CENG 434 Kriptoloji – 7. Ders

Alper UĞUR

CENG 507 :
KRİPTOGRAFİK ALGORİTMALAR VE SİSTEMLER

CENG 434:
KRİPTOLOJİ

Sayısal İmzalar

- Elektronik imza: Genel
  - El ile atılmış imzanın sayısallaştırılması
  - Biyometrik özelliklerin eklenmesi
  - Kriptografik yöntemler
- **5070 Sayılı Elektronik İmza Kanunu'nda elektronik imza,**
  - ***"Başka bir elektronik veriye eklenen veya elektronik veriyle mantıksal bağlantısı bulunan ve kimlik doğrulama amacıyla kullanılan elektronik veri"***

MADDE 4. — Güvenli elektronik imza;

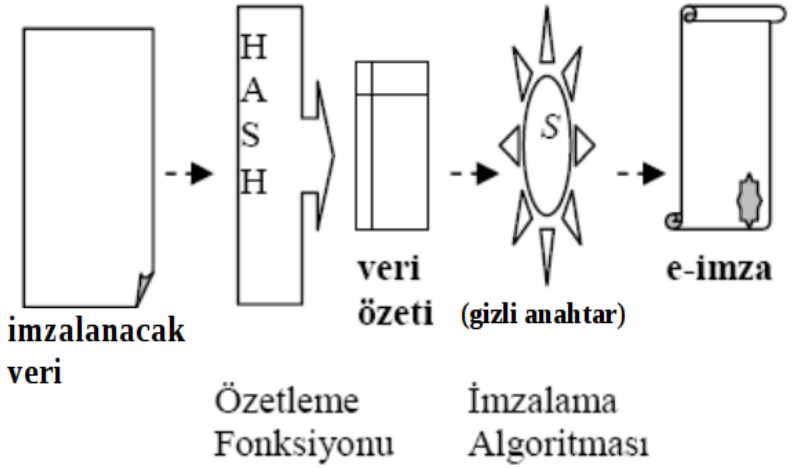
- **a) Münhasıran imza sahibine bağlı olan,**
- **b) Sadece imza sahibinin tasarrufunda bulunan güvenli elektronik imza oluşturma aracı ile oluşturulan,**
- **c) Nitelikli elektronik sertifikaya dayanarak imza sahibinin kimliğinin tespitini sağlayan,**
- **d) İmzalanmış elektronik veride sonradan herhangi bir değişiklik yapılıp yapılmadığının tespitini sağlayan,**

Elektronik imzadır.

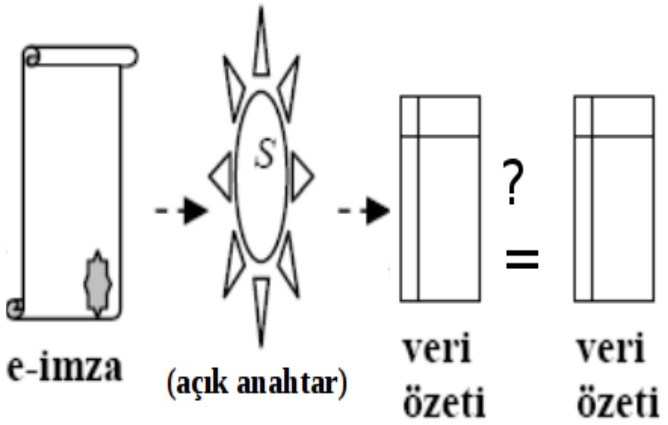
Sayısal İmzalar

- Özetlenmiş metnin gizli anahtarla şifrelenmesi ve karşı tarafta açık anahtarla doğrulanması işlemi

Gerçekleme

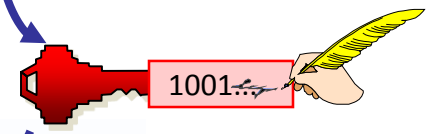


Doğrulama

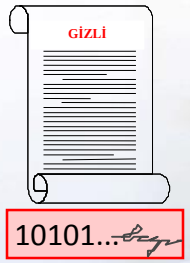


İmzalama Süreci

1. Orijinal verinin mesaj Özetini (hash değerini) hesapla

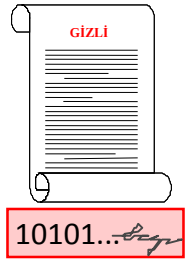


2. Mesaj özetini kendi özel anahtarınla şifrele (imzala)



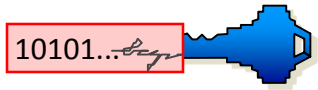
3. İmzalanmış orijinal veri meydana gelir.

