BİLGİSAYAR AĞLARI

💡 **BİLGİSAYAR AĞLARI**

⚠️ pka paylaşımı ska ile compromise (uzlaşma anlaşma) sağlamaz. pka dan ska yı bulamayız.

PKa = public key ⇒ verify signature

SKa=secret key ⇒ sign with

cryptography - study of encryption principles/methods
• cryptanalysis (codebreaking) - study of principles/ methods
of deciphering ciphertext without knowing key
• cryptology - field of both cryptography and cryptanalysis

cryptography

type of encryption operations used
• substitution // yerine koyma "plaintext are replaced by other letters or by numbers or symbols" **caeser cipher"**

• transposition
• product

computational security : the cipher cannot be broken

general approaches:
– cryptanalytic attack
– brute-force attack

simetrik encrption

gönderici - alıcı aynı keyi kullanır (single-shared key) “common key” private key ile şifrelenir

-requirements-

– a strong encryption algorithm
– a secret key known only to sender / receiver

- mathematically have:
- K=KEY X=PLAIN TEXT
  Y = E(K, X)
  X = D(K, Y)

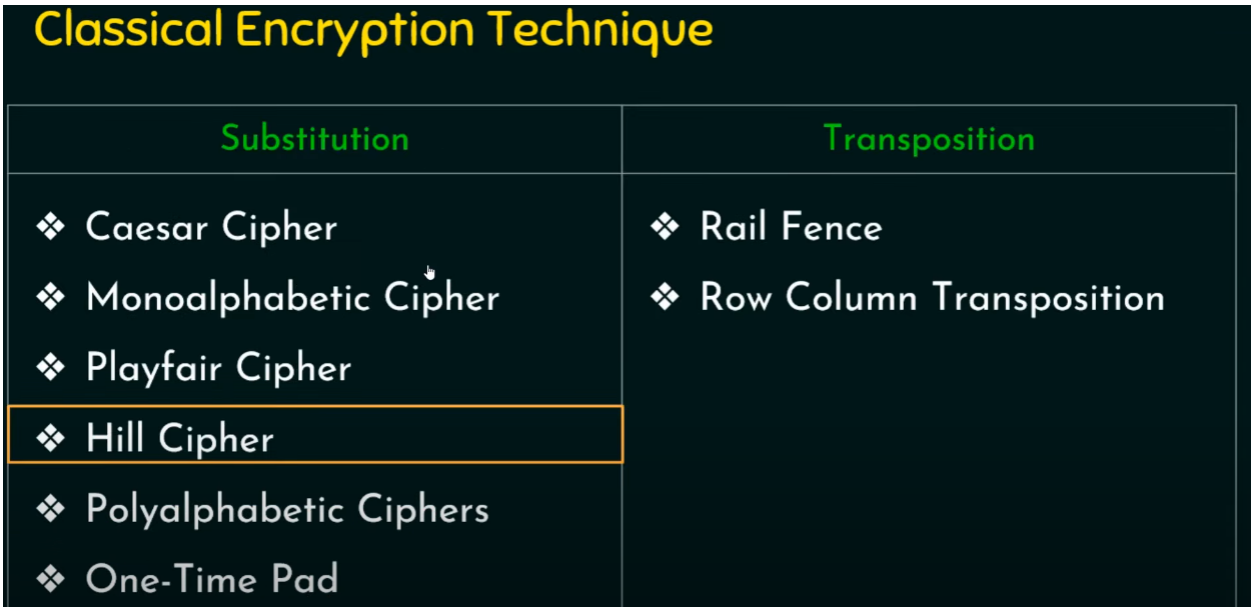
implies a secure channel to distribute key

Monoalphabetic

shuffle (jumble) the letters arbitrarily
• each plaintext letter maps to a different random
ciphertext letter key is 26 letters long

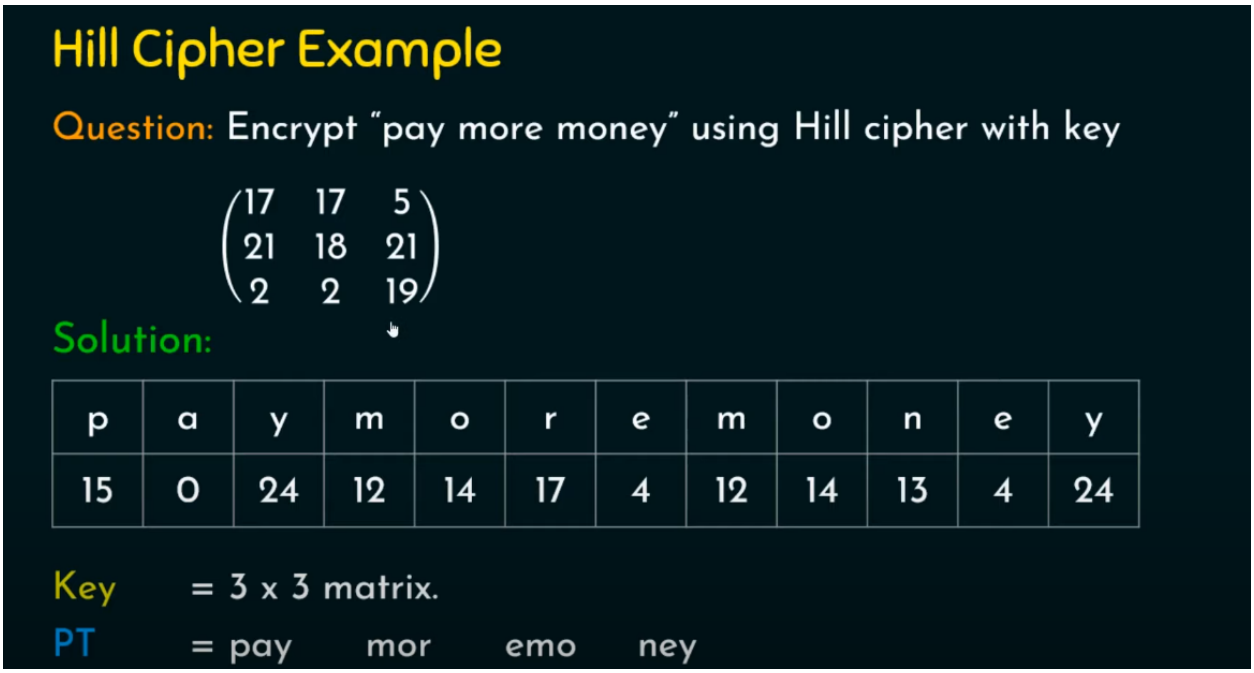
now have a total of 26! = 4 x 1026 keys for security

