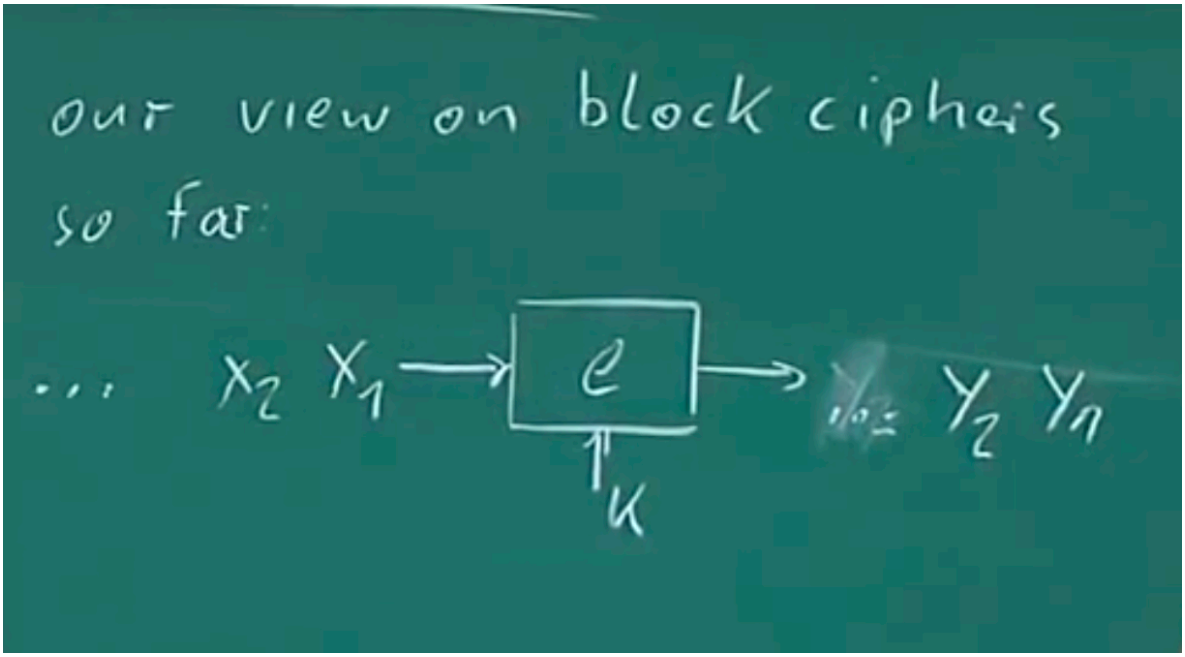
Lecture 9 Mode of Operation for Block Ciphers:

We have not done the Key Schedule and Decryption in AES.

Introduction:



Above is the case where we are having more than one block of data and we are encrypting block by block.
so Question is :- is this the good way to encrypt data which is more than one block ? or is there any other good way to do the same.

As in general practise we are always going to encrypt the data which is more than the block size as block sizes are very small eg for email encryption, 128 bit or 16 bytes(characters) is a very small. So how we are going to do that ?

