GENETİK KOD YÖNTEMİ İLE KRİPTOLOJİ UYGULAMASI

Mustafa ZENGİN

2021
YÜKSEK LİSANS TEZİ
BİLGİSAYAR MÜHENDİSLİĞİ

Tez Danışmanı
Dr. Öğr. Üyesi Zafer ALBAYRAK

GENETİK KOD YÖNTEMİ İLE KRİPTOLOJİ UYGULAMASI

Mustafa ZENGİN

Karabük Üniversitesi
Lisansüstü Eğitim Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalı'nda
Yüksek Lisans Tezi
Olarak Hazırlanmıştır

Tez Danışmanı
Dr. Öğr. Üyesi Zafer ALBAYRAK

KARABÜK
Ocak 2021

Mustafa ZENGİN tarafından hazırlanan "GENETİK KOD YÖNTEMİ İLE KRİPTOLOJİ UYGULAMASI" başlıklı bu tezin Yüksek Lisans Tezi olarak uygun olduğunu onaylarım.

Dr. Öğr. Üyesi Zafer ALBAYRAK ….....
Tez Danışmanı, Bilgisayar Mühendisliği Anabilim Dalı

Bu çalışma, jürimiz tarafından oy birliği ile Bilgisayar Mühendisliği Anabilim Dalı'nda Yüksek Lisans tezi olarak kabul edilmiştir. 19/01/2021

Unvanı, Adı SOYADI (Kurumu) İmzası

Başkan: Doç. Dr. İlker TÜRKER (KBÜ) ….....

Üye : Dr. Öğr. Üyesi Zafer ALBAYRAK (KBÜ) ….....

Üye : Doç. Dr. Numan ÇELEBİ (SÜ) ….....

KBÜ Lisansüstü Eğitim Enstitüsü Yönetim Kurulu, bu tez ile Yüksek Lisans derecesini onamıştır.

Prof. Dr. Hasan SOLMAZ ….....
Lisansüstü Eğitim Enstitüsü Müdürü

"Bu tezdeki tüm bilgilerin akademik kurallara ve etik ilkelere uygun olarak elde edildiğini ve sunulduğunu; ayrıca bu kuralların ve ilkelerin gerektirdiği şekilde, bu çalışmadan kaynaklanmayan bütün atıfları yaptığımı beyan ederim."

Mustafa ZENGİN

ÖZET

Yüksek Lisans Tezi

GENETİK KOD YÖNTEMİ İLE KRİPTOLOJİ UYGULAMASI

Mustafa ZENGİN

Karabük Üniversitesi
Lisansüstü Eğitim Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalı

Tez Danışmanı:
Dr. Öğt. Üyesi Zafer ALBAYRAK
Ocak 2021, 70 sayfa

İnsanlar, geçmişten günümüze kendileri için özel saydıkları bilgilerin istenmeyen kişilerin ellerine geçmesini önlemek amacıyla çeşitli şifreleme metot ve yöntemleri geliştirmişlerdir. Gelişen bilgisayar teknolojisi karşısında bu yöntem ve metotların çok kısa bir sürede deşifre edilebilir olması, kullanılan yöntemlerin çok zayıf kaldığını göstermiştir.Bu durumda insanlar, gelişen teknoloji karşısında farklı çözüm arayışları içine girmişlerdir.

Günümüzde bilgi ve iletişim araçlarının yaygın olarak kullanılması, bilgiye erişimi kolaylaştırmıştır. Ancak bu gelişmeler, bilgi ve veri güvenliğinin önemini ortaya çıkartmıştır. Veri güvenliği ve güvenli veri aktarımını sağlamak amacıyla çeşitli çalışmalar yapılmıştır. Yapılan çalışmalar sonucunda DES, 3DES, AES, RSA ve DSA vb. birçok şifreleme algoritmaları geliştirilmiştir.

Bu çalışmada, DNA (Deoksiribonükleik asit) yapısından esinlenerek yeni bir Genetik Şifreleme Algoritması (GEA) geliştirilmiştir. Geliştirilen Genetik Şifreleme Algoritması; (GEA) Standart Şifreleme Algoritması (DES), Gelişmiş Şifreleme Algoritması (AES) ve (RSA) Asimetrik Şifreleme Algoritmalarıyla performans karşılaştırması yapılarak kaba kuvvet kırılma süreleri, zaman karmaşıklığı, işlemci ve hafıza kullanımı bakımından incelenmiştir. Sonuçlar tablo ve grafiklerle gösterilerek kısa bir değerlendirilmesi yapılmıştır.

Anahtar Kelimeler **:** Kriptoloji, Simetrik Şifreleme, Asimetrik Şifreleme
Bilim Kodu **:** 92403

ABSTRACT

M. Sc. Thesis

CRYPTOLOGY APPLICATION WITH GENETIC CODE METHOD

Mustafa ZENGİN

Karabük University
Institute of Graduate Programs
Department of Computer Engineering

Thesis Advisor:
Assoc. Prof. Dr. Zafer ALBAYRAK
January 2021, 70 pages

From past to present, scientists have developed various encryption methods in order to prevent the information considered as private from falling into the hands of unwanted people. The fact that these methods can be deciphered in a very short time with the aid of developing computer technology has shown that the methods used are very weak. In this case, people had to search for various solutions across developing technology.

The widespread use of information and communication tools today has facilitated access to information. However, these developments revealed the importance of information and data security. Various studies have been carried out in order to ensure data security and secure data transfer. As a result of these studies, DES, 3DES, AES, RSA and DSA etc. many encryption algorithms have been developed.

In this study, a new Genetic Encryption Algorithm (GEA) was developed, inspired by the DNA (Deoxyribonucleic Acid) structure. Performance comparison of the developed GEA has been conducted with Standard Encryption Algorithm (DES), Advanced Encryption Algorithm (AES) and Asymmetric Encryption Algorithms (RSA). Results of these algorithms were evaluated in terms of brute force break times, time complexity, processor and memory usage. A brief evaluation was made by showing the results in tables and graphics.

Key Word **:** Cryptology, Symmetric Encryption, Asymmetric Encryption
Science Code : 92403

TEŞEKKÜR

Bu tez çalışmasının planlanmasında, araştırılmasında, yürütülmesinde, oluşumunda ilgi ve desteğini hiçbir zaman esirgemeyen, engin bilgi ve tecrübelerinden yararlandığım, kendisinin yapmış olduğu yönlendirme ve bilgilendirmeleriyle çalışmamı bilimsel temeller ışığında şekillendiren sayın hocam Dr. Öğt. Üyesi Zafer ALBAYRAK'a sonsuz teşekkürlerimi sunarım.

Sevgili aileme, manevi olarak hiçbir yardımı esirgemeden yanımda oldukları için tüm kalbimle teşekkür ederim.

İÇİNDEKİLER

Sayfa

Sayfa

ŞEKİLLER DİZİNİ

Sayfa

Sayfa

ÇİZELGELER DİZİNİ

Sayfa

KISALTMALAR

DES : Standart Şifreleme Algoritması
AES : Gelişmiş Şifreleme Standartı
IDEA : Uluslararası Veri Şifreleme Algoritması
TEA : Tiny Şifreleme Algoritması
RC2 : Ron Rivest Şifreleme Algoritması
DH : Diffie Helman Şifreleme Algoritması
RSA : Rivest Shamir Adleman Şifreleme Algoritması
DSA : Digital İmza Algoritması
DNA : Deoksiribo Nükleik Asit
GŞA : Genetik Şifreleme Algoritması
YSA : Yapay Sinir Ağları
NIST : Ulusal Teknoloji Standartları Enstitüsü
KEM : Anahtar Kapsülleme Mekanizması
DEM : Veri Kapsülleme Mekanizması
ASCII : Amerikan Standart Kodlama Sistemi

BÖLÜM 1

GİRİŞ

İnternet teknolojisinin gelişimine bağlı olarak, İnternet'in her alanda yaygın olarak kullanılması veri güvenliğinin önemini ortaya koymuştur. Özellikle e-ticaret, bankacılık, finans, güvenlik, eğitim vb. alanlarda İnternet kullanımı, bilgi güvenliği bakımından dikkat edilmesi gereken uygulamaların başında gelmektedir. Günümüzde insanların internette geçirdikleri süre dikkate alınarak yapılan bir araştırmada dünya ortalaması 6 saat 45 dakika iken Türkiye'de bu süre 7 saat 29 dakikayla ortalamanın üzerinde yeralmaktadır [1]. Bilgiye erişimin bu kadar kolay olduğu bir yerde veri güvenliğini sağlamak çok önemlidir [2]. Saldırılara veya tehditlere karşı koymak için verilerin güvenli bir şekilde korunması, başka yerlere iletilmesi önemli bir konu haline gelmiştir.Veri transferlerinde gizli kalması istenen bilgilerin yetkisiz kişi veya kişilerce iletişim ağına girerek, verilerin değiştirilmesi veya silinerek yok edilebileceği düşüncesi her zaman önemli bir problem oluşturmuştur [3]. Bilgisayarlar arasında güvenli veri alışverişlerinin sağlanması için yapılması gereken en basit yöntemlerin başında verilerin şifrelenmesi gelmektedir [3].Veri güvenliği ve verilerin korunması amacıyla geliştirilmiş birçok şifreleme metodu bulunmaktadır [2]. Bu yöntem ve metotlar, Kriptografi konusuyla açıklanmaktadır. Kriptografi, bir mesajın veya verinin çeşitli matematiksel işlemlerden geçirilerek geçici olarak anlamsız, okunamaz hale dönüştürülerek istenen hedefe gönderilmesi ayrıca istenen hedefe ulaşmış verinin orada tekrar çeşitli matematiksel işlemlerden geçirilerek verinin ilk hali olan normal okunabilir haline döndürülmesi işlemidir [3].

Günümüzde kullanılan Kriptografik algoritmalar, kullanılan anahtar yapısına göre simetrik ve asimetrik olarak iki kısımda incelenmektedir [4]. Simetrik şifreleme algoritmaları, verileri şifrelerken ve şifreli verileri çözerken tek bir gizli anahtar kullanır. Bu durum, Simetrik şifreleme algoritmalarının en büyük

güvenlik zafiyetini oluşturur [4]. Çünkü şifrelemede kullanılan anahtarın güvenli bir şekilde karşı tarafa iletilerek aynı mesajın çözümlenmesinde kullanılacak olması, anahtar güvenliğinin önemini ortaya çıkartmaktadır [5]. Simetrik şifreleme algoritmaları, Asimetrik şifreleme algoritmalarına göre daha hızlıdır ve performansları yüksektir. Fakat ortak tek bir anahtar kullanıldığı için güvenlik zafiyeti daha fazladır [5]. En yaygın kullanılan Simetrik şifreleme algoritmaları olarak DES, 3DES, AES gösterilmektedir. Simetrik şifreleme algoritmalarının çalışma prensibi genel olarak incelendiğinde istenilen mesajı bloklara bölüp bitlere dönüştürerek şifreleme işlemi gerçekleştirmeleridir [6].

Asimetrik şifreleme algoritmalarında herkese açık iki anahtar kullanılır. Verilerin şifrelenmesinde ve şifreli verilerin çözümlenmesinde farklı iki anahtar kullanılması, yüksek güvenlik önlemi sağlar. Fakat Simetrik şifreleme algoritmalarına göre çok yavaş ve işlem hızı uzun sürer [2]. En yaygın kullanılan Asimetrik Şifreleme Algoritması RSA ve DSA Algoritması'dır [7]. Asimetrik şifreleme algoritmalarının temel özelliği büyük asal sayılarla işlem yaptıkları için yapılması kolay, geri alması zor veya zaman alan işlemlerdir. Mesela iki sayıyı çarpmak kolay fakat çarpanlarını bulmak zordur veya zaman alır. Bir sayının karesini almak kolay fakat karekökünü bulmak zordur ya da çok zaman alır. Bu sebeple Asimetrik şifreleme algoritmaları en güvenilir şifreleme algoritmalarıdır [7].

Kriptografik algoritmaların performansı ve başarısı, şifrelemede kullanılan anahtar boyutu, işlem hızı ve kullanılan bellek miktarına göre belirlenmektedir [8]. Algoritmaların kaba kuvvet kırılma süreleri, kullanılan anahtar boyutuna göre değişiklik göstermektedir [8]. Örnek DES algoritması 56 bitlik anahtar yapısına sahiptir. Kaba kuvvet şifre kırılma süresi 2^{56} dır [8]. AES algoritması 128 bitlik anahtar yapısına sahip olduğu için kaba kuvvet kırılma süresi 2^{128} dir.

Çalışmanın amacı, bilgisayar ağları arasında veri güvenliği ve güvenli veri aktarımı sağlamak için günümüzde kullanılan şifreleme algoritmalarına göre daha iyi performans gösteren algoritmalar geliştirmektir. Yapılan çalışmalar sonucunda geliştirilen GEA (Genetik Şifreleme Algoritması), simetrik bir şifreleme, algoritma özelliği taşıdığı için işlem hızı Asimetrik şifreleme algoritmalarına göre daha hızlıdır.

Kullanılan anahtar boyutu 128 bit olduğu için de kaba kuvvet kırılma süresi 2^{128} olmasından dolayı kırılma süresi uzun ve zordur. GEA Şifreleme Algoritması DES, 3DES ve RSA şifreleme algoritmalarına göre daha hızlı ve DES algoritmalarına göre kırılma süresi uzun ve daha zordur.

1.1. AMAÇ

İnternet teknolojisinin insan hayatındaki önemi her geçen gün artarak devam etmektedir. Bilinçli kullanıldığında İnternet'in ne kadar çok faydası varsa, bilinçsiz kullanıldığında da o kadar çok zararı vardır. İnternet'in insanlara sağladığı faydalara bakarsak, e-ticaret, bankacılık, finans, güvenlik, eğitim vb. birçok alandaki kullanımı, hayatımızı kolaylaştırıcı bir teknoloji olduğunu görürüz. Zararlarını da en aza indirmek için bilinçli tüketiciler olmalıyız. Bu sebeple de teknolojiye uzak durup hayatımızdan çıkartmak yerine bilinçli kullanıcılar olarak istifade etmeliyiz. İnternet teknolojisinin gelişmesi ve bilgi teknolojisine giderek artan bağımlılık, bilginin erişilmesinde, korunmasında ve paylaşılmasında büyük önem taşımaktadır. Bu sebeple günümüzde veri güvenliği ve güvenli veri aktarımını sağlamak için geliştirilen çeşitli şifreleme yöntemleri çok önemli yer tutmaktadır.

Bu çalışmanın diğer bir amacı, iletişim araçları veya bilgisayar ağları arasında veri alışverişlerinin daha güvenilir biçimde yapılmasını sağlamak ayrıca toplumda teknolojiyi bilinçli kullanan bireylerin sayısının artmasına katkı sunmaktır. Kişisel verilerinin korunması, kişi veya kurumlar arasındaki iletişim sırasında veri güvenliğinin sağlanması, verilerin istenmeyen kişiler tarafından ele geçirilmesini önlemek amacıyla bilinen şifreleme algoritmalarına göre performansı daha yüksek şifreleme uygulaması geliştirmektir.

1.2. LİTERATÜR TARAMASI

"Şifreleme yöntemleri ve RSA Algoritması üzerine bir inceleme" adlı eserde şifreleme algoritmaları, Simetrik ve Asimetrik olmak üzere iki başlık altında incelenmiştir. Bu çalışma, Simetrik ve Asimetrik şifreleme algoritmalarının genel özelliklerini barındıran ancak literatürde önemli bir yere sahip olan RSA Algoritması'nın şifreleme yöntemleri üzerindeki etkisini analiz eden Asimetrik

şifreleme algoritmalarından biridir. Çalışmada, RSA Algoritması'nın yapısı, genel özellikleri, avantajları ve dezavantajları hakkında bilgi verilmiştir [7].

"Bilgi güvenliği kapsamında yeni bir veri şifreleme algoritması tasarımı ve gerçekleştirilmesi" Bu çalışmada, bilgi güvenliği kapsamında özgün bir veri şifreleme algoritması geliştirilmiştir. Geliştirilen algoritma, "tek harf değiştirme" teknolojisine göre tasarlanmış ve algoritmanın oluşturulmasında Sezar Şifreleme Teknolojisi, Çok Harfli Algoritma ve "gizem" kullanılmıştır. Bu araştırmanın amacı, yazılı metnin güvenliğini sağlamak ve gerektiğinde şifre çözme yoluyla bilgi güvenliğini sağlamaktır [9].

"Matematiksel ifadelerin üretimi ve çözümüne dayalı bir kriptoloji yöntemin tasarımı ve gerçeklemesi" Bu çalışmada, mesajları matematiksel ifadelerde gizleyebilen ve bunları matematiksel ifadelere dönüştürebilen bir yöntem önermektedir. Oluşturulan matematiksel ifade, güvenli metinle birlikte bir mesaj iletmek için bir kapak sayfası olarak kullanılabilir. Hedefimiz, matematiksel ifadelerde mesajları gizlemek için yeni yollar geliştirmektir. Metodolojinin ilk aşaması, matematiksel ifadelerin anlamını açıklayan biçimsel ifadelerin geliştirilmesini içerir. Söz dizimi bildirimine dayalı bir ayrıştırıcı oluşturmak için derleyici oluşturucu JavaCC kullanılmıştır. Ayrıştırıcı, belirli matematiksel ifadeleri, diğer yöntem aşamalarındaki düğümlerde gezinerek değerlendirilebilen soyut söz dizimi ağaçlarına (AST) dönüştürmek için kullanılır. Önerilen yönteme uygun bir dilbilgisi geliştirilerek çeşitli matematiksel ifadeler üretilmiştir. Ayrıca kağıtta daha güvenli olabilmesi için gizli mesaj gizlenmeden önce geliştirilen yeni bir yöntemle şifrelenmektedir [10].

"Veri şifreleme tekniklerinin incelenmesi ve uygulanması" Bu tez çalışmasında, şifreleme ve metin steganografisi kullanan şifreli bir gizli mesaj sunulur. AES Algoritması kullanılarak kapak sayfası, metne gömülüdür, bu nedenle saldırgan, kriptanaliz için bunları ayıramaz. "Şifreleme" bölümünde, AES Algoritması'na dayalı mesaj şifreleme için Gelişmiş Şifreleme Standardı (AES) kullanılır. AES Şifreleme Algoritması, 128 blokluk bit boyutu ve 256 anahtarlık bit boyutu kullanır. Veri depolamak için iki steganografik yöntem, yani "modifikasyon alanı" ve "kelime seçimi" kullanılır. "Modifikasyon Uzayları" yöntemi, "boşluk" kullanan gizli

mesajları gizlemek için "boşluk" kullanan bir steganografi aracı oluşturur. "Kelime seçimi" yöntemi, kelime listesi dosyasındaki her yontulmuş metin değerine bir anahtar kelime atamaya dayanır. Kelime listesi, zamanla ilgilidir ve her bir eşdeğer kelime, zaman kaydırma denklemine göre değişecektir. Deneysel sonuçlar, "kelime seçme" yeteneği açısından en iyi steganografi yönteminin olduğunu göstermektedir. Her iki yöntemin de doğrulanması zordur. Test ayrıca önerilen yöntemin hem steganografi yöntemleri hem de daha az hesaplama için hızlı bir şekilde çalışabileceğini doğrulamıştır. Önerilen yöntem, gönderici ve alıcı arasındaki yüksek güvenlik ve sağlamlığı karşılayabilen, hibrit kriptografi ve steganografiye dayalı yüksek güvenlikli bir veri iletim sisteminin tasarımına yol açmıştır [2].

"Şifreleme algoritmalarının hızını etkileyen faktörler" Bu tez çalışmasında, RSA Şifreleme Sistemi'nin şifreleme ve şifre çözme süresini etkileyen faktörleri incelemektir. Çalışmada şifreleme işlemlerinde kullanılan temel matematiksel kavramlar detaylı olarak açıklanmıştır. Daha sonra simetrik ve asimetrik şifreleme yöntemleriyle kullanılan algoritmalar anlatılmıştır. Bu algoritmaları uygulamak için Java programlama dili kullanılmış, şifreleme ve şifre çözme zamanı hesaplanarak karşılaştırma yapılmıştır. Bu algoritmaların avantajları ve dezavantajları tartışılmaktadır [11].

"Kriptoloji ve veri şifreleme teknikleri üzerine" Bu tez çalışmasında, güncel şifreleme sistemleri, bu sistemlerin uygulamalarında karşılaşılan sorunlar ortaya konarak, sorunlara karşı çözüm önerileri gözden geçirilmiştir. Bu durumda mevcut şifreleme sistemlerinin avantaj ve dezavantajları incelenmiştir. DES, 3DES, AES, Blowfish simetrik şifreleme yöntemleri ve RSA açık anahtar şifreleme yöntemleri Java'da uygulanmış ve karşılaştırılmıştır. Java ve Python dilinde yazılmış bu şifreleme sistemlerini kullanarak mesajları şifreleyebilir, aynı mesajların şifresini çözebilirsiniz. Java güvenlik paketi, şifreleme, şifre çözme, anahtar oluşturma, anahtar yönetimi altyapısı, kimlik doğrulama ve yetkilendirme işlevleri gibi birçok güvenlik çözümü sağlasa da Blowfish Algoritması'nı içermez. Tek bir yazılımda karşılaştırma yapabilmek için Blowfish Algoritması bir kütüphaneye dönüştürülmüş, bu kütüphane; harici olarak Java Güvenlik Kütüphanesi'ne eklenmiştir [5].

“Dalgacık dönüşümü tabanlı görsel kriptoloji” Bu çalışmada, dalgacık dönüşümü ile gizli paylaşımdan elde edilen paylaşım, uygulama için kapak resmine gömülür. Görsel gizli paylaşım yoluyla elde edilen gürültü benzeri görüntüler insanların ilgisini çektiğinden, bu paylaşımlar dalgacık dönüşümü yardımıyla anlamlı kapak görüntülerine gömülmüş ve bu paylaşımlar dikkat çekmemiştir. PSNR değeri, gerçek sır görüntüsü ile elde edilen sır görüntüsü karşılaştırılarak hesaplanır. Bunun için uygun kod yazılmıştır. Bu yöntemle şifreleme gerçekleştirilmiştir [12].

“Socketler üzerinden özel haberleşmede kriptoloji metotların kullanılması üzerine bir uygulama” Bu tez çalışmasında ilk şifreleme sisteminden günümüz teknolojisinde kullanılan şifreleme sistemine kadar tüm çalışma prensipleri açıklanmıştır. Rijndeal Algoritması son teknolojide kullanılan gizli anahtar şifreleme yöntemlerinden biri olduğu için ortadaki adamın saldırısı altında bile iletişim güvenliğinin sağlanmasını sağlayabilir. Burada Steganografi teknolojisi, iletişim sürecinde ortadaki adam saldırıları gerçekleştiren saldırganların şüphesini ortadan kaldırmak için uygulanmakta, iletişim şifrelenmektedir. Gizli Anahtar Şifreleme Sistemi’nin doğası gereği şifreleme ve şifre çözme anahtarları da sağlanmaktadır [13].

“Cebirsel Kriptoloji yöntemleri ve bazı uygulamaları” Bu tez çalışması Kriptolojiyi iki bölümde ele alarak ilk bölümde Kriptoloji, Kriptoloji’nin tarihsel gelişimi ve kullanılan şifreleme yöntemleri anlatılır. İkinci bölümde, Sezar Şifreleme, Polybius Dama Tahtası, Bifid ve Trifid Şifreleme, Playfair Şifreleme, Affine Şifreleme, Morse Mektup, Hill Şifreleme, Kuvvet fonksiyonu ve RSA Şifreleme Yöntemi’nin şifreleme yöntemleri, örnekler üzerinden incelenmiştir [14].

“Modern Blok Şifreleme Algoritmalarının Gücünün İncelenmesi” Blok şifreler modern şifreleme algoritmaları için oldukça önemlidir. Bundan dolayı veri iletişimindeki güvenlik düşünüldüğünde şifreleme algoritmalarının gücü oldukça önemli bir kriterdir. Diğer yandan, blok şifreler onlara gücünü veren önemli özelliklere sahiptir. Bu çalışmada blok şifrelerin gücünün detaylı bir analizini sunularak ve bu amacı gerçekleştirmek için AES (Rijndael) algoritması incelemişlerdir [15].