Doğrulama Süreci



1. Mesaj özetini yeniden hesapla

1001...



2. Karşı tarafın açık anahtarını kullanarak, şifrelenmiş orijinal mesaj özetini aç

1001... =? 1001...

3. Kendi hesapladığın mesaj özetiyle, orijinal mesaj özetini karşılaştır

RSA İmza

- ***Sign(M) = M^d mod n***
- ***İmza Kontrolü***
  - ***M ?= Sign(M)^e mod n***
  - ***Özetleme fonksiyonunu unutmayalım!***

- RSA şifreleme

- e: açık anahtar (n biliniyor)
- d: gizli anahtar (p ve q gizleniyor)

olmak üzere
e.d =1 mod n bulunabilirse
(d =e^{-1} *mod n)*

Şifreleme fonksiyonu
E(M) = M^e mod n

Şifre çözme fonksiyonu
D(E(M)) = (E(M))^d *mod n*

olarak tanımlanabilir.

7.6 ElGamal Açık Anahtarlı Şifrelemede Anahtar Oluşturma Algoritması

Her kişi kendi açık anahtarını ve buna bağlı gizli anahtarını oluşturur. Bunu oluşturmak için A şahsı şunları uygular:

1. Çok büyük rastgele bir p asal sayısı ve mod p ye göre tamsayıların oluşturduğu çarpım grubu Z_p^* nin bir jeneratörü α yı oluşturur.
2. $1 \leq a \leq p-2$ şeklinde olan bir a tamsayısı seçer ve $\alpha^a \, mod \, p$ değerini hesaplar.
3. A'nın açık anahtarı (p, α, α^a); A'nın gizli anahtarı ise a olur.

http://iam.metu.edu.tr/sites/iam.metu.edu.tr/files/UYGULAMALI%20MATEMAT_Kriptolojiye%20giris%20ders%20notlar%C4%B1.pdf

7.6.1 ElGamal Açık Anahtarlı Şifreleme Algoritması

B şahsı A için m mesajını şifrelesin.

1. **Şifreleme:** B mesajı şifreleme için şunları yapar:
   - A'nın açık anahtarını (p, α, α^a) alır.
   - mesajı $\{0, 1, \cdots, p-1\}$ aralığında m tamsayısı olarak ifade eder.
   - $1 \leq k \leq p-2$'yi sağlayan rastgele bir k tamsayısı seçer.
   - $\gamma = \alpha^k \bmod p$ ve $\delta = m \cdot (\alpha^a)^k \bmod p$ değerlerini hesaplar.
   - Son olarak $c = (\gamma, \delta)$ kapalı metnini A'ya gönderir.

2. **Deşifreleme:** c kapalı metninden m açık metine ulaşmak için A şunları yapar:
   - a gizli anahtarını kullanarak $\gamma^{-a} \bmod p$ değerini hesaplar $(\gamma^{-a} = \alpha^{-ak} \bmod p)$.
   - $\gamma^{-a} \cdot \delta \bmod p$ değerini hesaplayarak m'yi bulur.

$$\gamma^{-a} \cdot \delta \equiv \alpha^{-ak} \cdot m\alpha^{ak} \equiv m \pmod p$$

http://iam.metu.edu.tr/sites/iam.metu.edu.tr/files/UYGULAMALI%20MATEMAT_Kriptolojiye%20giris%20ders%20notlar%C4%B1.pdf

7.6.2 ElGamal İmzası

ElGamal kriptosisteminde imza RSA 'da olduğu gibi mesajın doğru kişiden geldiğini kontrol etmek için kullanılır. Sadece kapalı metin yerine imzalanmış kapalı metin gönderilerek o kapalı metnin istenen kişiden gelip gelmediği de kontrol edilmiş olur. A şahsının açık anahtarı $(p, \alpha, \alpha^a = y)$ ve gizli anahtarının da a olduğu düşünülsün.

