**T.C**
**FIRAT ÜNİVERSİTESİ**
**FEN BİLİMLERİ ENSTİTÜSÜ**

# ADLİ BİLİŞİMDE GÖRÜNTÜ ÜZERİNE KRİPTOGRAFİ UYGULAMALARI

**YÜKSEK LİSANS TEZİ**

**Murat AYDOĞAN**

**112131113**

**Anabilim Dalı: Elektronik Bilgisayar Eğitimi**

**Danışman: Doç. Dr. Engin AVCI**

**Tezin Enstitüye Verildiği Tarih: 7 Temmuz 2014**

**TEMMUZ-2014**

# ÖNSÖZ

Bu tez çalışması Veri şifreleme ve veri gizleme uygulamaları üzerine hazırlanmıştır. Günümüzde bilgisayar teknolojilerinin çok sık kullanılmasıyla birlikte bilgi güvenliğine duyulan ihtiyaçta oldukça artmıştır. Bu nedenle veri güvenliğini daha da geliştirmek amacıyla bu tez çalışması yapılmıştır. Bu çalışmada ki amaç veri güvenliği ve veri şifreleme bilimleri kullanılarak yeni yöntemler geliştirmek ve bu metotları da geliştirilen uygulamalar üzerinde kullanmaktır.

Tez çalışmam sırasında benden desteğini ve bilgisini esirgemeyen danışman hocam Sayın Doç. Dr. Engin AVCI' ya, yine çalışmalarım esnasında bana fikir ve destekleriyle katkıda bulunan değerli hocalarım ve mesai arkadaşlarım Öğr. Gör. İzzet TAMER' e, Öğr. Gör. Erkan GÜNERHAN' a, Uzm. Alpay ÖZTÜRK' e ve Arş. Gör. Murat AKBAY' a teşekkür ederim.

Her zaman yanımda olan ve benden desteğini esirgemeyen kıymetli aileme özellikle anneme ve değerli arkadaşlarıma sonsuz teşekkür ederim.

**Murat AYDOĞAN**

**ELAZIĞ – 2014**

## İÇİNDEKİLER

**Sayfa No**

# ÖZET

## Adli Bilişimde Görüntü Üzerine Kriptografi Uygulamaları

Günümüzde bilgisayar teknolojilerinin yaygın olarak kullanılmaya başlanmasıyla birlikte bilgi güvenliğine olan ihtiyaçta önemli derecede artmıştır. Bu tez çalışması güvenlik konusuyla yakından alakalı olan kriptografi ve steganografi tekniklerinden faydalanılarak bilgi güvenliğinin korunmasına katkı sağlamak amacıyla yapılmıştır. Şifre üretme bilimi olarak kabul edilen kriptografi ve veri gizlemenin önemli bir alt dalı olan steganografi konularının ele alındığı bu tezde, öncelikle kriptografinin temel kavramları, önemli bileşenleri ve belli başlı kriptosistemlerin yapıları incelenmiş daha sonra steganografi konusu açıklanmıştır.

Literatürde genellikle bilgi güvenliğinin sağlanması için birbirlerinin alternatifi olarak gösterilen veri şifreleme ve veri gizleme konularından bu tez çalışmasının en önemli kısmı olan uygulamalar bölümünde yararlanılmıştır. Geliştirilmiş olan iki uygulama da bu kavramlar birlikte kullanılmış ancak bu kez alternatif olarak değil birbirlerinin tamamlayıcısı olmuşlardır. Güvenliğin artırılmasına önemli bir katkı sağlayan bu yöntem dışında, tez içerisinde yer verilen iki uygulamada da hem veri gizleme hem veri şifreleme açısından yeni teknik ve metotlar geliştirilerek Visual C# programla dili ile kodlanmıştır. Geliştirilen uygulamalar sonucunda sayısal resim içerisine gizlenen verilerin her defasında farklı ve rasgele noktalara gizlendiği ardından gizlenmiş verilerin deşifre edilerek orijinal verilere ulaşıldığı görülmüştür.



**Anahtar Kelimeler**: Kriptografi, Veri Şifreleme, Steganografi, Veri Gizleme

**SUMMARY**

**Cryptography Applications on the image in Digital Forensics**

In the nowadays, with using of computer technology excessively, the necessity of secure information is increasing importantly day by day. This thesis is about cryptography and steganography which are generally related to security of information. In this thesis, the subjects of the cryptography which is known as a code producing science and steganography which is the important branch of hiding data are mentioned. Firstly, The basic concepts, the significant components and some forms of cryptosystem are analyzed. Then, stenography is explained shortly.

In the most important part of this thesis, in the practice part, It is benefited from the subjects of coding and hiding data which are generally mentioned alternative to each other for providing secure information in the literature. In these developed technics, for this time these concepts are used together for completing each other, not being alternative. In this thesis, except for this technic, it has important contribution that security of data, both of these technics are coded with visual C# for developing new methods from the aspects of data encryption and data hiding. Applications developed as a result of hidden data into digital image every time different and random points are hidden and then hidden data have been decrypted and original data is reached.

