T🡪Red pochne h umama ,aimen wagera se

🡪4,12

1. A stream cipher can be viewed as a generalization of a one-time pad.

Recall that the one-time pad is provably secure. Why can't we prove that a stream cipher is secure using the same argument that was used for the one-time pad?

--Yeh random nh hoti basically –wahan key har bar diff or random thi

Ans. About the one-time pad, the key is still large and the same size of the plaintext, so one-time pad is more secure than a stream cipher.

2. This problem deals with stream ciphers.

a. If we generate a sufficiently long keystream, the keystream must eventually repeat. Why?

That is because the stream cipher takes a small key and stretches it to the size of the plaintext before XOR’ing them together. And since it is using the same method to stretch the smaller key to the size of the plaintext, the keystream will repeat, sooner or later.

b. Why is it a security concern if the keystream repeats?

Because if the keystream repeats it will not be as secure as it could be. The one time pad uses a keystream that is all randomly generated, but if your keystream repeats, that means that it is a lot easier to crack(not truly random). Due to the repetitive nature, it creates a pattern, making it more feasible to recover the original key. Source: the book.

3. Suppose that Alice uses a stream cipher to encrypt plaintext *P,* obtaining ciphertext C, and Alice then sends *C* to Bob. Suppose that Trudy happens to know the plaintext *P,* but Trudy does not know the key *K* that was used in the stream cipher.

a. Show that Trudy can easily determine the keystream that was used to encrypt *P*

Alice XOR each bit of the plaintext(P) with the keystream(S).

To encrypt,

C0 = P0 XOR S0,

C1 = P1 XOR S1,

C2 = P2 XOR S2, ..., Cn = Pn XOR Sn

And to decrypt

,P0 = C0 XOR S0,

P1 = C1 XOR S1,

P2 = C2 XOR S2,

…, Pn = Cn XOR Sn

Trudy can do,

S0 = P0 XOR C0,

S1 = P1 XOR C1,

S2 = P2 XOR C2, …,

Sn = Pn XOR Cn*.*

b. Show that Trudy can, in effect, replace *P* with plaintext of her choosing, say, *P'.* That is, show that Trudy can create a ciphertext message C" so that when Bob decrypts *C'* he will obtain *P'.*

C’ = S XOR P’ and then, S = C XOR P, we can replace the second to the first such that C’ = C XOR P XOR P’

4. This problem deals with the A5/1 cipher. For each part, justify your answer.

a. On average, how often does the X register step?

3/4 because 2 register at least at 1 clock cycle

b. On average, how often does the Y register step?

3/4

c. On average, how often does the Z register step?

3/4

d. On average, how often do all three registers step?

5/6

e. On average, how often do exactly two registers step?

½

f. On average, how often does exactly one register step?

0

g. On average, how often does no register step?

0

5. Implement the A5/1 algorithm. Suppose that, after a particular step, the values in the registers are

*X = (x0,*xi,..., xis) = (1010101010101010101)

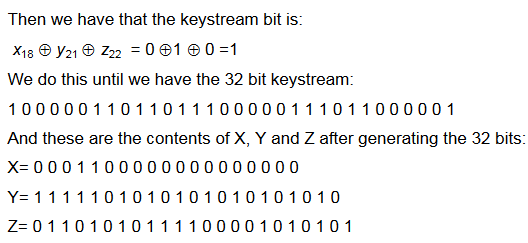
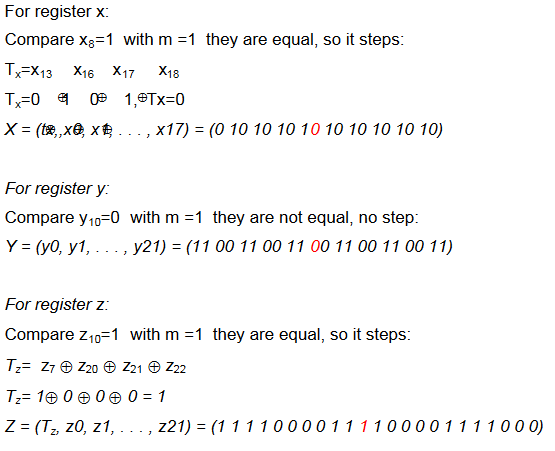
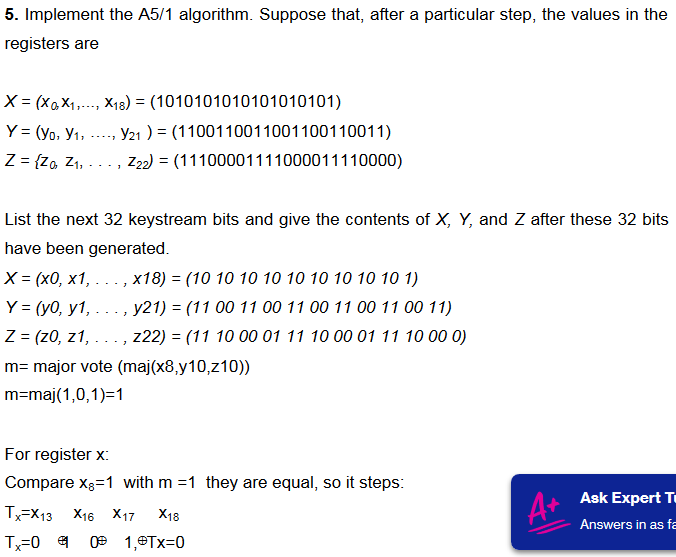
*Y* = (i/o, 2/1, \_··, 2/21 ) = (1100110011001100110011)

*Z = {z0,* z i , . . . , *z22)* = (11100001111000011110000)

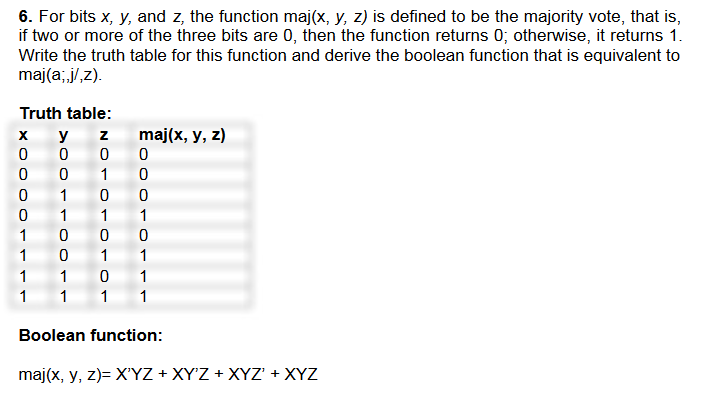
List the next 32 keystream bits and give the contents of *X, Y,* and *Z* after these 32 bits have been generated.

