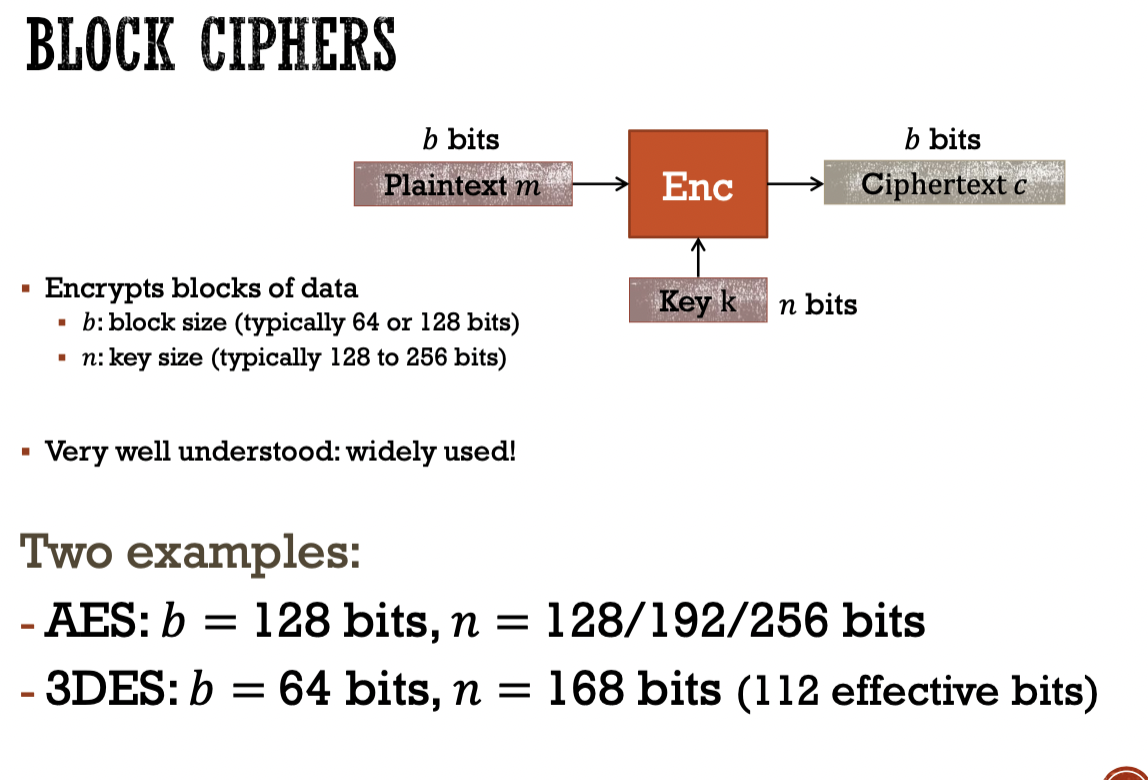
Class Notes

September 13th, 2021

**Block Ciphers**



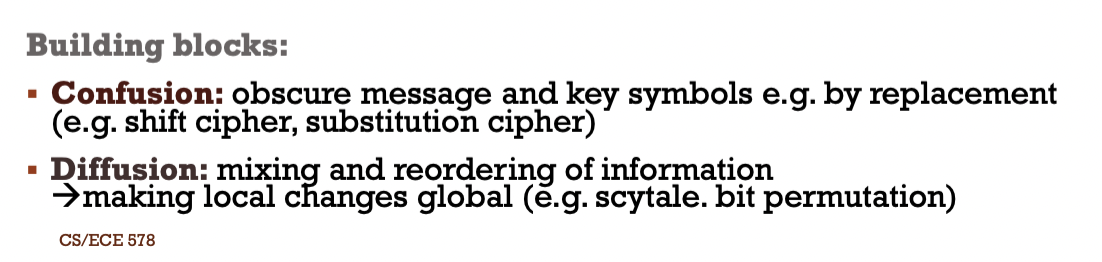
The size of the input, and the size of the output is the same.

DES block sizes are short, so there is a way to exhaustive search them. Which is why it is no longer in use anymore.

3DES is still around but it recommend to stay away from using it in your applications and servers.