7.6.3 İmza Algoritması

m mesajının Z_p nin bir elemanı olduğu düşünülür.Eğer değilse hash fonksiyonu kullanılarak m mesajının Z_p nin elemanı olması sağlanır. A şahsı m mesajını şu şekilde imzalar:

1. Rastgele bir t tamsayısı seçer öyleki $1 \leq t \leq p-2$ ve $gcd(t, p-1) = 1$ koşulunu sağlamalıdır.
2. $r = \alpha^t$ ve $s = t^{-1}(m - ra) \, mod \, (p-1)$ eşitliklerini kurar.
3. (m, r, s) A'nın imzalı mesajıdır.

Elgamal İmza Doğrulama

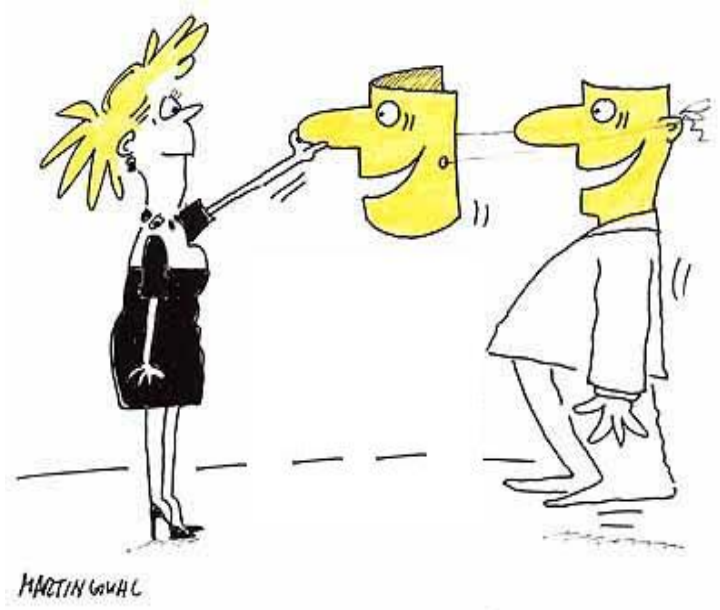
7.6.4 Doğrulama

(m, r, s) imzalı mesajı alan B şahsı aldığı mesajın A'dan geldiğini şu şekilde doğrular:

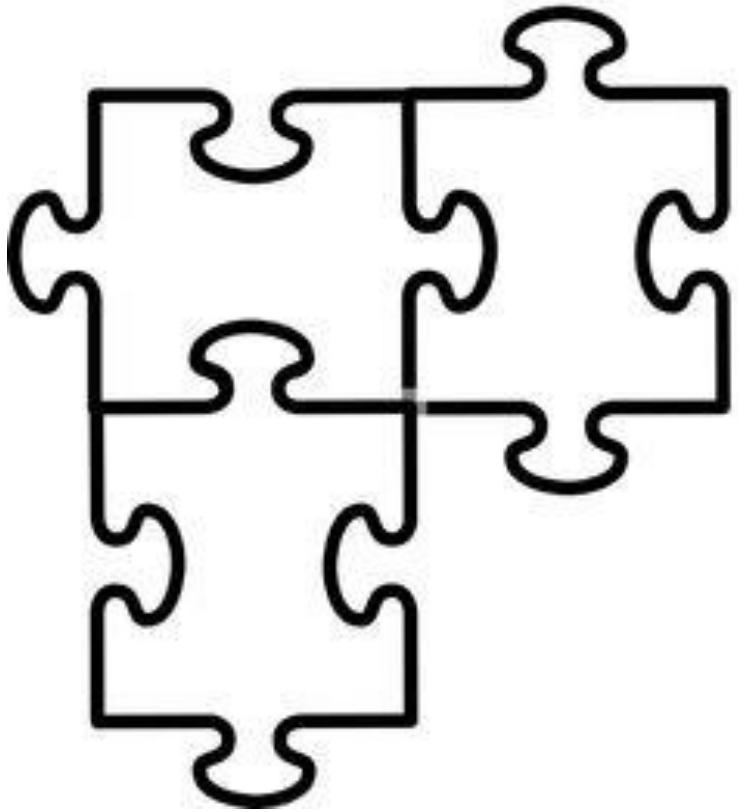
1. Öncelikle $1 \leq r \leq p-1$ olduğunu kontrol eder.Eğer değilse imzayı reddeder.
2. Daha sonra $v = \alpha^m$ ve $w = y^r r^s$ değerlerini hesaplar (Buradaki y sayısı A'nın açık anahtarındaki y sayısıdır.)
3. Eğer $v = w$ eşitliği sağlanıyorsa imza kabul edilir, aksi taktirde reddedilir.