A5/1 is based around a combination of three linear feedback shift registers (LFSRs) with irregular clocking. The three shift registers are specified as follows:

|  |  |  |
| --- | --- | --- |
| LFSR no. | Regs Length in bits | Feedback polynomial |
| X | 19 | x18 + x17 + x17 + x13 + 1 |
| Y | 22 | y21 +y20 + 1 |
| Z | 23 | z22 + z21 + z20 +z7 + 1 |

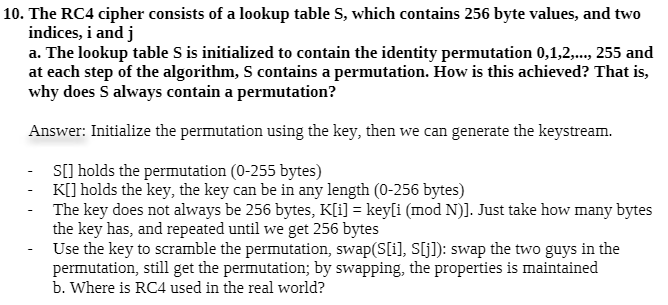


6. For bits *x, y,* and *z,* the function maj(x, *y, z)* is defined to be the majority vote, that is, if two or more of the three bits are 0, then the function returns 0; otherwise, it returns 1. Write the truth table for this function and derive the boolean function that is equivalent to maj(a;,j/,z).



7. The RC4 cipher consists of a lookup table *S,* which contains 256 byte values, and two indices, *i* and *j .*

a. The lookup table *S* is initialized to contain the identity permutation 0,1,2,..., 255 and at each step of the algorithm, *S* contains a permutation. How is this achieved? That is, why does *S* always contain a permutation?



b. Where is RC4 used in the real world?

SSL,WEP,internet explorer

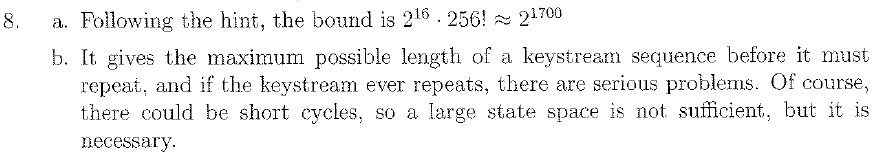
8. This problem deals with the RC4 stream cipher.

a. Find a reasonable upper bound on the size of the RC4 state space.

That is, find an upper bound for the number of different states that are possible for the RC4 cipher. Hint: The RC4 cipher consists of a lookup table S, and two indices i and j . Count the number of possible distinct tables S and the number of distinct indices i and j , then compute the product of these numbers.

b. Why is the size of the state space relevant when analyzing a stream

cipher?



**MANUAL**

9. Implement the RC4 algorithm. Suppose the key consists of the following

seven bytes: (OxlA, 0x2B, 0x3C, 0x4D, 0x5E, 0x6F, 0x77). For each of the

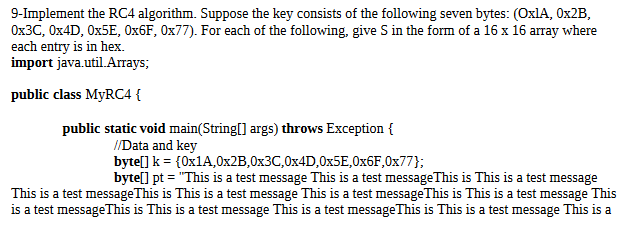
following, give S in the form of a 16 x 16 array where each entry is in hex.

1. List the permutation S and indices i and j after the initialization phase has completed.
2. List the permutation S and indices i and j after the first 100 bytes

of keystream have been generated.

c. List the permutation S and indices i and j after the first 1000

bytes of keystream have been generated.



10. Suppose that Trudy has a ciphertext message that was encrypted with the RC4 cipher—see Tables 3.1 and 3.2. For RC4, the encryption formula is given by c; = pi È fcj, where fc; is the ith byte of the keystream, Pi is the ith byte of the plaintext, and Cj is the ith byte of the ciphertext.

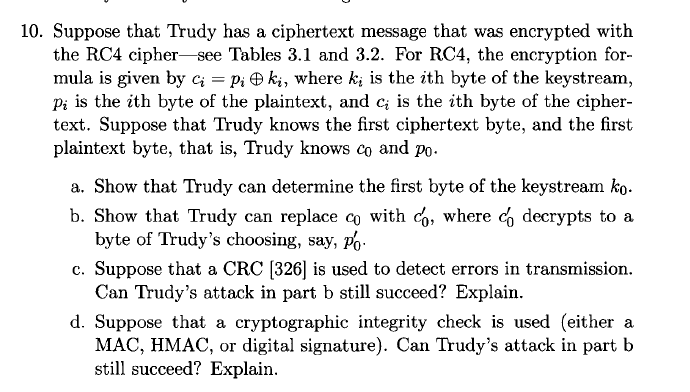
Suppose that Trudy knows the first ciphertext byte, and the first plaintext byte, that is, Trudy knows Co and poa.

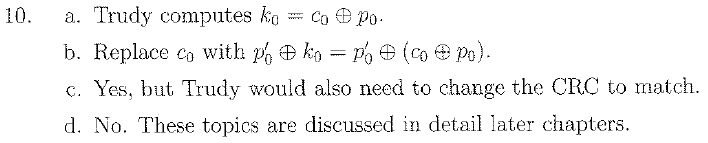
Show that Trudy can determine the first byte of the keystream ko.

b. Show that Trudy can replace CQ with CQ, where c0 decrypts to a byte of Trudy's choosing, say, p'Q.

c. Suppose that a CRC [326] is used to detect errors in transmission. Can Trudy's attack in part b still succeed? Explain.

d. Suppose that a cryptographic integrity check is used (either a MAC, HMAC, or digital signature). Can Trudy's attack in part b still succeed? Explain.