The key size doesn’t give it good enough secuirty

Block Ciphers are based on doing this:

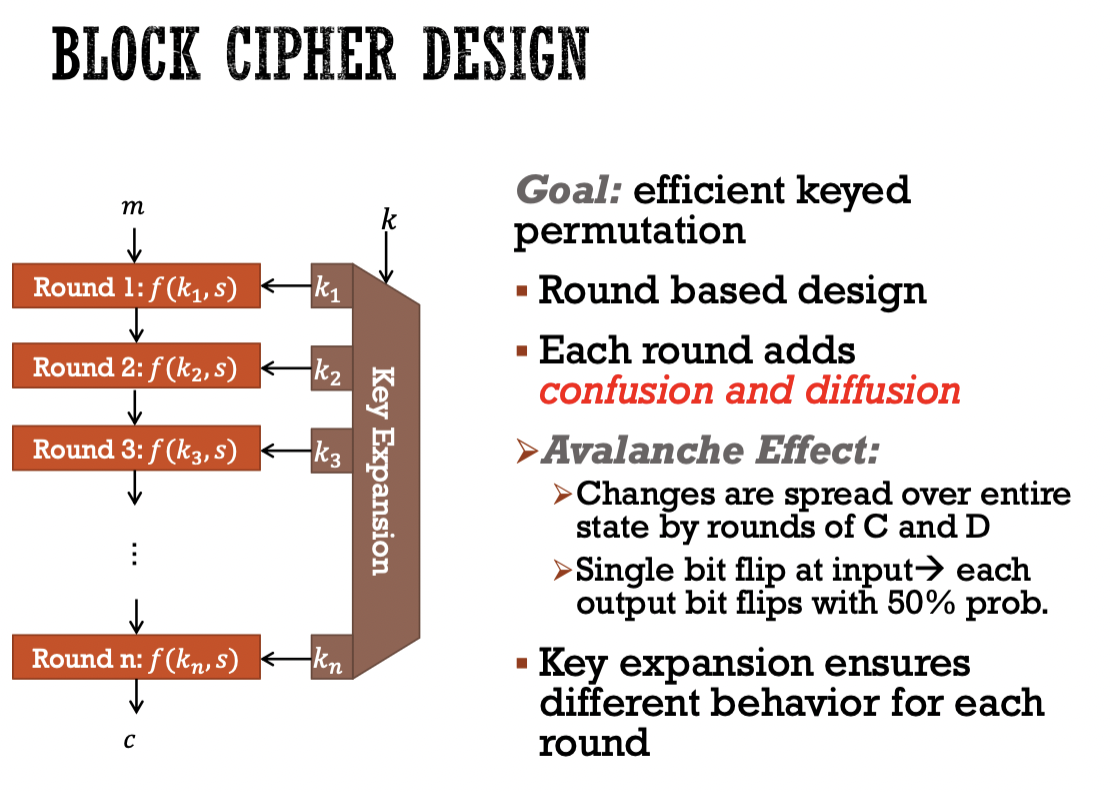


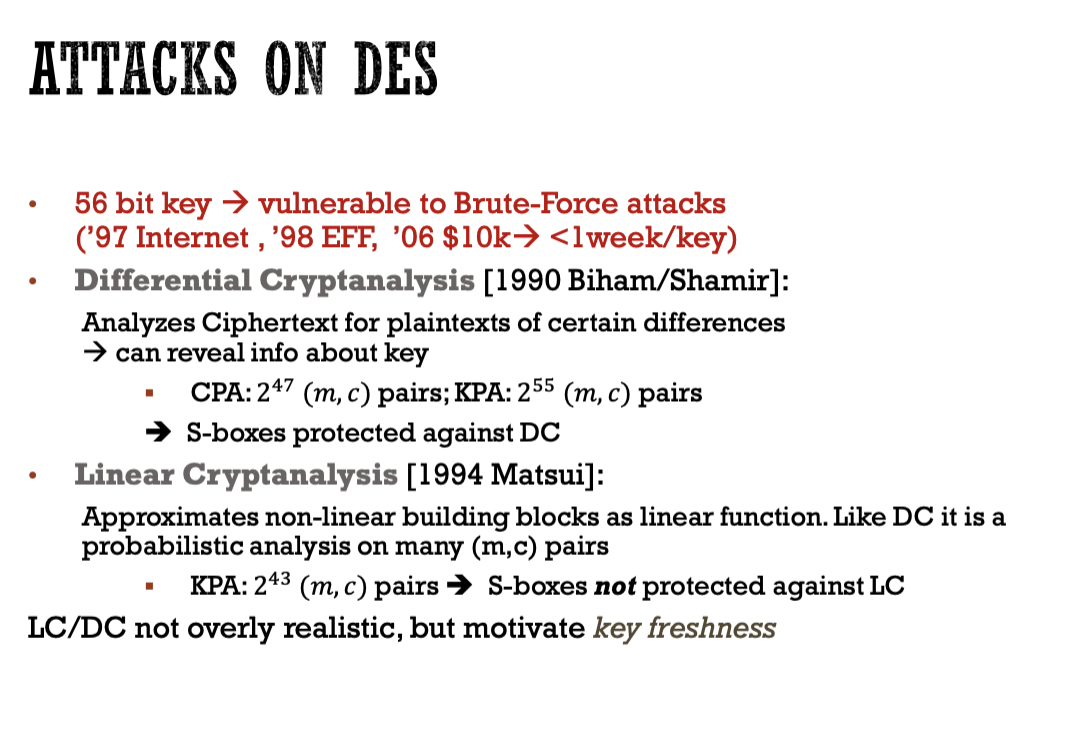
The design goal is efficient KEY permutation.