"Kriptoloji sistemleri ve uygulamaları üzerine" Bu tez çalışmasında, RSA parametrelerinin seçiminde dikkat edilmesi gereken konuların detaylı bir çalışma prensibi anlatılarak günlük hayatta en güvenli açık anahtar şifreleme algoritması olduğu vurgulanmıştır. RSA Sistemi'nin güvenliği iki büyük asal sayının çarpımı tarafından oluşturulan yeni bir sayının ayrıştırılmasının zorluğuna bağlıdır. Temel olarak gösterilebilecek etkili bir matematiksel çözüm yöntemi, Fermat Çarpanlara Ayırma Yöntemi'dir. Diğer mevcut çarpanlara ayırma yöntemleri (örneğin, sayısal alan filtreleme) makul bir sürede etkili bir şekilde çalışamaz. Fermat Ayrıştırma Yöntemi'ne direnen asal veya ikili dosyalar büyük bir küme oluşturacaktır. Küme içinde bilinen belirli saldırılara direnebilecek bir bileşik sayı oluşturmak için bazı araştırmalar yapılmıştır. Bu kapsamda, p ve q asalları seçilirken bu asal sayılar arasındaki uzaklık ölçümü tanımlanarak, bunun için uygulamalı olarak çeşitli sınırlar belirlenmiştir [16].

"Kriptoloji kullanımının fonksiyon kavramının anlaşılmasına etkisi" Bu çalışmada, Kriptoloji'nin öğretim işlevlerinin kullanılmasında ve bunların günlük hayata uygulanmasıyla öğrenme güçlüğü çeken öğrencilere destek sağlayıp sağlamadığı incelenmiştir. Çalışmada temel işlevsel kavramların öğretiminde Kriptografi kullanmanın avantaj ve dezavantajlarını ortaya çıkarmayı amaçladığından nitel veriler toplanmakta ve analiz edilmektedir. Bu çalışma için veri toplama aracı, gözlemci araştırmacılar, uygulayıcı araştırmacılar ve değerlendirme uzmanları tarafından hazırlanan 10 sorudan oluşan bir başarı testi bulunur. Bu başarı testi, ön test veya son test olarak kullanılabilir. Fonksiyonlar açısından sadece fonksiyon tanımları, ters fonksiyonlar, bire bir fonksiyonlar, alan değer setleri, görüntü setleri ve günlük hayatta kullanılabilecek fonksiyonlar kullanılmaktadır. Araştırma sonuçları, öğretmen adaylarının işlevsel farkındalığının arttığını ve işlevsel konuları günlük yaşamla bütünleştirmek için Kriptografi'nin kullanılmasının faydalı olabileceğini göstermektedir. Ancak bu olumlu durumun başarı testindeki tüm sorular için etkili olmadığı görülmüştür. Öğretmen adaylarının bire bir işlevlerde, üstün işlevlerde, tanımlarda ve değerlerde olumlu ya da olumsuz gelişme gösterdiği söylenemez. Dolayısıyla bu konuda Kriptografik etkinlik kullanımının dezavantajlı olduğu söylenebilir. Yapılan etkinlikler ve öğretilen ders süreci bize bu sorulara verilen ön test ve son test cevapları arasında fark olmadığını göstermiştir [17].

“Kriptoloji’ye Giriş” Bu tez çalışmasında, şifreleme ve bazı klasik şifreleme türlerinden bahsedilmiştir. Gopala-Hemachandra (GH) kodu, Fibonacci kodunun bir varyantıdır. Çeşitli sıralarda tanımlanmış ve GH kodları veya belirli sıralarda pozitif tamsayı gösterimleri elde edilmiştir. Şifrelenmiş örnekler, bu temsiller veya bu kodlar kullanılarak belirli koşullar altında gerçekleştirilir [4].

“Fermat sayılarının asal çarpanlarına ayrılması ve kriptoloji uygulamaları” Bu tez çalışmasında, kriptoloji bilgileri şifrelemeyi ve şifrelenmiş metnin şifresini çözmeyi amaçlamaktadır. Büyük asal sayılar ve asal sayı ayrıştırması kullanır. Fermat sayılarının asal çarpanları da çok büyük olduğundan bu sayıları asal sayılara ayırmak özellikle çok önemlidir. Modüler aritmetik yöntemi kullanarak fermat numarasını asmak için fermat çarpanlarına ayırma algoritmasının iyileştirilmesini inceler. 2n modunda farklı n sayıların kalan karelerinin alabileceği değerler dikkate alınarak fermat sayısının kare farkı şeklinde gösterilemeyen değer elimine edilir, tarama vakalarının sayısı azaltılır ve algoritmanın hesaplama süresi büyük ölçüde azaltılarak geliştirilmiş şifreleme yöntemidir [18].

“Yapay sinir ağları ile Kriptoloji uygulamaları” Bu çalışmada, yapay sinir ağlarının (YSA) Kriptografi’nin 3 farklı uygulama alanında kullanılabilirliği incelenmiştir. İlk olarak, YSA tabanlı bir sözde rastgele sayı üreteci tasarlandı ve rastgeleliği NIST (Ulusal Teknoloji Standartları Enstitüsü) istatistiksel testi ile test edildi ve 7 testi başarıyla geçti. YSA modelleme yoluyla, ağırlıkları nöron sayısını ve transfer işlevlerini gizli anahtarlar olarak kullanan halka açık bir değişim altyapısına sahip bir kriptografik sistem uygulaması hayata geçirildi. YSA tarafından modellenen Kriptografik sistemin hangi şifreleme algoritması kullanılırsa kullanılsın şifresinin çözülebileceği öğrenilmiştir. Daha sonra dijital imza sürecinde kullanılabilecek YSA tabanlı görüntü ve metin hash fonksiyonu uygulamaları geliştirilmiştir [19].

“Kriptoloji’de bazı şifreleme yöntemlerinde cebirsel yaklaşımlar” Bu çalışma, yaşadığımız bilgi çağında bilgi güvenliğinin sağlanmasının yanı sıra bilginin aktarılması ve depolanmasının önemini vurgulamıştır. Bu yöntemlerin çoğu bazı temel ilkelere dayanmaktadır. Bu yöntemler, matematiksel parametreler içeren algoritmalardan oluşur. Özellikle, günümüzde yaygın olarak kullanılan Açık Anahtar

Şifreleme Sistemleri'nde bulunan algoritmaların bilgisayarlarda bile şifresini çözmek zordur. Bu araştırmada geçmişten günümüze bazı şifreleme algoritmalarında kullanılan cebirsel yöntemler incelenmiş ve bu kodların nasıl kırılacağına odaklanılmıştır [20].

"Data şifreleme algoritmaları ve performans analizi" Bu çalışmada veri şifreleme algoritmaları tanımlanarak, farklı anahtarlarla Simetrik ve Asimetrik şifreleme karşılaştırılmıştır. Performans karşılaştırma sonuçları elde edilerek analiz edilir. RSA, AES ve DES kaynak kodları, analiz için ModelSim Altera Web Edition programı kullanılarak yazılmıştır. Ardından hız ve bellek gibi faktörler göz önünde bulundurularak algoritmanın şifreleme ve şifre çözme yetenekleri detaylı olarak kontrol edilmiştir. Analiz sonuçlarını kısaca analiz edersek RSA Algoritması, işlemleri daha uzun sürede gerçekleştirecektir. AES Algoritması, RSA ve DES'ten çok daha hızlıdır ve DES Algoritması RSA'dan daha hızlıdır. Ancak RSA daha güvenli olmasıyla diğer uygulamalardan bir adım öndedir [21].

"Kriptoloji'de eliptik eğri algoritması" Bu tez çalışmasında, eliptik eğri kodlamasının matematiksel temelini ve tanımını oluşturarak El-Gamal eliptik eğri kodlamasının uygulamasını geliştirmiştir. Eliptik Eğri Şifreleme Algoritması, RSA Şifreleme Algoritması ile karşılaştırılmıştır. Eliptik Eğri Şifreleme Algoritması'nı oluşturan temel matematiksel yapının özel anahtar uzunluğundan daha önemli olduğu vurgulanmıştır. Eliptik Eğri Şifreleme Algoritması'nın açık anahtar şifrelemede RSA Algoritması'na göre daha yüksek güvenlik sağladığı görülmüştür. El-Gamal Eliptik Eğri Şifreleme örnek programı, Java bilgisi kullanılarak yazılmıştır [22].

"Şifreleme algoritmalarının performans analizi" Bu çalışmada bilgi güvenliği amacıyla en çok kullanılan simetrik ve asimetrik algoritmalar; zaman karmaşıklığı, işlemci karmaşıklığı ve bellek karmaşıklığı açısından incelenmiştir. Geliştirilen uygulama yardımıyla algoritmalarının performans ve performans sıralaması karşılaştırılmıştır. Simetrik şifreleme algoritmasında, Blowfish, Twofish, IDEA, TEA, DES, AES, 3DES, RC2 şifreleme algoritması ve Asimetrik Şifreleme Algoritması'nda RSA Algoritması incelenmektedir [8].

“Kriptolojik uygulamalarda bazı istatistik testler” Bu tez çalışmasında, rastgele sayı üreticilerinin seçimi ve testinden bahsedilmektedir. Geliştirilen uygulama tanımına göre jeneratörlerin seçilmesi ve özelleştirilmesi için bazı kriterlerden bahsedilmektedir. Yapılan deneyler, uygulamanın etkinliğini kanıtlamıştır. Ayrıca istatistiksel test konusuna ve bunun şifreleme analiziyle ilişkisine değinilmiştir. Genel kuralları etkilemeden istatistiksel testlerin Kriptanaliz’in yerini alamayacağı çalışması gösterilmiştir [23].

“Verileri Nota Kullanarak Şifreleme ve Ses Dosyası İçerisine Gizleme” Bu çalışmada, veri şifrelemeye ve veri içerisine veri saklamaya farklı bir yaklaşım getirilmiştir. Önce AES şifreleme algoritmasıyla şifrelenen veriler daha sonra müzik notalarına dönüştürülmekte, dönüştürülen bu notalar AES’nin S-kutusu’nu (S-box) güncellerken aynı zamanda bir ses dosyasındaki müziğin notalarına dönüştürülmeleri için gerekli olan veriler aynı ses dosyası içerisine seçilen bir anahtar yardımıyla gizlenmektedir. Sonuçta ortaya şüphelenilmeyen fakat gizli verileri taşıyan bir şarkı çıkmaktadır [24].

“Temel şifreleme algoritmaları ve kripto analizlerinin incelenmesi” adlı tez çalışmasında, temel şifreleme algoritmasını ve şifreleme analizini açıklamaktadır. Tezin ilk bölümünde basit bir şifreleme tanımı ve şifrelemede kullanılan temel terimler, en eski şifreleme yolu tanımlanmıştır. Makalenin ikinci bölümünde şifrelemenin temelini oluşturan bazı matematiksel teoriler ve kanıtlar sunulmuştur. Sonraki bölümlerde temel şifreleme algoritmaları ve bu algoritmaların şifreleme analizlerinden bahsedilmektedir. Çalışmanın beşinci ve altıncı bölümlerinde bu temel algoritmalar Türkçe harflerle Kriptografik analizlere uygulanmıştır. Son kısım, temel kriptografik algoritma üzerinde Türk harflerinin kullanımının sonuçlarını içermektedir [25].

“Görüntü şifreleme algoritmaları ve performans analizleri” adlı tez çalışmasında, mevcut görüntü şifreleme algoritmasını incelemiştir. Hızlı bilgisayarların ve güçlü sistemlerin gelişmesi nedeniyle verilerin çok hızlı bir şekilde iletilebilirliği ve iyi şifreleme işlevlerine sahip olabileceği vurgulanmıştır. Kullanılan şifreleme algoritmasının güvenilirliğini ve hızlı veri iletimini test etmek çok önemlidir.

Resimleri yoğun bir şekilde kullanmaya başladığımızda resimlerin güvenliğini sağlamak çok önemli bir hale gelmiştir. Bu nedenle günümüzde çeşitli şifreleme yöntemleri geliştirilmiştir. Görüntü şifreleme algoritmalarının genel yapısı ve performansı hakkında bilgi verilmektedir [26].

"Veri şifrelemesinde Simetrik ve Asimetrik anahtarlama algoritmalarının uygulanması" Bu çalışmada, hibrit şifreleme algoritmalarından bahsedilmiştir. Şifrelemede kullanılan simetrik ve asimetrik anahtar algoritmalarının avantajları ve dezavantajları vardır. Bu iki dezavantajı birleştiren bir sisteme Hibrit şifreleme denir. Hibrit şifrelemede farklı yöntemler kullanılmaktadır. Burada anlatılan araştırmada, yeni anahtar kapsülleme mekanizması (KEM) ve veri kapsülleme mekanizması (DEM) yapıları önerilmiştir. Klasik KEM ve DEM yapılarında, bu yapılardan herhangi birinde kusur varsa, herhangi biri aynı anda her ikisine de saldırarak şifreyi kırabilir. Bu uygulamada bu tür saldırıları önlemek için klasik yapıya hash algoritması eklenmiştir. Bu Hash Algoritması'nda önce KEM ve DEM birleştirilir ve ardından baytlar simetrik bir anahtar kullanılıp değiştirilerek şifreleme yapılmaktadır. Buradaki amaç, ikili yapıyı tek bir bloğa dönüştürmektir. Ayrıca bu yöntemde her mesaj ayrı bir anahtar ile şifrelenerek tüm iletişim trafiği kontrol edilmiş olur [27].

"Nükleobazlar ve Nükleositlerde Tautomer kararlılığının moleküler modelleme yöntemleriyle belirlenmesi ve mutasyon etkisinin araştırılması" Bu çalışmada, DNA'daki Nükleositlerin, Nükleosit bazları ve olası İzomerleri hesaplanarak gaz ve su ortamlarındaki kararlı bazlar belirlenmiştir. DNA Nükleotid birimlerinden oluşan iki uzun polimer yapıdan oluşur. Bu Nükleotidler, Nükleobazlar, Deoksiriboz ve Fosfat gruplarının bir kombinasyonuyla oluşturulur. DNA zincirleri Nükleotidlerin bir araya gelmesiyle oluşturulur. Nükleosit bazları ve Nükleositlerle yapılan deneylerin yüksek izolasyon ve araştırma maliyeti gibi dezavantajları vardır. Bu sebeple bu tür biyomoleküler araştırmalar hesaplama biliminin konusu haline gelmiştir. Hesaplamalı araştırma, deneylerin yerini alamasa da düşük maliyetleri, çevreye zararsızlıkları ve bilgisayar teknolojisinin gelişmesiyle hızlanmaları nedeniyle giderek daha önemli hale gelmiştir. DNA hasarına neden olabilecek olası Tautomerler incelenerek nükleobazlar ile Nükleositlerin yapısındaki olası Tatomerler arasındaki enerji farkları hesaplanmıştır. Nükleobaz halindeki en kararlı Tautomerin

yapısı ile Nükleosit halindeki en kararlı Tautomer'in yapısının tam olarak aynı olmadığı tespit edilmiştir [28].

Bilgi ve veri güvenliğinin önemi üzerine çeşitli şifreleme algoritmaları kullanılmaktadır. Bu şifreleme algoritmalarının birbirlerine göre avantaj ve dezavantajları kullanılan anahtar yapısına göre değişiklik gösterir. Geliştirilen Genetik Şifreleme Algoritması (GEA), simetrik şifreleme algoritmaları gibi verilerin şifrelenmesinde ve şifreli verilerin çözümlenmesinde tek bir anahtar kullanır. Bu durum Genetik Şifreleme Algoritması'nın Asimetrik şifreleme algoritmalarına göre daha hızlı işlem yapmasını sağlar. GEA, 128 bitlik anahtar yapısına sahip olduğu için kaba kuvvet kırılma süresi 2^{128} dir. Bu durum GEA şifrelenen verilerin kaba kuvvet yöntemiyle çözülmesi çok uzun süre alacağını ve algoritmanın güvenilirliğini artırmaktadır.

BÖLÜM 2

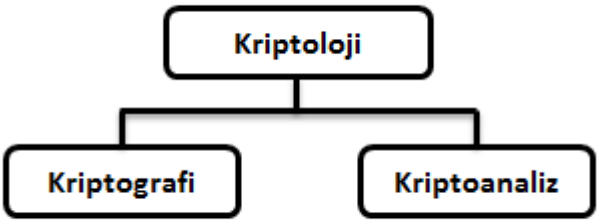
KRİPTOLOJİ

Kriptoloji terimi, Yunanca kryptos ve logos yani saklı, gizli anlamındaki kelimelerin birleşimidir. Literatürde şifreleme bilimi olarak bilinmektedir. Kriptoloji, Kriptografi (şifreleme) ve Kriptonaliz (şifre çözme) olarak iki bölümde incelenir. Kriptoloji, karşılıklı iki bilgisayar arasında güvenli veri alışverişini sağlayan temelinde matematiksel işlemlere dayanan uygulamalardır [4].

Şifreleme, bir mesajın veya verinin gizliliğinin sağlanması için güvensiz ortamlarda dahi istenilen yere ulaştırılmasını sağlamak amacıyla mesajın uygun bir anahtar yardımıyla şifrelenerek anlamsız, karmaşık hale dönüştürülme işlemidir. Şifrelenen mesaj, alıcı tarafından çözülebilmesi için şifrelemede kullanılan anahtar yardımıyla verinin tekrar ilk haline dönüştürülmesi sağlanır. Böylelikle şifreli mesaj başkasının eline geçse bile şifrelemede kullanılan anahtar olmadan mesajın içeriği anlaşılamaz veya anlaşılması zaman alır [4,8].

Günümüzde Kriptografi bilimi sadece mesajların şifrelenmesi veya şifreli mesajların çözümlenmesinin dışında kimlik denetimi (sayısal imza) imkanı da sunmaktadır. Sayısal imza artık veri güvenliği kadar önem taşımaktadır. Herhangi bir mesaja adımızı yazıp ağ üzerinden karşı tarafa iletmek istediğimizde kimliğimizin doğrulanması istenir. Kriptografi'nin buna sunduğu çözüm ise sayısal imzadır [8,21].

Kriptografi: Gizli tutulması istenen mesajın değişik matematiksel işlemler ve algoritmalar yöntemiyle şifrelenmesidir. Kısaca mesajın istenmeyen kişilerce anlaşılmayacak hale dönüştürülmesinde kullanılan tüm metot ve yöntemlerdir.
Kriptoanaliz: Şifrelenmiş mesajın tekrar eski haline dönüştürülmesi için yapılan tüm metot ve yöntemlerdir [21,29].



Şekil 2.1. Kriptoloji bilimi.

Veri güvenliğinin sağlıklı bir şekilde yapılabilmesi için şifreleme yöntemleri ve algoritmaların sağlaması gereken bazı güvenlik kavramları mecuttur. Şifreleme algoritmaları:

Gizlilik: İstenilen veriye sadece yetkisi olan kişilerin erişebilecek olması ve yetkisiz kişilerin erişememesini sağlaması gerekir.

Bütünlük: Bilginin değiştirilme veya yok edilmesine karşı korunması sağlanmalıdır.

Süreklilik: Bilginin güvenli bir şekilde ve zamanında kullanılması gereklidir.

Kimlik denetimi: Verilerin güvenilir kullanıcıdan ve kaynaktan geldiğinin kontrolü sağlanmalıdır.

İnkâr edilememeli: İşlem sırasındaki çeşitli bulgular yardımıyla veri alışverişinin yapılıp yapılmadığının sağlanması gerekir.

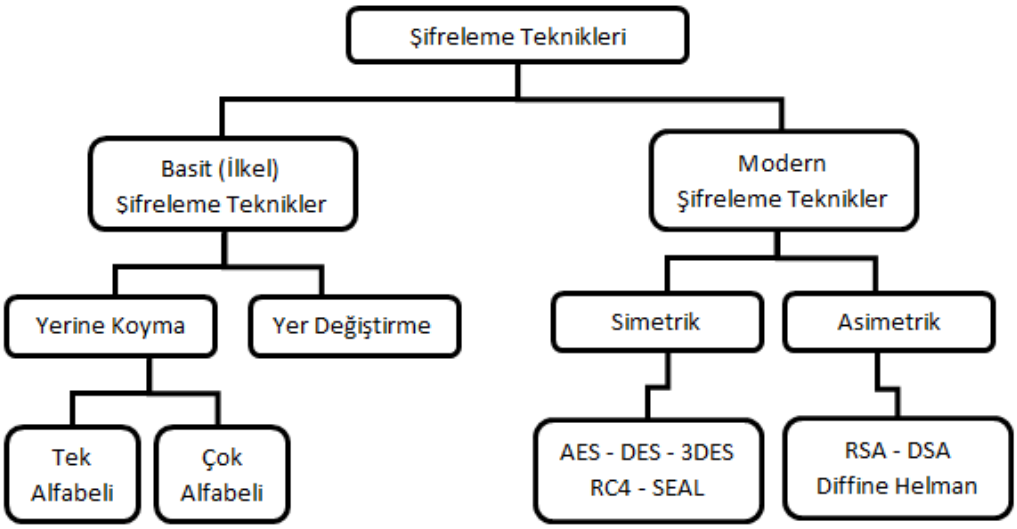
İzlenebilirlik: Yapılan güvenlik ihlallerinin sorumlusunu ortaya koymak için kullanıcıların yaptıkları işlemler takibinin yapılması gibi kavramları barındırmalıdır [8,30].

2.1. ŞİFRELEME YÖNTEMLERİ

Geçmişten günümüze veri güvenliğinin sağlanabilmesi için çeşitli şifreleme algoritmalarıyla karşılaşmaktayız. Şifreleme, bilgisayar sistemleri üzerinde veya

bilgisayar ağları arasında verilerin istenmeyen kişilerin eline geçmesini önlemek amacıyla yapılan çalışmalardır. Özellikle ticari, askeri, eğitim vb. birçok uygulamaların dijital ortamlarda aktarılmasında güvenli sistemlerin gereksinimine ihtiyaç duyulmaktadır [8].

Şifreleme yöntemleri incelendiğinde yapılan çalışmaların genelinde matematiksel işlemlere dayanan çeşitli metotlar geliştirilmiştir [4].



Şekil 2.2. Şifreleme algoritmalarının sınıflandırılması.

2.1.1. Basit (İlkel) Şifreleme Teknikleri

Basit şifreleme teknikleri tarihteki teknolojik gelişmelerin insanlara sunduğu imkânlar doğrultusunda veri güvenliği sağlamak amacıyla yapılan çalışmaları kapsamaktadır. İnsanlar tarihte bir dönem şifreli mesaj göndermek için kölelerinden istifade etmişler. Kölelerin saçlarını keserek gönderecekleri mesaj, kölelerin kafasına yazılır, saçlarının uzamasını bekledikten sonra mesajı, ulaşmasını istedikleri yerlere gönderirlermiş. Gittikleri yerde kölelerin tekrar saçlarını kestirerek iletilmesi istenen mesajın okunması sağlanırmış. Tarihte bu ve buna benzer birçok klasik şifreleme yöntemlerini görmekteyiz. İlkel şifrelemede genellikle yer değiştirme ve yerine koyma yöntemleri uygulanmıştır [4,8].

2.1.1.1. Sezar Şifreleme Yöntemi (Shift Ciphers)

İlkel şifreleme yöntemlerinden ilk şifreleme tekniği olan Sezar Şifreleme Yöntemi'nde şifrelenecek mesajdaki her bir karakter, kullanılan anahtar kadar kaydırılarak şifreli mesaj oluşturulur. Şifreli mesajın çözümlenmesi için de anahtar kadar her karakterin geri kaydırılması ile çözümlenir [8,30].

Örnek 1: Aşağıdaki oluşturulan tabloda Türk alfabesindeki tüm harfleri içine alacak şekilde yazılmıştır.
Şifrelenecek Mesaj: " b a h ç e "
Anahtar: 5

Yukarıda anahtarın 5 olması demek; şifrelenecek olan mesajdaki her karakterin kendisinden sonra gelen 5. karakterin yazılması anlamına gelmektedir. (Not: Türk alfabesinde 29 harf olduğu için Z harfinden sonra tekrar baştan başlanır.)
Şifreli Mesaj: " f e l ğ ı " olur.

A	B	C	Ç	D	E	F	G	Ğ	H	I	İ	J	K	L	M	N	O	Ö	P	R	S	Ş	T	U	Ü	V	Y	Z
E	F	G	Ğ	H	I	İ	J	K	L	M	N	O	Ö	P	R	S	Ş	T	U	Ü	V	Y	Z	A	B	C	Ç	D

Şekil 2.3. Sezar şifrelemesi.

2.1.1.2. Yerine Koyma Şifreleme Yöntemi (Substitution Ciphers)

Yerine koyarak şifrelemede istenilen bir alfabede bulunan karakterlerin her birisi için aynı alfabede bulunan farklı karakteri yerine koyarak şifreleme yöntemidir. Buna göre alfabedeki karakterlerden oluşan bir tablo oluşturulur. Tablodaki her bir karaktere karşılık gelen alternatif karakterler tabloda tutulur. Şifreli mesaj oluşturmak için önceden bilinen tablodaki karekterlerin karşılığındaki karekterler yardımıyla şifreli mesaj oluşturur. Şifreli mesajı çözümlemek için ise tabloda yapılan işlemin tersi işlem uygulanarak şifreli mesaj çözümlenir [8,30].