**MANUAL**

11.

This problem deals with a Feistel Cipher.

a. Give the definition of a Feistel Cipher.

b. Is DES a Feistel Cipher?

c. Is AES a Feistel Cipher?

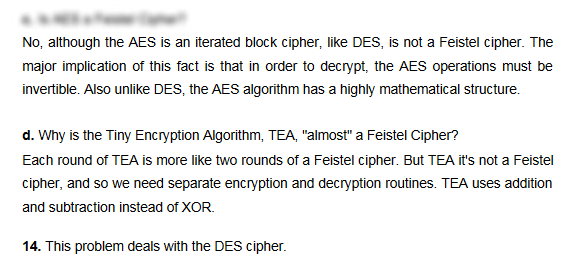
d. Why is the Tiny Encryption Algorithm, TEA, "almost" a Feistel

Cipher?

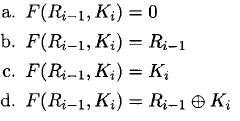
a)

b) Yes, DES is just one example of a Feistel Cipher. A cryptographic system based on Feistel cipher structure uses the same algorithm for both encryption and decryption.

c)



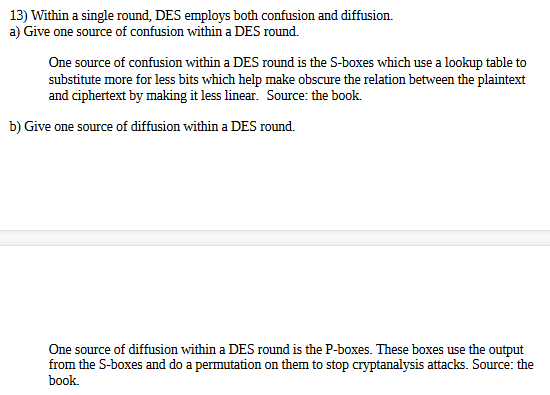
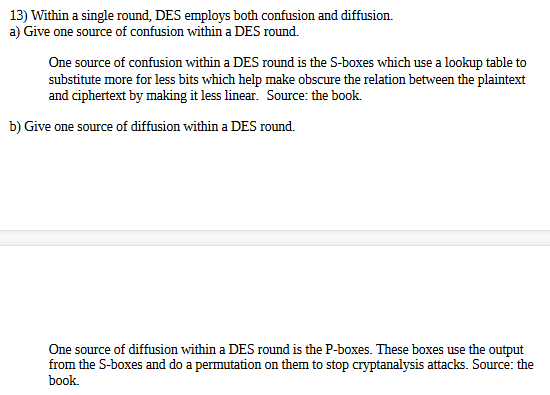
12. Consider a Feistel cipher with four rounds. Then the plaintext is denoted as P = (Lo, -Ro) and the corresponding ciphertext is C = (L4, R4).

What is the ciphertext C, in terms of Lo, RQ, and the subkey, for each of the following round functions? 

13. Within a single round, DES employs both confusion and diffusion.

a. Give one source of confusion within a DES round.

b. Give one source of diffusion within a DES round.



14. This problem deals with the DES cipher.

a. How many bits in each plaintext block?

b. How many bits in each ciphertext block?

c. How many bits in the key?

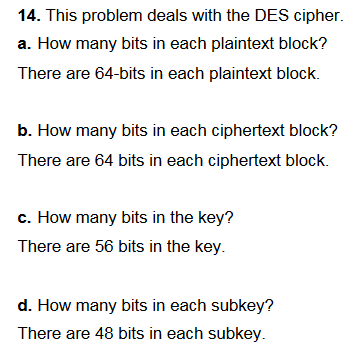
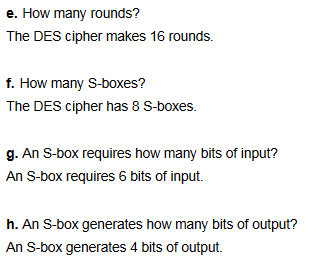
d. How many bits in each subkey?

e. How many rounds?

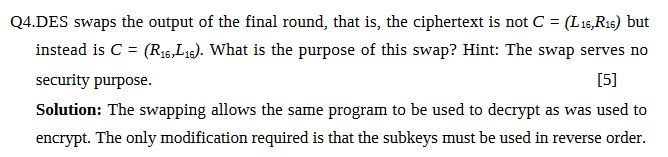
f. How many S-boxes?

g. An S-box requires how many bits of input?

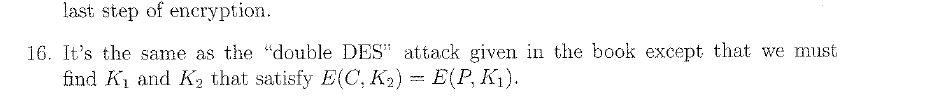
h. An S-box generates how many bits of output?

15. DES swaps the output of the final round, that is, the ciphertext is not C = {Lie, RIQ) but instead it is C = (Rie, Lie). What is the purpose of this swap?



16. Recall the attack on double DES discussed in the text. Suppose that we instead define double DES as C — D(E(P,Ki),K2). Describe a meet-in-the-middle attack on this cipher.