A Key is generated for each round in order to increase security. They’re called Round Keys

The rounds are effective for ensuring the randomness of the cipher text

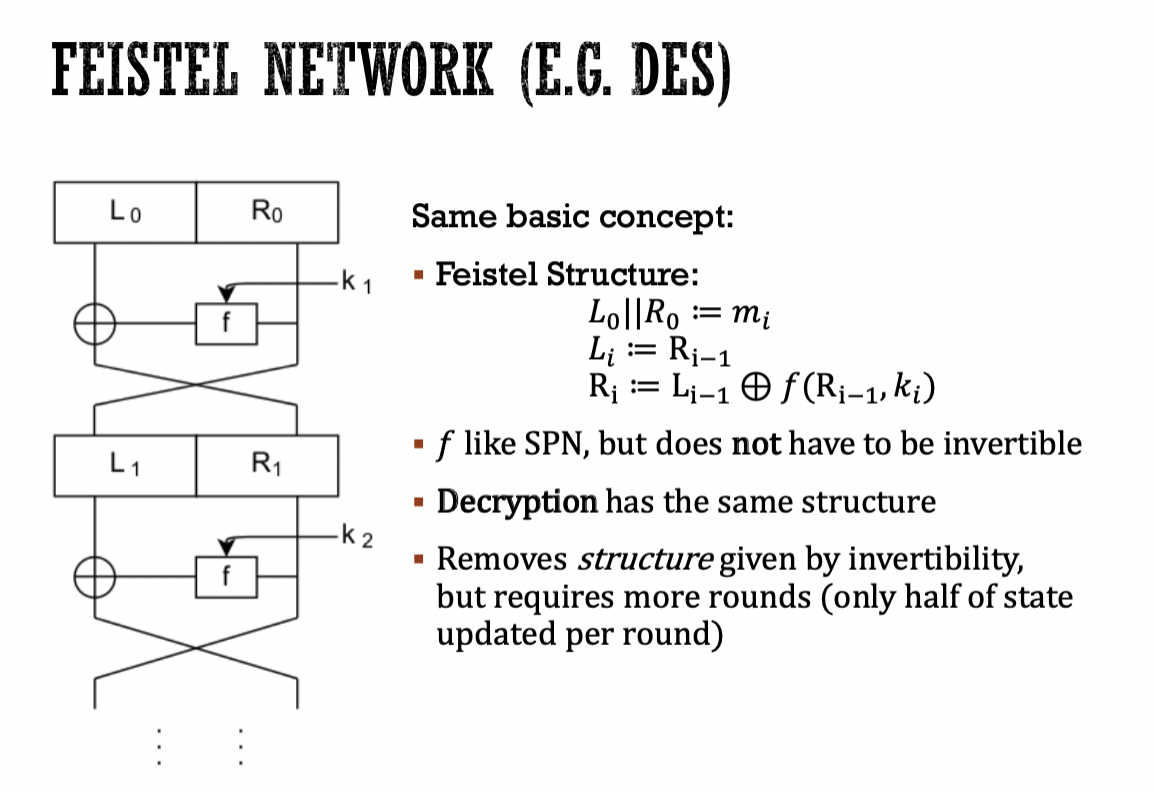
Encryption and Decryption only differ in the key schedule





If you do not change your key, then DC and LC can crack the cryptography. Which is why AES became the standard later down the line.





Called a phase out structure.