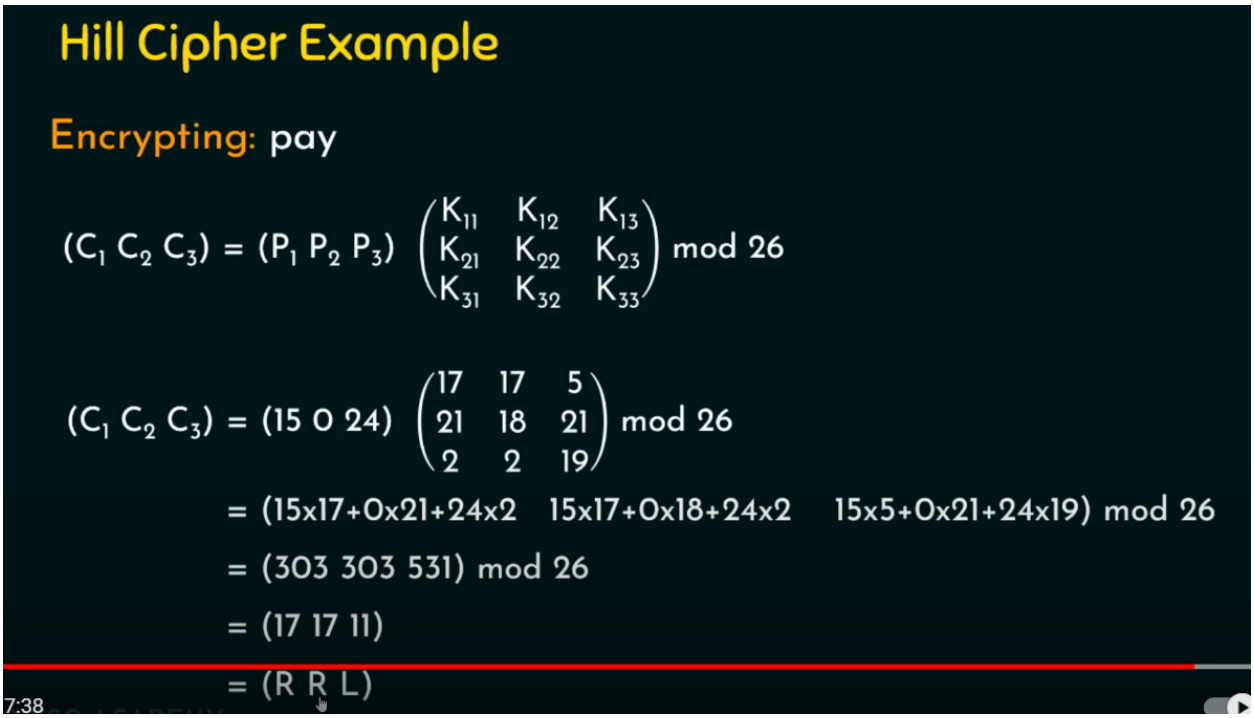
⚠️ key cok ama secure değil tamamen dilin karakteristiği Language Redundancy and Cryptanalysis



Hill Cipher

3 lü verdiği için 3 3 bol sonra çarp her birini her kolonla sonra **TOPLA** c1 c2 c3 bul mod 26 al





PLAYFAIR CIPHER

• multiple letter encry. a şifreledik b oldu b yi de şifreledik z oldu

5 x5 matrix keyword kullanılrak doldurulur . repeat edenleri atla boş alanları diğer harflerle doldur (a dan başlayıp) a b c d … b varsa doldurudgumzu yerde atlarız . I ve j combine yapabiliriz nednei ise 5 x5 matrixte 25 kare var 26 harf old için alfabede 1 alana 2 harf yazdık.

Rules for encryption using playfair cipher

1- diagrams

2- Repeating letters - filler letter

3- same column (⬇️) wrap around (en altta ise harf matrixin colum basına geçeriz)

4- same row ➡️ wrap around “ BİR YANINDAKİNİ ALIRIZ SAĞDAKİ “ SONDA İSSE BAŞA DÖNER

5- rectangle ↔ swap “ ROW SONUNDAKİNİ ALIRIZ KARE DE”

💡 örneğin hello seçtik ve kareye bakıyoruz h ve e aynı row veya columd da değil o zaman 5. kuralı uygularız rectangle alanı seçeriz .

⚠️ diagram oluşturmak için örn: attack plaintext , ayırırız (at ta ck) aynı yok (aa) olsa aynı olurdu . matrixte olmayan varsa diagram olmaz .

filler karakter ile diagram yaparız.