**Keywords**: Cryptography, Data Encryption, Steganography, Data Hiding

## ŞEKİLLER LİSTESİ

## TABLOLAR LİSTESİ

**Sayfa No**

## KISALTMALAR LİSTESİ

**AES** : Advanced Encryption Standart

**DES** : Data Encryption Standart

**RSA** : Rivest- Shamir- Adleman

**ASCII** : The American Standard Code for Information Interchange

**BMP** : Bitmap

**CMYK** : Cyan, Magenta, Yellow, Black

**GIF** : Grafik Değiştirme Biçimi

**RGB** : Red Green Blue

**LSB** : Least Significant Bit Insertion

# 1. GİRİŞ

Güvenlik, insanoğlu için geçmişten günümüze kadar her zaman en önemli gereksinimlerinden biri olmuştur. Özellikle günümüzde bu ihtiyaç gittikçe artmış bilim dünyasının da bu konuyla yakından ilgilenmesini sağlamıştır.

Son yıllarda gelişen bilgisayar teknolojileriyle birlikte bilgisayar sistemlerinin güvenliği özellikle veri güvenliği oldukça önemli bir hale gelmiştir. Özellikle internet dünyasının gelişmesine paralel olarak kullanıcıların ihtiyaçları ve bilgi paylaşımı oldukça artmıştır. Kullanıcılar, internet sayesinde istedikleri anda zamandan ve mekândan bağımsız olarak alışveriş yapabilme, dünyanın birçok yerinden insanlarla haberleşme, çeşitli türden verileri birbirleriyle paylaşabilme olanaklarına sahip olmuşlardır.

Kullanıcıların hayatlarını kolaylaştıran ve bu kadar popüler olup birçok kişi tarafından kullanılan ortak bir platformun olması tabi ki güvenlik açıklarının ortaya çıkmasına neden olmuştur. Bu zafiyette illegal yoldan çıkar sağlamak isteyen art niyetli kişiler içinde oldukça cazip bir hale gelmiştir.

Birbirleriyle ortak bir ağ üzerinden iletişim sağlayan iki kullanıcı arasında ki iletişim yetkisi olmadığı halde bir üçüncü kişi tarafından erişilebilir hatta bu ortamda ki bilgiler değiştirilebilir hale gelmiştir. Bu nedenle günümüzde de oldukça popüler kavramlar olan Kriptoloji ve Steganografi diğer bir deyişle veri şifreleme ve veri gizleme kavramları ortaya çıkmış ilerleyen zamanlarda da oldukça gelişme göstermişlerdir.

Bu tez çalışmasının ana hattını oluşturan iki uygulama geliştirilmiştir. Daha çok bu uygulamalar ele alınacağı için geliştirilen uygulamalarda kullanılan yöntem ve metotlar her ne kadar özgün olsa da temel olarak veri şifreleme ve veri gizleme prensiplerine dayandığı için bu tez çalışmasında Kriptoloji, Steganografi, Kriptosistem konularına yer verilmiştir.

Tezin birinci bölümünde giriş kısmına yer verilmiş, ikinci bölümde ise tezin ilerleyen bölümlerinde yer alan uygulamalarda veri şifrelemesi konularına yer verilip özgün algoritmalar paylaşıldığı için bu bölümde kriptoloji kavramına, amacına, tanımı ve tarihçesine ve alt bileşenlerine yer verilmiş kısaca bahsedilmiştir. Geliştirilen uygulamalarda şifre üretme bilimi olan kriptografi kavramı daha çok kullanılmış olsa da kriptolojinin bir bileşeni olarak kabul edildiği için bu bölümde birlikte yer verilmiştir.

Tezin üçüncü bölümünde ise kriptosistemler konusu ele alınmıştır. Yapılan uygulamada RSA şifreleme algoritması kullanıldığı için bu bölümde de kriptosistemlerden

bahsedilmiştir. Simetrik ve Asimetrik kriptosistemler açıklanmış, avantaj ve dezavantajları birbirlerine karşı üstünlükleri ve en önemli algoritmalar irdelenmiştir.

Tez çalışmasının dördüncü bölümünde ise veri gizlemenin önemli bir alt dalı olan steganografiden bahsedilmiştir. Steganografi konusunun tanımı ve tarihi incelenmiş alt bileşenleri açıklanmış, beşinci bölümde de veri gizleme yöntemleri kısaca ele alınmıştır.

Beşinci bölümde ise tez çalışmasının en önemli kısmı olan uygulamalar yer almaktadır. İki adet uygulamanın yer aldığı bu bölümde; İlk olarak diş hekimliği alanında kullanılan ve OPT adı verilen radyografik resimler üzerine hasta bilgileri, tedavide kullanılacak teşhis ve tanıların kriptografi yöntemiyle şifrelenerek güvenlik zafiyetinin giderilmesi ardından da şifrelenmiş verilerin OPT resimlerinin üzerine gömülerek gizlenmesi böylece saklama ve kaybolma problemlerinin ortadan kaldırılması amacıyla bir uygulama geliştirilmiştir. Uygulama sırasında veri şifreleme ve gizleme basamaklarında özgün yöntemler kullanılmıştır. İkinci uygulama olarak, kimlik sahteciliğinin günümüzde ulaştığı boyutlar ve yarattığı mağduriyet göz önüne alındığında bu alanda bir uygulama yapma ihtiyacı hissedilmiş ve sahte kimlik kullanımını tespit eden bir uygulama geliştirilmiştir. Beşinci bölümde bu uygulamaların çalışma ilke ve prensiplerinden, önerilen yöntemlerden ve elde edilen sonuçlardan bahsedilmiştir.

Altıncı ve son bölümde ise tez sonunda elde edilen sonuçlar paylaşılmış, geliştirilen uygulamalarda alınan sonuçlara yer verilmiştir.