Örnek 1: Aşağıdaki yerine koyma tablosu rastgele oluşturulmuştur. Bu tablo, alfabedeki tüm harfleri içine alacak şekilde daha uzun düzenlenebilir.

Şifreleme Tablosu : a b c ç d e f g ğ h ı i

Şifreleme Anahtarı : r m k s z ü l n ş t p z

Yukarıdaki tabloya göre şifrelenecek mesaj : " b a h ç e " olarak belirlensin.

Anahtar : r m k s z ü l n ş t p z

Şifreli Mesaj : " m r t s ü " olacaktır.

2.1.1.3. Doğrusal Şifreleme Yöntemi (Affine Ciphers)

Doğrusal şifreleme yöntemi günümüzde tramvay işletmelerinde kullanılan ve mors alfabesi ile iletişim sağlanarak kullanılan bir yöntemdir. Matematikte doğrunun denklemi olarak öğrendiğimiz y=ax+b fonksiyonunun şifreleme işleminde kullanılması yöntemidir. Buna göre x şifrelenecek, y ise şifrelenmiş mesajı göstermektedir. a ve b ikilisi ise kullanılan anahtarı oluşturmaktadır [8].

Örnek 1: Doğrusal şifreleme için aşağıdaki örneği inceleyelim.

Şifrelenecek Mesaj: " b a h ç e "

Anahtar (3,2) olsun. Burada a=3, b=2 olarak belirlenmiştir.

X: harfin alfabedeki sırasını gösterir.

b harfi için; b alfabede 2. harf ve y=ax+b ise y= 3x2+2 = 8 b harfi yerine alfabedeki 8. harf gelecektir. b=g gelir.

a harfi için; a alfabede 1. harf ve y=ax+b ise y= 3x1+2 = 5 a harfi yerine alfabedeki 5. harf gelecektir. a=d gelir.

h harfi için; h alfabede 10. harf ve y=ax+b ise y= 3x10+2 = 32 h harfi yerine alfabedeki 3. harf gelecektir. h=c gelir. (Not: Alfabemizde 29 harf olduğu için tekrar başa dönecektir.)

ç harfi için; ç alfabede 4. harf ve y=ax+b ise y= 3x4+2 = 14 ç harfi yerine alfabedeki 14. harf gelecektir. ç=k gelir.

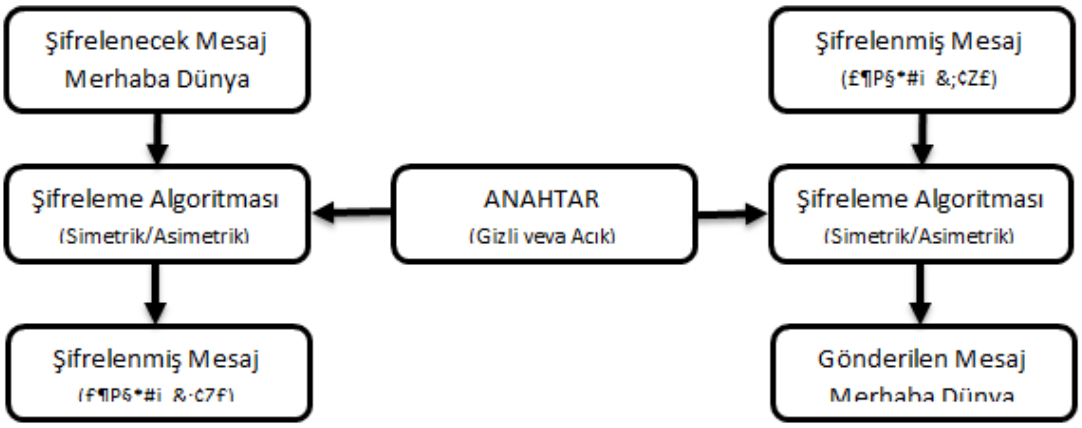
e harfi için; e alfabede 6. harf ve y=ax+b ise y= 3x6+2 = 20 e harfi yerine alfabedeki 20. harf gelecektir. e=p gelir.

Şifreli Mesaj: " g d c k p "

Yukarıdaki işlemlerin sonucunda doğrusal şifreleme yöntemi aslında yerine koyma yöntemidir. Sadece alfabedeki karakterlerin yerine hangi karakterin geleceğini belirlemek için matematiksel işlem yapılmaktadır [8,31].

2.1.2. Modern Şifreleme Teknikleri

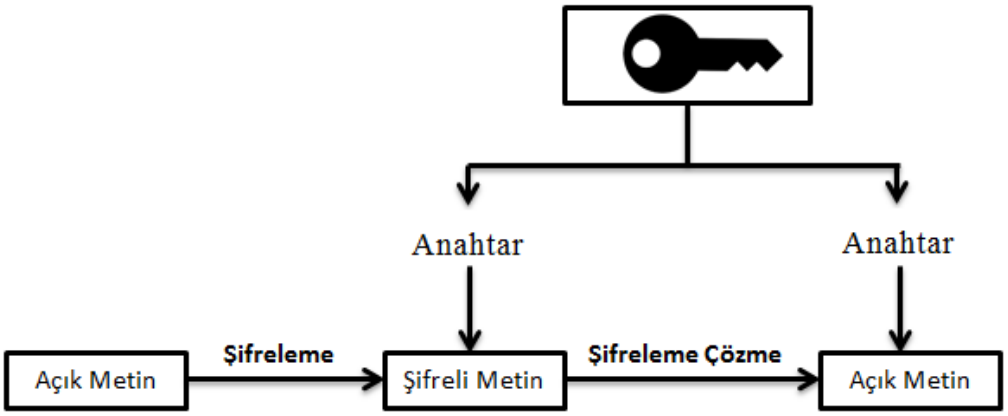
Bilgisayar kullanım alanlarının artarak hayatımızın her alanına girmiş olması, bilgi güvenliği ve şifreleme teknolojisinin gelişimine büyük katkı sunmuştur. İlk kez şifreleme biliminde matematiksel işlemler kullanılarak çeşitli algoritmalar geliştirilmiştir. Mesajlar bit, byte'lara dönüştürülerek şifreleme teknikleri kullanılmaya başlanmıştır. Modern şifreleme yöntemlerinde şifrelemede kullanılacak anahtar özelliğine göre Simetrik şifreleme algoritmaları ve Asimetrik şifreleme algoritmaları olarak ikiye ayrılmıştır [4].



Şekil 2.4. Modern şifreleme yöntemi.

2.1.2.1. Simetrik Şifreleme Algoritmaları

Simetrik şifreleme algoritmaları, verilerin şifrelenmesinde ve şifreli verilerin çözümlenmesinde tek bir gizli anahtar kullanılmaktadır. Güvenlik bu gizli anahtar üzerine kuruludur. Şifrelemede kullanılan bu anahtar aynı zamanda şifre çözümleme işleminde de kullanılacağı için gizli tutulmalıdır. Şifrelenmiş mesajla birlikte bu gizli anahtar, mesaja eklenerek ya da farklı bir yöntemle karşı tarafa iletilir. Şifre çözümleme işlemi, iletilen gizli anahtar yardımıyla gerçekleştirilir. Simetrik şifreleme algoritmalarının çalışma prensibi aşağıdaki gibidir [27].



Şekil 2.5. Simetrik şifreleme algoritmaları.

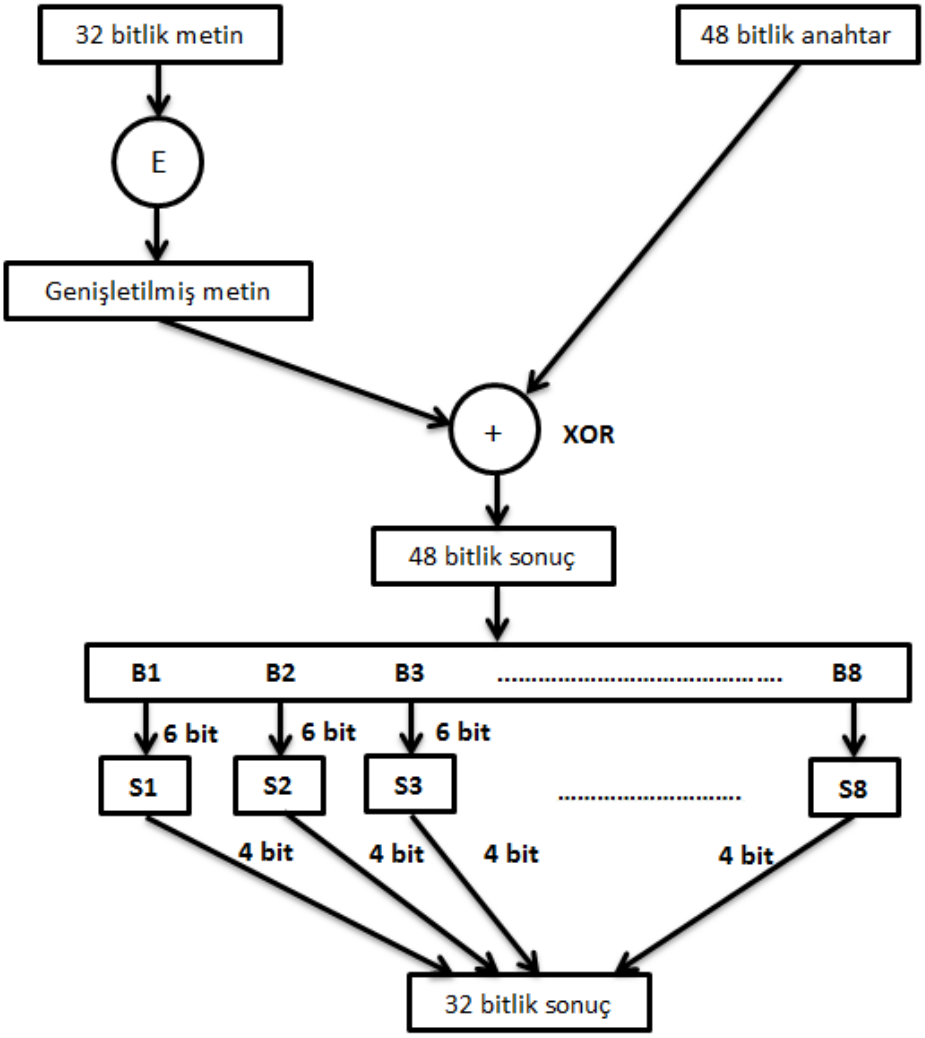
Simetrik şifreleme algoritmalarında işlem süresinin asimetrik şifreleme algoritmalarına göre daha hızlı olduğu için asimetrik şifreleme algoritmaları büyük verilerin şifrelenmesinde tercih edilmez. Simetrik şifreleme algoritmalarında kullanılan yöntem ve işlemlerden dolayı performansı düşük olan elektronik cihazlarda kullanılabilir olması çok önemlidir. Ayrıca simetrik şifreleme algoritmalarında kullanılan anahtarın uzunluğu ve anahtarın bit sayısı diğer algoritmalara göre daha az yer kaplar [21].

DES (Data Encryption Standart) Veri Şifreleme Standardı.

1974 yılında IBM ile NSA'nın (National Security Agency) birlikte çalışarak elektronik verilerin güvenliğini sağlamak amacıyla DES (Data Encryption Standart) Simetrik şifreleme algoritmasını geliştirmişler. Bu algoritmanın güvenlik eksikliklerinin çok olmasına rağmen Kriptoloji'nin gelişimine çok katkısı olmuştur. DES (Standart Şifreleme Algoritması) Blok Şifreleme Algoritması olarak da literatürdeki yerini almıştır. Algoritma, şifreleme ve deşifreleme işlemlerini 64 bitlik bloklar halinde gerçekleştirmektedir [6,8].

DES Algoritması, verileri şifrelerken 64 bitlik bloklar halinde işleme alarak 56 bitlik anahtar yardımıyla simetrik şifreleme yöntemine göre şifreleyerek gene 64 bitlik şifrelenmiş veri elde edilir. DES (Data Encryption Standart) Algoritması, şifrelenmiş verileri tekrar eski haline dönüştürmek için verilerin şifrelenmesindeki gibi aynı yöntemi kullanarak şifreli veriyi 64 bitlik bloklara bölerek gene aynı 56 bitlik anahtar

yardımı ile şifreleme yöntemine göre 64 bitlik bloklar şeklinde normal düz metinlere çevirir. DES Algoritması'nın çalışma prensibi aşağıdaki şekilde gösterilmiştir [**6**].



Şekil 2.6. DES şifreleme algoritması.

Yukarıdaki Standart Şifreleme Algoritması'nın (DES) çalışma prensibi incelendiğinde şifreleme ve şifre çözümleme işlemleri sırasında şifrelenecek veya deşifrelenecek verilerin yer değiştirmesi gibi bir takım işlemler gerçekleştirilmektedir. Bu işlemler şifreleme ve şifre çözümlemeleri için birbirlerinin tam tersi şeklinde gerçekleşmektedir [8].

Simetrik şifreleme algoritmalarının en temel özelliği, verileri gizli bir anahtar yardımıyla işleme alarak gerçekleştirmesidir. DES Simetrik Şifreleme Algoritması da

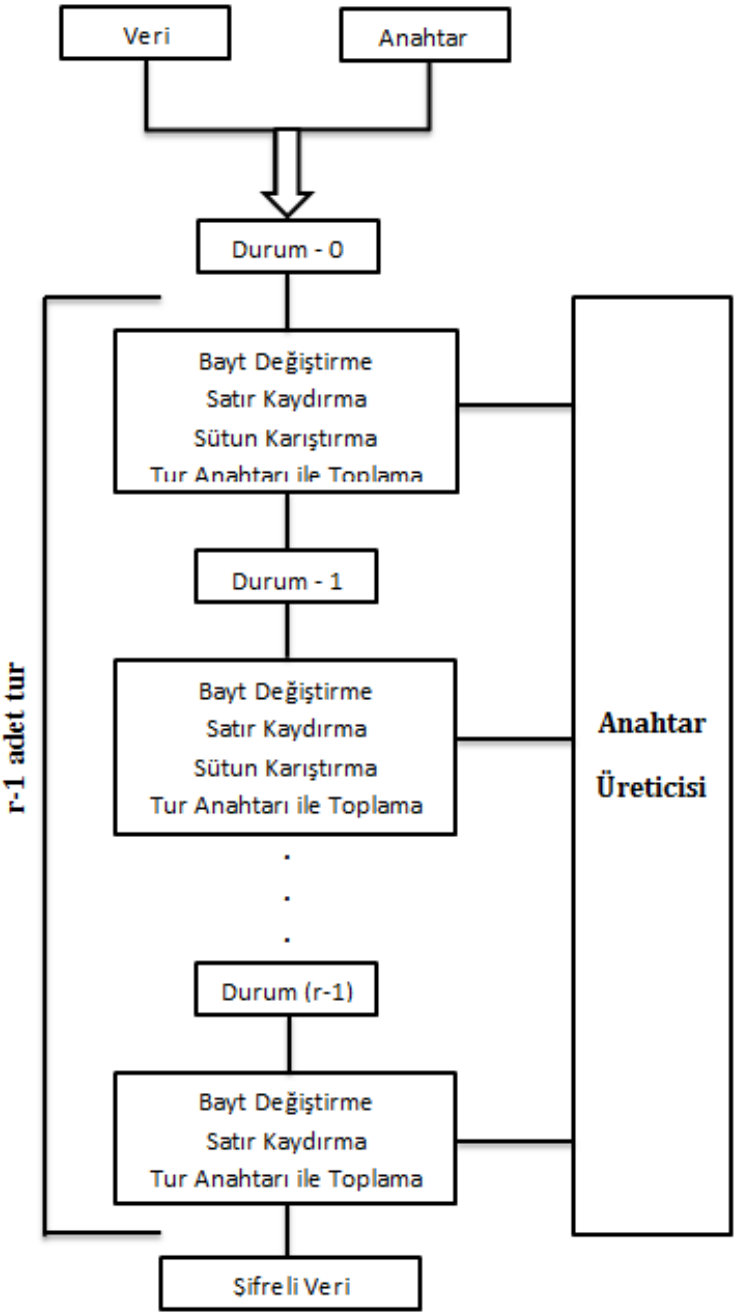
görünümde 56 bitlik gizli anahtar kullanarak işlevini gerçekleştirir. Fakat işlem sırasında 48 bitlik bir kısım kullanılarak geri kalan 8 bitlik kısmı parite kontrol etmek amacıyla kullanılır. Bu sebeple de kullanılan anahtar boyutu 56 bittir. İşleme alınacak veriler blok şeklinde iki parçaya bölünerek her defasında sadece bir blok üzerinde işlem yürütülür. Bir sonraki işlemde ise verinin diğer bloku üzerinde işlem yapılarak sarmal bir yapı şeklinde şifreleme işlemi boyunca devam eder. Her blok üzerinde 16 tur veya etap sonucunda şifreleme gerçekleştirilir [6,8].

DES Algoritması günlük hayatımızın bir parçası haline gelmiş olan, MASTERCARD, VİSA vb. kullanılan kart sistemlerinde şifreleme işleminin de temelini oluşturmaktadır. Standart Şifreleme Algoritması, kaba kuvvet saldırılarında kolayca deşifre edilebilir olması, güvenlik zafiyetinin fazla olmasına bakmayarak 1978 yılında IBM tarafından Triple DES (3DES) adında yeni bir şifreleme algoritması geliştirmiştir. Bu algoritma, DES Algoritması'nın çalışma işlevini 3 defa gerçekleştirerek şifreleme ya da şifre çözümleme işlemi gerçekleştirmektedir. Bu işlem, algoritmanın performansının üç kat yavaş çalışmasına neden olmaktadır. Ayrıca algoritmanın güvenliğinin tamamen kullanılan anahtara bağlı oluşu da birçok güvenlik zafiyeti oluşturmaktadır. 3DES Algoritması başta devlet dairelerinde, bankacılık işlemlerinde, elektronik ödeme işlemleri ve yazılım anahtarı geliştirme olmak üzere birçok yerde kullanılmaktadır [25].

AES (Advanced Encryption Standard) Şifreleme Algoritması

DES Simetrik Şifreleme Algoritması'nın teknolojik gelişmeler karşısında basit kalması, güçlü bilgisayarlar tarafından kaba kuvvet saldırılarıyla gizli anahtarın öğrenilebilir olması ciddi güvenlik açıkları vermiştir. Ulusal Standartlar Enstitüsü (NIST) tarafından 2001 yılında gerçekleştirilen bir yarışma sonucunda Belçikalı bilim adamları Joan Daemen ve Vincent Rijmen tarafından AES (Gelişmiş Şifreleme Standarttı) Şifreleme Algoritması geliştirilmiştir. DES Algoritması'nın eksik yönlerinin düzeltilip geliştirilmesiyle 128, 192 ve 256 bit anahtar kullanımına göre AES 128, AES 192 ve AES256 gibi isimlendirilmektedir [8].

AES Şifreleme Algoritması, verileri şifrelerken ve şifreli verileri çözmede aynı anahtar kullanıldığı için simetrik şifreleme algoritmasıdır. AES Algoritması'nın standartlaştırma süresi beş yıl sürmüştür. Bu süreçte birçok farklı çalışma önerilmiş ve bu algoritma çalışmalarında güvenlik ve performans açısından değerlendirilerek en başarılı şifreleme algoritması AES olarak belirlenmiştir. AES (Gelişmiş Şifreleme Standardı) Simetrik Şifreleme algoritması, bir döngüsel dönüştürme bloğu ve bir anahtar oluşturma bloğundan oluşur. Gelişmiş Şifreleme Standartı, tekrarlı algoritma olduğu için şifreleme veya şifre çözme işleminde 128 bitlik bir anahtar kullanılırsa 10 tekrar, 192 bitlik anahtar kullanımında 12 tekrar ve 256 bitlik anahtar kullanımında ise 14 tekrarlama işlemi gerçekleştirerek sonuçlanır. AES Şifreleme Algoritması'nın esnek bir yapıya sahip olması, farklı anahtar kullansa bile onun işlem hızını ve performansını olumsuz olarak etkilemez. AES Algoritması'nın çalışma prensibi aşağıdaki şekilde gösterilmiştir [8,30].



Şekil 2.7. AES şifreleme algoritması.

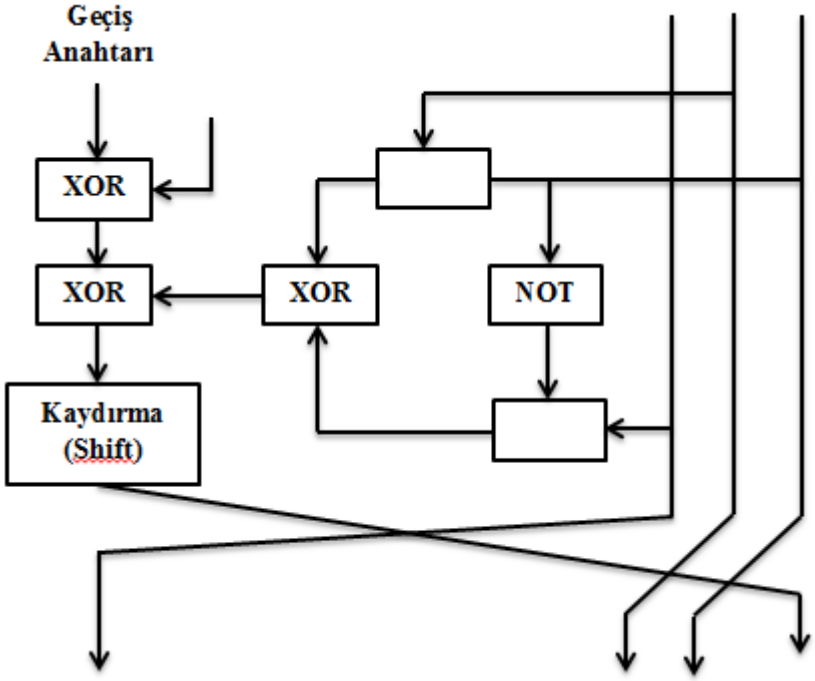
Gelişmiş Şifreleme Standartı Algoritması (AES) daha önce kullanılan veri şifreleme standardı olan DES Algoritması'nın geliştirilmiş halidir. Gelişmiş Şifreleme Standartı Algoritması, veri işleme ve karşılaştırma dediğimiz bir çalışmaya dayanmaktadır. DES Algoritması, Feistel yapısını kullanır. AES Algoritması'nın en önemli işlevlerinden biri de donanım ve performans açısından oldukça etkili olmasıdır. AES Algoritması, verileri sütun karıştırma adımlarıyla birleştirip tablolara

dönüştürerek 32 veya daha fazla veriye sahip sistemlerde bayt kaydırma ve satır kaydırma adımlarını hızlandırabilir. Bir algoritmanın şifrelenmiş biriminin çözülmesi (kırılması), onun kaba kuvvet saldırılar karşısında daha zor kırılabileceği anlamına gelir. Bu bakımdan anahtar uzunluğu 256 bit olan AES Algoritması, 2 ile 256 işlemlik ataklar gerektirir. Günümüz teknolojisiyle AES Algoritması'nın kırılması, çok uzun yıllar gerektirmektedir. Bu durum, AES Algoritması'nın ne kadar güvenli olduğunu ve başarısını göstermektedir [30].

AES Algoritması, İnternet'te gizlilik gerektiren gizli verileri veya işlemleri işlemek için kullanılır. Bunu bugün kullanılan birçok teknolojide görebiliriz. Özellikle kablosuz ürün teknolojisi altyapısında (klavye, fare, ağ teknolojisi vb.) AES 256 kullanılarak güvenlik önlemi sağlanır. Ayrıca VPN tüneli, AES-256 Algoritması'nı kullanarak İnternet'teki güvenlik ve gizlilik sorunlarımızıçözebiliriz [8,30].

RC2 Şifreleme Algoritması

1987 yılında Ron Rivest tarafından RSA güvenlik şirketi için geliştirilmiş bir simetrik şifreleme algoritmasıdır. Bir "Blok Şifreleme Algoritması" olan RC2, sonraki süreçlerde gelişerek RC4,RC5 ve RC6 şifreleme algoritmalarının başlangıç noktasını oluşturmuştur. RC2 Şifreleme Algoritması, DES Simetrik Şifreleme algoritmasının yerini alması için geliştirilmiş bir algoritmadır. 64 bitlik anahtar kullanarak şifreleme işlevini gerçekleştrir. DES Algoritması'na göre üç kat daha hızlı bir işlevselliği vardır. RC2 Algoritması'nın çalışma prensibi aşağıdaki gibidir [8,32].