F = Round Function. 1 Input & 1 Key

L0 R0

| |

XOR---- F -------|

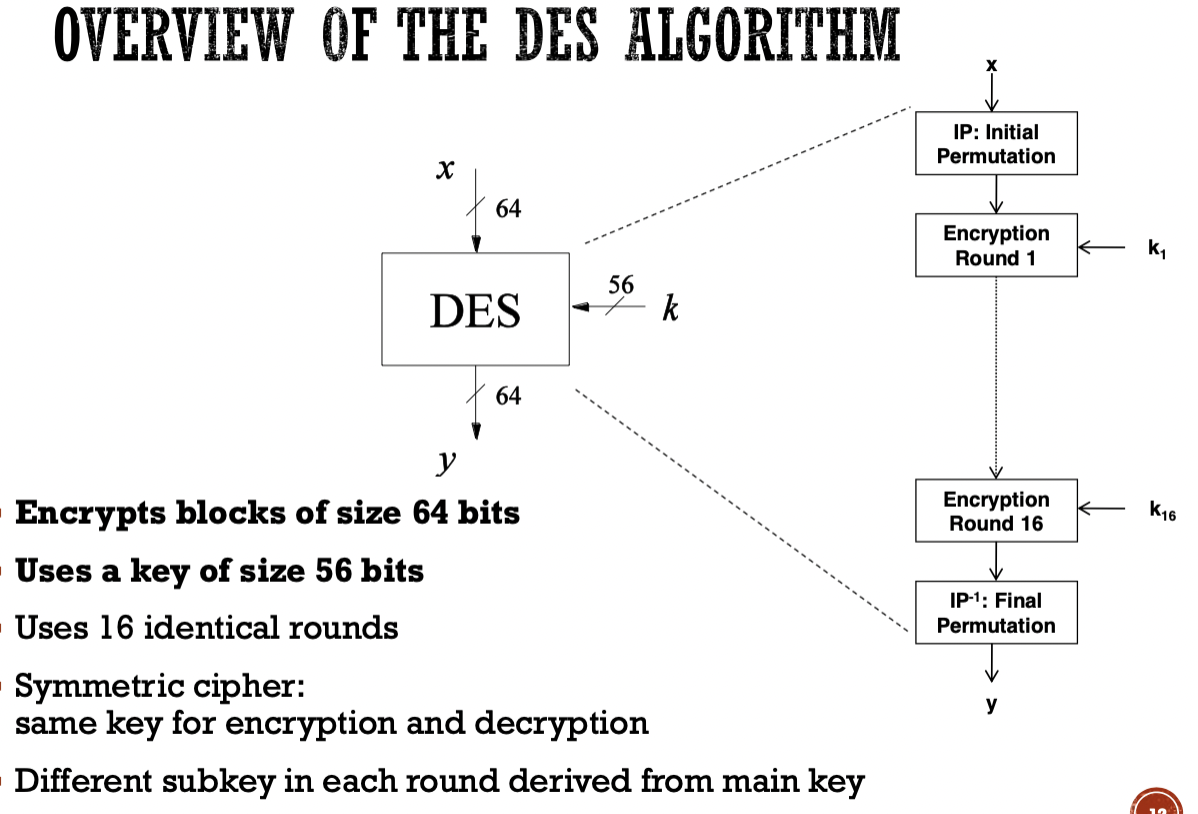
X

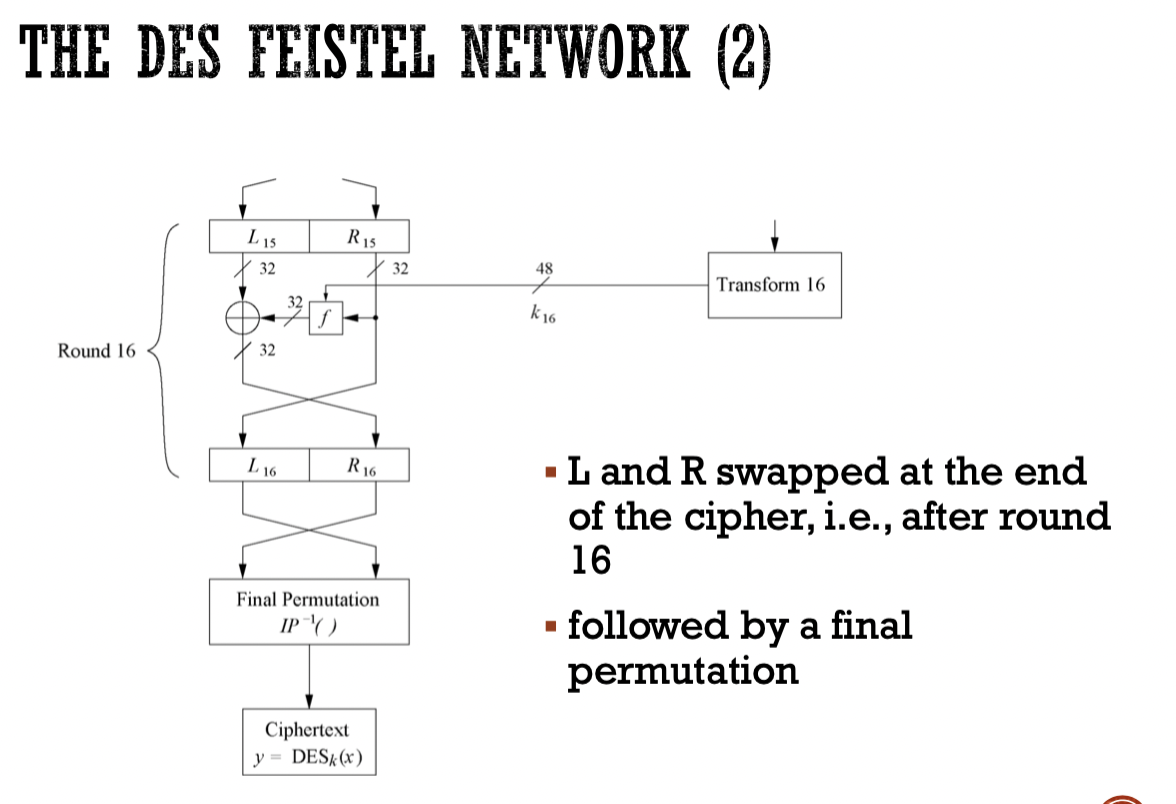
| |

L1 R1