🕐 boş olan vrsa örn plaintext: ballon ⇒ ba ll oo n

digrams ⇒ ba lx lo on diğer o için x kullanmadık l yi kullandık sonuncu tek olan için de diğerini “o” yu kullandık. başka bir harf kullanmak baska bir diagramı işaret edebilir o yüzden plaintexten devam ediyoruz

❗ filler karakter sender ve reciever ikisi karar verir

örn: plaintext: attack ⇒ diagram : at ta ck

“at” // ne aynı column ne aynı rowda kare içine alıyoruz sonra aynı kare içinde yanındakiyle değiştiriyoruz a ⇒ r t ⇒ s t en solda o yüzden sağdakini aldık kare içinde SONDA OLDUKLARI ICIN	**ta**	**ck**
RS	SR	

❗ I/J OLANDAN İKİSİNDEN BİRİ OLABİLİR HER HALUKARDA DECRPTION AYNI OLUR

Vigenère Cipher

plaintext ile key aynı uzunlukta olacak plain text teki ilk harf ile keyin ilk harfi eşleşecek 2. harifiyle 2. harfi yeni text çıkıcak

plaintext : (9 karakter)muzbanana key:lemonlemo(9oldu) m kolonunun l harfi 1. harf , u kolonunn e harfi 2.harf…….

Vernam Şifrelemesi (Vernam Cipher)

plaintext ile cipher xor a tabi tut oluşan 10100 ı key ile xor a tut bunu plaintextte eşleştir

bir bilgi rastgele herhangi başka bir bilgi ile XOR işlemine tabi tutulursa, orjinal bilginin geri elde edilmesi imkansızdır.

veri ile aynı boyuta sahip bir anahtar gerektirmesidir.

Dolayısıyla örneğin 1 GB boyutunda bir veriyi şifrelemek için 1GB boyutunda bir anahtara her iki tarafta da ihtiyaç duyulur.bir üretici fonksiyonu her iki tarafta doğru şekilde beslerse (seed) sonuçta elde ettikleri veri uzunluğundaki anahtar aynı olur.teorik olarak kırılması imkansız olan vernam şifrelemsinin zaafiyeti ortaya çıkar. Yani vernam şifrelemesinin tek zayıf noktası anahtar üreten fonksiyonudur.

The one-time pad offers complete security but, in practice, has two fundamental difficulties:
1.There is the practical problem of making large quantities of random keys.
2.And the problem of key distribution and protection, where for every message to be sent, a key of equal length is needed by both sender and receiver.

FİNALSS

Block Cipher

block ciphers process messages in blocks

most symmetric block ciphers are based on a Feistel Cipher Structure

for an n-bit general substitution block cipher, the size of the **key** is n x 2 yanii

would need table of 264 entries for a 64-bit block,
the key size is 64 (bits) x 264 (rows) = 270 bits

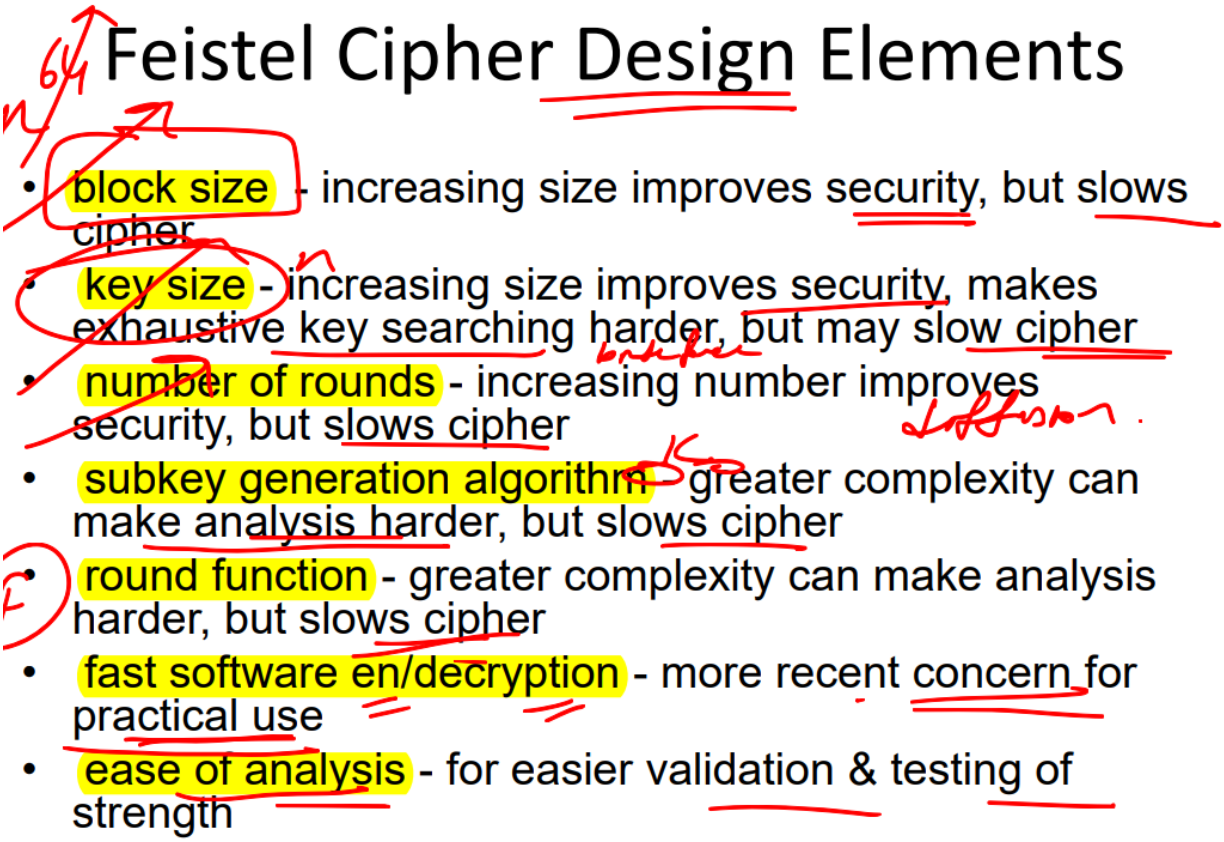
shanon substition permutation cipher

substitution (S-box)
 permutation (P-box)
 provide confusion “karmaşa”& diffusion “yayılma”of message & key