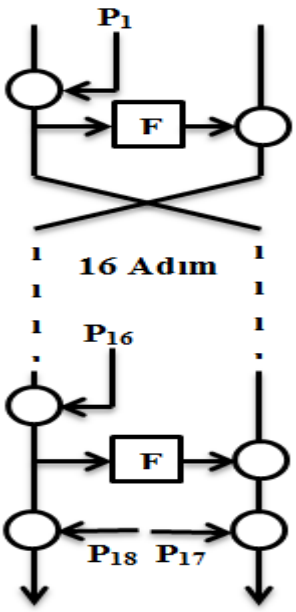
Şekil 2.8. RC2 şifreleme algoritması.

RC2 Simetrik Şifreleme Algoritması, giriş çıkış toplam 34 blok bulunan ve her blok 64 bitten oluşmaktadır. Kullanılan anahtar boyutu 48 bit olmasına rağmen normalde 8 bit ile 128 bite kadar kullanılabilir. Basitçe 64 bitlik (ikil) bir feistal yapı kullanarak 18 geçişte (etap) değişken uzunluklu bir anahtarla veri şifrelemesi yapmaktadır. 18 geçişin 16 tanesi karıştırma, 2 tanesi ise ezme işlemi için kullanılır [32].

Blowfish Şifreleme Algoritması

Blowfish Şifreleme Algoritması, 1993 yılında Bruce Schneier tarafından tasarlanmış Feistel ağ yapı kullanan bir "Blok Şifreleme Algoritması"dır. Blowfish Şifreleme Algoritması, veri şifreleme standartı veya uluslararası veri şifreleme algoritmalarının yerine kullanılabilen bir simetrik şifreleme algoritmasıdır. Blowfish Algoritması, farklı uzunlukta anahtar kullanabilir bir algoritması vardır. Şifrelemede kullanılan anahtar boyutu 32 bit ile 448 bit arasında değişebilir. Genel olarak 32 bitlik anahtar kullanımı için tasarlanmış ve ücretsiz olarak piyasaya sürülmüştür. Daha özel ve üst seviyede güvenlik söz konusu olduğunda şifrelenecek verilerin anahtar boyutu da yüksek seviyede kullanılmıştır [8].

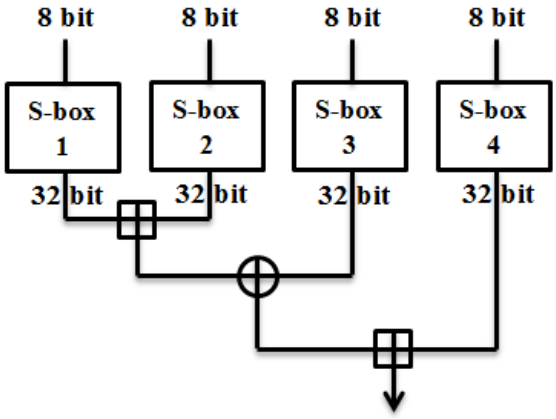
Hatta ABD, 40 bitten daha büyük anahtar boyutu ile şifreleme yapılabilen algoritmanın ihracatını yasaklamıştır. Blowfish Şifreleme Algoritması çalışma prensibi aşağıdaki gibidir [29].



Şekil 2.9. Blowfish şifreleme algoritması.

Blowfish, 16 adımlı bir Feistel ağı kullanan simetrik bir algoritmadır. Mesaj boyutu 64 bittir ve anahtar boyutu 32 ile 448 bit arasında bir değişkendir. Her adım 32 bit işlemedir. Blowfish Algoritması, iki alt öge içerir. Anahtarlar; 18 bağlantı noktalı P satırı ve dört 256 bağlantı noktalı S kutusu vardır. Burada, S kutusu 8 bit veri girişini kabul eder ve aynı zamanda 32 bit veri çıkışı üretir. Her aşamada P hattının bir girişi kullanılır. Son olarak veri bloğunun her bir yarısı, kalan iki kullanılmayan P girişinden biri tarafından XOR'lanır.

Yukarıda yapılan açıklamada her adımda yer değiştirme dizileri kullanılmıştır. Son kısımdaki işlemlerden sonra veri bloğunun iki yarısı alınarak arta kalan dizilerde kullanılmaktadır. Kısaca toplamda 18 dizi vardır. 16 adımın her biri, son adımda bir değiştirme dizisi ve iki değiştirme dizisi kullanır [8].



Şekil 2.10. Blowfish algoritması s-box şeması.

Yukarıdaki şekilde S-box şemasında kullanılan yöntem verilmiştir. Buna göre mesajın yarısı yani 32 bitlik uzunluktaki veriler 4 adet 8 bitlik parçalara bölünerek S-box kutularına yerleştirilir. S-box kutusundan çıkan her sonuç yukarıdaki gibi işlenmektedir. Blowfish'in bazı uygulamalarında 4 kilobayt RAM biraz daha büyük hafıza ile çalışabilir olması, bu şifreleme algoritmanın eski masaüstü bilgisayarlarda ve dizüstü bilgisayarlarda çalışılabilir olduğunu göstermektedir [15].

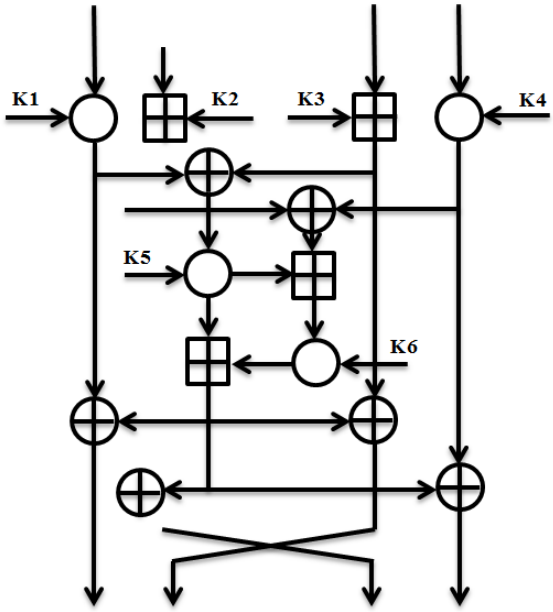
Twofish Şifreleme Algoritması

Twofish Şifreleme Algoritması, 1998 yılında John Kelsey, David Wagner, Bruce Schneier, Chris Hall, Doug Whiting, ve Niels Ferguson kişilerince geliştirilen AES Şifreleme Algoritması kadar hızlı bir algoritmadır. DES Şifreleme Algoritması'na benzeri Feistel yapı kullanılmıştır. DES'in farklı yönü ise anahtar kullanarak S-box (değiştirme kutuları)'lara sahip oluşudur. Twofish Şifreleme Algoritması 128 bitlik veriyi 32 bitlik parçalara bölerek işlemlerini genellikle 32 bitlik değerler üzerinde gerçekleştirir. Twofish Algoritması 128, 192 ve 256 bitlik değişken anahtar uzunluklarına sahiptir [8].

Twofish Şifreleme Algoritması'na eklenen iki adet bir bitlik rotasyon, AES Şifreleme Slgoritması'ndan farklı kılmaktadır. Bu eklenen iki bitlik rotasyon, uygulamanın maliyetini ve yazılımın yavaşlamasına sebep olmuştur.

IDEA Şifreleme Algoritması

Uluslararası Veri Şifreleme Algoritması (IDEA), 1991 yılında Xuejia Lai ve James Massey tarafından geliştirilen bir blok şifreleme algoritmasıdır. IDEA Şifreleme Slgoritması Zürih, İsviçre'de geliştirilmiş ve patenti Ascom Systec Ltd.'ye aittir. IDEA Şifreleme Algoritması'nın PGP programıyla birlikte kullanılması onu, kullanışlı ve güçlü bir algoritma haline getirdi. IDEA Algoritması, PGP programının temelini oluşturan iki önemli algoritmadan biridir. Algoritmanın çalışma prensibi aşağıda gösterilmiştir [8].



Şekil 2.11. IDEA şifreleme alogoritması.

IDEA (International Data Encryption Algoritma) Şifreleme Algoritması'nda şifrelenecek veri 64 bitlik metinler halinde 16 bitlik 4 eşit parçaya bölünür. 128 bitlik bir anahtarla 52 adet ve her biri 16 bit olan alt anahtarlar yardımıyla 8 adımda matematiksel işlemler yapılarak şifreleme işlemi gerçekleştirilir.

IDEA Algoritması'nda kullanılan 52 tane alt anahtar oluşturulmasında kullanılan 8 adet anahtarda kaydırma yapılarak oluşturulması, şifreleme algoritmasının dezavantajı olarak görülmektedir. Fakat yapılan araştırmalarda matematiksel olarak herhangi bir dezavantaj durum görülmemiştir [8].

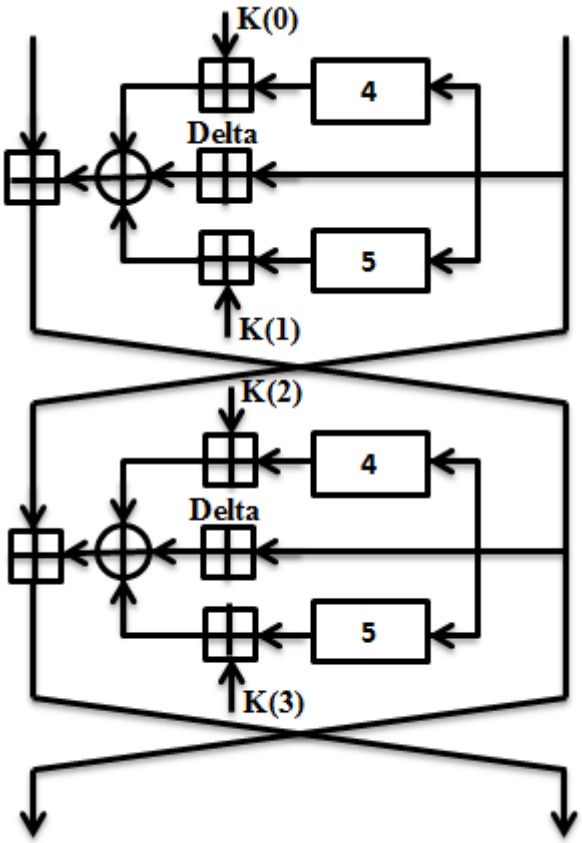
IDEA Simetrik Şifreleme Algoritması'nın avantaj ve dezavantajlarını şöyle sıralayabiliriz;

Avantajları:
IDEA Şifreleme Algoritma, hızlıdır.
IDEA Şifreleme Algoritması'nın donanımla gerçekleştirilmesi kolaydır.
Bu algoritma "gizlilik – güvenlik" ilkesini sağlamaktadır.

Dezavantajları:
IDEA Şifreleme Algoritması, ölçeklenebilir değildir.
IDEA Şifreleme Algoritması'nda anahtarın güvenli dağıtılması çok zordur.
"Bütünlük" ve "Kimlik Doğrulama", güvenlik ilkesini sağlayamamaktadır.

TEA Şifreleme Algoritması

(Tiny Encryption Algorithm) TEA Şifreleme Algoritması, 1994 yılında Needham Wheeler ve David Roger Cambridge Bilgisayar Laboratuvarı'nda geliştirilmiştir. Bu küçük şifreleme algoritması blok şifreleme yöntemini kullanmaktadır. Bu algoritmanın çok basit olmasının nedeni birçok şifreleme algoritmalarına göre kısa kod satırından oluşmasıdır. Bu algoritma karışık cebirsel işlemler kullanarak veri şifrelemesi gerçekleştirmektedir. TEA Şifreleme Algoritması, hafızada az yer kaplamak ve veri şifreleme işlemini çok hızlı yapabilmek için geliştirilmiş bir algoritmadır. Bu sebeple de genellikle kod boyutunun sınırlı oluşu, gömülü sistemlerde görülen kullanışlı bir şifreleme algoritmasıdır. TEA Algoritması'nın çalışma prensibi aşağıdaki gibidir [8,30].



Şekil 2.12. TEA şifreleme algoritması.

TEA Şifreleme Algoritması 64 bitlik veri blokları kullanır. 64 bitlik veri bloklarını 128 bitlik anahtar yardımıyla şifreleme gerçekleştirir. 128 bitlik anahtar 32 bitlik dört bloka bölünür.

Blok şifreleme metodu, Shannon'un ortaya koyduğu yayılma ve karıştırma tekniğine dayanır. TEA Şifreleme Algoritması, Shannon'un sunduğu blok şifreleme işlemlerinde kullanmak için gerekli olan karıştırma ve yayılma özelliği sağlayan önemli bir şifreleme algoritmasıdır. Bu algoritma, verileri karıştırma yaparak şifreli verilerle şifrelenecek olan veriler arasındaki ilişkinin gizlenmesini hedeflerken verilerin istenilen yerlere iletilmesinde ise umuma açık verideki izlerin şifreli veride gözükmemesini sağlamak için kullanır. Bu işlem de tam bir yayılma sağlar. Şifrelenecek metinde yapılan tek bir bit'in değişikliği şifreli mesajda 32 bitlik değişikliğe sebep olur [30].

TEA Algoritması'nda şifreli verilerin şifrelenmesi ve şifre çözümleme süreci temel olarak aynıdır. Şifreli veriyi normal okunabilir veriye dönüştürmek için şifreli mesaj, algoritmanın girdisi olarak kullanılır ve alt anahtarlar ters sırada kullanılarak şifre çözümleme işlemi gerçekleştirilir [8,30].

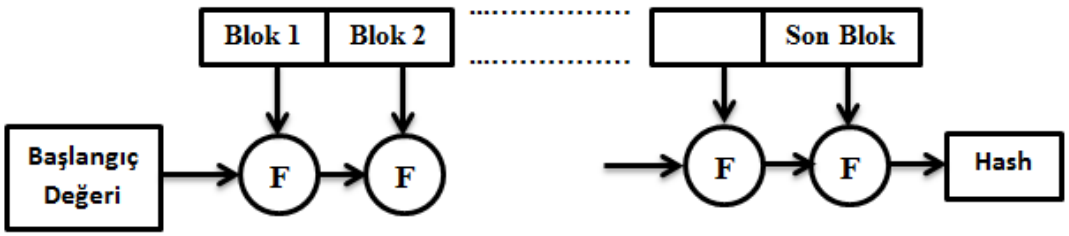
Hash Şifreleme Algoritması

Hash (Özetleme) Algoritması, temel olarak veri bütünlüğünün sağlanmasını amaçlayan bir algoritmadır. Literatürde tek yönlü algoritma olarak bilinir. Tek yönlü algoritmalarda, algoritmanın ürettiği sonuçtan tekrar metnin ilk haline dönüş mümkün değildir. Şifreleme algoritmalarından farklı oluşu bundandır. Şifreleme algoritmaları çift yönlüdür yani şifrelenen verileri tekrar asıl metne dönüştürebilirsiniz. Ama Hash Algoritması'nda özeti alınmış veri tekrar geri dönüştürülemeyecek şekilde sonuçlandırılır [21].

Özetleme fonksiyonları genellikle iki amaçla kullanılır:

1- Veri bütünlüğünün kontrol edilmesiyle verinin değişmediğinden emin olmak için,

2- Büyük boyutlu verilerin boyutunu daha küçük boyuta indirerek hedefe gönderilecek verinin boyutu düşürmek için.



Şekil 2.13. Hash şifreleme algoritması.

Hash (Özetleme) fonksiyonunda, dönüştürülecek veri, önce blok zincirine dönüştürülür. Bir başlangıç değeri mesaj bloğu zincirinin her elemanı ile bir işleme tabi tutulur. Bu işlem bütün bloklar bitene kadar tekrarlanır. Elde edilen son değer, mesajın özetidir.

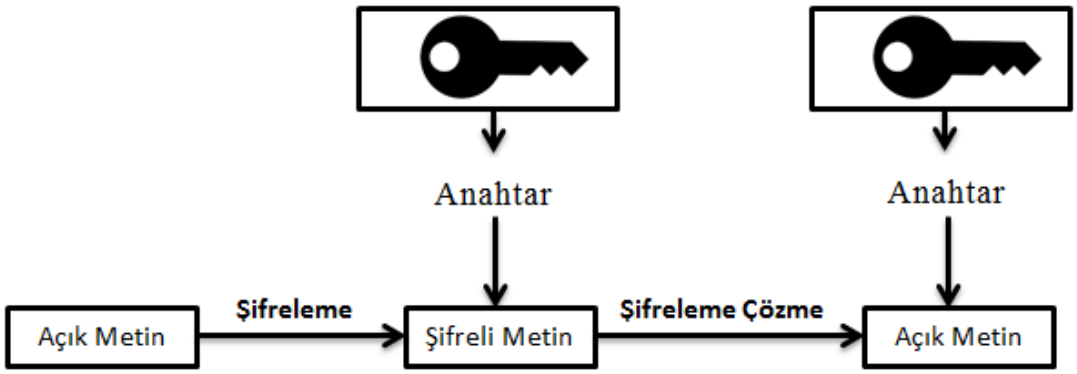
Günümüzde en yaygın kullanılan özetleme algoritmalarının başında MD5 ve SHA1 olmakla birlikte SHA256, SHA384 ve SHA512 algoritmalarıdır. SHA384 Algoritması, Facebook tarafından kullanılmaktadır. Hash işlevleri diğer standart şifreleme yöntemleriyle birleştirilerek verilerin kaynağını doğrulayabilir. Hash işlevleri, karma algoritmalarla birleştirildiğinde özel mesaj özetleri oluşturarak verilerin kaynağını tanımlar. Bu özel mesaj özetlerine mesaj doğrulama kodları denir. Özetleme Algoritmaları'nın bazı önemli hususları vardır:

Hash (Özetleme) Algoritmaları, simetrik veya asimetrik şifreleme grubuna girmez. Çünkü Hash Algoritmaları, anahtar kullanmazlar. Özetleme fonksiyonları tek yönlü çalışırlar. Bu sebeple özetlenen veri, tekrar eski haline dönüştürülemez. Aynı veri özetleme algoritmasıyla işlem yapılsa bile sürekli aynı sonuç ortaya çıkar. Bu, sonuçta algoritmanın bütünlük kontrolünü gösterir. Özetleme Algoritması'yla metin üzerinde yapılacak küçük bir değişiklik, alınan çıktıda büyük değişikliğe sebep olur [8].

Hash (özetleme) Algoritmaları'nın kullanım alanları ise mesaj doğrulama kodunda, şifrelerin saklanmasında, veri gizliliğinin sağlanmasında, veri boyutunun küçültülmesinde ve sayısal imza alanı gibi birçok alanda kullanılmaktadır [21].

2.1.2.2. Asimetrik Şifreleme Algoritmaları

Asimetrik şifreleme algoritmaları, 1976 yılında Stanford Üniversitesinden Diffie ve Hellman'ın çalışmalarıyla geliştirilmiştir. Bu şifreleme algoritmalarında iki farklı anahtar kullanılmaktadır. Bu anahtarlar, verilerin şifrelenmesinde ve şifre çözme işleminde kullanılır. Bu açık şifreleme anahtarları birbirinden bağımsız olarak üretilirler. Açık Anahtarlı Şifreleme Algoritması'nda anahtar, Simetrik şifrelemedeki anahtarlar gibi gizli değildir. Ayrıca Simetrik şifreleme algoritmalarında bir tek gizli anahtar kullanılırken Asimetrik şifleme algoritmalarında farklı olarak iki tür anahtar kullanılır. Biri herkese açık bir anahtar diğeri ise bir gizli anahtardır. Umuma açık olan anahtar, mesajı şifrelemek için kullanılırken gizli anahtar ise şifrelenmiş verileri çözmek için kullanılır. Asimetrik şifreleme algoritmalarının çalışma prensibi aşağıda gösterilmiştir [29].



Şekil 2.14. Asimetrik şifreleme algoritması.

Asimetrik şifreleme algoritmalarının güvenliği, şifre çözümünde kullanılacak anahtarın ilgili alıcı tarafından bilinmesidir. Şifrelemede ve şifre çözmede kullanılan bu iki anahtar birbirinden bağımsız olsa da şifreleme anahtarı bilindiğinde şifre açma anahtarını elde etmek teorik olarak ihtimal dahilinde olsa bile bu, pratikte mümkün gözükmemektedir. Bu sebeple de Asimetrik şifreleme algoritmaları güvenlik yönüyle simetrik şifreleme algoritmalarına göre daha başarılıdır. Asimetrik şifreleme algoritmalarının genel olarak iki kullanım alanı vardır: Şifreleme ve dijital imzadır [4,8].

DH (Diffie Helman) Şifreleme Algoritması

Asimetrik Şifreleme Algoritması'nın ilk temelini oluşturan Diffie Helman, bir anahtar değişim algoritmasıdır. Tüm Asimetrik şifreleme algoritmalarında olduğu gibi burada da açık ve gizli anahtarlar kullanılır. Bu şifreleme yöntemindeki amaç herkese açık umumi bir şifre ile iki kişinin bildiği anahtarları karşılıklı birbirlerine ulaştırmaktır. Sistemin çalışma prensibi qab=qba matematiksel işlemine dayanır [8].

Diffie-Helman Şifreleme Yöntemi aşağıdaki gibi çalışır; Anahtar değişimi yapacak kişileri ortak bir p ve q sayılarını kararlaştırıyorlar. (p=11, q=7)
Birinci taraf gizli anahtar olarak a=4 sayısını seçsin ve (qa mod p) işleminden 74mod11= 3 burada çıkan 3 diğerini ikinci tarafa göndersin.
İkinci taraf ise gizli anahtar olarak b= 6 sayısını seçsin ve (qb mod p) işleminden 76mod11= 4 burada çıkan 4 diğerini birinci tarafa göndersin.

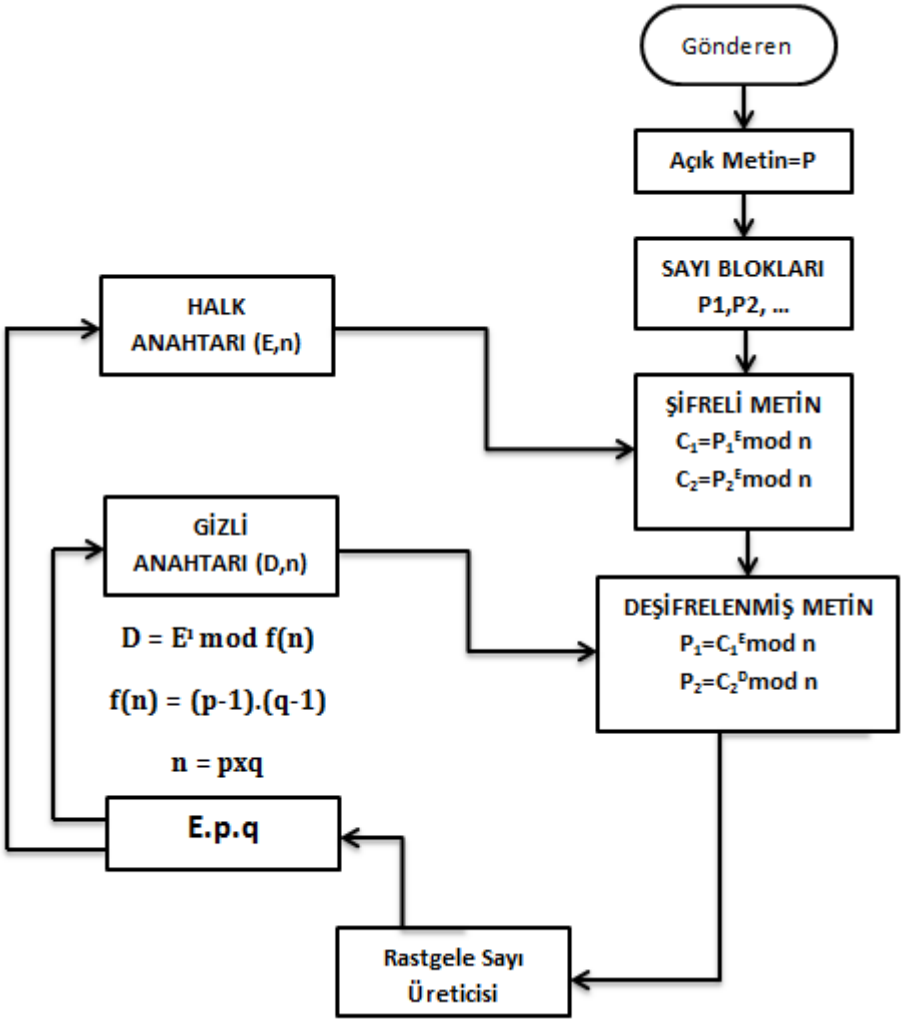
Birinci taraf gelen 4 sayısı ile kendi seçtiği sayının mod11 göre alarak gizli anahtar oluşturan sayı bulmuş olur. 44 mod 11 = 3
İkinci taraf ise birinci taraftan gelen 3 sayısı ile kendi belirlediği sayının mod11 göre alarak oda aynı gizli 36 mod 11 = 3

Örnek 1: Yukarıda açıkladığımız Diffie Helman Algoritması'nı sayısal örnekle açıklayalım.
Ahmet ile Ayşe 13 ve 5 saylarını belirliyorlar. Kısaca p=13 ve q=5 değerleri veriliyor. Ahmet'in gizli sayısı 7 ve Ayşe'nin gizli sayısı 9 olsun.
57 mod13 = 8 çıkan 8 sonucu Ayşe'ye gönderir. Ayşe de gelen sayının üstüne kendi gizli sayısını ekleyerek mod13'e göre değerini alır.
89 mod13 = 8 çıkan sonuçlar aynı olduğu gözükmektedir. Bu sayı Ahmet ve Ayşe'nin gizli anahtarlarıdır. Bu anahtar ile simetrik şifreleme algoritmalarında kullanılabilirler.