Farklı Uygulamalar

- Kör İmzalar
- Vekil İmzalar
- Kimlik tabanlı İmzalar
- Çoklu İmzalar

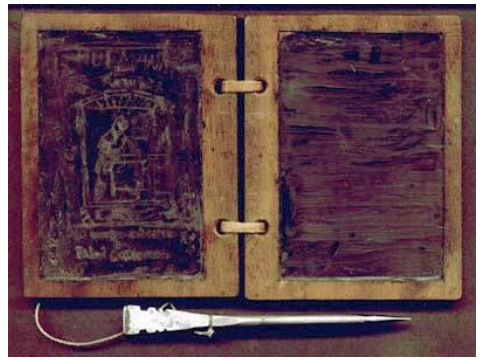


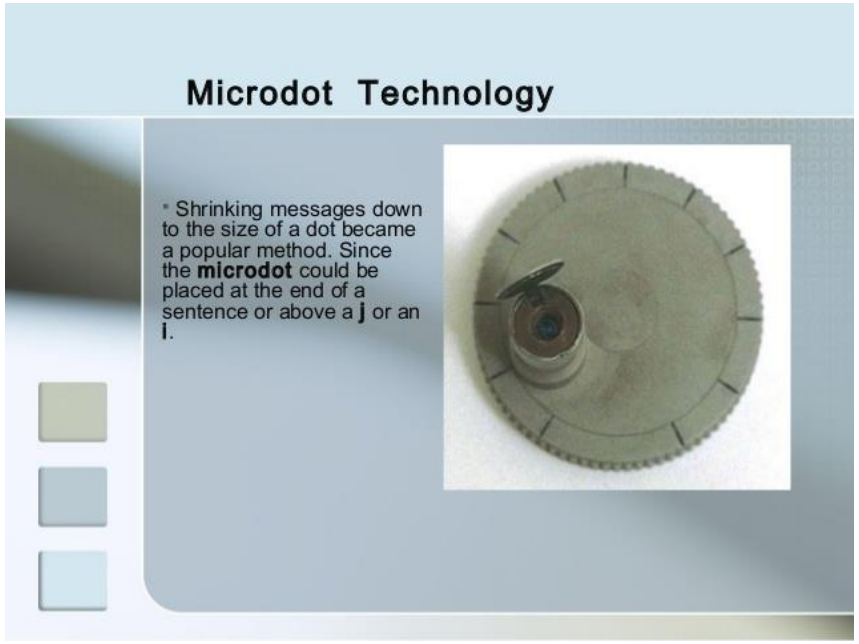
Ara - 15dk



Steganografi

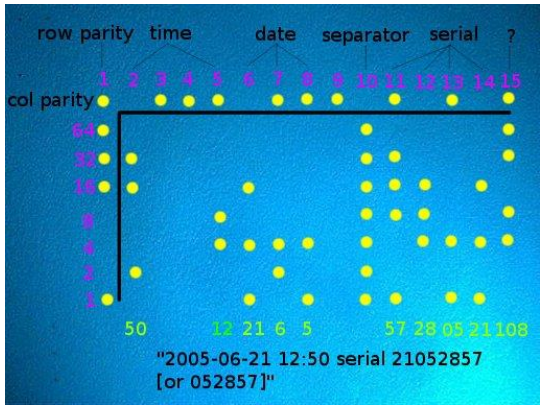
- Gizli yazı
- Watermarking





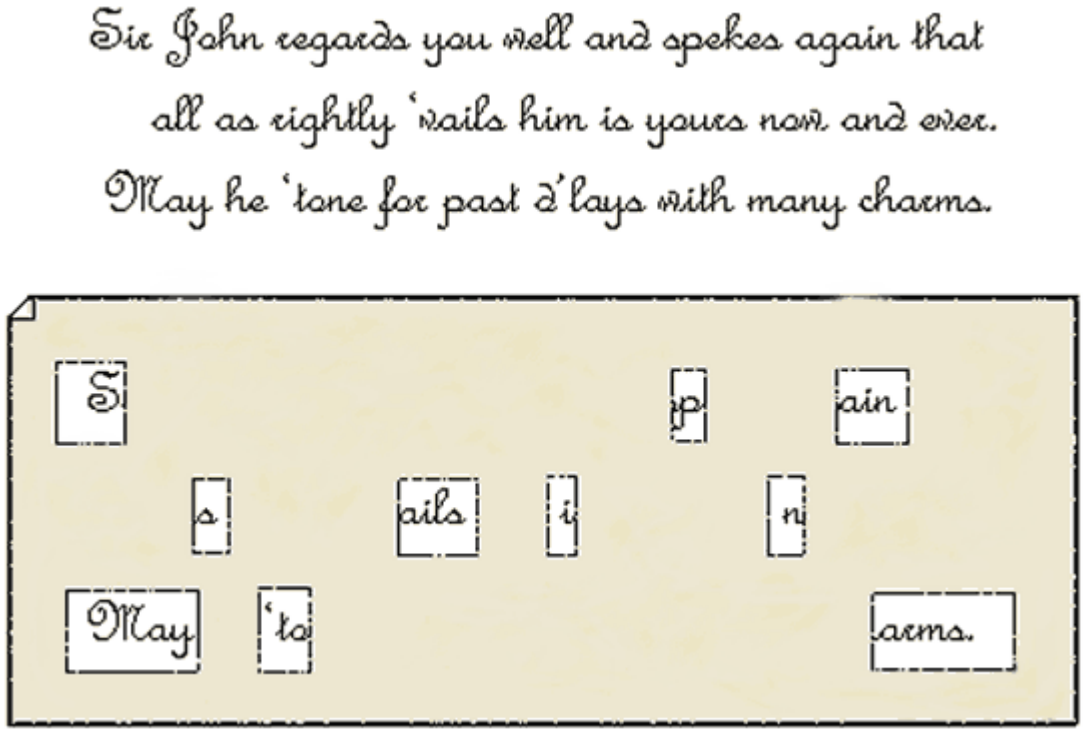
Steganografi

- Gizli yazı
- Watermarking

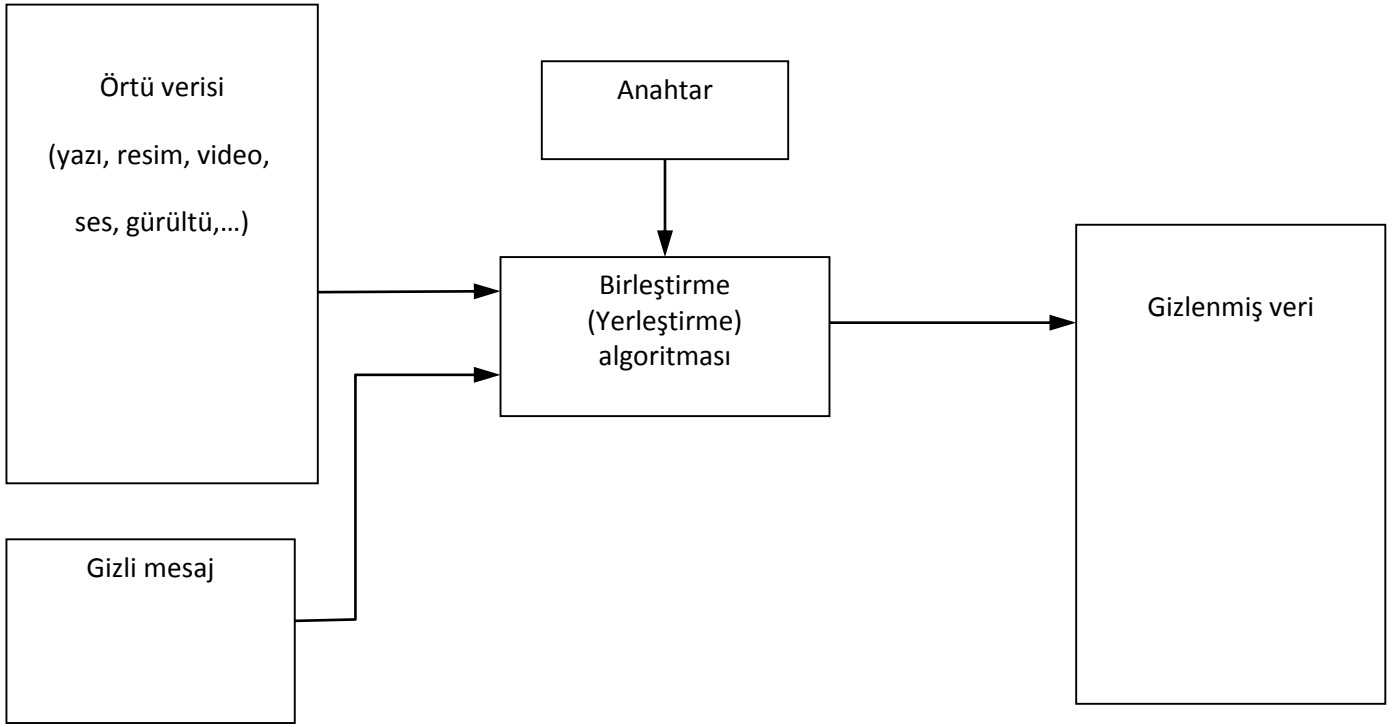


Steganografi

- Gizli yazı
- Watermarking



Genel Stegosistem



