**KLASİK ŞİFRELEME ALGORİTMALARININ KARŞILAŞTIRILMASI**

Klasik şifreleme algoritmaları modern kriptografiden önce kullanılan daha basit ve ilkel şifreleme yöntemlerinden oluşmaktadır. Bu algoritmalar özelliklerine göre farklı tercih sebepleri sunarlar. En önemli etken güvenlik olmakla birlikte, şifreleme ve deşifreleme süreleri, anahtar büyüklüğü, karmaşıklık seviyeleri, kırılma zorlukları gibi kriterler seçim sürecini etkilemektedir. Bu kriterleri Sezar şifreleme, Monoalfabetik şifreleme, Polialfabetik şifreleme, Playfair şifreleme, Hill şifreleme, Permütasyon şifreleme üzerinden değerlendirelim.

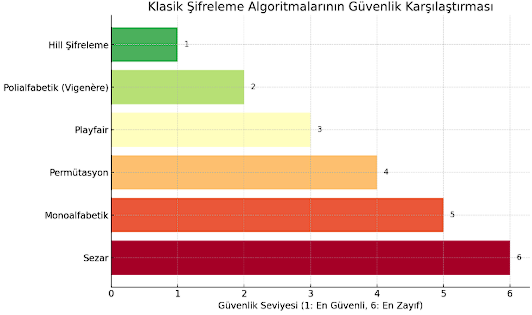
1. **Güvenlik**

Güvenlik, şifrelemeler için her zaman birinci öncelik olmuş ve bir şifrenin kullanılması veya kullanılmasının bırakılmasında en önemli etken olduğunu söyleyebiliriz. Klasik şifreleme algoritmaları için güvenliği tetikleyen en büyük etkenler anahtarın kolay bulunabilmesi veya şifrelemenin kolay kırılabilmesidir.

* Anahtarın kolay bulunabileceği senaryoda bu açık, şifrenin yeterince uzun olmamasından veya anahtarın yeterince karmaşık olmamasından kaynaklanır. Bu yüzden anahtarı oluşturan algoritma şifreyi oldukça karmaşık hale getirmelidir.
* Şifrelemenin kolayca kırılabildiği senaryo ise bazı saldırılar sonucu şifreleme deseninin tespit edilmesi ya da algoritmaların tersten gidilerek şifrelenen verinin keşfedilebilmesi üzerine gelişecek bir olaydır.

**>>>** Bu açılardan güvenliği en yüksek olan klasik şifreleme algoritmaları Viegenere(polialfabetik) şifreleme, Playfair ve Hill şifrelemedir. En düşük olanlar ise Sezar şifreleme ve Monoalfabetik şifrelemedir.

* Blok tabanlı şifrelemeler frekans analizi gibi analiz yöntemlerini zorlaştırır. Permütasyon, Playfair ve Hill şifreleme blok tabanlıdır.
* Matematiksel algoritmalar, matrisler basit analizlerle şifre kırmayı zorlaştırır. Hill şifreleme matris tabanlıdır ve matematiksel işlemler içerir.
* Monoalfabetik ve Sezar şifreleme, sabit harflerle oluşturulan şifrelemeler oluşturması sebebiyle kolay kırılabilirler. Bu yüzden güvenlik seviyesi en düşük olanlardır.

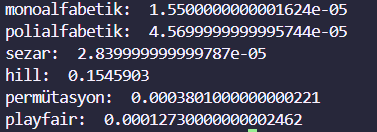


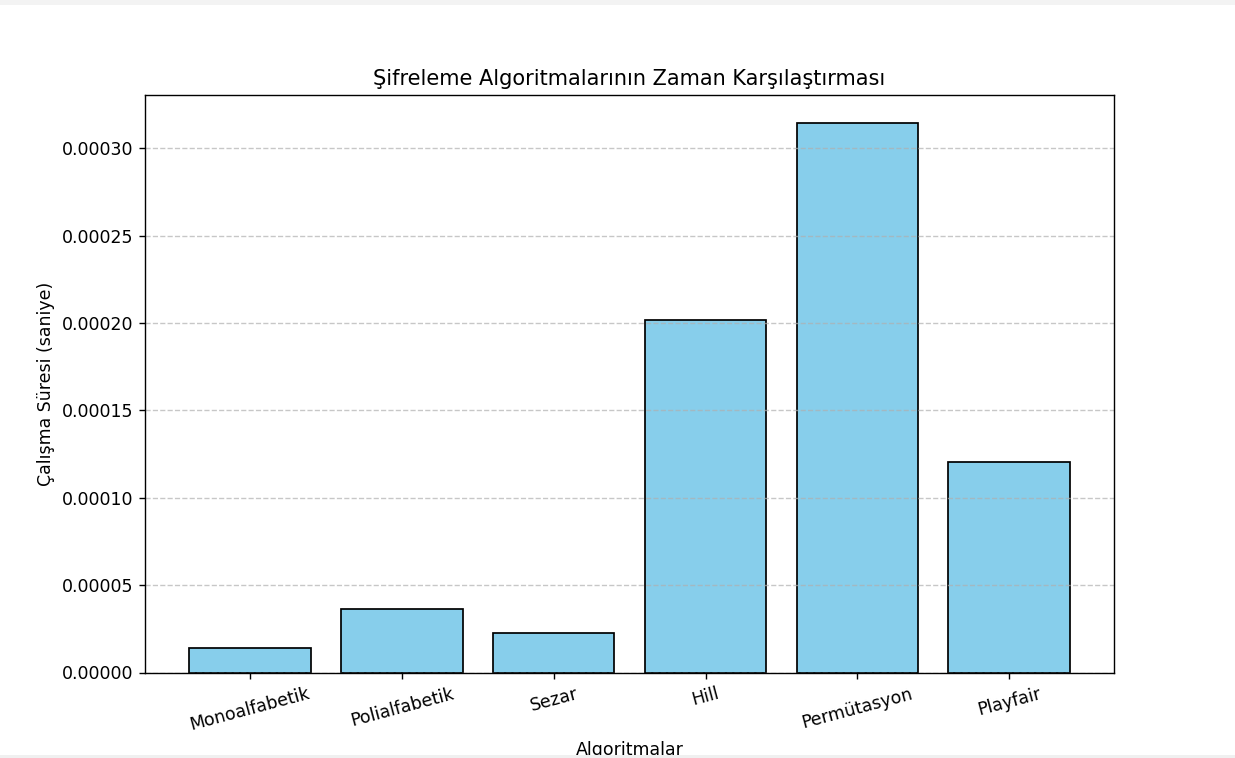
Tabloda olduğu gibi en karmaşık şifrelemeyi oluşturması ve en zor kırılabilen şifre olarak Hill şifreleme tercih edilebilir.