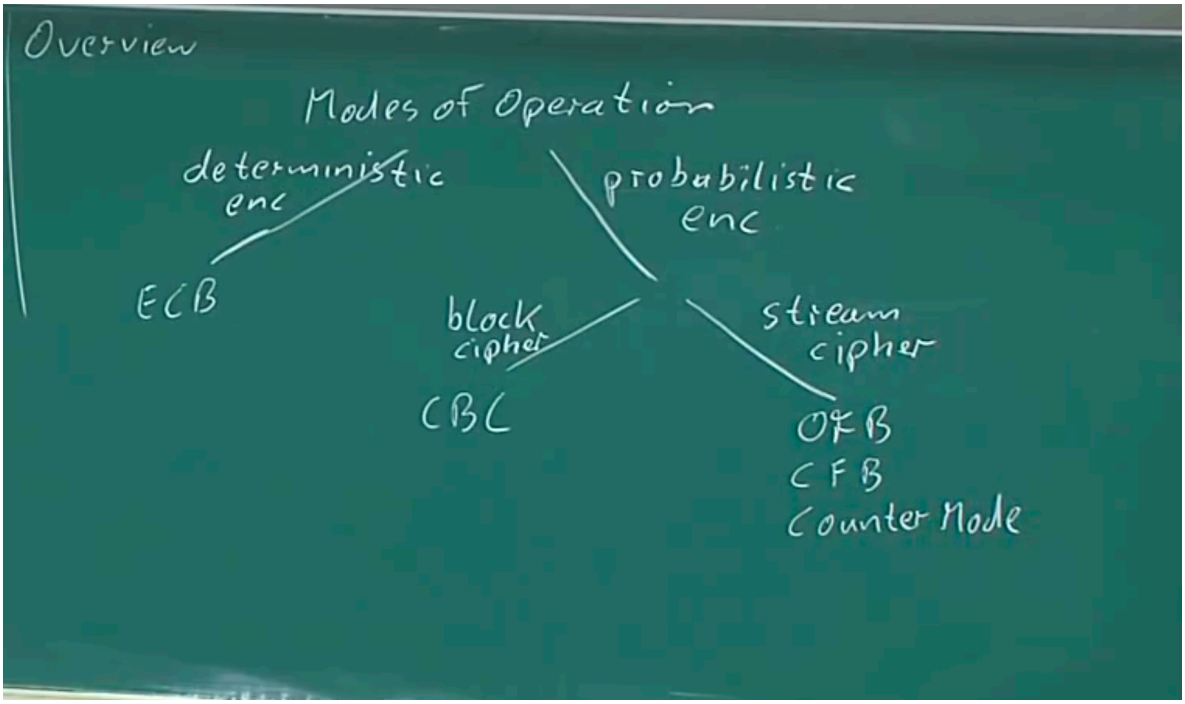
Remarks :
Block Ciphers can be used for many other tasks:

- Different Encryption Schemes (the above picture is one of the stupid way of doing encryption. we will see later in chapter.)
- Using Block Cipher as Stream Ciphers. ie we can take block cipher and turn it into Stream Cipher
- Using Block Cipher to create PRNG
- to create Hash functions
- to create MACs

So block cipher are not only used for Encryption or Decryption but they can be used for other multiple purposes as mentioned above.

In this chapter we are going to do Different Encryption Schemes (ECB, CFB) and Block Cipher as Stream Cipher (OFB).

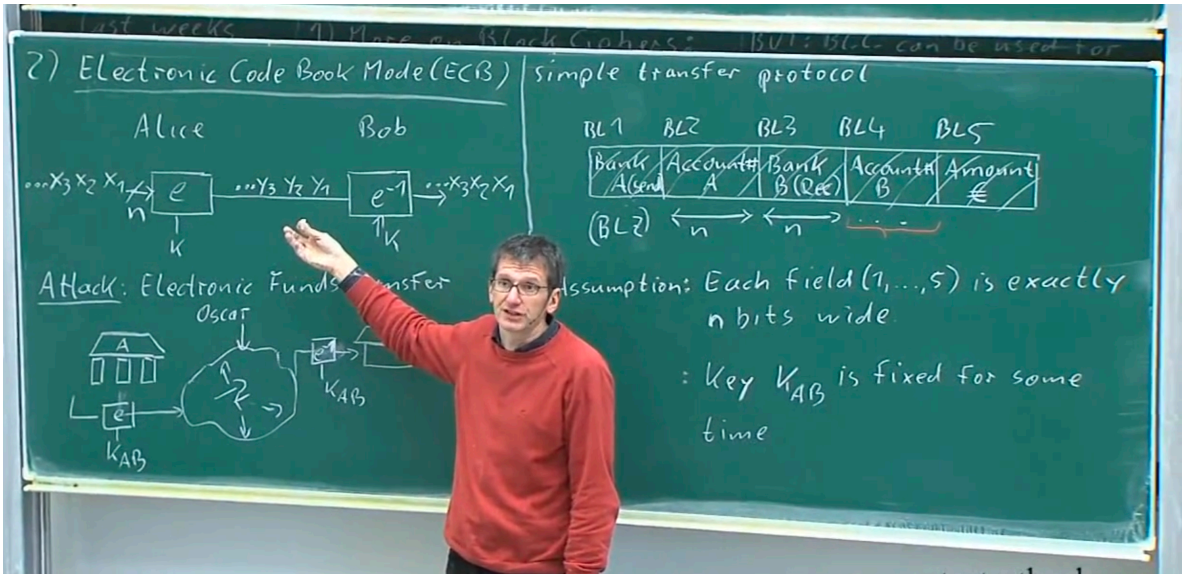
When we talk about modes of Operations, it means ways for using block cipher for encryption ie say i have given a big PDF or Movie file for encrypting and a AES software then how will you encrypt it?
One way is as mentioned in above pic. we are also going to learn why that is a bad way for doing encryption.