obscure(gizli) olmalı şifre

Feistel Cipher Structure

Use the ciphertext as input to the algorithm, but use the subkeys Ki in reverse order.
•That is, use Kn in the first round, Kn–1 in the second round, and so on until K1 is used in the last round.

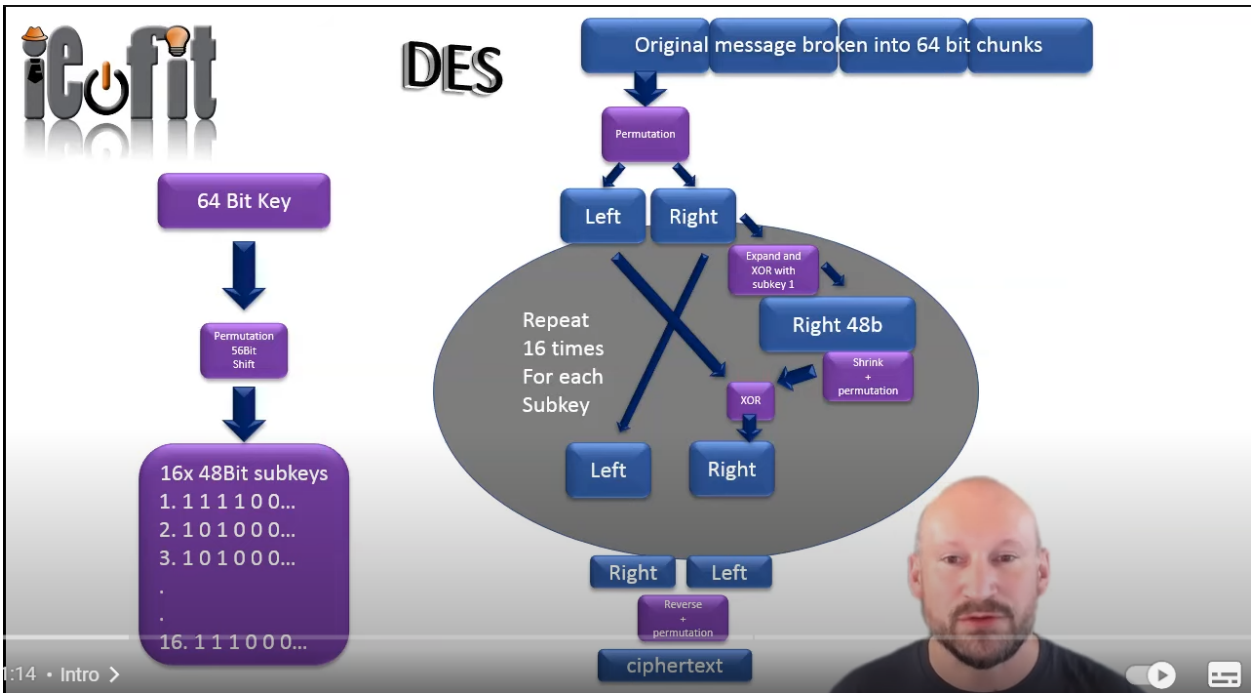


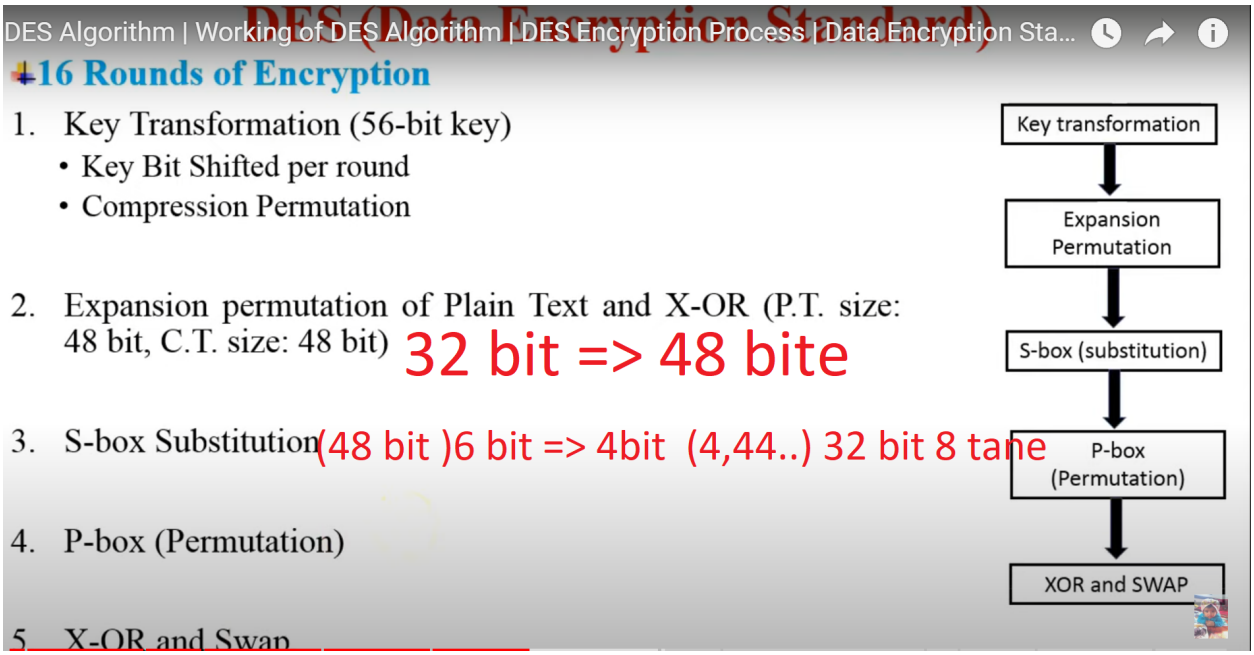
DES(DATA ENCRPTION STANDARD) most widely block cipher 64 bit 56 bit key ile