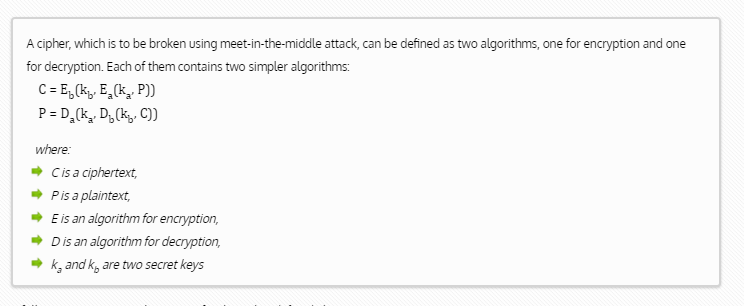


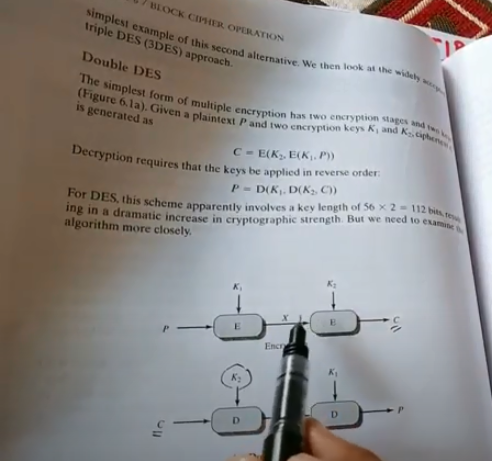
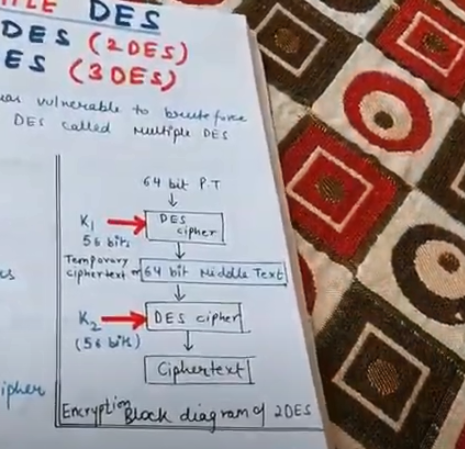
**meet-in-the-middle attack** (**MITM** :

The **meet-in-the-middle attack** (**MITM**) is a generic space–time tradeoff cryptographic attack against encryption schemes that rely on performing multiple encryption operations in sequence. The MITM attack is the primary reason why [Double DES](https://en.wikipedia.org/wiki/Double_DES) is not used and why a [Triple DES](https://en.wikipedia.org/wiki/Triple_DES) key (168-bit) can be bruteforced by an attacker with 256 space and 2112 operations

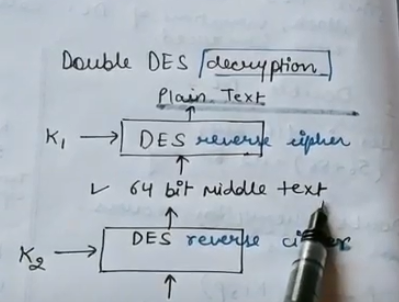
**🡺Manual**

The meet-in-the-middle attack is one of the types of [known plaintext attacks](http://www.crypto-it.net/eng/attacks/known-plaintext.html). The intruder has to know some parts of plaintext and their ciphertexts. Using meet-in-the-middle attacks it is possible to break ciphers, which have two or more secret keys for multiple encryption using the same algorithm.



Kuch ciper or kuch plain text hota h or use key generate ho sktey h. ek end se encryption dosree end se description phir job anta h wo meet in the middle attack hota h



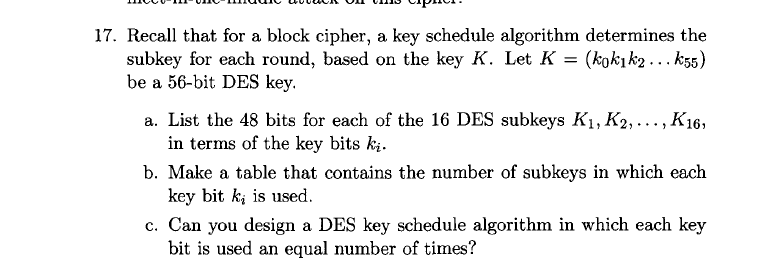
It is advantage of Double DES

17. Recall that for a block cipher, a key schedule algorithm determines the subkey for each round, based on the key K. Let K = (fcofci&2 · · · ^55) be a 56-bit DES key.

a. List the 48 bits for each of the 16 DES subkeys K\, Ki, , Kie, in terms of the key bits fcj.

b. Make a table that contains the number of subkeys in which each key bit ki is used.

c. Can you design a DES key schedule algorithm in which each key bit is used an equal number of times?



18. Recall the meet-in-the-middle attack on double DES discussed in this chapter. Assuming that chosen plaintext is available, this attack recovers a 112-bit key with about the same work needed for an exhaustive search to recover a 56-bit key, that is, about 255.

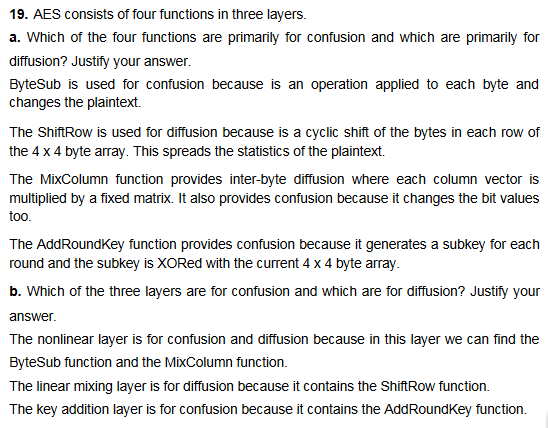
a. If we only have known plaintext available, not chosen plaintext, what changes do we need to make to the double DES attack?

b. What is the work factor for the known plaintext version of the meet-in-the-middle double DES attack?

19. AES consists of four functions in three layers.

a. Which of the four functions are primarily for confusion and which are primarily for diffusion? Justify your answer.

b. Which of the three layers are for confusion and which are for diffusion? Justify your answer.



20.Implement the Tiny Encryption Algorithm (TEA).

a. Use your TEA algorithm to encrypt the 64-bit plaintext block 0x0123456789ABCDEF

using the 128-bit key

0xA56BABCD00000000FFFFFFFFABCDEF01.

Decrypt the resulting ciphertext and verify that you obtain the original plaintext.

b. Using the key in part a, encrypt and decrypt the following message using each of the three block cipher modes discussed in the text

(ECB mode, CBC mode, and CTR mode).

Four score and seven years ago our fathers brought forth on this continent, a new nation, conceived in Liberty, and dedicated to the proposition that all men are created equal.