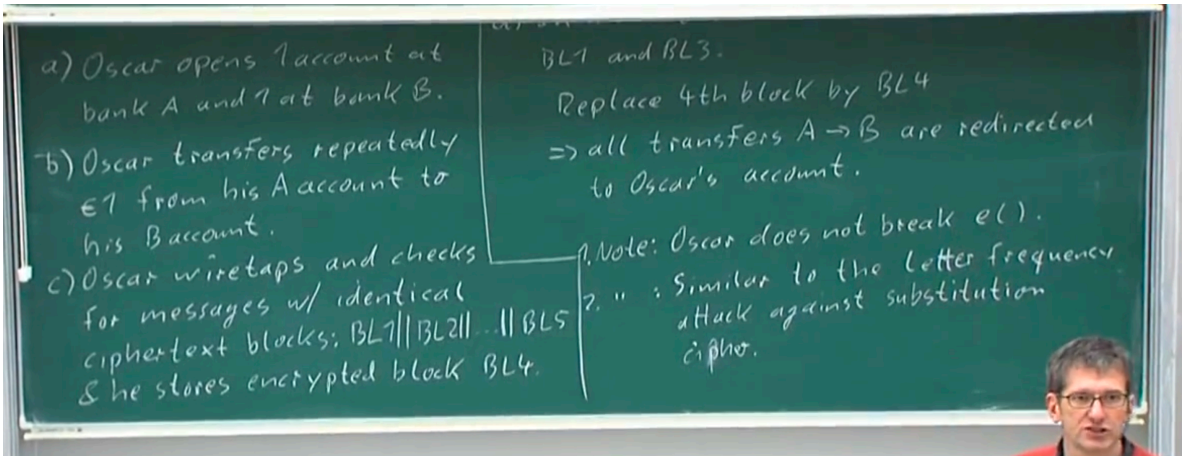


We are going to talk about ECB, CBC and OFB.
Important thing is we are going to learn more about "Probabilistic Encryption".

Electronic Code Book mode (ECB) :
Problem statement is AES encrypts 128 bits but now say 1 MB file we need to encrypt, then how can we do that ?
ECB mode is same as the above described, intuitive way.

Attacks on ECB:





So Summary is there are 2 banks A and B and they are transfering transactions/ Amount using ECB mode.
Assumption: Now a transaction is divided into 5 parts with each part is of same size and that too equal to Block size for Encryption Algorithm say AES or DES

Now Oscar, who can listen on the wire tape and also can modify contents of the packets, opens account in both A and B bank. After that he does multiple transfer from A to B and analysed the traffic, he noted down the block 4 which is his bank B's account number in encrypted form.

Now in every transfer from A to B bank he will just replace the block4 and can become rich.
Note: to know if transfer is from A to B bank, he can prerecord the encrypted A and B bank blocks.