mesajlar bitlere çevrilir, 64 bitlik parçalara ayrılır (1 byte 8 bit 8 kelime yani) 64 bitlik şifre alır bide anahtar aynı.

Initial Permutation IP: first step of the data computation even “çift” bits to LH half, odd bits to RH half

must now consider alternatives to DES, the most
important of which are AES and triple DES

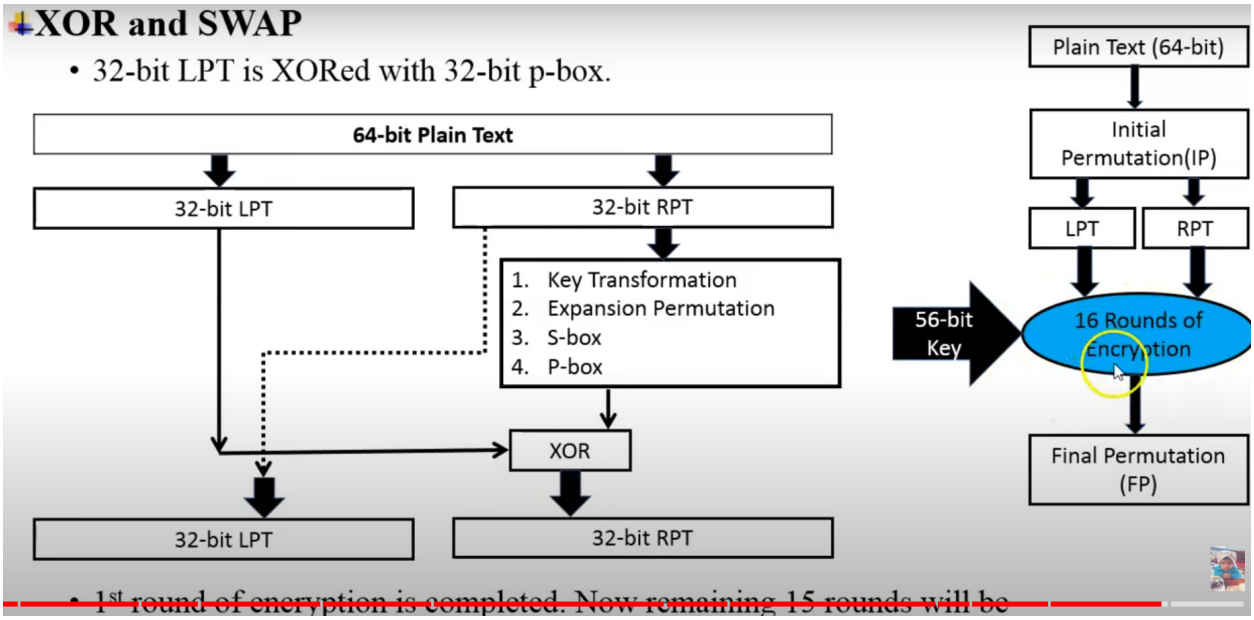




⚠️ key biti her roundda (16) shift edilir

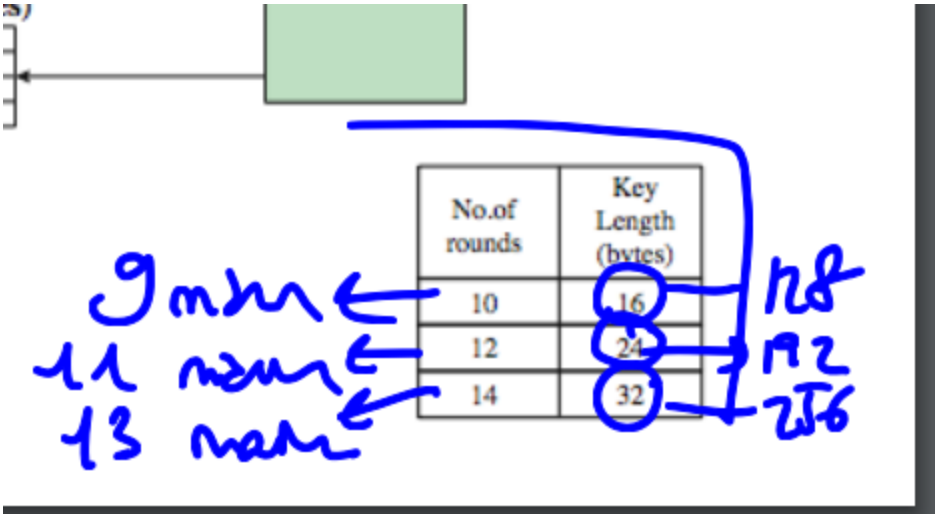
substitionda 6 bit 4 bite dönerken 2 bit row 4 bit column olur