# 2. KRİPTOLOJİ

## 2.1 Kriptolojinin Tanımı ve Tarihçesi

Günümüz teknolojisinin hiç durmadan hızlı bir şekilde ilerleyişi göz önüne alındığında, ortaya çıkan güvenlik açığının ne kadar önem arz ettiği görülmektedir. Kriptoloji, şahıslar arası veya devlet kurumları arasındaki haberleşmelerden, mesajlaşmalardan ve bunların oluşumunun her aşamasında güvenlik boşluklarını dolduran önemli bir daldır ve bu bilim dalı geçmişten bu yana farklı yöntemlerle kullanılmaktadır.

Kriptoloji terimi, Yunan dilinde *'kryptos'* (saklı – gizli) ve *'logos'* (sözcük) kelimelerinden türetilmiştir. Bundan dolayı kriptoloji terimi, dilimizde 'gizli sözcük' olarak açıklanmıştır. Bu tanıma göre, kriptolojideki temel amacın, belirli sözcüklerin anlamını gizlemek, sözcüklerin güvenliğini sağlamak, gizliliğini korumak olduğu bilinmektedir [1].

Kriptolojinin özellikle matematik biliminde yer aldığı ve ayrıca çalışma alanının kriptografiyi ve kriptoanalizi kapsadığı yapılan araştırmalarla ortaya konulmuştur. Bu araştırma sonuçlarına göre, kriptolojinin iki ana kolu olan kriptografi ve kriptoanaliz başlıklarının anlamlarını açmak gerekmektedir [1].

Kriptografi (şifreleme) işlemi, veriyi anlamsız hale getirme-dönüştürme, veri üzerinde saptanamayan değişimi engelleme ya da verinin yetkisiz kişilerin kullanımından koruma maksatlı yapılmaktadır. Bu haliyle kriptografi, verinin şifrelenmesi ve sonrasında tekrardan orijinal haline dönüştürülmesi konusuyla ilgilenmiştir [1-2].

Kriptoanaliz, Yunan dilinde 'krypto' (saklı – gizli) ve 'analyein' (çözme) kelimelerinden oluşturulmuştur. Bu yönüyle kriptoanaliz, saklı bilginin ya da şifrelenmiş verinin çözülmesi anlamına gelmektedir. En genel manada deşifreleme işlemi olarak tanımlanmaktadır. Ayrıca kriptoanaliz (deşifreleme), şifreleme tekniklerinin ve sistemlerinin kırılması olasılıklarıyla ve veri güvenliği konularında yoğunlaşmaktadır [1-3].

Kriptografinin tarihte görüldüğü ilk belirgin örnek olarak Julius Caesar'ın MÖ 60-50 yıllarında roma alfabesindeki harflerin yerini değiştirerek oluşturduğu şifreleme yöntemini devlet haberleşmesinde kullanması örnek gösterilir. Bu yöntem şifrelenecek olan metindeki her harfin alfabede kendisinden 3 sonraki harfle değiştirilmesine dayanmaktadır [4].

Kriptolojinin bundan sonraki ilerlemesini özetlersek [4-5] :

- 1586, Blaise de Vigenère (1523-1596) şifreleme hakkında bir kitap yazdı. İlk kez bu kitapta açık metin ve şifreli metin için otomatik anahtarlama yönteminden bahsedildi [6].
- 1623'de Sir Francis Bacon, 5-bit ikili kodlamayla karakter tipi değişikliğine dayanan stenografı buldu.
- 1920 ve 1930'larda Amerika'da Federal Bureau of Investigation (FBI) içki kaçakçılarının haberleşmesini çözebilmek için bir araştırma ofisi kurdu.
- Bir başka önemli gelişme ise yine aynı devirde Almanların Scherbius tarafından icat edilmiş olan Enigma makinesini kullanmasıydı. Bu makine ile gerçekleştirilen Alman ordusuna ait şifreleme sistemi, İngiliz Alan Turing ve ekibi tarafından kırıldı ve bu gelişme savaşta İngilizlere üstünlük tanıyan anahtar bir faktör oldu.
- 1970'lerde Horst Feistel, Data Encryption Standard'ın (DES) temelini oluşturan Lucifer algoritmasını geliştirdi [7].
- 1976'da DES, Amerika Birleşik Devletleri (ABD) tarafından Federal Information Processing Standard 46 (FIPS 46) standardı olarak açıklandı.
- 1976 Whitfield Diffie ve Martin Hellman, Açık Anahtar sistemini anlattıkları makaleyi yayınladılar.
- 1978'de Ronald L. Rivest, Adi Shamir ve Leonard M. Adleman, Rivest - Shamir-Adleman (RSA) algoritmasını buldular [8].
- 1990'da Xuejia Lai ve James Massey, International Data Encryption Algorithm (IDEA) algoritmasını buldular.
- 1991'de Phil Zimmerman, Pretty Good Privacy (PGP) sistemini geliştirdi ve yayınladı.
- 1992'de Heisenberg, belirsizlik kuralına dayanan Kuantum Kriptografi araştırması yaptı. İlerleyen yıllarda ise Kuantum kriptografik sistemler tasarlanmaya çalışıldı [10].
- 1995'de Secure Hash Algorithm (SHA-1) özet algortiması, ABD Ulusal Teknoloji ve Standartları Enstitüsü (NIST) tarafından standart olarak yayınlandı [11].
- 1997'de ABD'nin NIST kurumu DES'in yerini alacak bir simetrik algoritma için yarışma açtı.

