**kAYDIRMA ve Permütasyon Şifreleme Kullanarak Basit Bir Şifreleme Sistemi Oluşturmak**

**Kaydırma Şifrelemesi Nedir ?**

* Her harf alfabede belirli bir sayı kadar kaydırılır.
* Örneğin: Alfabetik 3 kaydırma ile "A" → "D", "B" → "E", vs.
* Örnek: **"HELLO" → "KHOOR"**

**Permütasyon Şifreleme Nedir?**

* P, C ∈ Z26m  öyle ki m bir tamsayı olsun.
* Anahtar K = {1, …, m} kümesi üzerindeki permütasyonlar.
* Olası anahtar sayısı = m!

**Şifreleme:**

Eπ(P1, …, Pm) = (Pπ (1), …, Pπ (m)) = C, Pi ∈ Z26

**Şifre Çözme;**

Dπ(C1, …, Cm) = (Cπ-1(1), …, Cπ-1(m)) = C, Ci ∈ Z26

Ben projem için permütasyon şifreleme için anahtarı [3, 1, 2] den oluşan 3 bloklu bir anahtar seçtim. Fakat siz blok sayısını istediğiniz şekilde arttırabilirsiniz. Ayrıca permütasyon şifrelemeden sonra kaydırma yönetimi kullanarak index numaraları tek olanları 1 çift olanları 2 ileri kaydırdım.

Örnek verecek olursam:

Girilen kelimem **KRİPTOGRAFİ** olsun

Permütasyon şifre için blok uzunluğum 3 anahtarım ise (3, 1, 2) idi.

Kelimeyi belirli blok uzunluğuna bölersek

* 1. Blok : **KRİ**
  2. Blok : **PTO**
  3. Blok : **GRA**
  4. Blok : **FİX** (eksik bloğu dolgu harfi ile doldurduk.)

Sonrasında permütasyon anahtarını uygularız, yani:

Yeni Pozisyon Eski Harf

1. 3. Harf
2. 1. Harf
3. 2. Harf

**1.blok :** KRİ 3. harf: İ 1. harf: K 2. harf: R sonuç itibariyle **İKR** olur

**2.blok :** PTO aynı şekilde permütasyon uyguladığımızda **OPT** olur.

**3.blok :** GRA aynı şekilde permütasyon uyguladığımızda **AGR** olur.

**4.blok :** FİX aynı şekilde permütasyon uyguladığımızda **XFİ** olur.

**Permütasyon Şifrelememizin sonucunda;**

**KRİPTOGRAFİ** olan metnimiz **İKROPTAGRXFİ** metnine dönüşür

Daha sonra ikinci işlemimiz olan kaydırma şifresini uygulamaya geçeriz

Burada kuralımız 0 dan başlayarak Tek index numaraları harfler 1 ileri kayarken, çift index numaralı harfler 2 harf ileri kayar.

\*İndex sistemi Türkçe alfabe kullanarak, döngüsel kaydırma gerçekleştirir. Bu yüzden diğer alfabedeki harfler ve semboller değişmeden kalır. Fakat kodlamada istediğimiz harf ve sembolü ekleyebiliriz.\*

**Şifreleme Kısmında:**

İndex Harf Çift/Tek Kaydırma Yeni Harf

0 İ ÇİFT +2 K

1 K TEK +1 L

2 R ÇİFT +2 Ş

3 O TEK +1 Ö

4 P ÇİFT +2 S

5 T TEK +1 U

6 A ÇİFT +2 C

7 G TEK +1 Ğ

8 R ÇİFT +2 Ş

9 X TEK +1 X(X Türk alfabesinde yok o yüzden x olarak kalsın)

10 F ÇİFT +2 Ğ

11 İ TEK +1 J

**Sonuç Olarak Şifreli Metinimiz:**

**KLŞÖSUCĞŞXĞJ** olur

Şifreyi çözmek için ise adımları tersten uygulayarak gelmeliyiz yani önce şifreli metnimizin harflerini index numaralarına göre ters kaydırıp sonrasında ise permütasyonun tersini alarak asıl metine ulaşabiliriz.

**Çift indexler 2 harf geri giderken tek indexe sahip harfler 1 har geri gider yani;**

İndex Harf Çift/Tek Kaydırma Yeni Harf

0 K ÇİFT +2 İ

1 L TEK +1 K

2 Ş ÇİFT +2 R

3 Ö TEK +1 O

4 S ÇİFT +2 P

5 U TEK +1 T

6 C ÇİFT +2 A

7 Ğ TEK +1 G

8 Ş ÇİFT +2 R

9 X TEK +1 X(önceki dolgu harfi)

10 Ğ ÇİFT +2 F

11 J TEK +1 İ

Kaydırma çözümü sonucu **İKROPTAGRXFİ** metnine ulaşırız

Daha sonra bu metni bloklara ayırıp permütasyon tersi uygularız;

Metin: **İKROPTAGRXFİ**  
Bloklara ayır:

* İKR
* OPT
* AGR
* XFİ

Blok 1: IKR → **(2, 3, 1)** → K → R → İ → **KRİ**  
Blok 2: OPT → **(2, 3, 1)** → P → T → O → **PTO**  
Blok 3: AGR → **(2, 3, 1)** → G → R → A → **GRA**  
Blok 4: XFİ → **(2, 3, 1)** → F → İ → X → **FİX**

**KRİPTOGRAFİX** metnine ulaşırız

X dolgu harfini çıkardığımızda ise metnimiz olan **KRİPTOGRFİ** ye ulaşmış oluruz.

