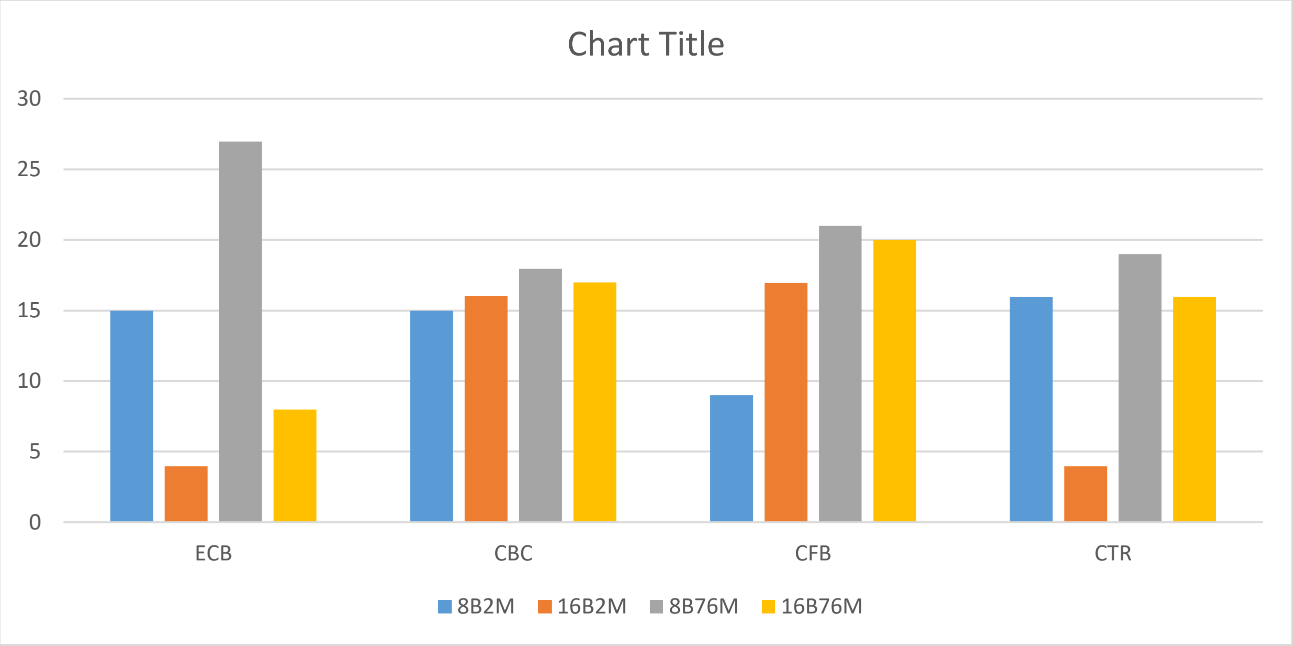
Chart Title
ECB
CBC
CFB
CTR
8B2M
16B2M
8B76M
16B76M