- 2001'de NIST'in yarışmasını kazanan Belçikalı Joan Daemen ve Vincent Rijmen'e ait Rijndael algoritması, Advanced Encryption Standard (AES) adıyla standart haline getirildi. [2-4-11]

Türkiye"de kriptolojiden bahsedecek olursak, Orta Doğu Teknik Üniversitesinde yer alan ve kuruluşu 2002 yılına dayanan Uygulamalı Matematik Enstitüsüne bağlı Kriptografi anabilim dalı bugün bu alanda yüksek lisans ve doktora dereceleri veren bir kurum olarak göze çarpmaktadır. Mezunların çalışabileceği kurumlar arasında NASA'nın Türkiye karşılığı olarak düşünebileceğimiz Ulusal Elektronik ve Kriptoloji Araştırma Enstitüsü (UEKAE), Milli İstihbarat Teşkilatı (MIT), elektronik imza şirketleri ve üniversiteler ile çeşitli araştırma enstitüleri yer almaktadır. Benzer motivasyonla, Atılım Üniversitesi Matematik Bölümü'nde, matematiksel içeriği ağırlıklı olan bir Kriptografi Sertifika Programı başlatılmıştır.

## 2.2 Kriptolojinin Amacı

Günümüzde kriptografik dönüşüm (şifreleme) , bilgisayar ağları ve haberleşme sistemlerinde, birçok farklı veri koruma problemlerinin çözümünde en etkin ve yaygın bir metot olarak kullanılmaktadır. Genel anlamda şifrelemenin üç temel hedefi bulunmaktadır [12]:

- Veri güvenliğinin sağlanması,
- Mesaj kaynağının ve bilginin doğruluğunun tespiti,
- Kullanıcının gerçek isminin saklanması olarak açıklanmaktadır.

Sistemler arası bağlantılarda ya da herhangi iki nokta arasındaki haberleşmede verinin güvenli bir şekilde gittiğinden emin olmak gerekir. Bunun güvenilir bir şekilde sağlanması ise gönderilen verinin şifrelenmesi ile olur. Böylece açık haberleşme kanalları kullanılarak verinin güvenli bir şekilde ulaştırılması sağlanır. İletişimde, açık bir haberleşme kanalı kullanılıyorsa gizli tutulmak istenen bilginin yetkisiz bir kişi tarafından dinlenebileceği veya haberleşme kanalına girip (araya girme) veriyi bozabileceği ya da değiştirebileceği (yanlış verinin gönderilmesi) düşüncesi her zaman için önemli bir problem oluşturur [13].

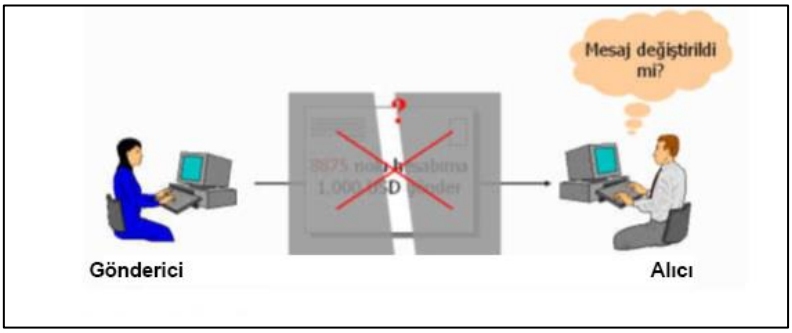
## 2.3. Modern Kriptolojide Güvenlik Prensipleri

### 2.3.1. Gizlilik

Bilginin istenmeyen kişilerin eline geçmesinin engellenmesidir. Gizlilik, hem kalıcı ortamlarda (disk, vb.) saklı bulunan veriler, hem de ağ üzerinde bir göndericiden bir alıcıya gönderilen veriler için söz konusudur. Saldırganlar, yetkileri olmayan verilere birçok yolla erişebilirler: Parola dosyalarının çalınması, sosyal mühendislik, bilgisayar basında çalışan bir kullanıcının, ona fark ettirmeden özel bir bilgisini ele geçirmesidir. Bunun yanında trafik analizinin, yani hangi gönderici ile hangi alıcı arası haberleşmenin olduğunun belirlenmesine karsı alınan önlemler de gizlilik hizmeti çerçevesinde değerlendirilir [2-14].

### 2.3.2. Veri Bütünlüğü

Veri bütünlüğünün amacı, veriyi göndericiden çıktığı haliyle alıcısına güvenli bir şekilde ulaştırmaktır. Bu haliyle veri, haberleşme sırasında izlediği yollarda değiştirilmemiş, araya yeni veriler eklenmemiş, belli bir kısmı ya da tamamı tekrar edilmemiş ve sırası değiştirilmemiş şekilde alıcısına ulaşır. Veri bütünlüğünü, geri dönüşümü olan ve olmayan şekilde verebiliriz; alıcıda iki tür bütünlük sınaması yapılabilir: Bozulma sınaması ya da düzeltme sınaması. Bozulma sınaması ile verinin göndericiden alıcıya ulaştırılması sırasında değiştirilip değiştirilmediğinin sezilmesi hedeflenmiştir. Düzeltme sınamasında ise, bozulma sınamasına ek olarak eğer veride değişiklik sezildiyse bunu göndericiden çıktığı haline döndürmek hedeflenmektedir (Şekil 2.1.) [2-14].




**Şekil 2.1.** Veri bütünlüğü

### 2.3.3. Süreklilik

Süreklilik hizmeti, bilişim sistemlerini, kurum içinden ve dışından gelebilecek başarım düşürücü tehditlere karsı korumayı hedefler. Süreklilik hizmeti sayesinde, kullanıcılar, erişim yetkileri dâhilinde olan verilere, veri tazeliğini yitirmeden, zamanında ve güvenilir bir şekilde ulaşabilirler.