56 bit input key 48bitlik key oluşturur expansion substiton kullanılır



Advanced Encryption Standard(AES)

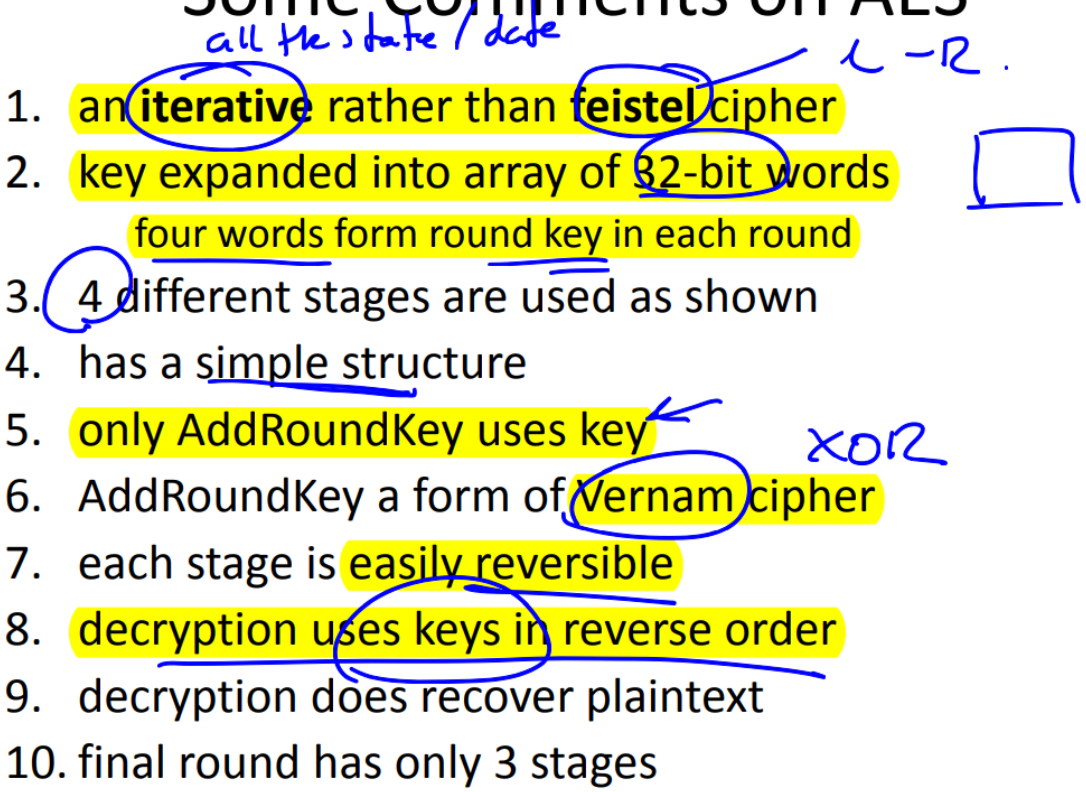
an iterative rather than feistel cipher
– processes data as block of 4 columns of 4 bytes
– operates on entire data block in every round



AES Structure

- data block of 4 columns of 4 bytes is state
- key is expanded to array of words
- has 9/11/13 rounds in which state undergoes:
  - byte substitution (1 S-box used on every byte)
  - shift rows (permute bytes between groups/columns)
  - mix columns (subs using matrix multiply of groups)
  - add round key (XOR state with key material)
    - view as alternating XOR key & scramble data bytes
- initial XOR key material & incomplete last round
- with fast XOR & table lookup implementation

Some Comments on AES



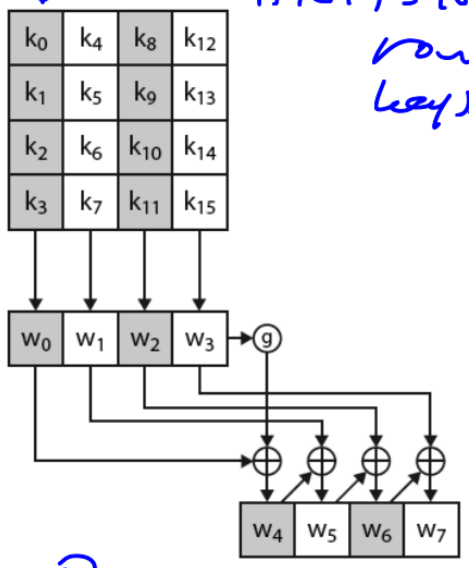
1. an **iterative** rather than **feistel** cipher
2. key expanded into array of 32-bit words
   four words form round key in each round
3. 4 different stages are used as shown
4. has a simple structure
5. only AddRoundKey uses key
6. AddRoundKey a form of Vernam cipher
7. each stage is easily reversible
8. decryption uses keys in reverse order
9. decryption does recover plaintext
10. final round has only 3 stages