Diffie-Hellman, çeşitli İnternet servislerinin güvenliğinin sağlanması için kullanılır. Ayrıca açık İnternet ağları üzerinden yapılın alışveriş için gizli iletişim kurmamızı sağlamaktadır.

RSA (Rivest Shamir Adleman) Şifreleme Algoritması

Rivest, Adleman ve Shamir(RSA) 1977 tarafından geliştirilen bir şifreleme algoritmasıdır. RSA, verileri istenen bir konuma güvenli bir şekilde iletmek için yaygın olarak kullanılan açık anahtarlı bir asimetrik şifreleme algoritmasıdır. RSA Şifreleme Algoritması'nda şifreleme için kullanılan anahtarlar umumi ve geneldir. Ayrıca şifre çözmek için kullanılacak olan gizli anahtardan farklıdır. RSA Algoritması'nın güvenliği tam sayıların ayrıştırılması için algoritmanın zorluğuna bağlıdır. Bu Kriptografik algoritmalarda, açık anahtar olarak kullanılır ve iki büyük asal sayı ile çarpılarak başka bir değer seçilir. Genel anahtar, mesajı şifrelemek için kullanılabilir ancak genel anahtar yeterince büyükse şifrelenmiş mesajın şifresi yalnızca asal numara bilindiğinde çözülebilir. RSA Şifreleme Algoritması'nın çalışma prensibi aşağıda gösterilmiştir [7,8].



Şekil 2.15. RSA şifreleme algoritması.

RSA Algoritması, Netscape tarafından gerçekleştirilen SSL (Secure Socket Layer) anahtar değişimi, dosya transferi sırasında, dijital imza ve İnternet ortamında birçok işlemde güvenli bir şekilde kullanılır.

RSA Algoritması'nda şifreleme işlemi aşağıdaki yöntemler takip edilerek gerçekleşir:

- ✓ Yeterince büyük rastgele iki farklı asal sayı seçilir. (p ve q asal sayıları)
- ✓ n=pxq ve t=(p-1)x(q-1) değerleri hesaplanır. Burada n, seçilen iki asal sayının çarpımı olup, şifrelemede taban (modulus) olarak adlandırılır. t ise Totient fonksiyonu temsil eder.

- ✓ 1 < e < t ve gcd(e,t)=1 olacak şekilde rastgele bir e sayısı seçelir. Burada gcd en büyük ortak bölen sayıyı ifade eder.
- ✓ Öklid Algoritması kullanılarak 1< d < t ve exd = mod(t) koşulunu sağlayan d sayısı hesaplanır.
- ✓ Herkese açık anahtar (n,e) sayılarıdır. Gizli anahtar ise (d) olur.

RSA Algoritması'nda şifrelenen veri, aşağıdaki yöntemler takip edilerek çözümlenir:

- ✓ Kendisine gönderilen açık anahtarı alır. (n,e)
- ✓ m metnini [0, (n-1)] aralığında yazar.
- ✓ c şifreli mesajı oluşturma: c = me (mod n)
- ✓ d gizli anahtarını kullanarak şifreli mesaj açmak için: m = cd (mod n) işlemini uygulayarak m açık mesaja ulaşırız.

Örnek 1: Yukarıda açıkladığımız RSA Algoritması'nı sayısal örnekle açıklayalım.
Elma kelimesini RSA Algoritması ile önce şifreleyelim sonra da geri dönüştürelim.
Elma kelimesinin Asc11 kodu (E=69, l=108, m= 109, a=97,)
Rastgele iki asal sayı seçiyoruz. p= 3 ve q= 7 ise n=21 ve T(n)=12 olur, 1< e < T(n) olduğu için e=5 seçelim.
E- Mesaj şifrelemek için c = me mod(n) ise c= 695 mod21 ve c=15
l- Mesaj şifrelemek için c = me mod(n) ise c=1085 mod21 ve c=12
m-Mesaj şifrelemek için c = me mod(n) ise c=1095 mod21 ve c=16
a-Mesaj şifrelemek için c = me mod(n) ise c=1095 mod21 ve c=13
Şifreli Mesaj: 15 12 16 13 oldu.,
Şifre Çözümü: M = cd mod(n) ise;
M = cd mod(n) ise m=15d mod21 m= 69 = E
M = cd mod(n) ise m=12d mod21 m=108 = l
M = cd mod(n) ise m=16d mod21 m=109 = m
M = cd mod(n) ise m=13d mod21 m= 97 = a

DSA (Digital Signature Algorithm) Şifreleme Algoritması

(Digital Signature Algorithm) DSA şifreleme Algoritması, RSA gibi açık anahtarlı şifreleme algoritmasıdır. DSA Algoritması, Ulusal Standartlar ve Teknoloji Enstitüsü (NIST) tarafından geliştirilmiş ve günümüzde birçok yerde kullanılan sayısal imza teknolojisidir. DSA Algoritması ayrık logaritmik işlemlerine dayanır. RSA Şifreleme Algoritması'ndan farkı ise şifreleme yapmaması sadece dijital imza amaçlı kullanılmasıdır. DSA ilk olarak ABD tarafından kullanılmaya başlanmış olsa da günümüzde de çok yaygın olarak kullanılmaktadır [8,30].

Dijital imza, kişiye ait yetkinin bir göstergesi ya da bir doküman içeriğinin kabul edildiğinin bir göstergesi olarak kullanılmaktadır. Bir dokümanı dijital olarak imzalamak için açık anahtarlı Kriptoloji algoritması kullanılabilir. Bazı şifreleme algoritmalarında şifreleme ve şifre çözme işleminin durumuna bağlı olarak şifreleme işlemi için açık anahtar veya gizli anahtar kullanılabilir. Bu algoritmalar dijital olarak imzalanan dokümanlarda kullanılmaktadır. Böyle bir sistemde kişinin kendi gizli anahtarını kullanarak mesaj içeriğini imzalamasıyla güvenli bir dijital imza elde edilir. DSA gibi sistemlerde ise dijital imzalar için şifreleme algoritmasından farklı başka bir algoritma kullanılır. Dijital imzanın çalışma protokolü basit olarak su şekilde çalışır [30].

- ✓ Gönderici, mesajı imzalamak suretiyle kendi gizli anahtarı ile mesaj içeriğini şifreler. Gönderici imzalı mesajı alıcıya gönderir.
- ✓ Alıcı, dijital imzayı doğrulamak için gönderilen mesajı göndericinin açık anahtarı ile çözer.
- ✓ Eğer alıcı, üçüncü adımı gerçekleştiremiyorsa, "Dijital imza geçerli değildir." denilmektedir.
- ✓ Bu protokol ayrıca ideal bir imzada olması gereken aşağıdaki özellikleri de sağlamaktadır.
- ✓ İmza gerçektir, eğer alıcı, göndericinin açık anahtarı ile mesajı doğrulayabiliyorsa mesajın gönderici tarafından imzalandığını bilir.
- ✓ Gönderici gizli anahtarını sadece kendisi bildiği için imza taklit edilemez.

- ✓ İmza tekrar kullanılamaz, imza değeri mesaj içeriğinin bir fonksiyonu olacağı için bu imza, diğer mesajlar için kullanılamaz.
- ✓ İmzalı doküman değiştirilemez, eğer mesaj içeriğinde herhangi bir değişiklik olursa imza gönderenin açık anahtarı ile doğrulanamaz.
- ✓ İmza reddedilemez, alıcı; gönderenin yardımına ihtiyaç duymadan imzayı doğrulayabilir.

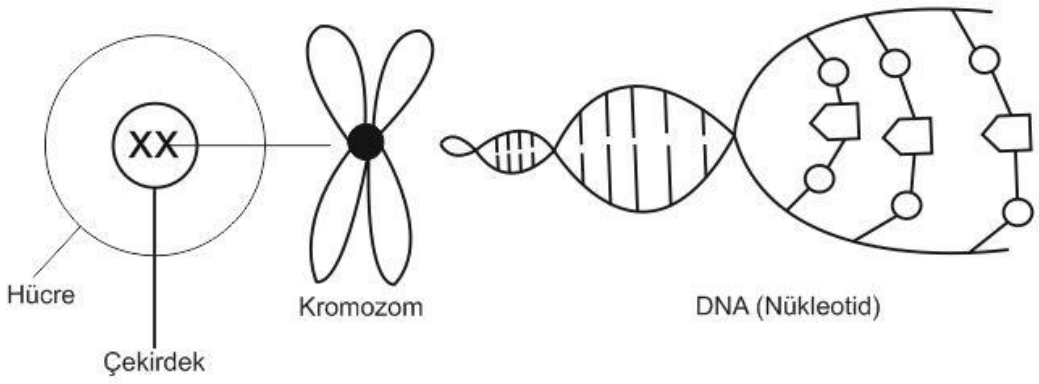
Genelde, dijital imzalama ve doğrulama işlemleri, kullanılan algoritmanın detaylarından bağımsız bir şekilde çalışırlar. Dijital imzalama gerçekleştirildikten (dokümanın gizli anahtar ile şifrelenmiş mesaj özeti) sonra dokümana eklenen bit dizisine dijital imza denir. Alıcıyı mesajın göndericisi ve mesaj içeriğini öğrenmesini sağlayan protokol yetkilendirme olarak adlandırılır. Bununla birlikte, günlük hayatta kullandığımız elle attığımız imzalar için bu ifadelerin hiçbiri tamamıyla doğru değildir. İmzalar taklit edilebilir, dokümanlardan çıkartılabilir ve bu şekilde doküman içeriği imzalandıktan sonra bile değiştirilebilir [30].

BÖLÜM 3

DNA'NIN YAPISI VE MOLEKÜLLERİN ÖZELLİKLERİ

3.1 DNA NEDİR?

Deoksiribonükleik asit (DNA) genetik kodlu bir nükleik asittir.Tek hücreli veya çok hücreli tüm organizmaların canlılığı ve biyolojik gelişimi için gerekli olan genetik bir özelliktir. DNA'nın en önemli rolü, organizmaların genetik özelliklerinin nesilden nesile aktarılmasını sağlamaktır. DNA, hücrenin diğer bileşenlerini (proteinler ve Ribonükleik asit gibi) oluşturmak için gerekli bilgileri içerir. Standart bir kalıp veya şablon yapısına benzer. Biyolojik genetik bilgiyi taşıyan deoksiribonükleik aside gen denir. Deoksiribonükleik asit dizileri Kromozomların şeklini tanımlama ve genetik bilginin hangi hücrelerde ve hangi koşullarda nasıl kullanılacağını düzenleme gibi işlevsel özelliklere sahiptir [28].

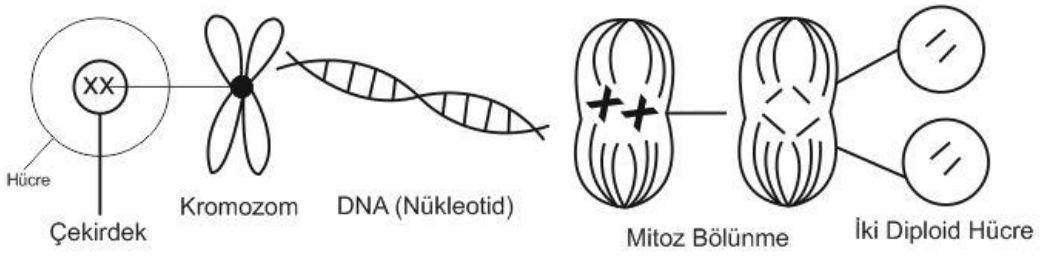


Şekil 3.16. Hücrenin yapısı.

Kimyasal olarak konuşursak Deoksiribonükleik asit, nükleotid adı verilen iki uzun Polimer'den oluşur. Halat şeklindeki bu polimerlerin yapısı ester bağlarıyla birbirine bağlanan şeker ve Fosfat gruplarından oluşur. Polimer, zıt yönlerde iki uzun çizgi olarak görünür.

Baz olarak adlandırılan dört tür molekülden biri, her şeker grubuna bağlanır. DNA yapısındaki bazların oluşturduğu dizi organizmaların genetik bilgilerini kodlar. Bu dizilerin Protein birleştirme işlemi sırasında bilgiler, proteinin Amino asit dizisini tanımlayan genetik kod aracılığıyla okunur. Bu süreçte DNA'daki bilgiler DNA ile aynı yapıya sahip bir nükleik asit olan Ribonükleik asite kopyalanır. Bu olaya Transkripsiyon denir [28].

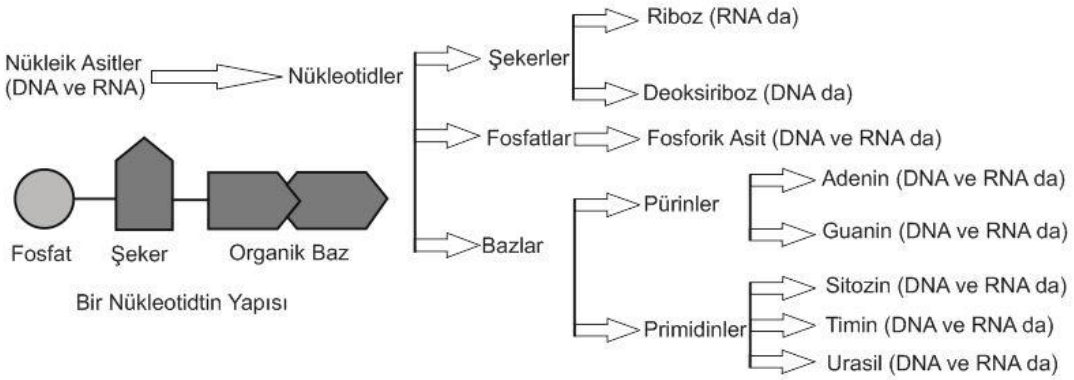
Deoksiribonükleik asit, Kromozom adı verilen yapılarda bulunur. Hücre bölünmesi gerçekleşmeden önce Kromozomlar eşleştirilir. Bu durumda DNA replikasyonu meydana gelir. Ökaryotlar yani hayvanlar, bitkiler, mantarlar ve protistler çekirdekte DNA içerirken Prokaryotlarda, bakterilerde ve arkelerde DNA, hücrenin sitoplazmasında rol oynar. Kromozom yapısında bulunan kromatin proteinleri DNA'yı Histonlar gibi sıkıştırır ve düzenler. Bu kalabalık yapılar DNA ile farklı proteinler arasındaki etkileşimi DNA'nın hangi kısımlarının okunacağını belirler [28,19].



Şekil 3.17. Genetik bilginin transferi.

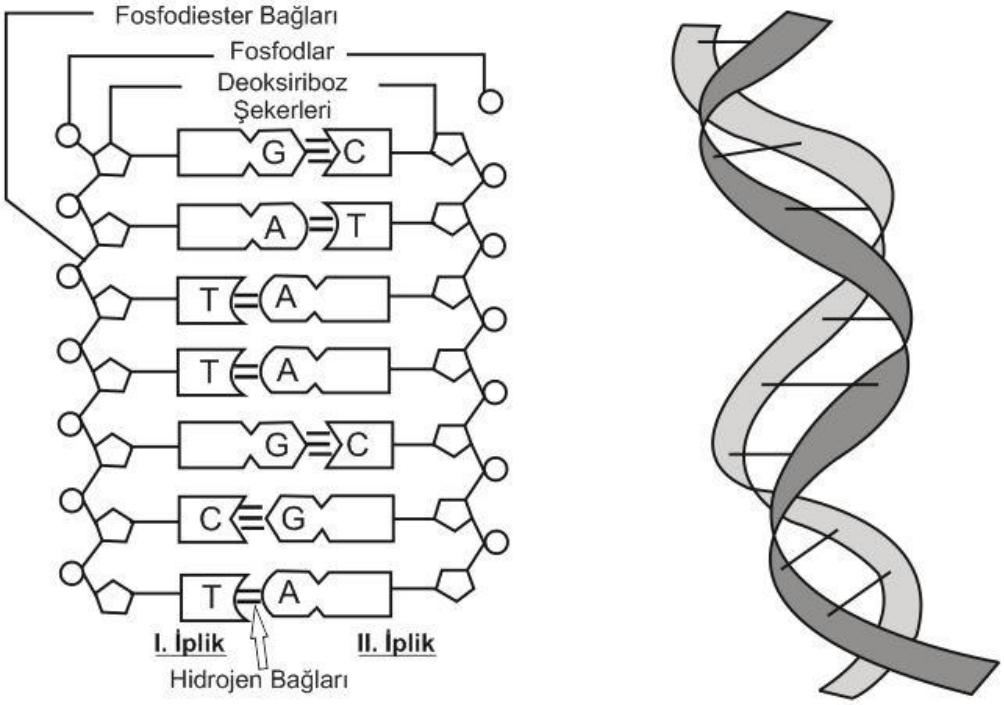
Nükleotidler birimlerden oluşan polimerlerdir. DNA sarmalının genişliği 22 ila 26 angstromdur ve aralık 2,2 ile 2,6 nm'dir. Bir nükleotid birimi 3,3 angstrom veya 0,33 nm uzunluğundadır. Her birim küçük olmasına rağmen Deoksiribonükleik asit polimerleri milyonlarca Nükleotit'ten oluşan devasa moleküllerdir. En iyi örnek olarak en büyük insan Kromozomunu verebiliriz. İnsan Kromozomu yaklaşık 220 milyon baz çifti uzunluğundadır [28].

Bir organizma, soyunu sürdürdüğünde DNA'nın yarısı kadınlardan ve yarısı erkek organizmalardan gelir. Biyoloji'de bulunan Deoksiribonükleik asit tek bir molekül değildir. Ancak birbirine sıkıca sarılmış bir çift molekülden oluşur. Bu moleküller birbirine iki uzun iplik gibi dolanır ve sarmaşık şeklinde sarılarak çift sarmal oluşturur. Nükleotid birimi şeker, Fosfat ve Baz'dan oluşur. Şeker ve Fosforik asit, Deoksiribonükleik asit moleküllerinin omurgasını oluşturur. Bu baz çift sarmalda başka bir Deoksiribonükleik asit ipliği ile etkileşime girer. Genellikle şekerlere bağlanan bazlara Nükleosit denir ve şekerlere bağlı bir veya daha fazla Fosfat'a bağlanan bazlara Nükleotid denir. Birbirine bağlı çoklu Nükleotidlere Polinükleotidler denir [28,19].



Şekil 3.18. Nükleik asitlerin yapısı.

Deoksirüboükleik asit ipliğinin yapısı şeker ve Fosfat'tan oluşur. Deoksiribonükleik asitte bulunan şeker 2 deoksiriboz veya pentozdur. Bu beş karbonlu bir şekerdir. İki bitişik şekerden biri, bir fosfodiester bağı oluşturarak 3 numaralı Karbon atomları arasındaki Fosfat ester grubu ve 5 numaralı Karbon atomları arasındaki diğer Fosfat ester grubu ile şekere bağlanır. Fosfodiester bağı asimetrik olduğu için DNA zincirinin bir yönü vardır. Bir çift sarmalda bir sarmaldaki nükleotidlerin bağlantı yönü diğer sarmaldaki nükleotidlerin yönünün tersidir. DNA zincirinin kaliperleri antiparalel olarak adlandırılır. Deoksiribonükleik asit zincirinin asimetrik uçlarına 5 bert ve 3 bert adı verilir. Bu 5 lif uç Fosfat gruplarına ve üç bert uç Hidroksil gruplarına sahiptir. Deoksiribonükleik asit ile RNA arasındaki temel farklardan biri içerdikleri şekerdir. RNA 2 Deoksiriboz yerine 2 Deoksiriboz içerir.



Şekil 3.19. DNA'nın yapısı.

DNA iki ipliği birbirine bağlayan bazlar arasındaki hidrojen bağları yoluyla çift sarmalı stabilize eder. Deoksiribonükleik asitte bulunan dört baz Adenin A, Sitozin C, Guanin G ve Timin T olarak adlandırılır. Bu dört baz nükleotidler oluşturmak için şeker ve Fosfat ile birleşir. Örneğin: Adenozin, monofosfat bir nükleotiddir [28].

DNA yapısındaki bazlar, pürinler ve pirimidinlere ayrılır. Pürin grubunda yer alan Adenin ve Guanin beş ve altı alt halkanın kaynaşmasıyla oluşan heterosiklik bileşiklerdir. Sitozin ve Timin pirimidin türevleridir ve altı kullanıcı halkasından oluşurlar. Sitozinin yıkımı nedeniyle Urasil başka bir baz olan Deoksiribonükleik asitte nadiren bulunur. Kimyasal olarak Deoksiribonükleik asite benzeyen RNA'da Timin yerine Urasil bulunur [28].

3.2 DNA'NIN ÖZELLİKLERİ

Deoksiribonükleik asit; Adenin, Guanin, Sitozin ve Timin Nükleotidlerinden oluşur. Deoksiribonükleik asit çift sarmallı bir sarmal yapıya sahiptir. DNA nükleer hücreli

organizmaların çekirdeğinde ve nükleolar hücreli organizmaların stoplazmasında bulunur.

Nükleotidler birbirine hidrojenle bağlıdır. Guanin ve Sitozin arasındaki üçlü bağ, Adenin ve Timin arasındaki çift hidrojen bağıyla bağlanır. Deoksiribonükleik asit zincirinde Adenin, Timin'e karşılık gelir ve Guanin Sitozin'e karşılık gelir. Deoksiribonükleik asit molekülleri tüm organizmalarda Adenin, Timin, Guanin ve Sitozin bazlarından oluşmasına rağmen Nükleotidlerin sayı ve dizisindeki değişiklikler organizmaları birbirinden farklı kılar [28].

DNA'nın yapısını incelerken aşağıdaki denklem görülebilir.
Adenin Sayısı = Timin Sayısı, Guanin sayısı = Sitozin Sayısı
Nükleotid, şeker, baz ve Fosfat sayılarının birbirine eşit olduğu görülmüştür.

3.3 DNA'NIN ANALİZİ

James Watson ve Francis Crick, 1953'te yapılan bir çalışmada DNA'nın yapısının çift sarmal olduğuna dikkat çekmiştir. Watson ve Crick'in bu söylemlerinin gelişmesinin nedenleri iki kaynak olarak gösterilmiştir. Bunlardan biri hidrolize Deoksiribonükleik asit örneklerinin kompozisyon analizidir ve Deoksiribonükleik asidin X-ışını kırınım çalışmalarıdır. Erwin Chargaff ve arkadaşları, birçok organizmadan Deoksiribonükleik asit elde etmek için 1949 ve 1953 yılları arasında Kromatografi'yi kullanmışlardır. Ribonükleik asit numunesinde dört baz ayırt edilmiştir [28].