Sistem sürekliliği, yalnızca kötü amaçlı bir Hacker'in, sistem başarımını düşürmeye yönelik bir saldırısı sonucu zedelenmez. Bilgisayar yazılımlarındaki hatalar, sistemin yanlış, bilinçsiz ve eğitimsiz personel tarafından kullanılması, ortam şartlarındaki değişimler (nem, ısı, yıldırım düşmesi, topraklama eksikliği) gibi faktörler de sistem sürekliliğini etkileyebilir [2-14].

### 2.3.4. İzlenebilirlik

İzlenebilirliğin başlıca amacı, verileri daha sonra analiz edebilmek üzere kayıt altına almaktır. Burada olay dendiğinde, bilgisayar sistemi ya da ağı üzerinde olan herhangi bir faaliyeti anlayabiliriz. Bir sistemde olabilecek olaylara, kullanıcının parolasını yazarak sisteme girmesi, bir web sayfasına bağlanmak, e-posta almak göndermek ya da elektronik ortam üzerinden mesaj yollamak gibi örnekler verilebilir. Toplanan olay kayıtları üzerinde yapılacak analiz sonucunda, bilinen saldırı türlerinin örüntülerine rastlanırsa ya da bulanık mantık kullanılarak önce rastlanmayan ve saldırı olasılığı yüksek bir aktivite tespit edilirse alarm mesajları üretilerek sistem yöneticileri uyarılır [2-14].

### 2.3.5. Kimlik Sınaması

Ağ güvenliği açısından kimlik sınaması; alıcının, göndericinin iddia ettiği kişi olduğundan emin olmasıdır. Bunun yanı sıra, bir bilgisayar programını kullanırken bir parola girmek de kimlik sınaması çerçevesinde değerlendirilebilir. Günümüzde kimlik sınaması, sadece bilgisayar ağları ve sistemleri için değil, fiziksel sistemler için de çok önemli bir hizmet haline gelmiştir. Akıllı karta ya da biyometrik teknolojilere dayalı kimlik sınama istemleri yaygın olarak kullanmaya başlanmıştır [2-14].

### 2.3.6. Güvenilirlik

Sistemin beklenen davranışı ile elde edilen sonuçlar arasındaki tutarlılık durumudur. Başka bir anlamda güvenilirlik, sistemden ne yapmasını bekliyorsak, sistemin de eksiksiz ve fazlasız olarak bunu yapması ve her çalıştırıldığında da aynı şekilde davranması olarak tanımlanabilir [2-14].

### 2.3.7. İnkâr Edememe

Bu hizmet sayesinde, ne gönderici alıcıya bir mesajı gönderdiğini ne de alıcı göndericiden bir mesajı aldığını inkâr edebilir. Bu hizmet, özellikle gerçek zamanlı işlem gerektiren finansal sistemlerde kullanım alanı bulmaktadır ve gönderici ile alıcı arasında ortaya çıkabilecek anlaşmazlıkların en aza indirilmesini sağlamaya yardımcı olmaktadır. Bu hizmetler, zaman içinde bilgisayar sistemlerine karsı ortaya çıkmış tehditler ve yasanmış olaylar sonucunda ortaya konmuştur. Yani her bir hizmet, belli bir grup potansiyel tehdide karsı sistemi korumaya yöneliktir, denilebilir [2-14].

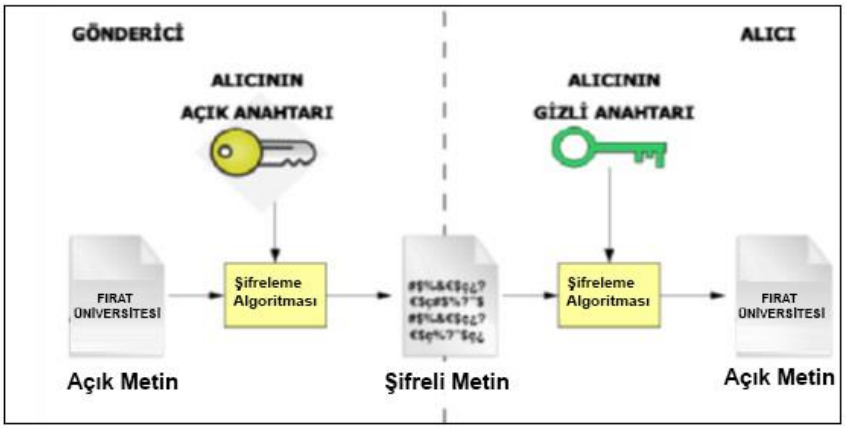
# 3. KRİPTOSİSTEMLER

Kriptosistemler temel olarak simetrik ve asimetrik kriptosistemler olmak üzere iki ana başlıkta incelenir.