Oscar has not broken Encryption and also this is similar to letter frequency attack ie here instead of letters, blocks are doing the same.

ECB because fixed mapping if key is same and it is similar to book with 2^128 entries.

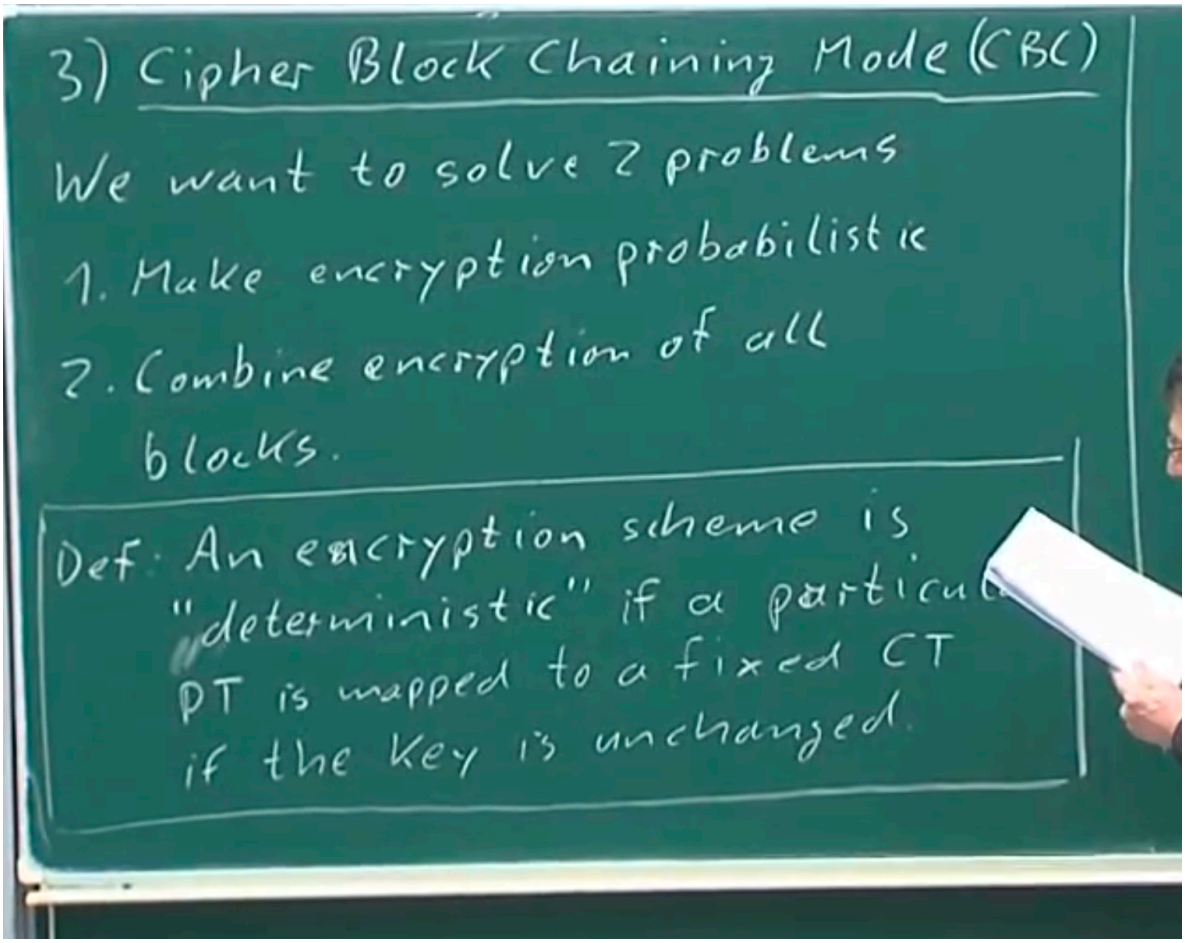
So what we want is if we encrypt same plain text twice we want to get different cipher text. This might make you think that AES is deterministic algorithm then how this can happen.