21. Give a diagram analogous to that in Figure 3.2 for the TEA cipher.

**TEA:** TEA is that it's not a Feistel cipher, and so we need separate encryption and decryption routines. However, TEA is about as close to a Feistel cipher as it is possible to be without actually

being one—TEA uses addition and subtraction instead of XOR. But the need for separate encryption and decryption routines is a minor concern with TEA, since so few lines of code are required, and the algorithm is reasonably efficient even with the large number of rounds. The TEA decryption algorithm, assuming 32 rounds .There is a somewhat obscure related key attack on TEA [163]. That is, if a cryptanalyst knows that two TEA messages are encrypted with keys that are related to each other in some very special way, then the plaintext can be recovered. This is a low-probability attack that in most circumstances can probably safely be ignored. But in case you are worried about this attack, there is a slightly more complex variant of TEA, known as extended TEA, or XTEA [218], that overcomes this potential problem. There is also a simplified version of TEA, known as STEA, that is extremely weak and is used to illustrate certain types of attacks

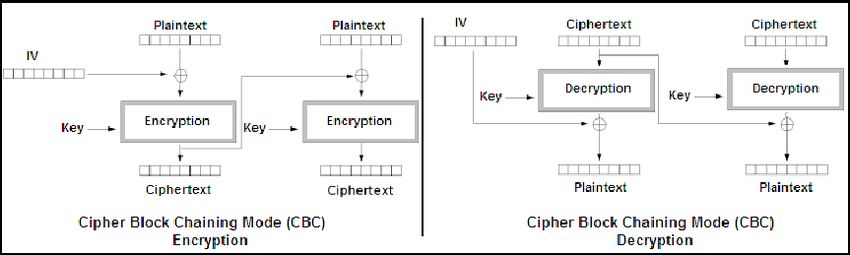
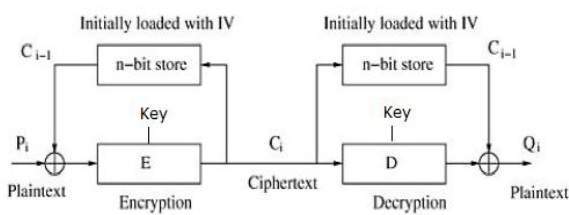
22. Recall that an initialization vector (IV) need not be secret.

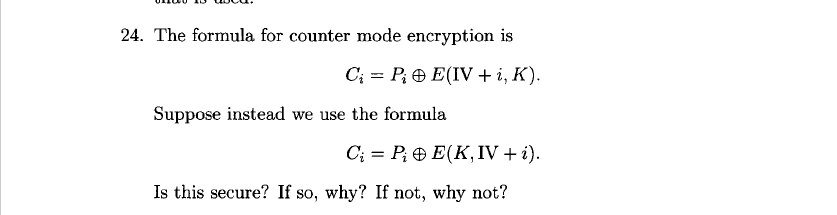
a. Does an IV need to be random?

b. Discuss possible security disadvantages (or advantages) if IVs are selected in sequence instead of being generated at random.

23. Draw diagrams to illustrate encryption and decryption in CBC mode.

Note that these diagrams are independent of the particular block cipher that is used.

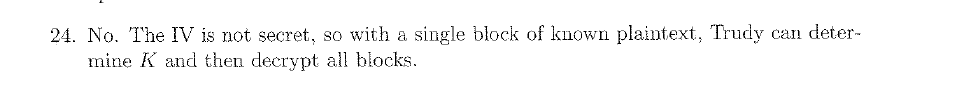




24. The formula for counter mode encryption is Ci = Pi®EQV + i,K).

Suppose instead we use the formula Ci = Pi®E(K,W + i).

Is this secure? If so, why? If not, why not?



🡺Agar sirf ek bho plaintext Trudy ko milgaya toh use k paat chl jayegi or wo har block ko decrypt kr payega

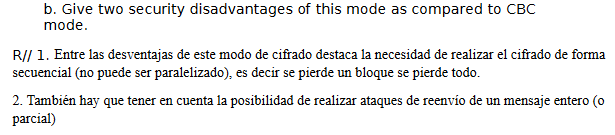
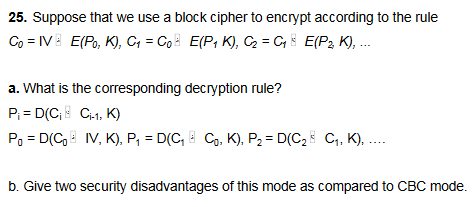
25. Suppose that we use a block cipher to encrypt according to the rule

Co = IV È E(PQ, K), d = Co e E(PU K), C2 = C1® E{P2, K), ...

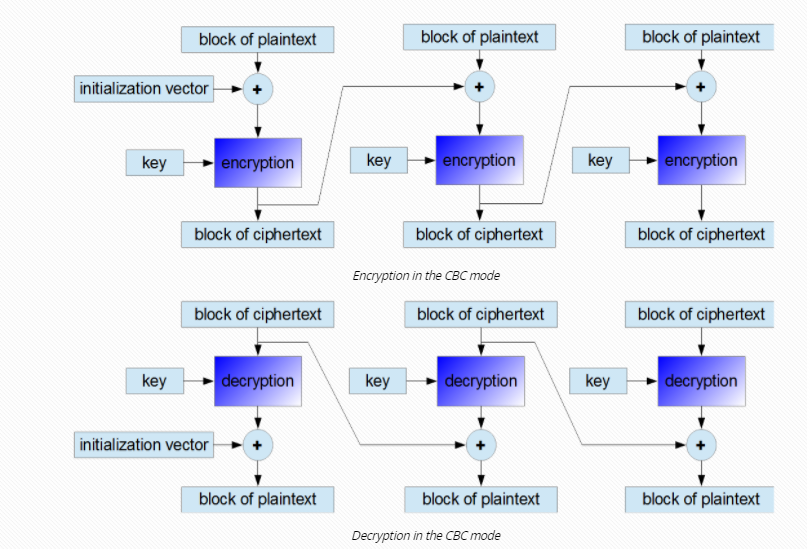
a. What is the corresponding decryption rule?

b. Give two security disadvantages of this mode as compared to CBC

mode.



26. Suppose that ten ciphertext blocks are encrypted in CBC mode. Show that a cut-and-paste attack is possible. That is, show that it is possible to rearrange the blocks so that some of the blocks decrypt correctly, in spite of the fact that the blocks are not in the correct order.