## 3.1 Açık Anahtarlı (Asimetrik) Sistemler

Gizli anahtarlı kriptosistemlerinin aksine açık anahtarlı kriptosistemlerinin kullanımı henüz çok yenidir. Açık anahtarlı kriptosistemleri üzerine ilk öneri,1976 yılında Diffie ve Hellman tarafından yapılmıştır. Ardından 1977 yılında Rivest, Shamir ve Adleman RSA Kriptosistemi adlı yeni bir açık anahtarlı kriptosistemini bulmuşlardır. 1978 yılından beri kripto dünyasına değişik teklifler yapıla gelmiştir. Bunlardan en önemlileri El-Gamal tarafından tasarlanan El-Gamal açık anahtarlı kripto algoritması ve eliptik eğri açık anahtarlı kriptosistemleridir. Temelde açık anahtarlı kriptosistemlerinin gayesi belli bir anahtar üzerinde anlaşmanın ve karşı tarafa bu anahtarı güvenli olarak ulaştırabilmenin zorluğunu ortadan kaldırmaktır. Burada tek yönlü bir mesajlaşma söz konusudur. Mesaj alıcısı sadece kendisinin bileceği gizli anahtar ve diğer kişilere dağıtabileceği bir açık anahtardan oluşan anahtar çifti belirler. Kullanılan anahtar üretim algoritmasına göre bu iki anahtar arasında matematiksel bir bağlantı mutlaka olabilecektir fakat asıl amaç, bilinen açık anahtardan gizli anahtarın hesaplanmasının polinomsal zamanda imkânsız olabilmesidir [15].

Simetrik şifreleme tekniğinde anahtar dağıtım problemi vardır. Bu sorunu gidermek için şifreleme ve deşifreleme işlemlerinde ayrı ayrı anahtar kullanılır. Bu sistemde, şifreleme işlemi herkes tarafından bilinen açık anahtarla yapılır. Şifreleme ve çözme işlemi birbirinin simetriği olmayan (yani aynısı olan) algoritmalarla yapıldığından dolayı asimetrik şifreleme sistemi olarak bilinir [16-17].




**Şekil 3.1.** Asimetrik Şifreleme modeli

## 3.2. Açık Anahtarlı Sistemlerin Avantajları

Bilgisayar bilim ve teknolojisinin eristiği yüksek düzey göz önüne alındığında, simetrik kriptosistemlerin mutlak biçimde korumak zorunda oldukları anahtarların koruma ve dağıtım maliyetinin ne kadar yüksek ve koruma işleminin ne kadar zor olduğu kolayca görülebilir. Sırf bu nedenden ötürü, karşılıklı haberleşme içinde olan iki tarafın güvenli dağıtım kanalları oluşturması özellikle güncel bankacılık sisteminde, yaygın görülen bir örnektir [18].

Öte yandan, şifreleme ve deşifreleme dönüşüm fonksiyonlarının kullandıkları anahtarlar birbirinden ayrılarak anahtar güvenliği sorunu kesin biçimde çözülebilir. Asimetrik kriptosistemlerin tüm güvenliği deşifre anahtarının sadece yetkili alıcı tarafından bilinmesinde yatar. Öte yandan, her ne kadar her iki anahtar birbirinden farklıysa da şifreleme anahtarından gidilerek deşifre anahtarını oluşturmak, teorik olarak olası, ancak pratikte çözümsüz bir problemdir [18].

Asimetrik kriptosistemlerin, simetrik kriptosistemlere göre önemli yararı da anahtar yönetimidir. Gerçekçi olunursa, n kişinin özel anahtar şifrelemesini kullanması durumunda grup içerisinde her kişi için bir farklı özel anahtar ihtiyacı olmaktadır. Böylece n(n–1) adet anahtar yönetimi olacaktır. Eğer n binlerce kullanıcı olursa o zaman milyonlarca anahtar yönetimi söz konusudur. Bundan başka gruba yeni bir kullanıcı eklemesi kolay bir iş olmamaktadır. Yeni kullanıcının gruptaki herkesle iletişim kurabilmesi için n adet yeni anahtar yönetimi söz konusu olacaktır. Daha sonra, yeni anahtarların gruba yollanması gerekmektedir. Aksine, asimetrik sistemlerde, kullanıcıların n adet genel anahtarları genel bir

dizinde tutulur, yeni bir kullanıcı eklemesinde kullanıcının genel anahtarı dizine eklenmesi yeterli olacaktır [18].

Asimetrik kriptosistemlerin en büyük avantajı, veri şifreleme ve deşifreleme işlemlerini yapmasının yanında dijital imza gerçekleştirilmede kullanılmasıdır. Aldığımız bir şifreli metinde deşifreme işlemini gerçekleştirdikten sonra karşılaşacağımız en büyük problem bu şifreli metnin bize doğru kişiden gelip gelmediğidir. Bunu da dijital imzalar sayesinde doğrulamaktayız.

Dijital imza sayesinde bir otorite dokümanların doğruluğu denetler. İmzalama, doküman ile otoritenin girişinin belirli işlemleri ile oluşan bir kaç bitin veya MD5 (Mesaj Özetleme Algoritması 5 - Message Digest Algorithm 5) ile SHA benzeri karıştırma algoritmalarının kullanımıyla oluşan bitlerin dokümana eklenmesiyle elde edilir. Dokümana erişim yetkisi olan herhangi bir kişi, imzanın gerçekten otorite tarafından atıldığını doğrulayabilir. Bu iş için imza şemaları kullanılır. İmza semalarının en ünlüsü El-Gamal imza şemasıdır [19].