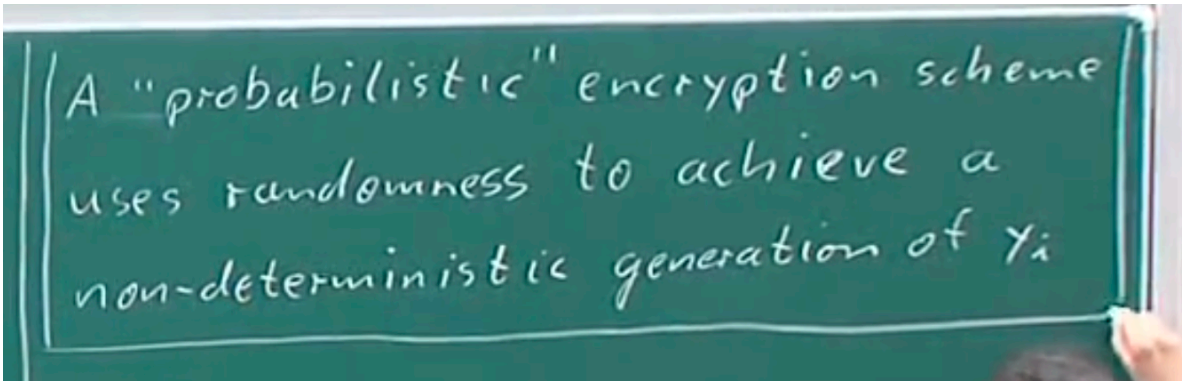
Lets see other modes :

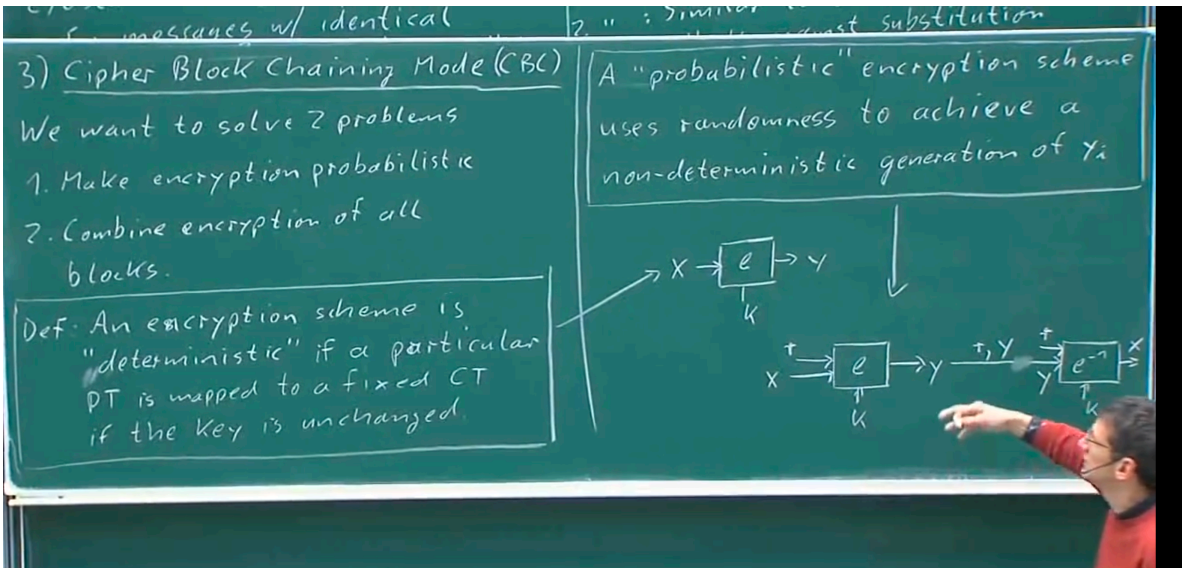
Cipher Block Chaining (CBC):
we are solving 2 problems with CBC:

1. Make Encryption **Probabilistic**
2. There are no link between one block and other, CBC combine the encryption of all blocks (Chaining)

*Definition: An Encryption schema is "**deterministic**" if a particular PlainText is mapped to a FIxed Cipher text if key remains the same.*







here in above diagram, in case of probabilistic encryption, r (randomness) is passed to encryption scheme with plain text and then it is distributed through the channel so that decryption function can use the same r to decrypt it.