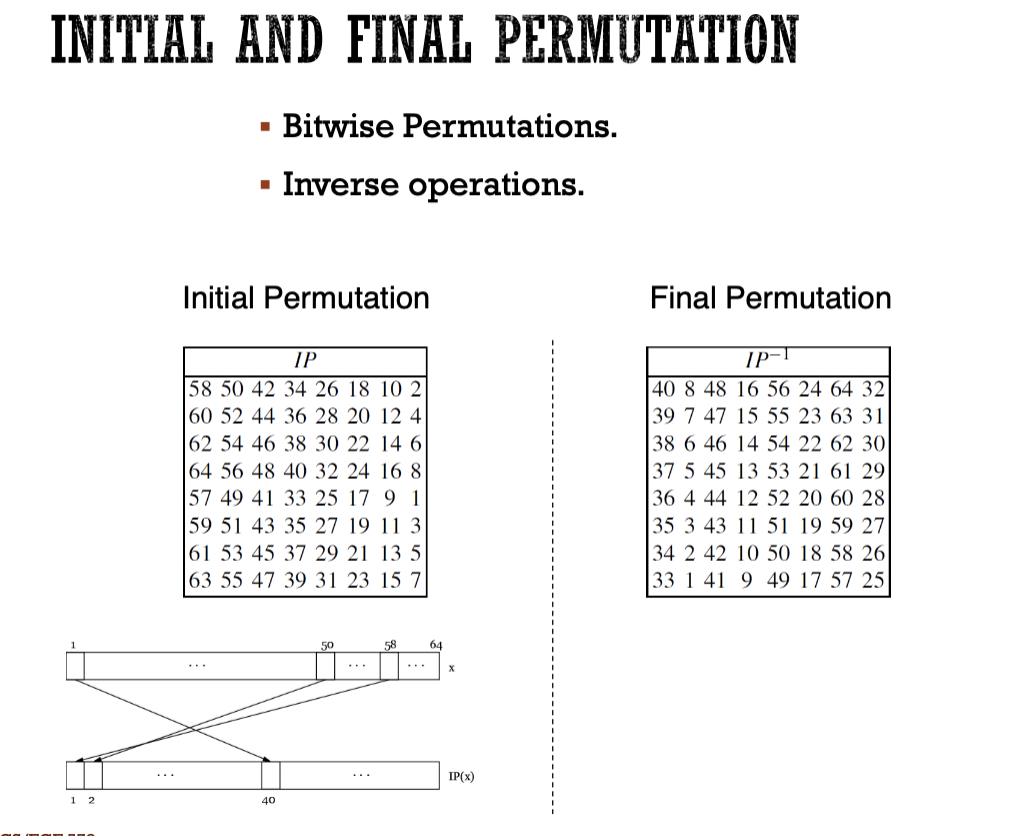
If you just XOR’d R1 and L0, you would find the key, which is why we need multiple rounds, otherwise it is very trivial to crack the keys.