3.4 X-IŞINI KIRINIMI ANALİZİ

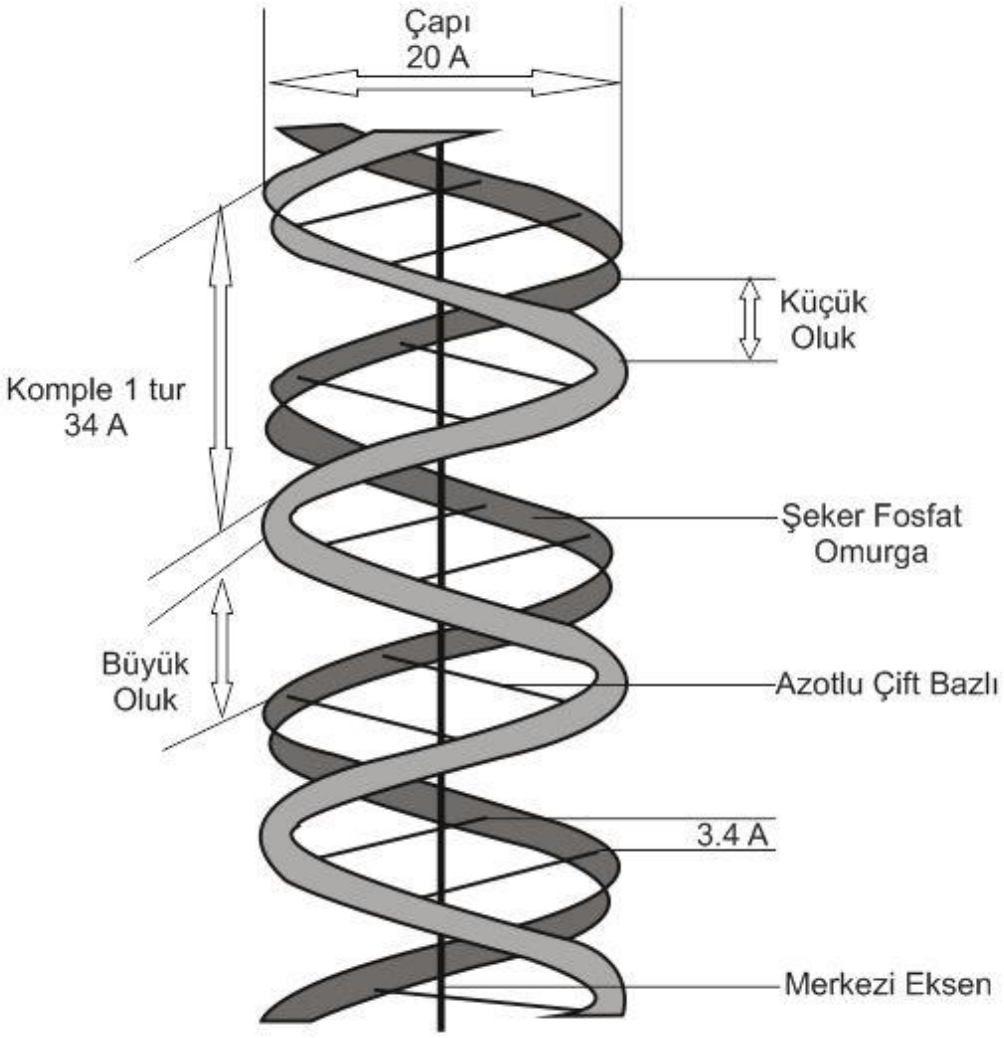
DNA zincirleri güçlü X ışınlarına maruz kaldıklarında molekülün atomik yapısına göre ışıma yaparlar. Bu saçılmaların sonucu ana hattın film üzerinde toz olarak görünmesidir. Moleküllerde özellikle kirli yapılar ve genel görünümleri ortaya çıkacaktır. 1938'de William Astbury, teknolojiyi DNA üzerinde test etmiştir. 1947'de Astbury, ham DNA'da 3.4 aralıklarını tekrarlayan bir yapıyı tespit etmek için bir

kumpas kullanmıştır. 1950 ile 1953 arasında Rosalind Franklin (Rosalind Franklin), daha saf ham DNA örneklerinden daha karmaşık X-ışını bilgileri elde etmiştir. Rosalind'in çalışmasında Astbury, 3.4 tekrar yapısının varlığını doğrulamış ve DNA'nın sarmal bir yapıya sahip olduğunu iddia etmiştir [28].

3.5 WATSON-CRİCK MODELİ

Watson ve Crick 1953'de yaptıkları çalışmalarla Deoksirübonükleik asitin yapısının ikili helezoni biçiminde olduğunu ortaya koymuşlardır.Buna göre;

1- İki uzun Polinükleotit zinciri, bir merkez dingil çevresinde kıvrılarak, sağ el ikili helezoni yapısını oluşturur.
2- İki zincir birbirine ters konumludur. Şöyle ki iki zincirin C-5 ucundan C-3 ucuna doğru olan istikametleri birbirine göre terstir.
3- Her iki zincirin bazları düzlemsel yapıdadır. Dizilimleri ise aksa dik, bazlar arasında 3.4 Å0.34 nm mesafe olacak biçimde birbiri arkasına dizilir ve helezoninin içinde yer alır.
4- Karşı zincirdeki Azotlu bazlar, Hidrojen bağları ile bağlanarak birbirleri ile eşleşirler, Deoksirübonükleik asitte yalnızca A=T ve G=C eşleşmesi muhtemeldir.
5- Helezoni'nin her bir bütün dönümü 34 Å3.4 nm'dir. Böylece Deoksirübonükleik asitin her bir dönümünde 10 baz yer alır.
6- Molekülün rastgele bir kısmında aks üzerinde gizeme ile daha geniş olan büyük majör oluklar ve daha dar olan minik minör oluklar yer alır.
7- Helezoni'nin çapı 20 Å 2 nm'dır.



Şekil 3.20. Watson crick modeli.

Bazların birbiriyle uyuşması bu modelin en önemli genetik özelliğidir. Bir zincir, 5'in birincil ucundan 3'ün birincil ucuna diğeri 3'ün birincil ucundan 5'in birincil ucuna kadar uzanır.

Watson ve Crick'in en önemli keşifleri, temel eşleşmeleri bulmaktır. Chargaff'a göre Adenin Timin'e, Guanin ise Sitozin'e eşdeğerdir. Kısaca, A = T ve G = C baz eşleşmesi tamamlayıcılık kavramının temelidir. Watson ve Crick'e göre, A = G ve C = T baz eşleşmesi gibi diğer baz eşleşmelerinin olasılığını kabul etmiyorlar. Çünkü bunlar pürin ve pürin ile pirimidin ve pirimidin arasındaki eşleşmelerdir. Bu eşleşmede bazı parçaların spiral çapı 20'den büyük veya küçük olacaktır. Hidrojen bağının önemi burada ortaya çıkıyor.

Bir Hidrojen bağı, kovalent olarak bağlanmış bir Hidrojen atomu ile eşleşmemiş elektronlar içeren başka bir atom arasındaki çok zayıf bir elektrostatik çekimdir. Baz'ın çift sarmal yapıdaki konumuna göre A ve T iki hidrojen bağı G ve S ise üç Hidrojen bağı oluşturur. Tek başına iki veya üç hidrojen bağı çok zayıftır. Ancak iki veya üç bin tanesi sürekli göründüğünde spirale büyük bir dayanıklılık sağlar. Güçlü bir ölçüm yapıldı. Sonuçlar DNA'nın Watson ve Crick'in önerdiği 10 yerine tek turda 10.4 bc içerdiğini göstermiştir. Eğitim modelinde her bir baz çifti bitişik baz çiftine göre spiral eksen etrafında 36 ° döndürülür ve yeni ölçüm 34.6 ° 'dir [28].

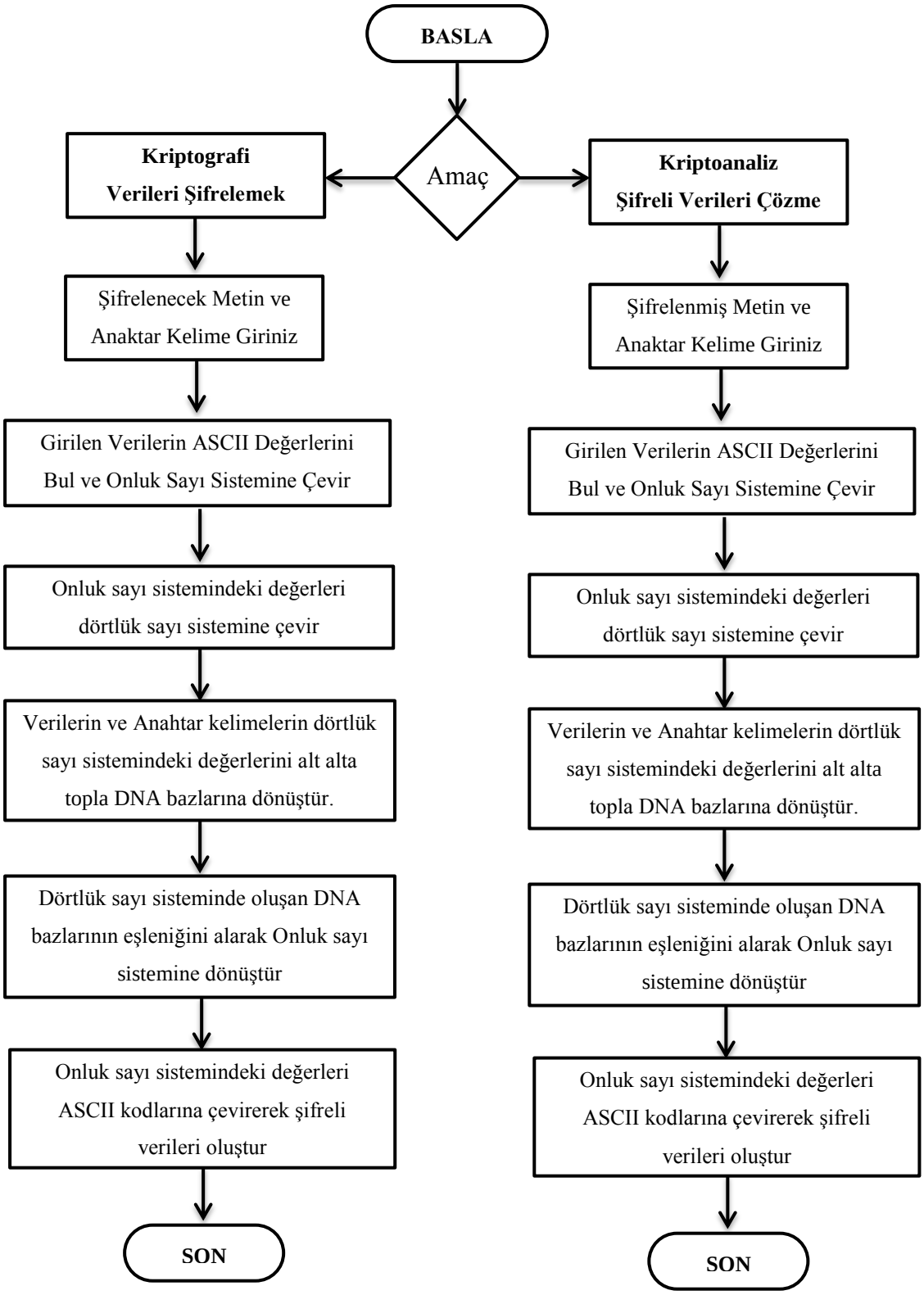
BÖLÜM 4

GENETİK KOD YÖNTEMİ İLE KRİPTOLOJİ UYGULAMASI

4.1 GENETİK ŞİFRELEME ALGORİTMASI

Canlıların biyolojik özelliklerinin taşındığı, korunduğu veya şifrelendiği DNA yapısından yola çıkarak şifreleme algoritmamızın temelini oluşturmaktadır. Yukarıda 3. bölümde geniş açıklaması yapılan, (Deoksiribonükleik Asit) DNA'nın yapısı incelendiğinde Adenin (A), Guanin (G), Sitozin (C) ve Timin (T), gibi organik bazlar bulunmaktadır. Bu organik bazlarının birbirleri ile olan ilişkilerinden yola çıkarak farklı bir şifreleme algoritması geliştirilmiştir [28].

Geliştirilen uygulama programının algoritması aşağıdaki şekilde gösterilmiştir.

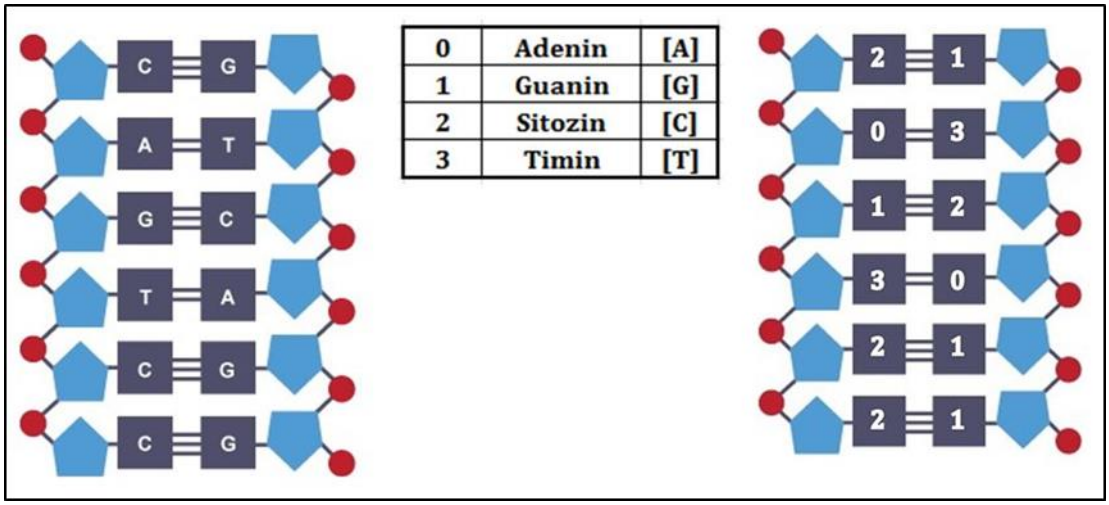


Şekil 4.21. Genetik şifreleme algoritması.

4.2 GENETİK ŞİFRELEME ALGORİTMASININ UYGULANMASI

Bu çalışmada, Genetik Şifreleme Algoritması (GEA), DES, AES ve RSA şifreleme algoritmalarının birlikte bulunduğu bir uygulama programı geliştirilmiştir. Bu uygulama program çalıştırılarak karşımıza gelen şifreleme algoritmalarından hangisiyle işlem yapmak istiyorsak o algoritmayı tercih etmeliyiz. Tercih edilen şifreleme algoritmasına göre 64 veya 128 bitlik rastgele anahtar kelime oluşturulur. Şifrelenmesini istediğimiz metin veya verilerimizi ilgili bölüme girerek şifrelenmesi sağlanır. Girilen verilerin ve anahtar kelimenin her karakterinin ASCII değerleri hesaplanır. Onluk tabanda hesaplanan bu ASCII değerleri, bir genetik koda dönüştürülebilmesi için her karakterin ASCII değerinin dörtlük tabana çevrilerek sayısal değerleri oluşturulur. (Onluk sayı sistemde verilen bir sayıyı dörtlük sayı sistemine çevirdiğimizde kalan değerler 0,1,2,3 değerlerini oluşturur.)

ASCII değerlerimizi dörtlük sayı sistemine dönüştürmemizin asıl sebebi DNA yapısında bulunan organik bazlar ile dörtlük sayı sisteminde kalan değerleri eşleştirmektir. Bu eşleşme aşağıdaki tabloda gösterilmiştir.



Şekil 4.22 DNA organik bazlarının kodlanması.

Yukarıda sol taraftaki şekilde DNA bazlarının birbirleri ile olan eşleşmesi gösterilmiştir. Ortadaki resimde ise DNA bazlarının her birine sayı değeri atayarak, hangi Baz'ın hangi sayı ile eşleştiği gösterilmiştir. Sol taraftaki şekilde ise DNA

bazlarının oluşturduğumuz sayı değerleri ile birbirleri arasında nasıl bir eşleşme olduğu gösterilmiştir.

Böylelikle herbir organik Baz'a dönüşen rakamlarımız tıpkı DNA eşlenmesi gibi kendisine karşılık gelen organik bazlarla eşlenmiş olur. (Adenin'le Timin, Guanin'le Sitozin eşlenir). Yani 0'ın karşısına 3, 1'nin karşısına 2 gelir. Burada tekrar eşleme yaparak yeni DNA zinciri oluşturulup, akılda tutulur.

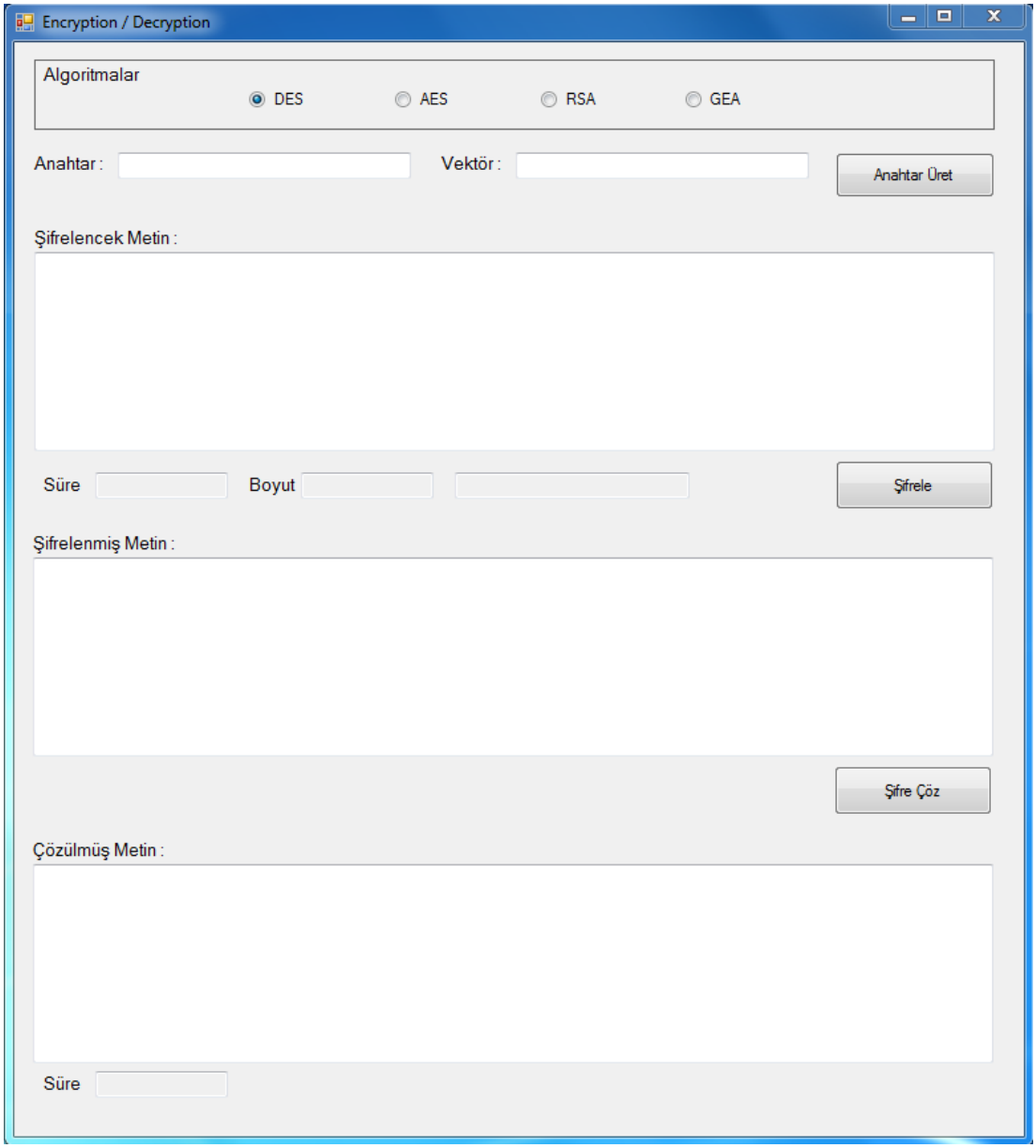
Şifrelenmesi için programa girilen verilerin dörtlük sayı sistemindeki DNA eşleşmesi ile anahtar şifremizin DNA eşlemeleri dörtlük sayı sisteminde toplanarak tek bir DNA zinciri oluşturulur. Dörtlük sayı sisteminde yeni oluşan bu sayı değerlerinin onluk sayı sitemindeki karşılıklarına çevirilir. Son olarak elimizde bulunan onluk sistemdeki sayıların ASCII tablosundaki karakter değerleri elde edilir. Böylelikle şifrelenmesini istediğimiz metin veya verilerimiz şifreli verilerin bulunduğu bölüme yazdırılması sağlanır.

Şifreli verilerin çözümlenmesi ise şifreleme yapmak için yapılan adımların tekrardan tersi yapılarak şifrelenmiş olan veriler başlangıç yani ilk haline dönüştürülür.

Yukarıda algoritma üzerinden açıkladığımız, şifreleme yöntemini örnek uygulama üzerinden açıklayalım.

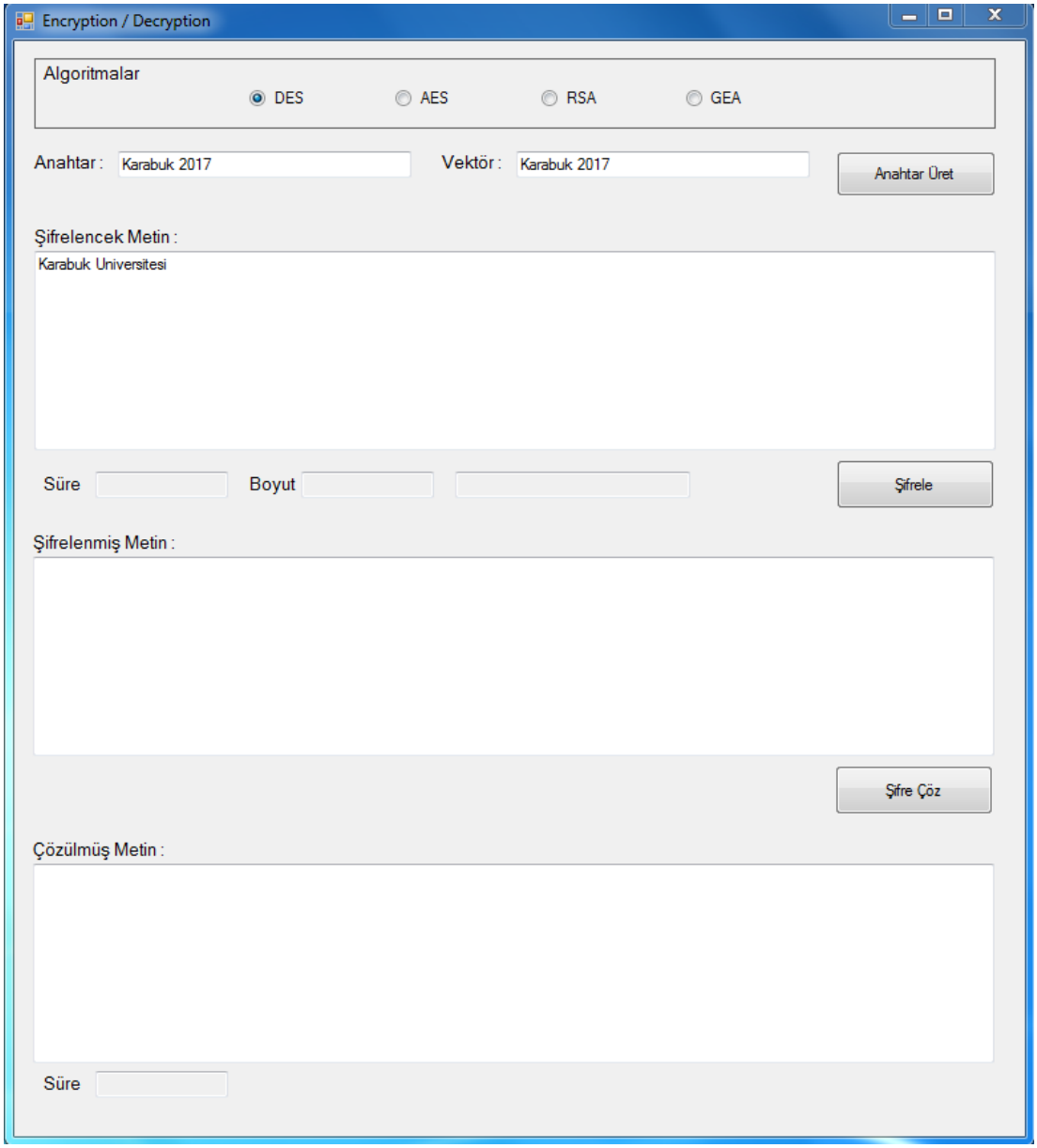
Örnek Uygulama

Genetik şifreleme projemiz, çalışma prensibi yapılan örnek çalışma üzerinde gösterilmiştir. Uygulama programımızı başlatıyoruz.



Şekil 4.23. Uygulama programı.

Yukarıdaki şekilde görüldüğü gibi metin veya verilerimizi ilgili bölüme girerek şifreleme ya da şifre çözme işlemleri yapılır. Burada ayrıca girilen verilerin kaç bytelık bir veri olduğu, şifrelenirken ve şifre çözümlerken ne kadar zaman aldığı da mili saniye olarak hesaplanmaktadır.



Şekil 4.24. Verilerin Şifrelenmesi.

Örnek çalışma uygulamamızda şifrelemek istediğimiz veri "Karabuk Üniversitesi", şifrelemede kullanacağımız anahtar kelimemiz de "Karabuk 2017" olsun.