Here point to consider is that r is not key, it can be sent over the channel. There is no secrecy in r. Now question is why r is not a secret ?
ok analogy is y1 is the IV for y2 and y2 is IV for y3 so even incase r is secret, for other blocks IV is known. so all the n-1 IV's are exposed to public.

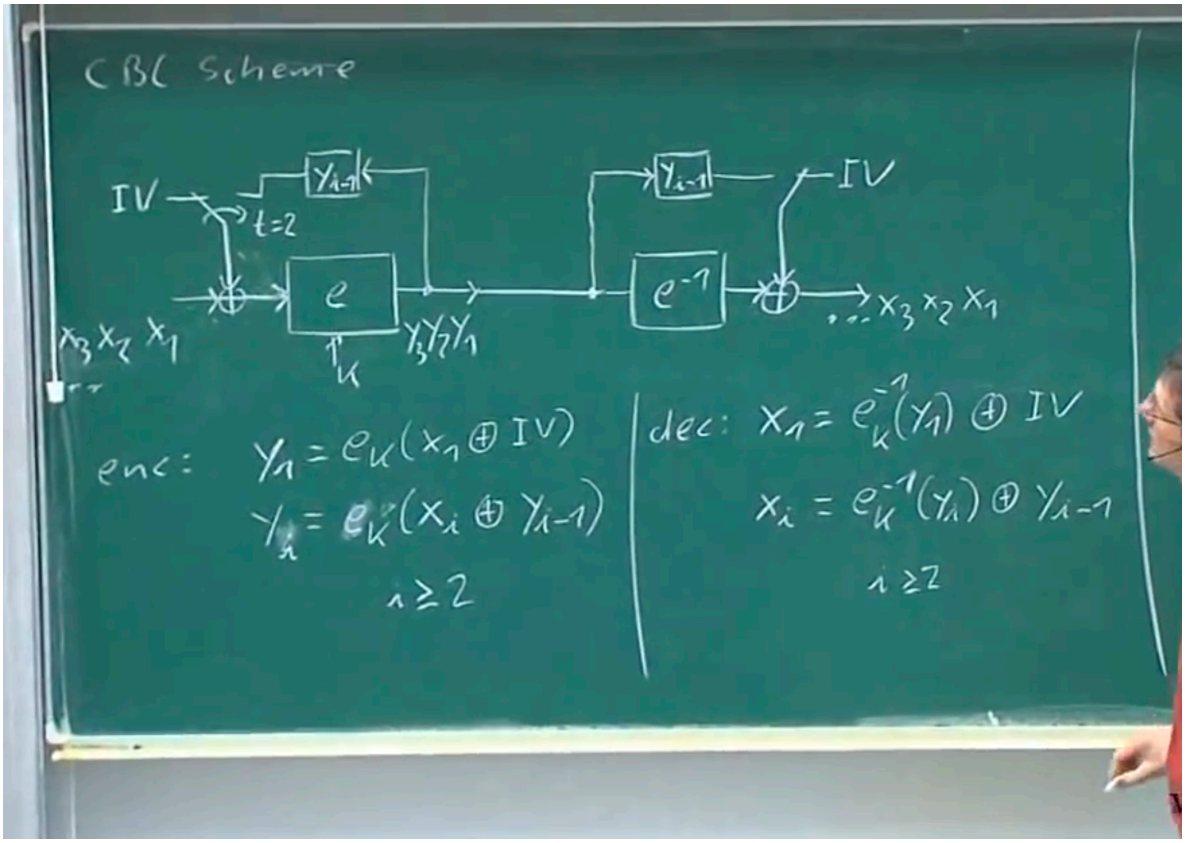
thinking on if it is not secret does it exposes any risk ?
Same bank case, now if IV is not secret and random then say hacker has the cipher text and iv both still he cannot do anything reason being he needs key to decrypt and then do XOR with iv.
Also incase attacker has plain text and cipher text for multiple transactions, they are useless as due to chaining he cannot replace one plaintext with another without the key.