Mix Columns : each column is processed separately

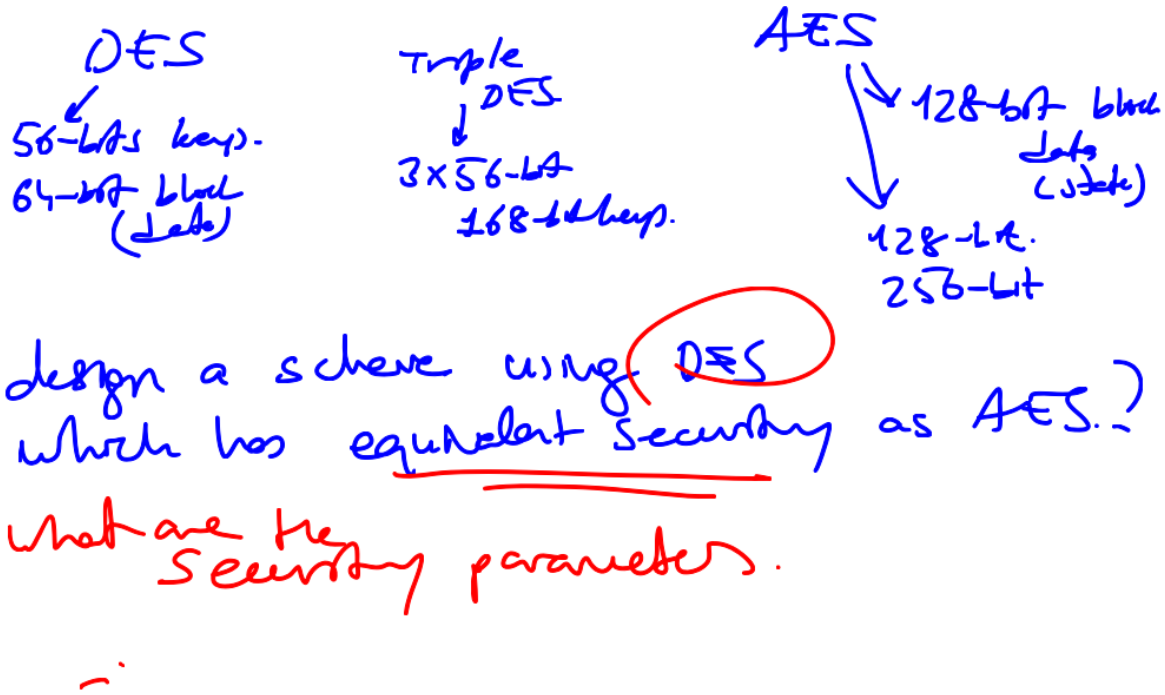
Add Round Key: XOR state with 128-bits of the round key □again processed by column (though effectively a series of byte operations) □inverse for decryption identical □since XOR own inverse, with reversed keys □designed to be as simple as possible □a form of Vernam cipher on expanded key □requires other stages for complexity / security

AES Key Expansion

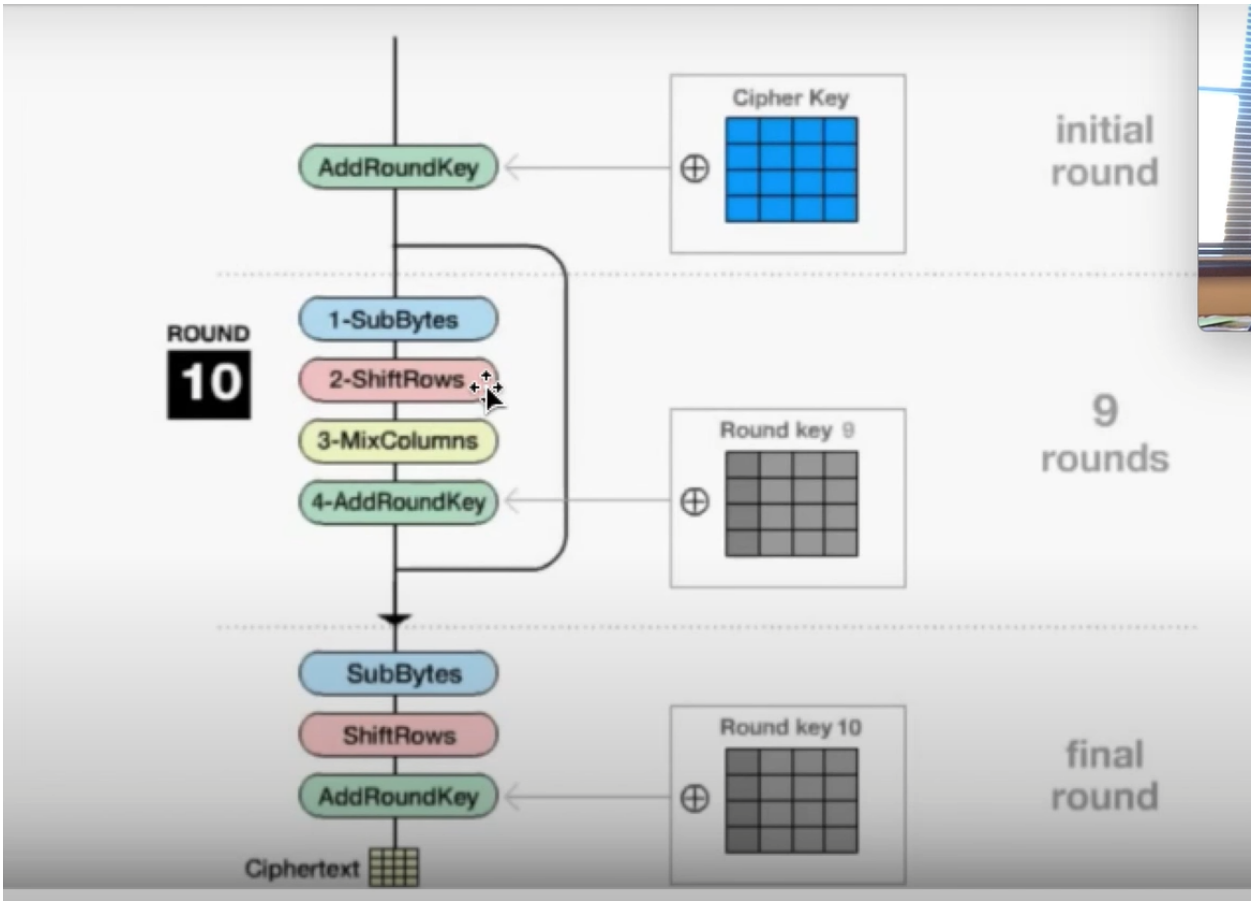
- takes 128-bit (16-byte) key and expands into array of 44/52/60 32-bit words
- start by copying key into first 4 words
- then loop creating words that depend on values in previous & 4 places back
  - in 3 of 4 cases just XOR these together
  - 1st word in 4 has rotate + S-box + XOR round constant on previous, before XOR 4th back