Bu örnek uygulamamızda şifrelenecek metnin ve kullanılan anahtar kelimenin fazla yer tutmaması için mümkün olduğunca kısa metinler tercih edilmiştir. Normalde 128 bitlik rastgele anahtar kelime üreten programımız da mevcuttur. Bilgisayarınızın işlem kapasitesine bağlı olarak daha büyük şifrelemek istediğiniz metin dosyaları da seçebilirsiniz.

İlk aşamada şifrelenecek metnimiz ve anahtar kelimemizin tüm harflerinin ASCII değerlerini bulalım.

K	a	r	a	b	u	k	U	n	i	v	e	r	s	i	t	e	s	i
123	148	165	148	149	167	158	133	160	156	168	152	164	165	156	166	152	165	156

Şekil 4.25. Metnimizin ASCII değerlerinin karşılığı.

K	a	r	a	b	u	k	2	0	1	7
123	148	165	148	149	167	158	95	93	94	100

Şekil 4.26. Anahtar kelimemizin ASCII değerlerinin karşılığı.

Yukarıdaki onluk sayı sisteminde bulunan harflerin ASCII değerleri aşağıda dörtlük sayı sitemine dönüştürüyoruz.

K	a	r	a	b	u	k	U	n	i	v	e	r	s	i	t	e	s	i
1323	2110	2211	2110	2111	2213	2132	2011	2200	2130	2220	2120	2211	2212	2130	2213	2120	2212	2130

Şekil 4.27. Metnimizin dörtlük sayı sistemindeki karşılığı.

K	a	r	a	b	u	k	2	0	1	7	K	a	r	a	b	u	k	2
1323	2110	2211	2110	2111	2213	2132	1133	1131	1131	1210	1323	2110	2211	2110	2111	2213	2132	1133

Şekil 4.28. Anahtar kelimemizin dörtlük sayı sistemindeki karşılığı.

K+K	a+a	r+r	a+a	b+b	u+u	k+k	U+2	n+0	i+1	v+7	e+K	r+a	s+r	i+a	t+b	e+u	s+k	i+2
2202	0220	0022	0220	0222	0022	0220	3130	3331	3221	3030	3003	0321	0023	0200	0320	0333	0300	3223

Şekil 4.29. Sayıların dörtlük sayı sisteminde toplamı.

Aşağıda şekilde toplamları dörtlük sayı sisteminda yazılan değerlerin DNA eşleşmelerine dönüştürülmesi gösterilmiştir. Ayrıca burada eşleşen DNA organik bazlarının tekrar birbirleri arasındaki eşleştirilerek güvenlik artırılmıştır.

2202	0220	0022	0220	0222	0022	0220	3130	3331	3221	3030	3003	0321	0023	0200	0320	0333	0300	3223
SSAS	ASSA	AASS	ASSA	ASSS	AASS	ASSA	TGTA	TTTA	TSSG	TATA	TAAT	ATSG	AAST	ASAA	ATSA	ATTT	ATAA	TSST
GGTG	TGGT	TTGG	TGGT	TGGG	TTGG	TGGT	ASAT	AAAT	AGGS	ATAT	ATTA	TAGS	TTGA	TGTT	TAGT	TAAA	TATT	AGGA

Şekil 4.30. Toplanan sayıların DNA eşleşmesi.

Yukarıda DNA organik bazları incelendiğin de Adenin Timin ile Guanin ise Sitozin ile eşleşmektedir.

0=	Adenin	Timin=	3
1=	Guanin	Sitozin=	2
2=	Sitozin	Guanin=	1
3=	Timin	Adenin=	0

Şekil 4.31. DNA eşleşmesi.

1131	3113	3311	3113	3111	3311	3113	0203	0003	0112	0303	0330	3012	3310	3133	3013	3000	3133	0110
93	213	245	213	213	245	213	17	1	21	49	52	197	244	221	197	192	221	20
]	F	⌡	F	F	⌡	F	(dc1)	(soh)	(nak)	1	4	┼	⌠	▌	┼	└	▌	(dc4)

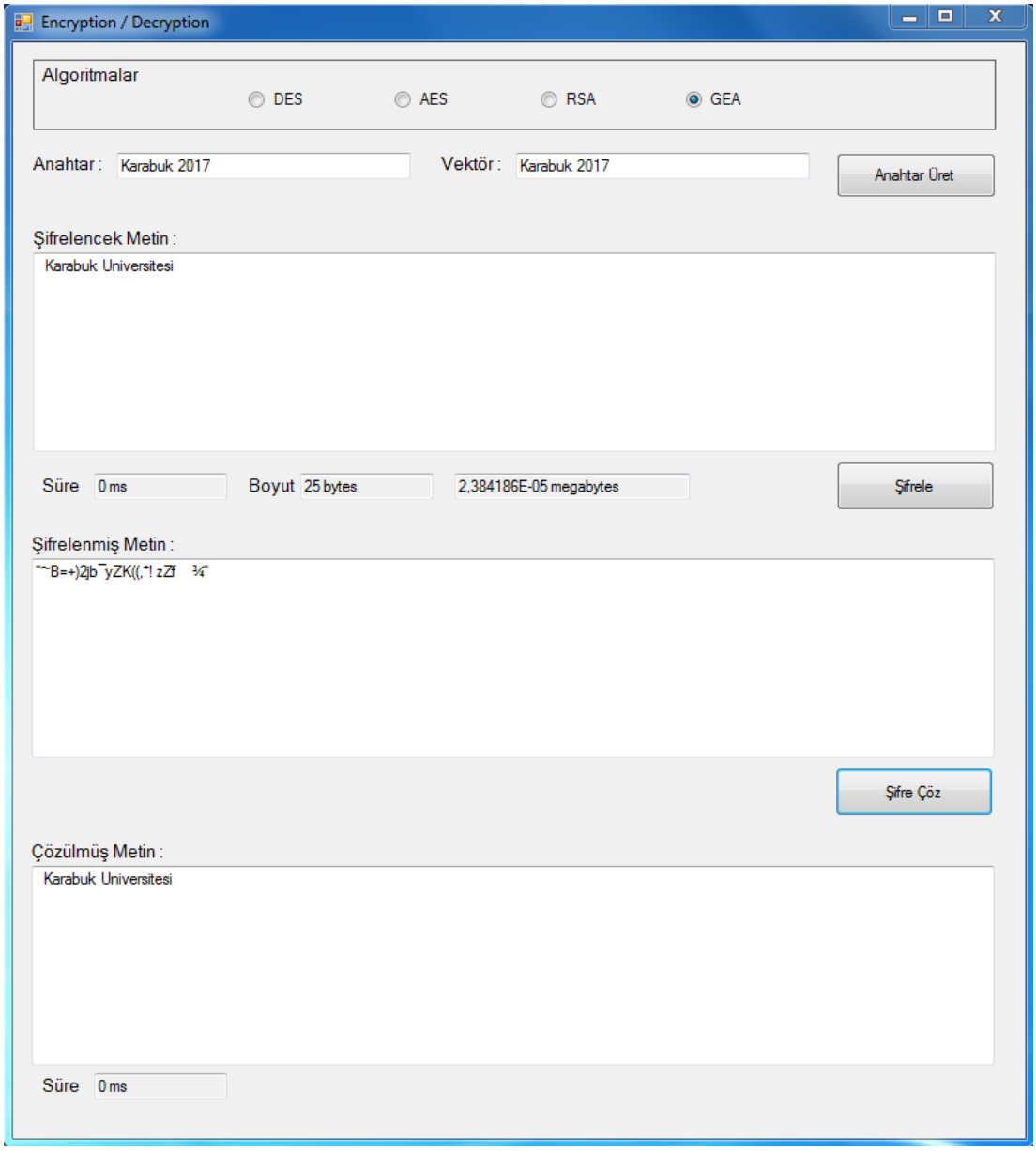
Şekil 4.32. Dörtlük sayı sisteminden onluk sayı sistemine geçiş.

Yukarıdaki şekil incelendiğinde algoritmada uygun eşleştirmeler yapıldıktan sonra dörtlük sayı sisteminde elde edilen verilerin tekrar onluk sayı sistemine ve oradan da ASCII değerlerine dönüştürülmüştür. Kısacası örnekte gösterildiği gibi şifrelenecek metin.gir dosyamızda "Karabuk Üniversitesi" ve Anahtar şifremiz "Karabuk 2017" olsaydı metin.cik dosyamıza aşağıdaki şekilde yazılırdı.

] F ⌡ F F ⌡ F (dc1) (soh) (nak) 1 4 ┼ ⌠ ▌ ┼ └ ▌ (dc4)

Şekil 4.33. Şifrelenmiş verimiz.

Genetik şifreleme programımızın en önemli özelliklerden birisi de şifrelenmiş karakterlerin tekrar eski haline çevrilebilmesidir. Şifrelenmiş metnimiz üzerine yukarıda açıklanan algoritmaları geriye doğru uyguladığımız takdirde tekrar eski metnimizi elde edebiliriz. Burada tek dikkat etmemiz gereken, şifre çözümlerken de aynı anahtar kelimemizi kullanmaktır.



Şekil 4.34. Şifre çözümleme.

4.3 GENETİK ŞİFRELEME ALGORİTMA TESTİ VE ÖLÇÜMLERİ

Bu tez çalışmasında şifreleme algoritmalarının performans analizleri, deneysel analiz yöntemiyle gerçekleştirilmiştir. Deneysel analiz yöntemi, karmaşık algoritmaların değerlendirilmesinde tercih edilen güvenilir yöntemlerden biridir [8,15,23].

Geliştirilen genetik şifreleme uygulamasının performans değerlendirmesini yapmak için DES (Standart Şifreleme Algoritması), AES (Gelişmiş Şifreleme Algoritması) ve RSA (Rivest Shamir Adleman) şifreleme algoritmaları karşılaştırılmıştır.

DES (Standar Şifreleme Algoritması) ve AES (Gelişmiş Şifreleme Algoritması) şifreleme algoritmaları Simetrik (gizli anahtarlı) şifreleme algoritmalarıdır. RSA (Rivest-Shamir-Adleman) ise Asimetrik (açık anahtarlı) bir şifreleme algoritmasıdır.

Asimetrik şifreleme algoritmaları büyük yer tutan metin dosyaların şifrelenmesi için genel olarak uygun değildir. RSA Algoritması 112 byte'e kadar olan verilerin şifrelenmesinde kullanılmaktadır. RSA algoritmaları büyük asal sayılarla işlem yaptıkları için genellikle sayısal işlemlerde ve bankacılık sektöründe çok yaygın kullanılmaktadır [8,15].

Verilerin şifrelenmesi için tercih edilen şifreleme algoritmalarının performanslarını ölçerken aşağıdaki kriterler doğrultusunda deneysel değerlendirmeler yapılmıştır [8,21,15].

1- Şifreleme algoritmalarının veriyi şifrelerken ve çözerken ne kadar zaman harcadığını ölçerken bu işlem sırasında işlemci (CPU) ve hafıza (RAM) kullanımı hesaplanmıştır.
2- Kullanılan şifreleme algoritmaları birbirleriyle işlem hızı, CPU ve RAM kullanımı bakımından kıyaslanarak performansları grafiksel olarak sunulmştur.

Bu deneysel analiz yönteminde işlem zamanı; dakika, RAM (Bellek) kullanımı; MegaByte, CPU (İşlemci) kullanımı ise % olarak verilmiştir. Ayrıca bu deneyde kullanılan masaüstü bilgisayarın özellikleri aşağıda verilmiştir.

Çizelge 4.1. Performans testi yapılan bilgisayarın özellikleri.

PC	**Test Yapılan Bilgisayarın Özellikleri**
İşlemci (CPU)	Intel (R) Core 2 DUO / 2.94 GHz
Bellek (RAM)	2048 MB
Kullanılan Sistemi	Windows7 Ultimate 64Bit

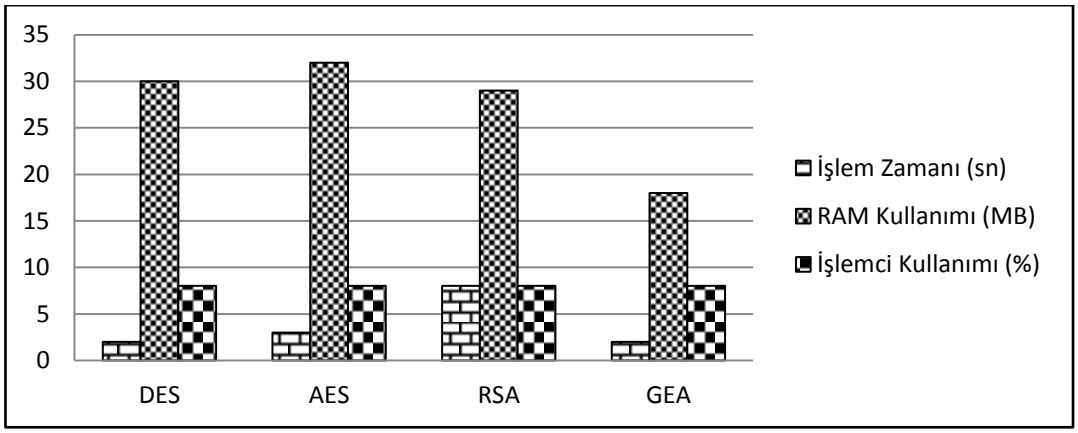
4.3.1. 58 Byte'lık Verilerin Şifreleme ve Şifre Çözme Analizi

58 Byte'lık karekter uzunluğuna sahip metnin şifrelenmesi ve çözümlenmesi sürecinde elde edilen işlem zamanı, işlemci kullanımı, bellek kullanımına ait değerler, aşağıdaki tablo ve grafikle gösterilmiştir.

Çizelge 4.2. 58 Byte'lık verinin şifreleme işlemi performans değerleri.

	Şifreleme Algoritmaları	İşlem Zamanı (sn)	Bellek (RAM) Kullanımı (MB)	İşlemci (CPU) Kullanımı (%)
Test Bilgisayarı	DES	2 (sn)	30 MB	% 8
	AES	3 (sn)	32 MB	% 8
	RSA	8 (sn)	29 MB	% 8
	GEA	2 (sn)	18 MB	% 8

Çizelge 4.2'de 58 byte'lık veriler şifrelenirken DES, AES, GEA ve RSA şifreleme algoritmalarının performans değerlerini göstermektedir. Şekil 4.15'deki grafik incelendiğinde, 58 byte'lık veri, şifrelenirken kullanılan işlemci değerlerinin aynı olduğu görülmüştür.



Şekil 4.35. 58 Byte'lık verinin şifreleme işlemi grafik gösterimi.

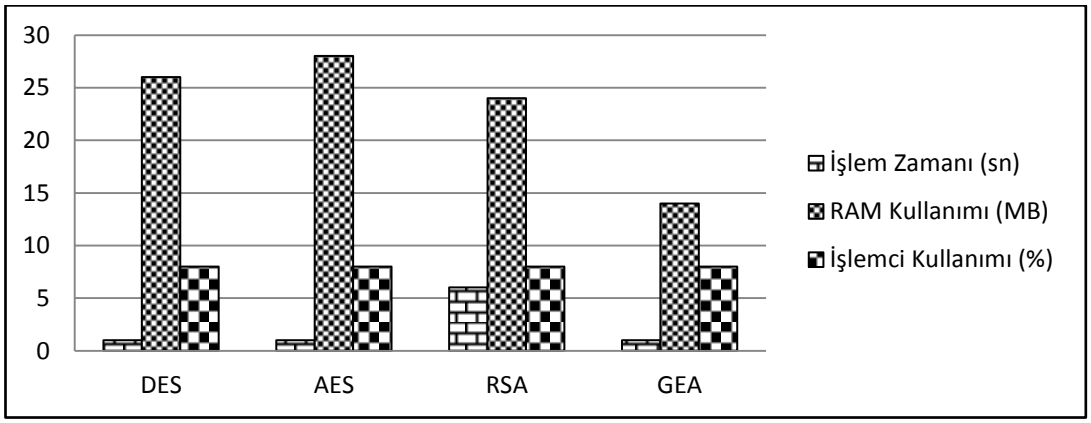
GEA Algoritması'nın işlem zamanı değeri DES Algoritması'yla aynı iken, bellek kullanım değerleri diğer DES, AES ve RSA algoritmalarına göre performansının daha iyi olduğu görülmektedir. Bunun sebebi ise kullanılan anahtar yapısı ve çalışma

prensibinden kaynaklanmaktadır. GEA 128 bit anahtar yapısı kullanırken DES Algoritması 56 bitlik anahtar yapısı kullanmaktadır. Bu da DES Algoritması'nın işlem hızı etkilemiş olsa da bellek kullanımını artırmaktadır.

Çizelge 4.3. 58 Byte'lık verinin şifre çözme işleminin performans değerleri.

	Şifreleme Algoritmaları	İşlem Zamanı (sn)	Bellek (RAM) Kullanımı (MB)	İşlemci (CPU) Kullanımı (%)
Test Bilgisayarı	DES	1 (sn)	26 MB	% 8
	AES	1 (sn)	28 MB	% 8
	RSA	6 (sn)	24 MB	% 8
	GEA	1 (sn)	14 MB	% 8

Çizelge 4.3'te 58 byte'lık verilerin şifre çözümlemesinde DES, AES, GEA ve RSA şifreleme algoritmaların performans değerlerini göstermektedir. Şekil 4.16'daki grafik incelendiğinde, 58 byte'lık şifreli veri çözümlenirken, simetrik şifreleme algoritmaları olan DES, AES ve GEA işlem zamanı ve işlemci kullanımı aynı değerlerde iken Asimetrik şifreleme olan RSA Algoritması'na göre performanslarının daha iyi olduğu görülmektedir.



Şekil 4.36. 58 Byte'lık verinin şifre çözme işleminin grafik gösterimi.

Genel olarak GEA Algoritması'nın bellek kullanım performansı DES, AES ve RSA algoritmalarına göre daha iyi olduğu görülmektedir. Bunun sebebi ise 128 bitlik anahtar kullanımı ve çalışma prensibinden kaynaklanmaktadır.

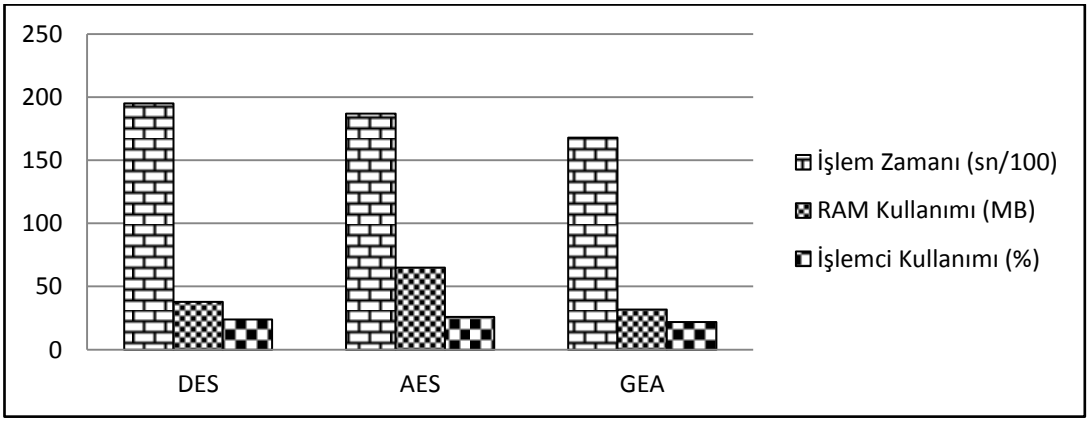
4.3.2. 102 KiloBytelık VerilerinŞifreleme ve Şifre Çözme Analizi

102 Byte'lık karekter uzunluğuna sahip metnin şifrelenmesi ve çözümlenmesi sürecinde elde edilen işlem zamanı, işlemci kullanımı ve bellek kullanımına ait değerler aşağıdaki tablo ve grafikle gösterilmiştir.

Çizelge 4.4. 102 KiloByte'lık verinin şifreleme işlemi performans değerleri.

	Şifreleme Algoritmaları	İşlem Zamanı (sn)	Bellek (RAM) Kullanımı (MB)	İşlemci (CPU) Kullanımı (%)
Test Bilgisayarı	DES	1750 (sn)	35 MB	% 24
	AES	1713 (sn)	65 MB	% 26
	GEA	1680 (sn)	32 MB	% 22

Çizelge 4.4'te 102 KiloByte'lık veri şifrelenirken DES, AES ve GEA simetrik şifreleme algoritmalarının performans değerlerini gösterilmiştir. RSA Algoritması büyük veri şifrelemeleri yapamaz. Algoritması buna uygun değildir. Genellikle büyük asal sayılar üzerinde işlem gerçekleştirir. Bu sebeple de yukarıdaki ve bundan sonraki veri paketlerinde gösterilmemiştir. Şekil 4.17'deki grafik incelendiğinde GEA Algoritması işlem zamanı, bellek ve İşlemci kullanımı diğer DES ve AES algoritmalara göre performansının daha iyi olduğu görülmektedir.



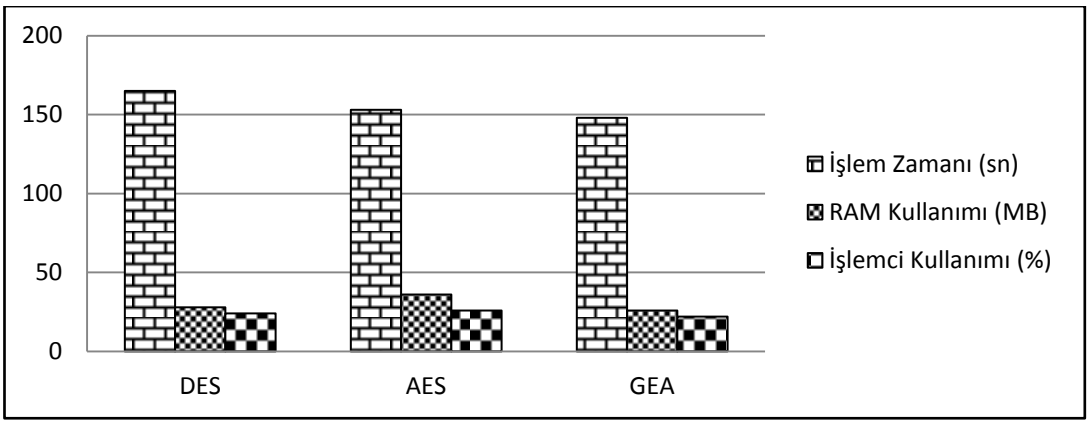
Şekil 4.37. 102 KiloByte'lık verinin şifreleme işlemi grafik gösterimi.

Bunun sebebi GEA simetrik şifreleme algoritmasında kullanılan anahtar boyutu ve algoritmanın çalışma prensibinden kaynaklanmaktadır.

Çizelge 4.5. 102 KiloByte'lık verinin şifre çözme işleminin performans değerleri.

	Şifreleme Algoritmaları	İşlem Zamanı (sn)	Bellek (RAM) Kullanımı (MB)	İşlemci (CPU) Kullanımı (%)
Test Bilgisayarı	DES	165 (sn)	28 MB	% 24
	AES	153 (sn)	36 MB	% 26
	GEA	148 (sn)	26 MB	% 22

Çizelge 4.5'te 102 KiloByte'lık verilerin şifre çözümlemesinde DES, AES ve GEA Simetrik Şifreleme algoritmalarının performans değerlerini göstermektedir.



Şekil 4.38. 102 KiloByte'lık verinin şifre çözme işleminin grafik gösterimi.

Şekil 4.18'deki grafik incelendiğinde, GEA Algoritması'nın işlem zamanı, bellek ve işlemci kullanım değerleri DES ve AES algoritmalara göre daha iyi olduğu görülmektedir. Sebebi aynı şifrelemedeki gibi anahtar yapısı ve çalışma prensibinden kaynaklanmaktadır.

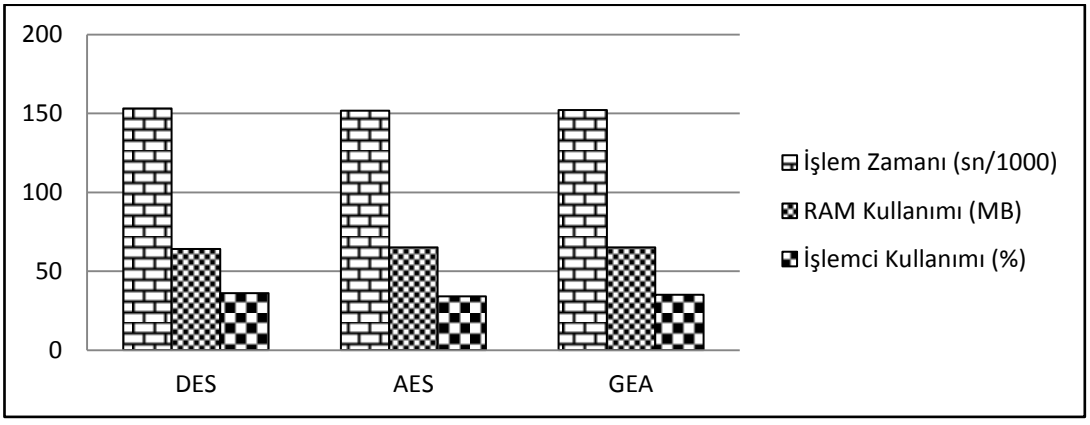
4.3.3. 1 MegaByte'lık Verilerin Şifreleme ve Şifre Çözme Analizi

1 MegaByte'lık karekter uzunluğuna sahip metnin şifrelenmesi ve çözümlenmesi sürecinde elde edilen işlem zamanı, işlemci kullanımı ve bellek kullanımına ait değerler aşağıdaki tablo ve grafikle gösterilmiştir.