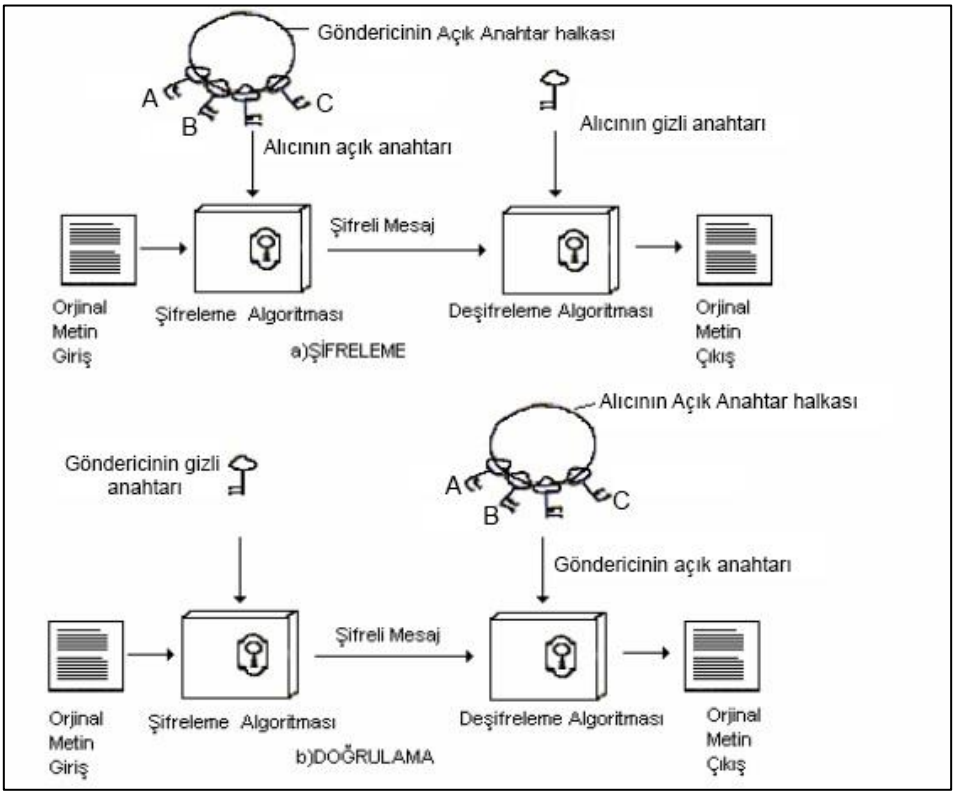
Açık anahtarlı kriptografide şifreleme işlemleri şekil 3.2 ‘de örnek olarak gösterilmiştir. Bu basamakları incelersek [20]:

**1)** Her kullanıcı şifreleme ve deşifreleme işlemleri için bir çift anahtar üretir.

**2)**Her kullanıcı şifreleme için kullanılan anahtarını, herkesçe erişilebilecek bir dosyaya kaydederek açık anahtarını yayınlar. Eş anahtarı özel olarak saklanır. Buda deşifreleme işleminde kullanılan özel anahtardır.

**3)** Eğer A, B’ye mesaj yollamak istiyorsa, B’nin açık anahtarını kullanarak mesajı şifreler.

**4)** B mesajı kabul ettiği zaman kendi özel anahtarını kullanarak onu deşifreleyecektir. B dışında hiçbir alıcı mesajı deşifreleyemez. Çünkü B’nin özel anahtarına yalnızca B sahiptir.




**Şekil 3.2**. Açık Anahtar Şifreleme- Doğrulama

Örnekten de anlaşıldığı üzere tüm kullanıcılar açık anahtarlara sahiptir ve onları kullanabilir. Fakat özel anahtar yalnızca sahibi tarafından kullanılır. Bundan dolayı dağıtılmaya ihtiyacı yoktur. Kullanıcılar kendi özel anahtarlarını kontrol ettiği sürece iletişim güvenlidir. Bir kullanıcı istediği zaman özel anahtarını değiştirebilir ve açık anahtarını yayınlayabilir [20].

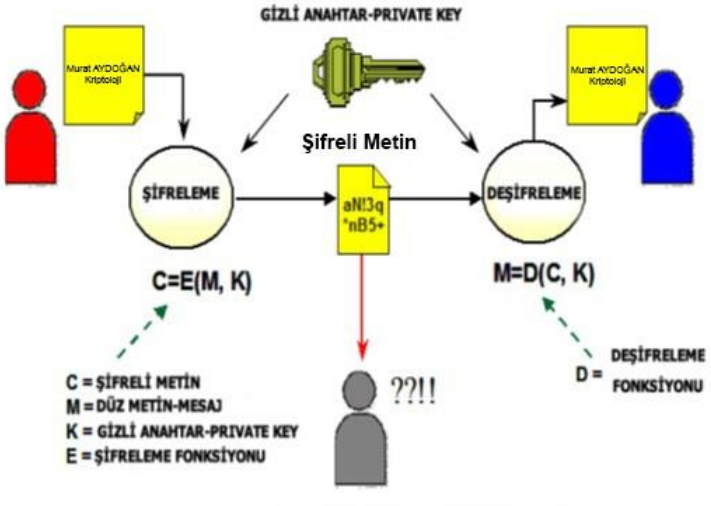
### 3.3. Gizli Anahtarlı (Simetrik) Sistemler

Simetrik algoritmalara bazen geleneksel algoritmalar da denir. Şifreleme anahtarının çözümleme anahtarı üzerinden hesaplanabildiği bir algoritmadır. Çoğu simetrik algoritmada şifreleme ve çözümleme anahtarları aynıdır. Bu algoritmalar, aynı zamanda gizli anahtar algoritmaları, tek anahtar algoritmaları veya bir (aynı) anahtar algoritmaları diye de adlandırılır, gönderici ile alıcının güvenle iletişime başlamadan önce bir anahtar üzerinde anlaşmalarını gerektirir. Bir simetrik algoritmanın güvenliği anahtara dayanır; anahtarın açığa çıkması herkesin mesajları şifreleyebileceği ve çözebileceği anlamına gelir. İletişimin gizli kalması gerektiği sürece, anahtar gizli kalmalıdır [18].