Since the F function in each round does not need to invertible, input size and output sizes do not have to be the same. Making encryption stronger because there are more options.

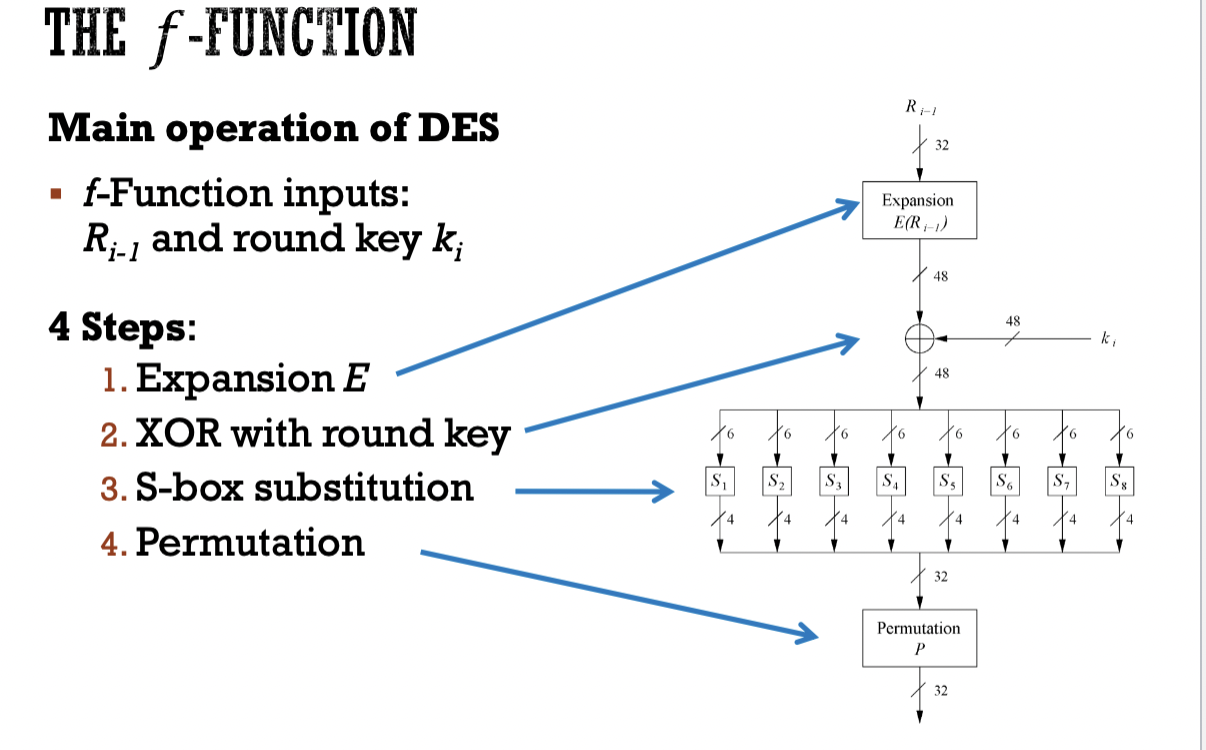




The final Permutation is just the inverse of the initial permutation.