**Peki Bu Şifreleme Güvenli mi?**

**Bu şifreleme frekans analizine karşı dayanıklı mı?**

**✅** Dayanıklılığı artıran yönler:

* Permütasyon her üçlü bloğun harflerini yer değiştirerek harf dizilişini bozar. Bu, kelime yapısını karıştırır ve tahmin etmeyi zorlaştırır.
* Kaydırma sabit değil, pozisyona bağlı. Bu da klasik Sezar şifresinden daha güçlü kılar.

**❌** Zayıf yönleri:

* Her harf hâlâ tek bir harfe dönüşüyor (deterministik). Yani 'A' harfi her zaman aynı pozisyondaysa, hep aynı şifreli harfe dönüşür.
* Uzun metinlerde frekans bilgisi korunmuş olur, çünkü harfler sadece sabit kurallarla değiştiriliyor. Yani örneğin 'A' harfi şifreli metinde de sık görünüyorsa, analizle tespit edilebilir.
* Permütasyon yapısı sabit, yani aynı blok şablonu sürekli tekrarlanıyor. Bu da çözüm için ipucu verir.

**Sonuç:**

* Sistem, klasik Sezar'dan daha güvenli ama yine de frekans analizine karşı tam dayanıklı değil. Gerçek hayattaki güvenlik için daha gelişmiş yöntemlere geçmek gerekir.

**Brute Force’a Karşı Dayanıklı mı?**

| **Bileşen** | **Brute Force'a Karşı Güçlü mü?** | **Neden?** |
| --- | --- | --- |
| Permütasyon | ❌ Hayır | 3 harfli blokta sadece 6 olasılık var. Sabit ve tekrarlı. Blok sayısı arttırılarak güvenlik sağlanabilir. |
| Kaydırma | ❌/✅ Orta seviyede | Sabit değil, ama algoritma biliniyorsa çözülür. Kurallar sabit |
| Genel Sistem | ❌ Hayır | Algoritma açık ve düşük karmaşıklıkta |

**Bilinen Açık Metin Saldırısına Dayanıklı mı?**

| **Bileşen** | **Dayanıklı mı?** | **Neden?** |
| --- | --- | --- |
| Permütasyon | ❌ Hayır | Sabit ve tekrarlı; birkaç örnekle çözülür |
| Kaydırma | ❌ Hayır | Kurallar sabit, birkaç harf eşleşmesiyle algoritma anlaşılır |
| Genel Sistem | ❌ Hayır | Deterministik yapı, açık metin eşleşmeleriyle kolayca çözülebilir |

**Seçilen Açık Metin Saldırısına Karşı Dayanıklı mı?**

| **Özellik** | **Durum** | **Neden?** |
| --- | --- | --- |
| Permütasyon sabit mi? | ✅ Evet (sabit) | Her zaman aynı bloklara aynı işlem uygulanıyor |
| Kaydırma sabit mi? | ✅ Evet | Kaydırma indekse bağlı, her zaman aynı |
| Rastgelelik var mı? | ❌ Hayır | Her açık metin aynı şifreli çıktıyı üretir |
| Dayanıklı mı? | ❌ Hayır | Saldırgan sistemin iç yapısını hızlıca çözebilir |

**İstatistiksel Saldırıya Dayanıklı mı?**

| **Özellik** | **Durum** | **Neden?** |
| --- | --- | --- |
| Harf frekansları bozuluyor mu? | ❌ Hayır | Permütasyon sadece yer değiştirir, sabit kaydırma yapar |
| Rastgelelik var mı? | ❌ Hayır | Her A harfi hep aynı C veya B oluyor |
| Aynı açık metin aynı şifreli metni mi üretir? | ✅ Evet | Bu da saldırıya zemin hazırlar |
| Dayanıklı mı? | ❌ Hayır | Saldırgan harf frekanslarını kullanarak çözümleme yapabilir |

**Sonuç olarak**

Frekans Analizine karşı dayanıklılık →→→→→→→→→ **ORTA SEVİYE DAYANIKLILIK**

Brute Force’a karşı dayanıklılık →→→→→→→→→ **DÜŞÜK SEVİYE DAYANIKLILIK**

Bilinen açık metin saldırısına karşı dayanıklılık →→→→→→→→→ **DÜŞÜK SEVİYE DAYANIKLILIK**

Seçilen açık metin saldırısına karşı dayanıklılık →→→→→→→→→ **DÜŞÜK SEVİYE DAYANIKLILIK**

İstatiksel saldırıya karşı dayanıklılık →→→→→→→→→ **DÜŞÜK SEVİYE DAYANIKLILI**