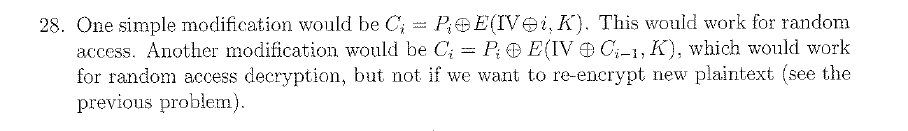
If one bit of a plaintext message is damaged (for example because of some earlier transmission error), all subsequent ciphertext blocks will be damaged and it will be never possible to decrypt the ciphertext received from this plaintext. As opposed to that, if one ciphertext bit is damaged, only two received plaintext blocks will be damaged. It might be possible to recover the data.

A message that is to be encrypted using the CBC mode, should be [extended](http://www.crypto-it.net/eng/theory/padding.html) till the size that is equal to an integer multiple of a single block length (similarly, as in the case of using the ECB mode).

27. Explain how to do random access on data encrypted in CBC mode. Are there any significant disadvantages of using CBC mode for random access as compared to CTR mode?

CBC m agar anpe text m replacement ki toh koi faida h par agar apko key or cipher tex ci and C(i-1) pata h toh you can easily got P (plain text)

28. CTR mode generates a keystream using a block cipher. Devise another method for using a block cipher as a stream cipher. Does your method support random access?



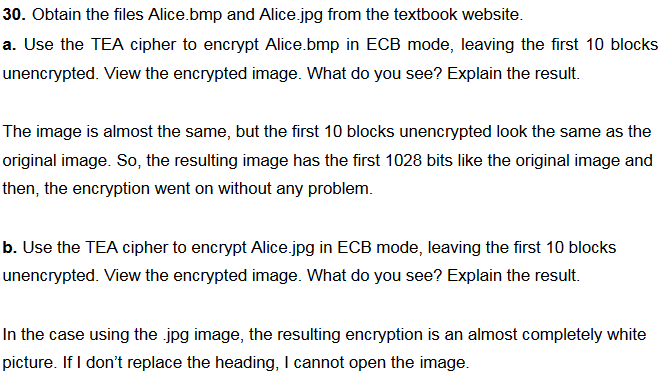
**Manual**

29. Suppose that the ciphertext in equation (3.8) had been encrypted in CBC mode instead of ECB mode. If Trudy believes ECB mode is used and tries the same cut-and-paste attack discussed in the text, which blocks decrypt correctly?

30. Obtain the files Alice.bmp and Alice.jpg from the textbook website.

a. Use the TEA cipher to encrypt Alice.bmp in ECB mode, leaving the first 10 blocks unencrypted. View the encrypted image. What do you see? Explain the result.

b. Use the TEA cipher to encrypt Alice.jpg in ECB mode, leaving the first 10 blocks unencrypted. View the encrypted image. What do you see? Explain the result.



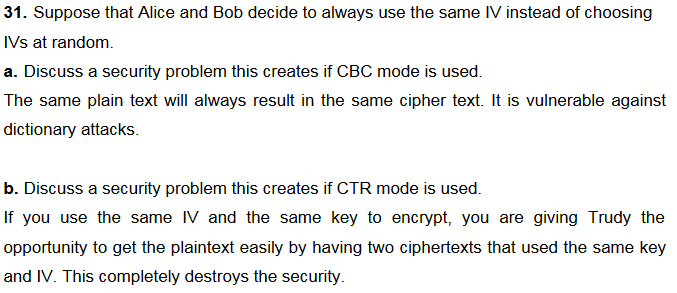
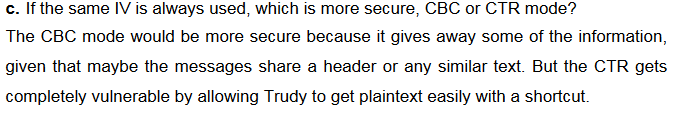
31. Suppose that Alice and Bob decide to always use the same IV instead

of choosing IVs at random.

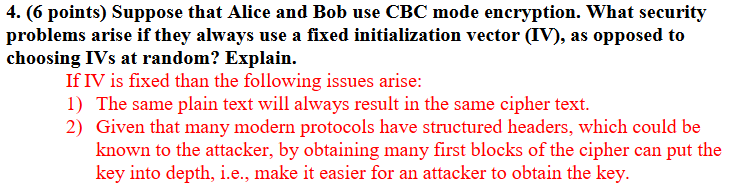
a. Discuss a security problem this creates if CBC mode is used.

b. Discuss a security problem this creates if CTR mode is used.

c. If the same IV is always used, which is more secure, CBC or CTR mode?

32. Suppose that Alice and Bob use CBC mode encryption.

a. What security problems arise if they always use a fixed initialization vector (IV), as opposed to choosing IVs at random? Explain. 

b. Suppose that Alice and Bob choose IVs in sequence, that is, they first use 0 as an IV, then they use 1 as their IV, then 2, and so on.

Does this create any security problems as compared to choosing the IVs at random?

🡺is case m koi masla nh hoga qk proctocl aise hi define tha

33. Give two ways to encrypt a partial block using a block cipher. Your first method should result in ciphertext that is the size of a complete block, while your second method should not expand the data. Discuss any possible security concerns for your two methods.

34. Recall that a MAC is given by the CBC residue, that is, the last ciphertext block when the data is encrypted in CBC mode. Given data X, key K, and an IV, define F(X) to be the MAC of X.

a. Is F one-way, that is, given F(X) is it possible to determine ×º

b. Is F collision resistant, that is, given F(X) is it possible to find a value Y such that F(Y) = F(X)7

35. Suppose Alice uses DES to compute a MAC. She then sends the plaintext, the IV, and the corresponding MAC to Bob. If Trudy alters one block of plaintext before Bob receives it, what is the probability that Bob will not detect the change?

**Manual**

36. Alice has four blocks of plaintext, Po,Pi,P2,P3, which she encrypts using CBC mode to obtain CQ,C\,C2,CZ. She then sends the IV and ciphertext to Bob. Upon receiving the ciphertext, Bob plans to verify the integrity as follows. He'll first decrypt to obtain the putative plaintext, and then he'll re-encrypt this plaintext using CBC mode and the received IV. If he obtains the same C3 as the final ciphertext block, he will trust the integrity of the plaintext.

a. Suppose that Trudy changes C\ to X, leaving all other blocks and the IV unchanged. Will Bob detect that the data lacks integrity?