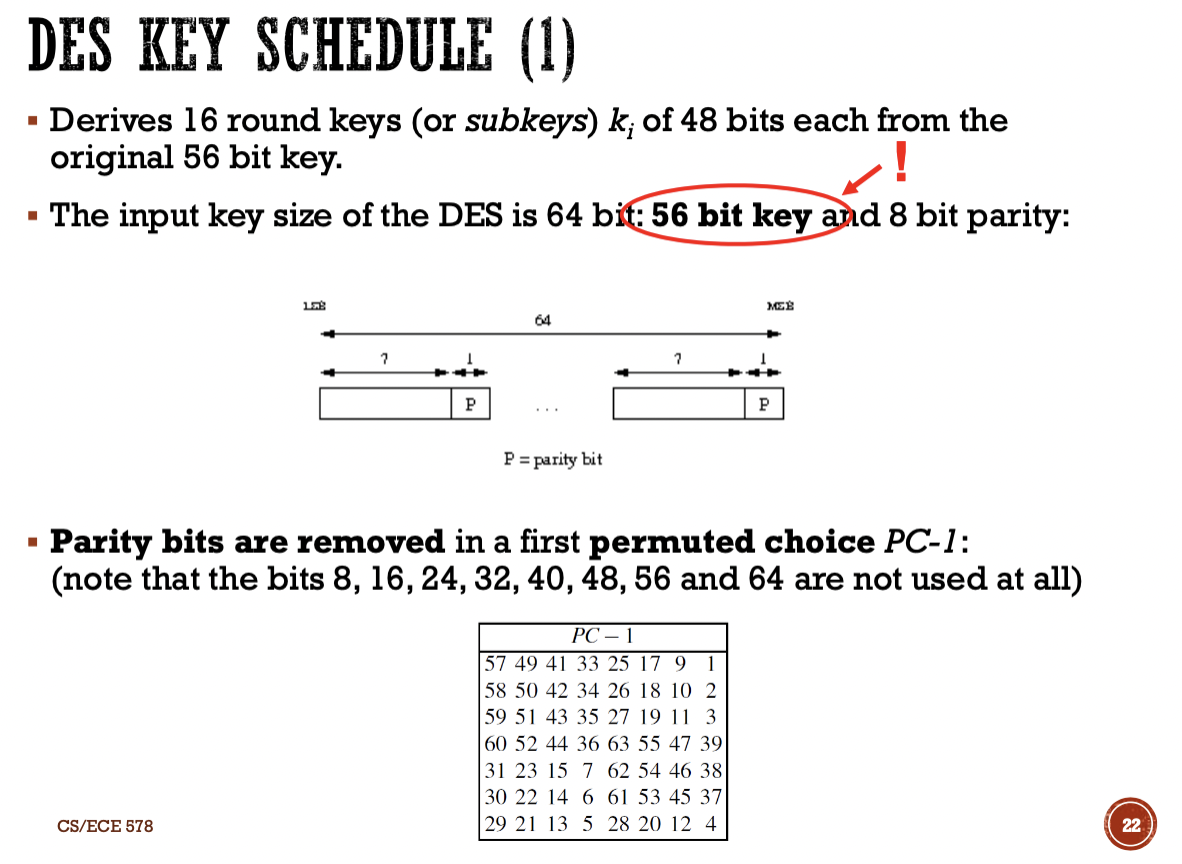
The bits in the table are split between the two halfs. Consecutvie bits will always be sperated betwene the two halves. Example bit 0 will be on left half, bit 1 will be on the right half.