Resme yazı yerleştirme

- Değişken renk seçenekleri çok
- Kararlaştırılmış Rasgele dönüşümlü pixel yer bilgileri R[]

1. i=0
2. yer bilgisini al , R[i] (i=0, R[0])
3. Mesajın i. bitini al, M[i] (i=0,M[0])
4. Resimdeki R[i]nci pixeldeki renk kodunun en az önemli bitini M[i] ile değiştir
5. i++;
6. i<=M.length ise 2'ye git
7. Değilse çık.

Örnek

- Mesaj= 0001 1010 1101 0111

Yerleştirilecek bit

Değişen bit

- Resim bit stream = …. 01101110 ….
- Yeni bit stream = …. 01101111 ….
- Renk 255 in içinde kalmalı ;)

Stegonografi komut satırı

- **$ cat** image.jpg archive.rar **>** newimage.jpg
- > copy /b image.jpg + archive.rar newimage.jpg

Steganografi Linux : steghide

$ steghide embed -cf tux.jpg -ef mytext.txt

Enter passphrase:

Re-Enter passphrase:

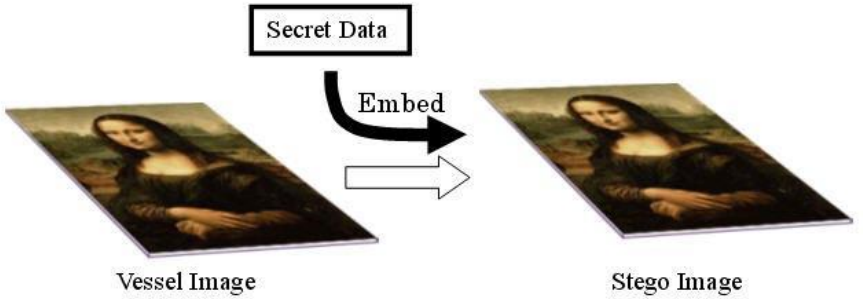
embedding "mytext.txt" in "tux.jpg"... done

$ steghide extract -sf tux.jpg
Enter passphrase:
wrote extracted data to "mytext.txt".