🡪nahi qk usne dobara se decrupt k k encrupt kia h toh hoskta h cipher change hua ho is case m detect nh kr payegaa

b. Suppose that Trudy changes C3 to the value Y, leaving all other blocks and the IV unchanged. Will Bob detect that the data lacks integrity?

c. Is Bob's integrity checking method secure?

🡪N0 :pagalpaan h k cipher ko decrupt kr k dobara eccrupt kr k check krrha h

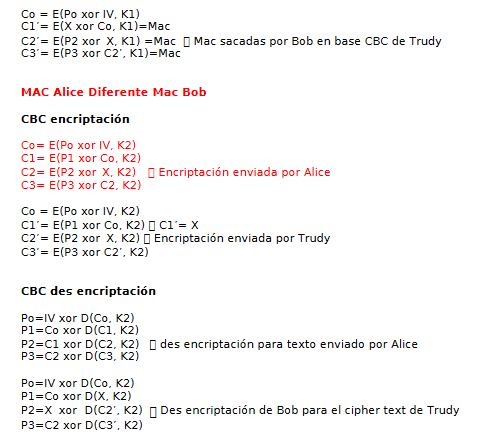
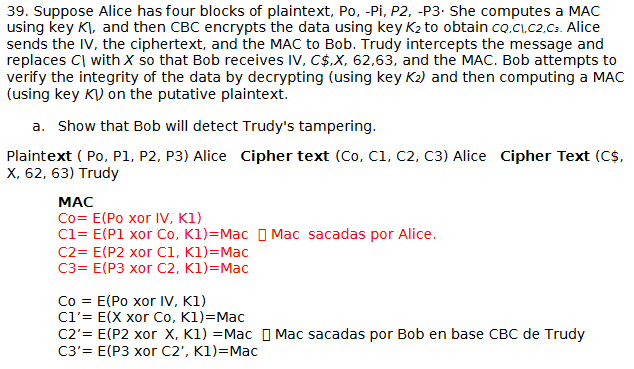
37. Using CBC mode, Alice encrypts four blocks of plaintext, PQ, P\, P2, P3 and she sends the resulting ciphertext blocks, CQ,C\,C2,CZ, and the IV to Bob. Suppose that Trudy is able to change any of the ciphertext blocks before they are received by Bob. If Trudy knows Pi, show that she can replace P\ with X. Hint: Determine C so that if Trudy replaces Co with C, when Bob decrypts C\, he will obtain X instead

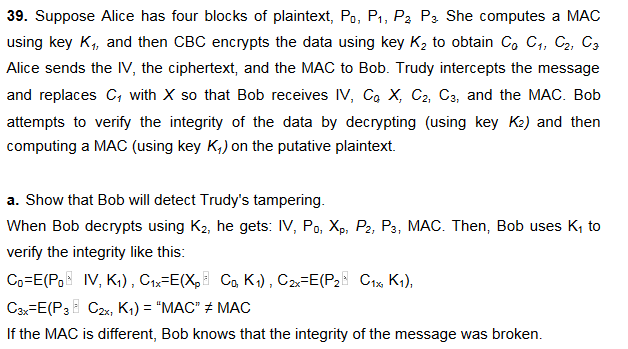
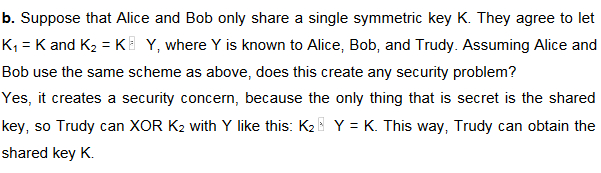
of Pi

38. Suppose we encrypt in CBC mode using the key K and we compute a MAC using the key K ® X, where X is a known constant. Assuming the ciphertext and the MAC are sent from Alice to Bob, show that Bob will detect a cut-and-paste attack.

39. Suppose Alice has four blocks of plaintext, Po, -Pi, P2, -P3· She computes a MAC using key K\, and then CBC encrypts the data using key K2 to obtain CQ,C\,C2,C3. Alice sends the IV, the ciphertext, and the MAC to Bob. Trudy intercepts the message and replaces C\ with X so that Bob receives IV, C$,X, 62,63, and the MAC. Bob attempts to verify the integrity of the data by decrypting (using key K2) and then computing a MAC (using key K\) on the putative plaintext.

a. Show that Bob will detect Trudy's tampering.

b. Suppose that Alice and Bob only share a single symmetric key K. They agree to let K\ = K and K2 = K (BY, where Y is known to Alice, Bob, and Trudy. Assuming Alice and Bob use the same scheme as above, does this create any security problem? 

40. Suppose that Alice and Bob have access to two secure block ciphers, say, Cipher A and Cipher B, where Cipher A uses a 64-bit key, while Cipher B uses a 128-bit key. Alice prefers Cipher A, while Bob wants the additional security provided by a 128-bit key, so he insists on Cipher B. As a compromise, Alice proposes that they use Cipher A, but they encrypt each message twice, using two independent 64-bit keys. Assume

that no shortcut attack is available for either cipher. Is Alice's approach as secure as Bob's?

41. Suppose that Alice has a secure block cipher, but the cipher only uses an 8-bit key. To make this cipher "more secure," Alice generates a random 64-bit key K, and iterates the cipher eight times, that is, she encrypts the plaintext P according to the rule

C = E(E(E(E(E(E(E(E(P, K0), Kx), K2), K3), KA), K5), K6), K7),

where Ko, K\,..., Êã are the bytes of the 64-bit key K.

a. Assuming known plaintext is available, how much work is required to determine the key K1

b. Assuming a ciphertext-only attack, how much work is required to break this encryption scheme?