Çizelge 4.6. 1 MegaByte'lık verinin şifreleme işlemi performans değerleri.

	Şifreleme Algoritmaları	İşlem Zamanı (sn)	Bellek (RAM) Kullanımı (MB)	İşlemci (CPU) Kullanımı (%)
Test Bilgisayarı	DES	153073 (sn)	64 MB	% 36
	AES	151655 (sn)	65 MB	% 34
	GEA	152056 (sn)	65 MB	% 35

Çizelge 4.6'da 1 MegaByte'lık verilerin şifrelenmesinde DES, AES ve GEA Simetrik şifreleme algoritmalarının performans değerleri gösterilmiştir. Şekil 4.19'daki grafik incelendiğinde, AES Şifreleme Algoritması'nın işlem zamanı, bellek ve işlemci kullanımı diğer DES ve GEA algoritmalara göre performansının daha iyi olduğu görülmektedir.



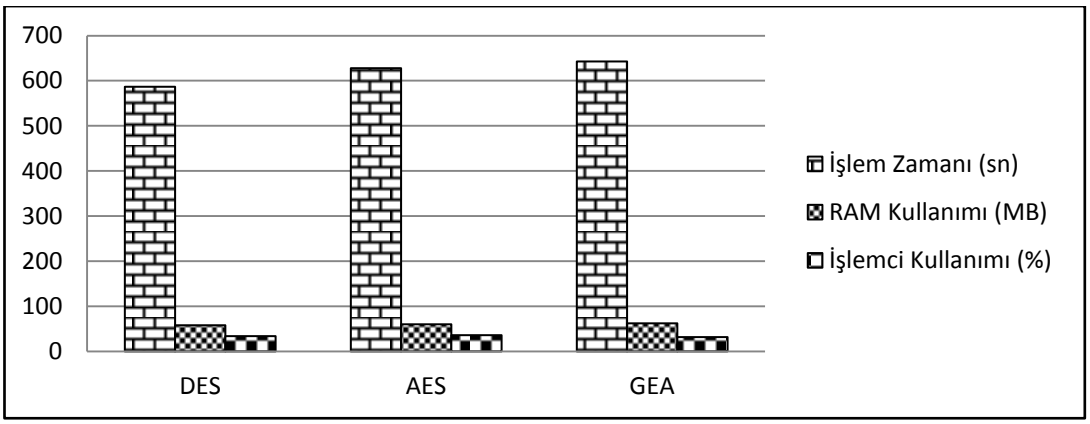
Şekil 4.39. 1 MegaByte'lık verinin şifreleme işlemi grafik gösterimi.

Bunun sebebi ise AES Algoritması'nın yapısı ve çalışma prensibi bloklar halinde verileri şifreleme prensibinden kaynaklanmaktadır. Ayrıca AES Algoritması esnek bir yapıya sahip olmasıdır. Farklı boyutta 128 bit, 192 bit veya 256 bit anahtarlar kullanılsa bile işlem hızı ve performansı değişmez.

Çizelge 4.7. 1 MegaByte'lık verinin şifre çözme işleminin performans değerleri.

	Şifreleme Algoritmaları	İşlem Zamanı (sn)	Bellek (RAM) Kullanımı (MB)	İşlemci (CPU) Kullanımı (%)
Test Bilgisayarı	DES	587 (sn)	58 MB	% 34
	AES	628 (sn)	60 MB	% 36
	GEA	643 (sn)	62 MB	% 32

Çizelge 4.7'de 1 MegaByte'lık verilerin şifre çözümlemesinde DES, AES ve GEA Simetrik Şifreleme Algoritmaları'nın performans değerlerini göstermektedir.



Şekil 4.40. 1 MegaByte'lık verinin şifre çözme işleminin grafik gösterimi.

Şekil 4.20'deki grafik incelendiğinde, şifrelemede olduğu gibi yine AES Şifreleme Algoritması'nın işlem zamanı, bellek ve işlemci kullanımı diğer DES ve GEA Algoritmalar'a göre performansının daha iyi olduğu görülmektedir. Bu da AES Şifreleme Algoritması'nın yapısı ve çalışma prensibinden kaynaklanmaktadır.

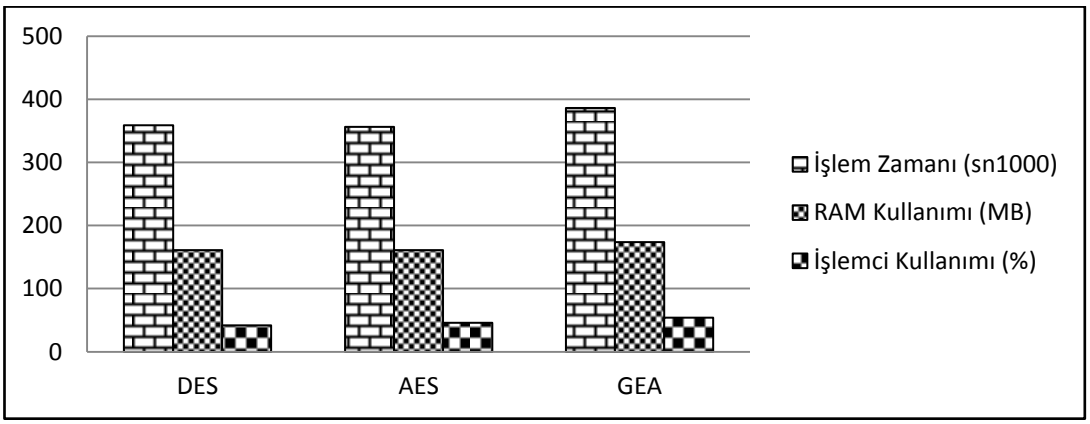
4.3.4. 5 MegaBytelık Verilerin Şifreleme ve Şifre Çözme Analizi

5 MegaByte'lık karakter uzunluğuna sahip verilerin şifrelenmesi ve şifre çözümlenmesi sürecinde elde edilen işlem zamanı, işlemci kullanımı ve bellek kullanımına ait değerler aşağıdaki tablo ve grafikle gösterilmiştir.

Çizelge 4.8. 5 MegaByte'lık verinin şifreleme işlemi performans değerleri.

	Şifreleme Algoritmaları	İşlem Zamanı (sn)	Bellek (RAM) Kullanımı (MB)	İşlemci (CPU) Kullanımı (%)
Test Bilgisayarı	DES	3586500 (sn)	161 MB	% 42
	AES	3561497 (sn)	162 MB	% 46
	GEA	3862428 (sn)	174 MB	% 54

Çizelge 4.8'de 5 MegaByte'lık verilerin şifrelenmesinde DES, AES ve GEA Simetrik şifreleme algoritmalarının birbirlerine göre performans değerlerini göstermektedir.



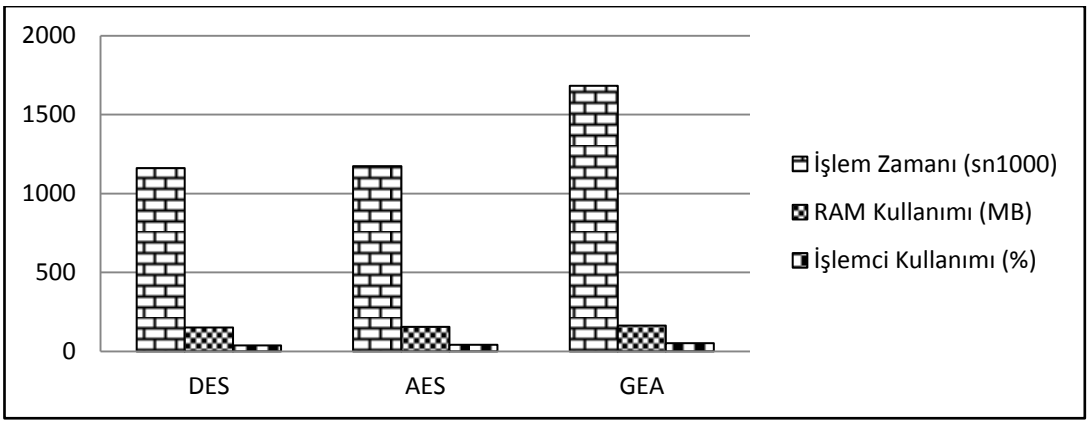
Şekil 4.41. 5 MegaByte'lık verinin şifreleme işlemi grafik gösterimi.

Şekil 4.21'deki grafik incelendiğinde, 5 MB'lık verilerin şifrelenmesinde DES ve GEA Algoritması işlem zamanı, bellek ve işlemci kullanımı birbirine çok yakın sonuçlar elde edilirken, AES Şifreleme Algoritması'nın performans değerlerinin daha iyi olduğu görülmektedir. Bunun sebebi ise büyük verilerin şifrelenmesinde AES Şifreleme Algoritma yapısı ve çalışma prensibi yönüyle daha kullanışlı olmasıdır.

Çizelge 4.9. 5 MegaByte'lık verinin şifre çözme işleminin performans değerleri.

	Şifreleme Algoritmaları	İşlem Zamanı (sn)	Bellek (RAM) Kullanımı (MB)	İşlemci (CPU) Kullanımı (%)
Test Bilgisayarı	DES	1161 (sn)	152 MB	% 38
	AES	1174 (sn)	156 MB	% 42
	GEA	1682 (sn)	164 MB	% 52

Çizelge 4.9'da 5 MegaByte'lık şifreli verilerin çözümlenmesinde DES, AES ve GEA Simetrik şifreleme algoritmalarının performans değerlerini göstermektedir.



Şekil 4.42. 5 MegaByte'lık verinin şifre çözme işleminin grafik gösterimi

Şekil 4.22'deki grafik incelendiğinde, 5 MegaByte'lık şifreli verilerin çözümlenmesinde AES Şifreleme Algoritması'nın performans değerleri DES ve GEA Şifreleme algoritmalarının performans değerlerinden daha iyi olduğu görülmektedir. Bunun sebebi ise AES Şifreleme Algoritması'nın yapısından kaynaklanmaktadır.

BÖLÜM 5

TARTIŞMA VE ÖNERİLER

Veri güvenliğinin sağlanması için simetrik ve asimetrik şifreleme algoritmaları incelenerek yeni bir şifreleme uygulaması geliştirilmiştir. Geliştirilen bu şifreleme uygulaması simetrik şifreleme algoritmaları gibi tek bir anahtar kullanılmaktadır. Simetrik şifreleme algoritmaları, verileri şifrelerken ve şifre çözerken tek bir anahtar kullanmaktadır. Bu sebeple de kullanılan anahtarın karşı tarafa güvenli bir şekilde gönderilmesi önem arz etmektedir. Bu durum Simetrik Şifreleme Algoritması'nın güvenliği için büyük bir dezavantajdır.

Asimetrik Şifreleme algoritmaları iki anahtar kullanıldığı için simetrik şifreleme algoritmalarına göre daha güvenlidir. Verileri şifrelemek için veriyi şifreleyen gönderici ve kendisine gönderilen şifrelenmiş verinin şifresini çözmek isteyen alıcı, kendi anahtarlarını oluştururlar. Bu nedenle, mesajı şifrelemek için iki anahtar kullanılır. Asimetrik şifreleme algoritmaları, büyük asal sayılarla işlemler gerçekleştirdiği için matematiksel hesaplamalara dayalı algoritmalardır. Bu nedenle, Simetrik Şifreleme Algoritması ile karşılaştırıldığında, Asimetrik Şifreleme Algoritması zaman açısından çok yavaştır.

Bu çalışmada yapılan testler sonucunda verilerin şifrelenmesi ve şifreli verilerin çözümlenmesinde simetrik şifreleme algoritmaları Asimetrik Şifreleme algoritmalarına göre performansları daha iyi olduğu tespit edilmiştir. Şifreleme ve şifre çözme işleminde kullanılan anahtar boyutuna göre kaba kuvvet kırılma süreleri de değişiklik göstermektedir. DES Algoritması'nda 56 bit anahtar kullanıldığı için kaba kuvvet kırılma süresi 2^{56}iken AES ve GEA algoritmaları 128 bit anahtar kullandığı için kırılma süresi 2^{128}dir.

Geliştirilen Genetik Şifreleme Algoritması (GEA), DES, AES ve RSA şifreleme algoritmalarına göre küçük boyutlu verilerin şifrelenmesi ve şifre çözümlenmesinde daha başarılı olduğu, büyük verilerin şifrelenmesinde ve şifre çözümlenmesinde ise AES Algoritması'nın daha başarılı olduğu görülmüştür.

Şifreleme algoritmaları günün şartlarına ve ihtiyaçlarına göre geliştirilebilecek bir bilim dalıdır. Simetrik veya Asimetrik şifreleme algoritmalarının dezavantajlarını ortadan kaldırarak günümüz teknolojisine uygun, daha hızlı ve performansı yüksek şifreleme algoritmaları geliştirilebilir.

KAYNAKLAR

[1] H. Güven, "Covid-19 pandemi krizi sürecinde e-ticarette meydana gelen değişimler," ***Eurasian Journal of Researches in Social and Economics,*** no. ISSN:2148-9963, (2020).

[2] N.H.S. Al-Sarray, "Veri şifreleme tekniklerinin incelenmesi ve uygulanması," *Yüksek Lisans Tezi,* ***Erciyes Üniversitesi Fen Bilimleri Enstitüsü, Kayseri,*** (2018).

[3] H. Al-Sanabani and Ü. Çavuşoğlu, "The performance comparison of lightweight encryption algorithms" ***Sakarya Unıversıty journal of computer and ınformatıon scıences*** vol. 2, no. 3, December (2019).

[4] Ç. Özyılmaz, "Kriptolojiye giriş," *Yüksek Lisans Tezi,* ***Karabük Üniversitesi Fen Bilimleri Enstitüsü,*** *Karabük*, (2014).

[5] M. Aghayev, "Kriptoloji ve veri şifreleme teknikleri üzerine," *Yüksek Lisans Tezi*, ***Ege Üniversitesi Fen Bilimleri Enstitüsü,*** *İzmir* , (2017).

[6] T. Jawahar and K. Nagesh, "DES, AES and blowfish: symmetric key cryptography algorithms simulation based performance analysis," ***International Journal of Emerging Technology and Advanced Engineering***, vol. 1, no. 2, ISSN 2250-2459, (2011).

[7] A. Beşkirli, D. Özdemir, and M. Beşkirli, "Şifreleme yöntemleri ve RSA algoritması üzerine bir inceleme," ***European Journal of Science and Technology***, pp. 284-291, (2019).

[8] Ü. Günden, "Şifreleme algoritmalarının performans analizi," *Yüksek Lisans Tezi,* ***Sakarya Üniversitesi Fen Bilimleri Enstitüsü****, Sakarya* , (2010).

[9] N. Topaloğlu, H.M. Calp, and B. Türk, "Bilgi güvenliği kapsamında yeni bir veri şifreleme algoritması tasarımı ve gerçekleştirilmesi," ***Bilişim Teknolojileri Dergisi,*** vol. 9, no. 3, (2016).

[10] S. Hosseınpour, "Matematiksel ifadelerin üretimi ve çözümüne dayalı bir kriptoloji yöntemin tasarımı ve gerçeklemesi ," *Doktora Tezi,* ***Karadeniz Teknik Üniversitesi Fen Bilimleri Enstitüsü****, Trabzon* , (2018).

[11] C. Aslanyürek, "Şifreleme algoritmalarının hızını etkileyen faktörler," *Yüksek Lisans Tezi, **Trakya Üniversitesi Fen Bilimleri Enstitüsü**, Edirne* , (2018).

[12] N. Sayın, "Dalgacık dönüşümü tabanlı görsel kriptoloji," *Yüksek Lisans Tezi, **Kocaeli Üniversitesi Fen Bilimleri Enstitüsü**, Kocaeli* , (2017).

[13] O. Abdullazada , "Socketler üzerinden özel haberleşmede kriptoloji metotlarının kullanılması ve bir uygulama," *Yüksek Lisans Tezi, **İstanbul Aydın Üniversitesi Fen Bilimleri Enstitüsü**, İstanbul*, (2017).

[14] E. Yeşilbaş, "Cebirsel kriptoloji yöntemleri ve bazı uygulamaları," *Yüksek Lisans Tezi, **Recep Tayyip Erdoğan Üniversitesi Fen Bilimleri Enstitüsü**, Rize* , (2016).

[15] A. Şahin, E. Buluş , and M.T. Saklı, "Modern blok şifreleme algoritmalarının gücünün incelenmesi," ***Bilgisayar Mühendisliği Bölümü, Mühendislik-Mimarlık Fakültesi, Trakya Üniversitesi**, 22100*, (2006).

[16] S. Nasibov, "Kriptoloji sistemleri ve uygulamaları üzerine," *Yüksek Lisans Tezi, **Ege Üniversitesi Fen Bilimleri Enstitüsü**, İzmir* , (2015).

[17] R. Erol, "Kriptoloji kullanımının fonksiyon kavramının anlaşılmasına etkisi," *Yüksek Lisans Tezi, **Hacettepe Üniversitesi Fen Bilimleri Enstitüsü**, Ankara*, (2015).

[18] L. Alizade, "Fermat sayılarının asal çarpanlarına ayrılması ve kriptoloji uygulamaları," *Yüksek Lisans Tezi, **Ege Üniversitesi Fen Bilimleri Enstitüsü**, İzmir* , (2014).

[19] A. Yayık, "Yapay sinir ağı ile kriptoloji uygulaması," *Yüksek Lisans Tezi, **Mustafa Kemal Üniversitesi Fen Bilimleri Enstitüsü**, Hatay*, (2013).

[20] F. Külen, "Kriptolojide bazı şifreleme yöntemlerinde cebirsel yaklaşımlar," *Yüksek Lisans Tezi, **Gaziosmanpaşa Üniversitesi Fen Bilimleri Enstitüsü**, Tokat*, (2013).

[21] İ. Ciğer, "Data şifreleme algoritmaları ve performans analizi," *Yüksek Lisans Tezi, **İstanbul Üniversitesi Fen Bilimleri Enstitüsü**, İstanbul*, (2012).

[22] A.M. Yücelen, "Kriptolojide eliptik eğri algoritması," *Yüksek Lisans Tezi, **Dicle Üniversitesi Fen Bilimleri Enstitüsü,** Kayseri* , (2011).

[23] R. Yılmaz, "Kriptolojik uygulamalarda bazı istatistik testler," *Yüksek Lisans Tezi, **Selçuk Üniversitesi Fen Bilimleri Enstitüsü,** Konya*, (2010).

[24] M.H. Yavuz and O. Ergin, "Verileri nota kullanarak şifreleme ve ses dosyası içerisine gizleme," ***3.Uluslararası katılımlı Bilgi Güvenliği ve Kriptoloji Konferansı****, Ankara,* (2008).

[25] H.N. Buluş, "Temel şifreleme algoritmaları ve kripto analizlerinin incelenmesi," *Yüksek Lisans Tezi,* ***Trakya Üniversitesi Fen Bilimleri Enstitüsü****, Edirne,* (2006).

[26] E. Güvenoğlu, "Görüntü şifreleme algoritmaları ve performans analizleri," *Yüksek Lisans Tezi,* ***Trakya Üniversitesi Fen Bilimleri Enstitüsü****, Edirne,* (2006).

[27] K. Yıldırım, "Veri şifrelemesinde simetrik ve asimetrik anahtarlama algoritmalarının uygulanması," *Yüksek Lisans Tezi,* ***Kocaeli Üniversitesi Fen Bilimleri Enstitüsü,*** *Kocaeli,* (2006).

[28] Y. Çelik, "Nükleobazlar ve nükleositlerde tautomer kararlılığının moleküler modelleme yöntemleriyle belirlenmesi ve mutasyon etkisinin araştırılması," *Yüksek Lisans Tezi,* ***Balıkesir Üniversitesi Fen Bilimleri Enstitüsü,*** *Balıkesir,* (2016).

[29] Z. Obaid, A. Sabonchi, and B. Akay, "Klasik kriptoloji yöntemlerinin karşılaştırılması," ***Engineering Sciences***, no. ISSN: 1308 7231, (2016).

[30] F. Şahin, "Modern blok şifreleme algoritmaları," ***İstanbul Aydın Üniversitesi Dergisi,*** no. 17, pp. 47-60, (2015).

[31] Z. Hercigonja, "Comparative analysis of cryptographic algorithms," ***International Journal Of Dıgıtal Technology & Economy,*** vol. 1, no. 2, (2016).

[32] N. Singhal and J.P.S. Raina, "Comparative analysis of AES and RC4 algorithms for better utilization," ***International Journal of Computer Trends and Technology,*** no. ISSN: 2231-280, p. 178, (2011).

[33] T. Yerlikaya, E. Buluş, and N. Buluş, "Kripto algoritmalarinin gelişimi ve önemi," ***Bilgisayar Müh. Bölümü, Trakya Üniversitesi,*** (2006).

[34] H.M. Kader and M.M. Hadhoud, "Performance evaluation of symmetric encryption algorithms," ***Communications of the IBIMA***, vol. 8, no. ISSN: 1943-7765, (2009).

[35] H. Kodaz and F.M. Botsalı, "Simetrik ve asimetrik şifreleme algoritmalarının karşılaştırılması," ***Selçuk Teknik Dergisi****,* vol. 9, no. 1-2010, ISSN 1302-6178, (2010).

[36] M. Yılmaz and S. Ballı, "Veri şifreleme algoritmalarinin kullanimi için akıllı bir seçim sistemi geliştirilmesi," ***Uluslararası Bilgi Güvenliği Mühendisliği Dergisi,*** vol. 2, no. 2, pp. 18-28, (2016).

[37] S. Tunçer and C. KARAKUZU, "Veri güvenliğini artırmak amacıyla bilgiyi şifreleme ve steganografik yöntemlerle görüntüye gizleme," ***Elektrik Elektronik ve Bilgisayar Sempozyumu****, 11-13 Mayıs*, (2016).

[38] C. Koçak, "Kriptografi ve stenografi yöntemlerini birlikte kullanarak yüksek güvenlikli veri gizleme," ***Erciyes Üniversitesi Fen Bilimleri Enstitüsü Dergisi,*** vol. 31, no. 2, ISSN 1012-2354, pp. 115-123, (2015).

ÖZ GEÇMİŞ

Mustafa ZENGİN, 1973 yılında Samsun'un Çarşamba ilçesinde doğdu; ilkokulu Alan köyünde, Ortaokulu Salıpazarı'ında ve Lise eğitimini Çarşamba Anadolu Lisesi Matematik Bölümü'nden mezun olarak tamamladı. 1993 yılında Azerbaycan Teknik Üniversitesi Bilgisayar Mühendisliği Bölümü'nde bir yıl hazırlık olmak üzere lisans eğitimine başlayıp, 1998 yılında iyi bir dereceyle üniversiteden mezun oldu. Bir çok özel eğitim kurumlarında bilgisayar öğretmeni ve bilgi işlem sorumlusu olarak çalıştı. Çalışma süreleri içinde Avrupa Birliği projeleri hazırlama, Tübitak Bilgisayar Olimpiyatı çalıştırmaları neticesinde çeşitli başarılar elde etti. 2013 yılında İstanbul Ticaret Üniversitesi'nden Pedagojik Formasyonu ve ICF Eğitim Koçluğu Eğitimleri alarak eğitim danışmanlığı yaptı. 2016 yılında Karabük Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği Anabilim Dalı'nda tezsiz yüksek lisansa başladı. 2017 yılında iyi bir ortalamayla aynı alanda tezli yüksek lisans bölümüne geçti. Şu an özel bir firmada yurt içi ve yurt dışı eğitim danışmanlığı yapmakta ve reklam ajansı işletmektedir.

ADRES BİLGİLERİ

Adres : Karadeniz Mahallesi 1128. Sokak Aydın Apartmanı No: 25 Daire:1
Gaziosmanpaşa / İSTANBUL

Tel : (549) 606 07 55

E-posta : mustafazengin@ogrenci.karabuk.edu.tr