Dikkat : **stegdetect**

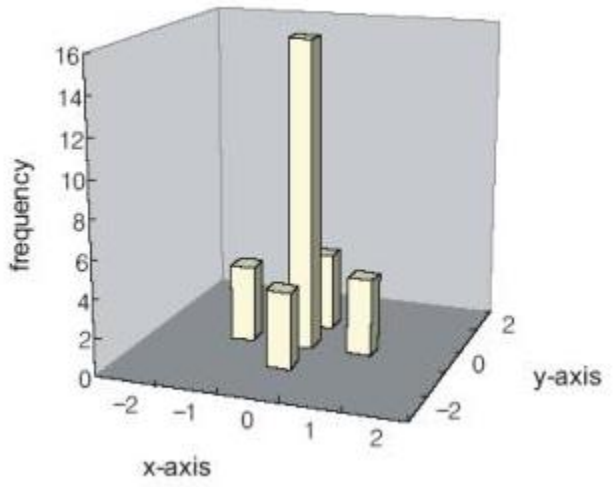
Dikkat edilecek hususlar ;)

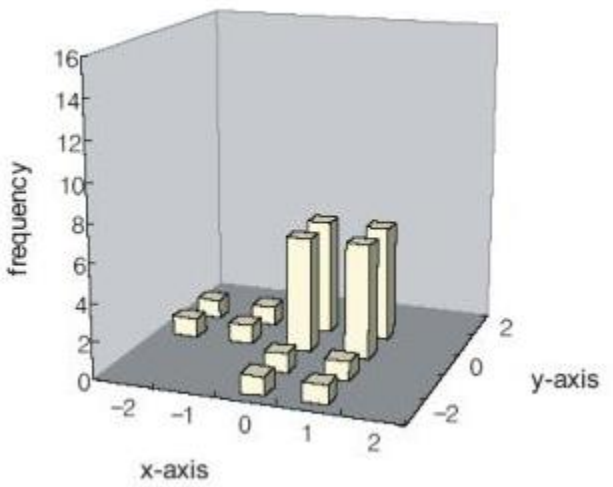
- Değişiklik gözle (!) farkedilmemeli
- Başlığı güncelleme
- Gizli mesaj örtü verinin içinde olmalı
- Örtü veri mesajdan büyük olmalı
- Dönüştürme işlemleri (jpeg,png) veriyi bozabilir
- Hata düzeltme kodu kullanılabilir
- Örtü veriyi bir daha KULLANMA!
- Gizli mesajı şifrele!

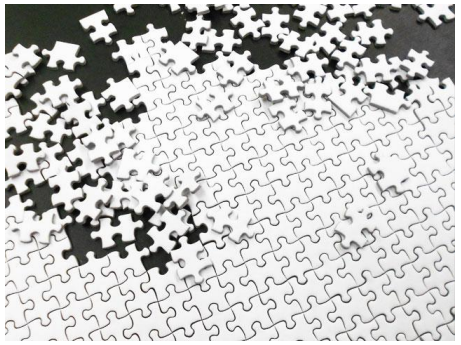


Steganaliz

- Görsel analiz
- Histogram analizi
- Komşu renklerin farkı
- Görüntü karmaşıklığı (!)







Proje

Senaryo 1- Görüntü şifreleme
Senaryo 2- Görüntüye metin gizleme
Senaryo 3- Görüntü için özet hesaplama
Senaryo 4- Orijinal görüntü ve metin gizlenmiş görüntü özetlerini karşılaştırma
Senaryo 5- Görüntü sayısal imzalama/doğrulama