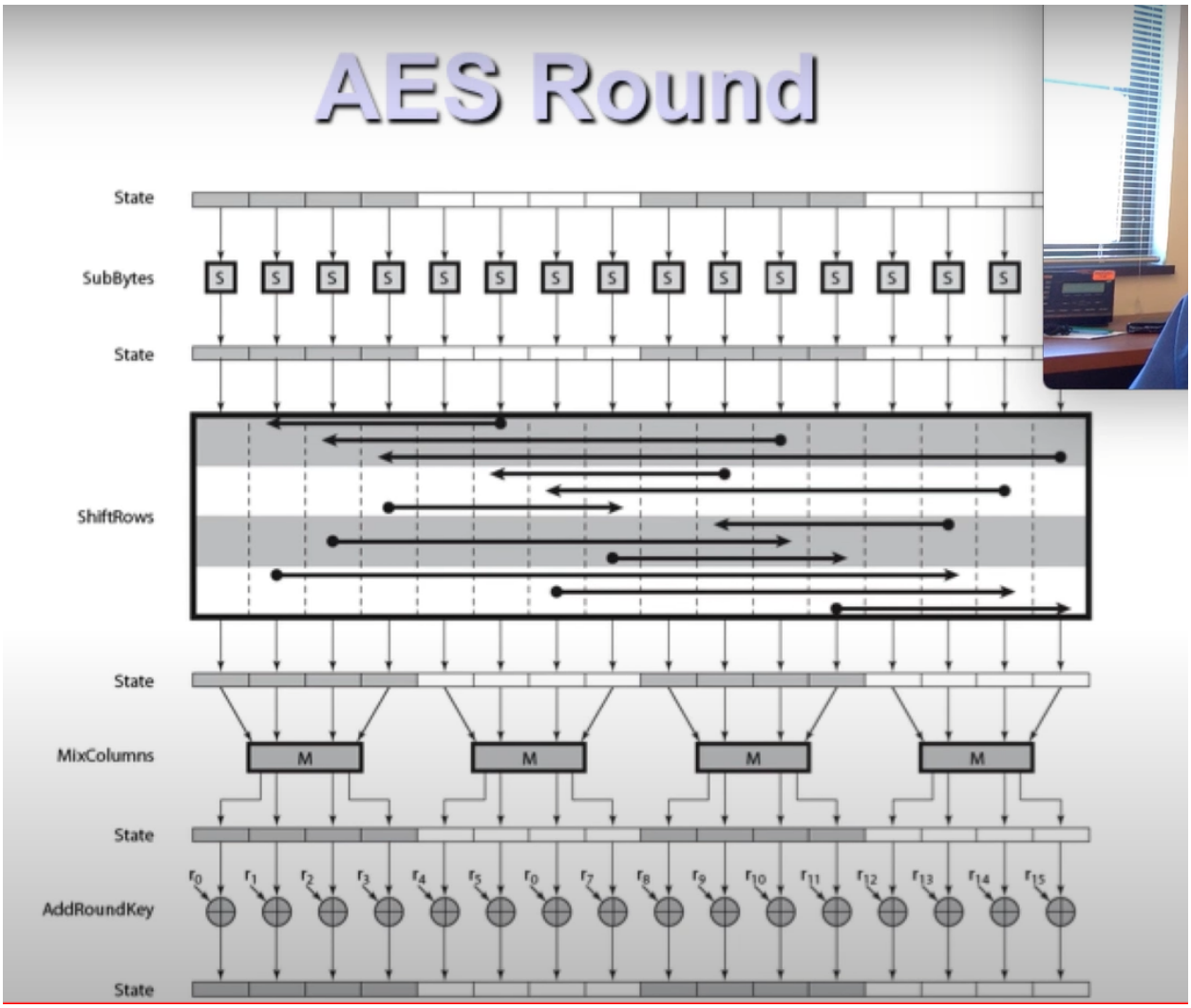


AES Decryption : inverses of each step – with a different key schedule


DES
56-bits keys.
64-bit block
(data)
Triple
DES
3×56-bit
168-bit keys.
AES
128-bit block
data
(state)
128-bit.
256-bit
design a scheme using DES
which has equivalent security as AES.?
what are the
security parameters.

Sadi : ⇒


Cipher Key
initial round
AddRoundKey
ROUND
10
1-SubBytes
2-ShiftRows
3-MixColumns
4-AddRoundKey
Round key 9
9 rounds
SubBytes
ShiftRows
AddRoundKey
Round key 10
final round
Ciphertext


AES Round
State
SubBytes
State
ShiftRows
State
MixColumns
State
AddRoundKey
State
S
M
r0 r1 r2 r3 r4 r5 r0 r7 r8 r9 r10 r11 r12 r13 r14 r15