Two transactions means one by hacker and one by victim.

CBC scheme:

rough:
y2 = enc(x2 xor y1)
y1 = enc(x1 xor IV)

decryption
dec(y1) xor IV
dec(y2) xor y1



Question why each block is not encrypted with r or IV ?, r or IV are making encryption as probabilistic where as we have another requirement of combining all encryption blocks so for that we need to chain in a way such that block Xi is encrypted using input from Xi-1.

Also there is an attack possibility, say attacker has cipher text and plain text for his transaction and victims transaction, and as IV is not secret, he found the say same IV is used then he can replace any block with its block but if chaining is there then it will make the entire encryption invalid. How ever CBC doesn't provide integrity (https://security.stackexchange.com/questions/9437/does-symmetric-encryption-provide-data-integrity)
something like below:
say IV + p is the cipher text sent then if i send IV as iv + p + p'

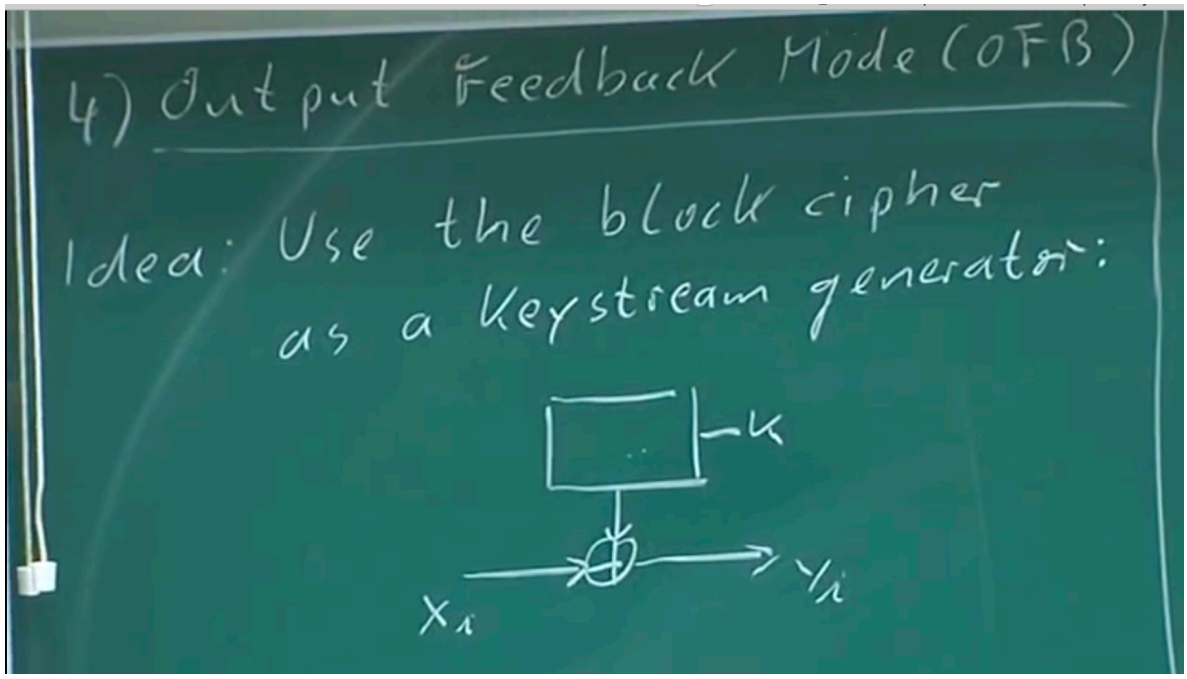
Initialization vector:

1. there is no need for IV to be secret
2. it should be nonce ie non repeating number/number only used once

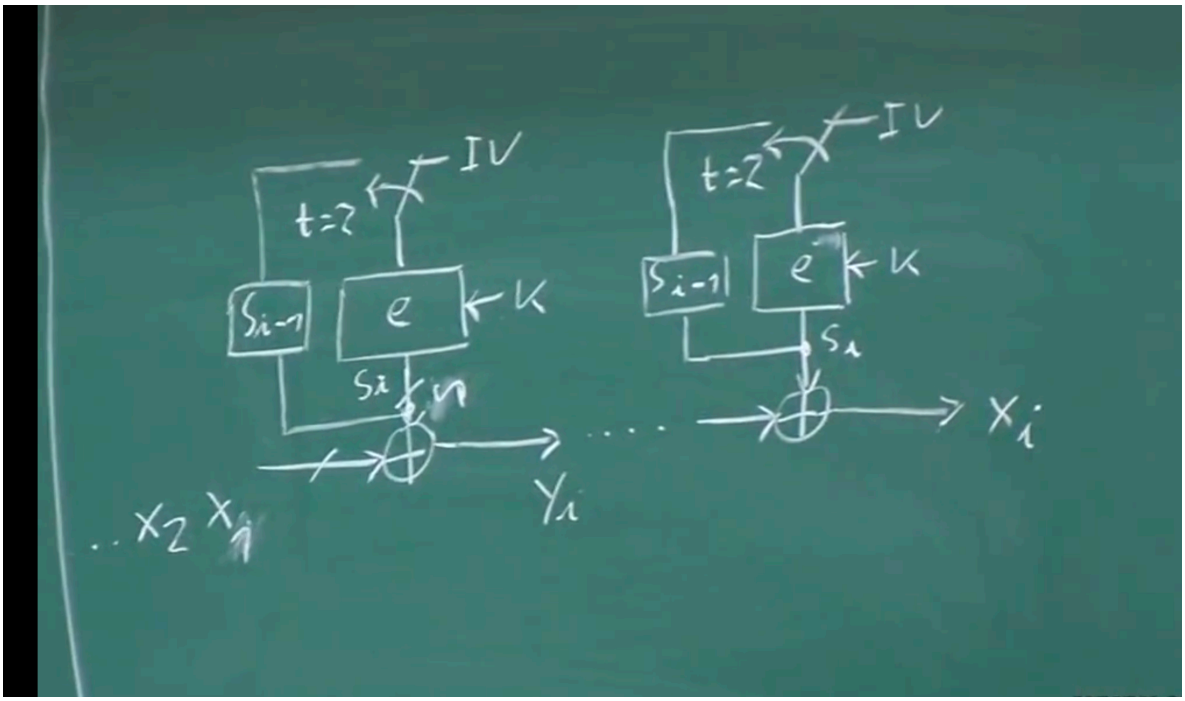
Ways for generating IV

1. True Random number generator —> issue is number might repeat
2. Counter/Sequence
3. using IP of Alice and IP of bob and time in millis concatenated.
4. there are many more ways.

How we can use Block cipher as Stream cipher:
Output Feedback Mode (OFB Mode):



Idea is to use Block cipher as a Keystream generator. some days back we have used LFSR's as Key Stream Generators but in this mode we are trying to use AES/triple DES as key stream generator.



So simple way is using a block cipher to generate Key stream passing some nonce, why nonce ? if we don't pass nonce then key stream generated will be same and it makes stream cipher insecure.

so similar to CBC we are here taking a nonce and generating 128 bits in case of AES and then using the output as the input IV for next block.

Note: we have not used decryption of block cipher in decrypting stream cipher why?
reason being is we want to generate the keystream not decrypting the block cipher so key stream generation process will remain same.

If Counter is used as IV then it is called CTR mode.