1. **Şifreleme Süreleri**

Şifreleme süreleri algoritmanın içerdiği döngüler, şifrelenecek verilerin büyüklüğü, anahtar uzunluğu, algoritmanın karmaşıklığı gibi kriterlere göre değişkenlik gösterebilir. En güvenli şifreleme en karmaşık ve şifreleme süresi en uzun olan algoritma olabilir. Burada önemli olan verimliliği ve güvenliği en yüksek seviyede tutmaktır. Bu yüzden şifreleme süreleri algoritmanın seçim kriteri için önemlidir.

Aynı sistem içinde, bütün algoritmaların şifrelemesi için aynı açık metnin verilmiş olduğu durumda şifreleme sürelerinin karşılaştırması için “timeit” kütüphanesi kullandım. Bu kütüphane algoritmaların çalışma süreleri ve şifrelenmiş metnin oluşturulması için geçen toplam süreyi hesaplamak içindir. Bunun sonucunda çıkan sonuç:



 jbhklllllllööç……………………… Burkmlfesşföesssssssssss Bnjkllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllllll

* Buradan çıkan sonuca göre zaman karmaşıklığı en düşük olan algoritma monoalfabetik şifrelemedir. Bunun sebebi algoritmasının oldukça basit olması ve karmaşıklığının oldukça düşük olmasıdır. Bu da işlem süresinin çok kısa olmasını sağlamaktadır. Fakat bu basitlik şifrenin kolay çözülebilmesine de sebep olur.

1. **Saldırılara Karşı Dayanıklılık**

Klasik şifreleme algoritmaları artık günümüz şartlarında belirli yöntemlerle tespit edilebilir olmasıyla kullanımı oldukça azalmıştır. Özellikle bazı klasik şifreleme algoritmaları diğerlerine göre çok daha kolay kırılabilen saldırıya açık algoritmalardır. Bunun sebebi basit algoritma kullanımı, yetersiz anahtar uzunluğu, analizlerle kolay deşifre edilebilmeleri gibi pek çok etkenden kaynaklanabilmektedir.

* Sezar Şifreleme 🡪 Şifrenin sadece 25 olası durumdan oluşması sebebiyle Brute Force saldırısıyla kolayca bulunabilir.
* Monoalfabetik Şifreleme 🡪 Her harf sabit başka bir harfe karşılık geldiği için frekans analiziyle anahtar kolayca tahmin edilebilir. Ayrıca bu durum anahtar uzayının sınırlı olmasıyla da ilişkilidir.
* Polialfalfabetik Şifreleme 🡪 Kasiski analizi ya da Friedman testi gibi yöntemler ile anahtar uzunluğu tespit edilebilir ve frekans analiziyle de harflerin tespiti yapılabilir. Metin içinde tekrarlayan anahtar parçalarının aynı şekilde şifrelenmesi şifre tespitini kolaylaştırır.
* Hill Şifreleme 🡪Bilinen düz metin saldırısı yöntemiyle, metin bloklarının matris terslemesiyle birlikte anahtar tespitini yapabilmesi mümkündür. Küçük matrisler kullanılıyorsa bu durum daha kolaylaşır.
* Permütasyon Şifreleme 🡪 Known-plaintext saldırıları şifrelemede kullanılan dizlimin bulunabilmesini sağlayabilir. Karışık bir yapı kullanılsa da çok fazla örnek ile karşılaştırma yaparak tahmin edilebilir. Çünkü harflerin içeriği değil sadece sırası değişmektedir.
* Playfair Şifreleme 🡪 İkili harf çiftleri analiziyle harflerin frekansları incelenebilir. Her ne kadar karmaşık bir algoritma ile şifrelense de frekans analizlerine açıktır.

**>>>** Aralarında karşılaştırma yaparsak düzgün yapılandırılmış bir Hill şifrelemesi diğer şifrelemelere göre kırılması en zor şifrelemedir. Güvenli olma sebebiyle aynı sebeplerden dolayı saldırıya karşı en dayanıklı algoritmadır.