42. Suppose that we define triple 3DES with a 168-bit key as

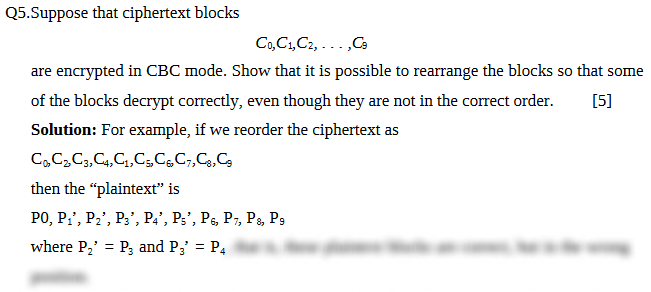
C = E(E(E(P,K1),K2),K3).

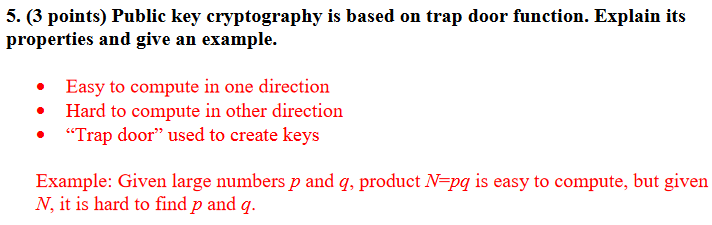
Suppose that we can compute and store a table of size 256, and a chosen plaintext attack is possible. Show that this triple 3DES is no more secure than the usual 3DES, which only uses a 112-bit key. Hint: Mimic the meet-in-the-middle attack on double DES.

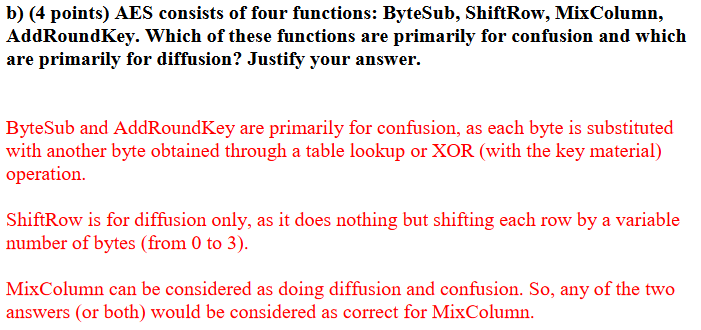
43. Suppose that you know a MAC value X and the key K that was used to compute the MAC, but you do not know the original message. (It may be instructive to compare this problem to Problem 16 in Chapter 5.)

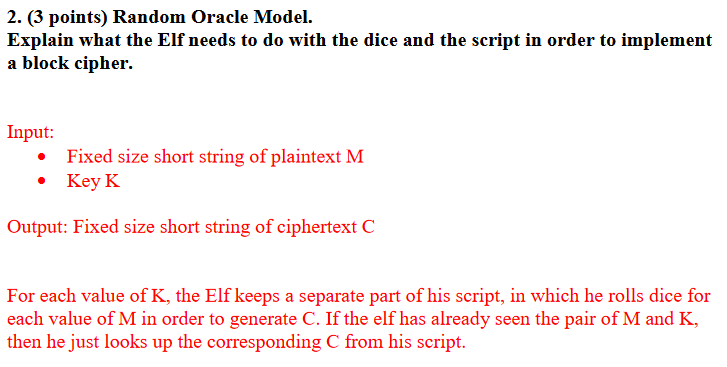
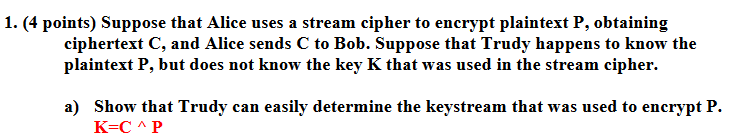
a. Show that you can construct a message M that also has its MAC equal to X. Note that we are assuming that you know the key K and the same key is used for both MAC computations.

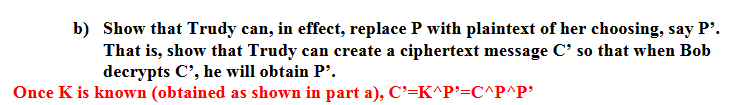
b. How much of the message M are you free to choose?



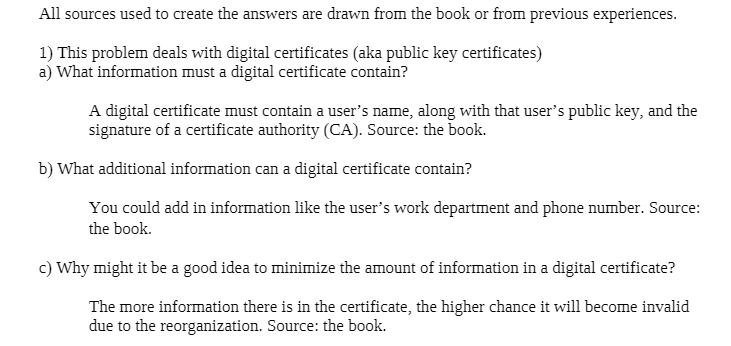


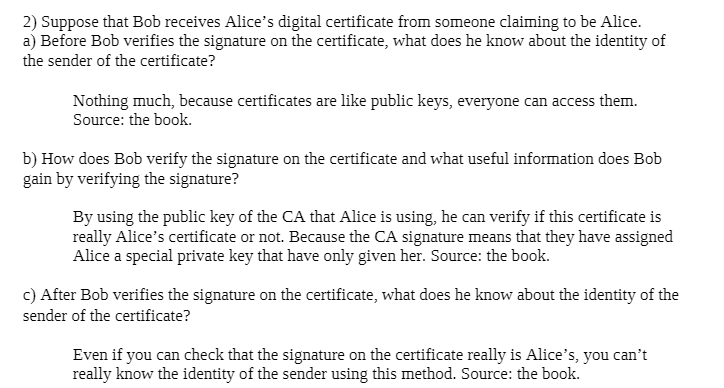


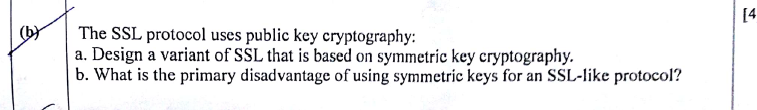
 



MID







https://www.digicert.com/ssl-cryptography.htm