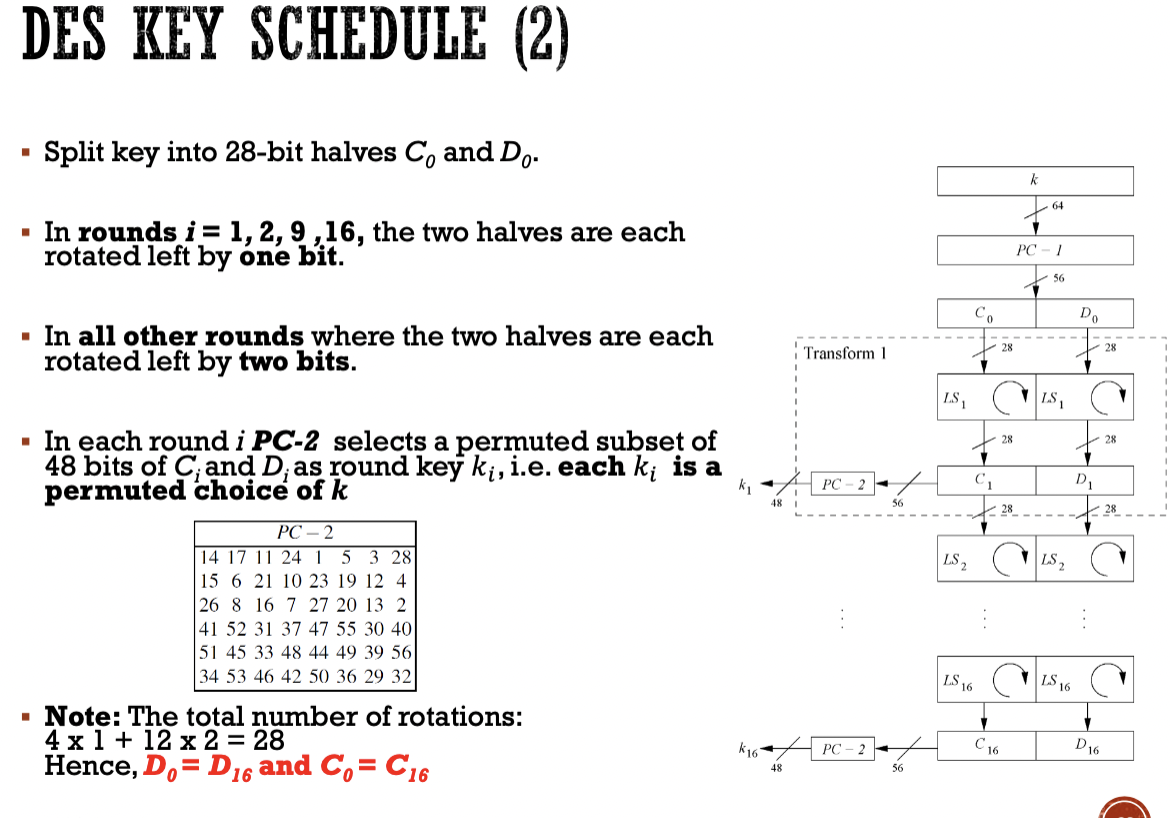


The purpose of the key expansion is to increase the diffusion.

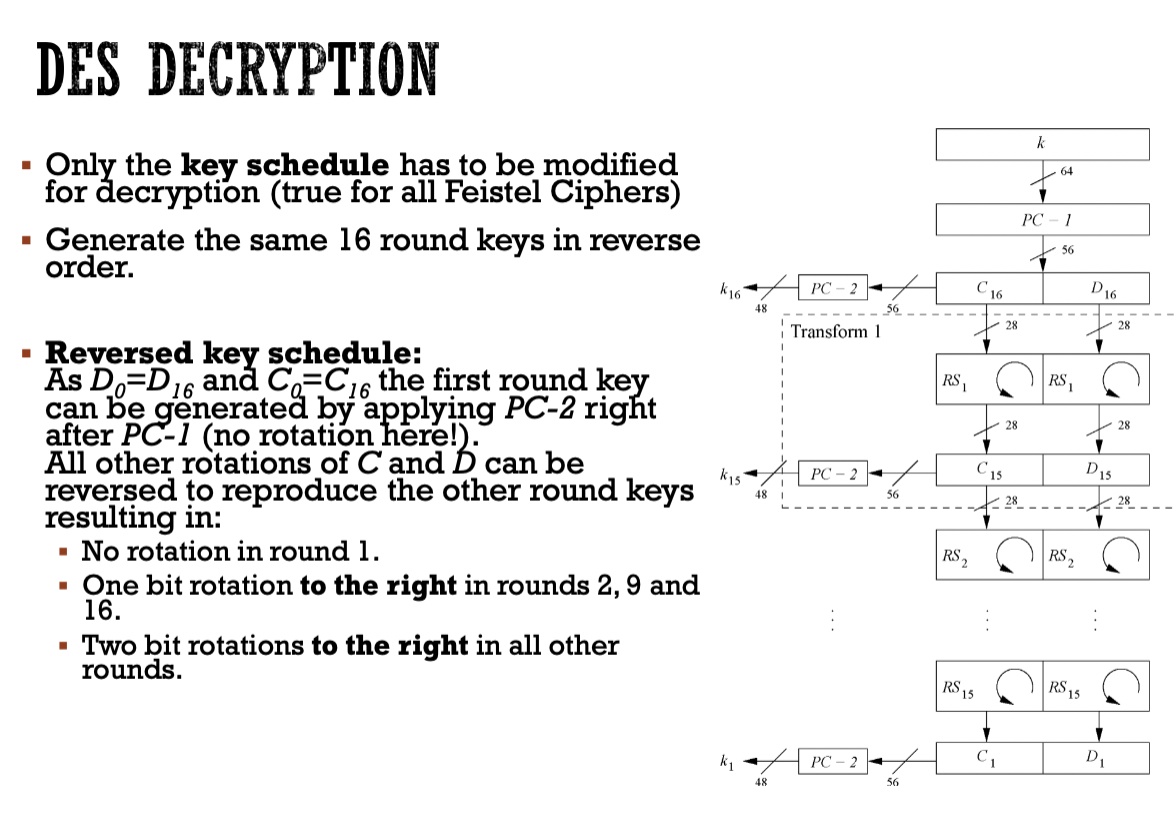
During the key expansion, the expanded bits are added to the front and back of each block.

S-Boxes are what make the f function invertible because the input size is different than the output size.





The bits are rotated within each half. So bits starting on one half will never end up in another half. Example bit 1 will swap with 28, and bit 29 will swap with bit 56. But never 28 will swap with 56.



To decrypt, the order of the round keys are just reversed. Your shifting to the right instead of to the left.

**Differential Cryptoanalysis:**