Simetrik kriptosistemlerde, şifreleme anahtarının deşifreleme anahtarından üretilmesi oldukça basittir. Çoğu simetrik algoritmalarda, şifreleme ve deşifreleme için kullanılan anahtar aynıdır. Ancak, bu algoritmaların güvenliği anahtara bağlı olduğundan anahtar gizli tutulmalıdır [18].

Simetrik algoritmalar iki sınıfa ayrılabilir. Bazıları belli bir anda bir bitlik acık metni şifreleyebilir; bunlar akış (stream) algoritmaları veya akış şifreleri olarak adlandırılır. Diğerleri acık-metini bit gruplarına bölerek işler. Bit grupları blok ve algoritmaları blok algoritmaları veya blok şifreleri diye adlandırılır. Çağdaş bilgisayar algoritmaları için, tipik bir bit bloğunun uzunluğu 64 bittir.

Simetrik şifrelemede, şifreleme ve deşifreleme işlemleri gizli bir anahtarla yapılır. Şifreleme işlemleri bittikten sonra şifreli metni (cipher text) alıcıya gönderirken ayrıca gizli anahtarın da güvenli bir şekilde gönderilmesi gerekir. Simetrik şifreleme algoritmaları çok hızlıdır. DES ve AES şifreleme sistemleri bu algoritmalara örnek olarak verilir [16-21, 22].




**Şekil 3.3**. Simetrik Şifreleme modeli

Simetrik algoritmaların avantajları [23];

1. Algoritmalar olabildiğince hızlıdır.
2. Donanımla birlikte kullanılır.
3. Güvenlidir.

Simetrik algoritmaların dezavantajları [23];

1. Güvenli anahtar dağıtımı zordur.
2. Kapasite sorunu vardır.
3. Kimlik doğrulama ve bütünlük ilkeleri hizmetlerini güvenli bir şekilde gerçekleştirmek zordur.

### 3.4. Açık ve Gizli Anahtarlı Kriptosistemlerin Karşılaştırılması

İki kriptosistemin karşılaştırılması genel hatlarıyla şöyledir [2]:

**1.** Simetrik anahtar kriptolama genellikle çok yüksek veri taşıma oranı düşünülerek tasarlanır. Bazı donanımsal uygulamalar saniyede yüzlerce megabayt veriyi kriptolamayı başarabilirken, yazılımsal uygulamalarda saniyede megabaytlar düzeyinde gerçekleşir. Açık anahtar kriptolamada (asimetrik kriptolama) işlemler simetrik anahtar kriptolamaya oranla daha düşük hızda gerçekleşir.

**2.** Simetrik anahtar kriptolamada anahtar uzunluğu, acık anahtar kriptolamaya göre nispeten kısadır.

**3.** Simetrik anahtar kriptolamada, iki farklı taraf arasında gerçekleşen iletişim için anahtarı her iki tarafında bilmesi ve gizli tutması zorunlu iken, açık anahtar kriptolamada tarafların sadece kendilerine ait özel anahtarı gizli tutmaları yeterlidir.

**4.** Simetrik anahtar kriptolamada anahtarın güvenlik acısından sık sık değiştirilmesi gerekirken, açık anahtar kriptolamada özel/genel anahtar çiftinin uzun süreler boyunca değiştirilmesine gerek duyulmaz.

**5.** Çoğu açık anahtar yapılarında, simetrik anahtar yapılarına göre oldukça verimli dijital imza mekanizmaları elde edilir.

**Tablo 3.1**. Simetrik ve Asimetrik Kriptosistemlerin Karşılaştırılması [2].

| KRİTER | KAPALI ANAHTARLI SİSTEMLER | AÇIK ANAHTARLI SİSTEMLER |
|---|---|---|
| **GİZLİLİK** | Sağlar | Sağlar |
| **VERİ BÜTÜNLÜĞÜ** | Sağlar | Sağlar |
| **KİMLİK SINAMASI** | - | Sağlar |
| **İNKÂR EDEMEME** | - | Sağlar |
| **HESPLAMA HIZI** | Yüksek | Düşük |
| **GÜVENİRLİK** | Anahtar gücüne göre | Anahtar gücüne göre |
| **DEĞERLENDİRME** | Mesaj şifrelemede tercih edildiğinde başarım oranı yüksektir. | Anahtar şifrelemede tercih edildiğinde başarım oranı yüksektir. |

**Tablo 3.2.** Geleneksel ve Açık anahtarlı Kriptografi [20].

| **GELENEKSEL KRİPTOGRAFİ** | **AÇIK ANAHTARLI KRİPTOGRAFİ** |
|---|---|
| **Çalışması için ihtiyaçları:**<br>1- Şifreleme için kullanılan anahtar ve algoritma aynı zamanda deşifreleme içinde kullanılır.<br>2- Gönderen ve alıcı, algoritma ve anahtarı paylaşmalılar. | **Çalışması için ihtiyaçları:**<br>1-Bir algoritma, deşifreleme için 1, Şifreleme için 1 olmak üzere; deşifreleme ve şifreleme için 1 çift anahtar kullanır.<br>2- Gönderici ve alıcıdan her birisi, eşlenen çift anahtarların birine sahip olmalıdır. |

| | |
|---|---|
| **Güvenlik için ihtiyaçları:**<br>1-Anahtar gizli tutulmalı.<br>2-Eğer diğer bilgi mevcut olmazsa mesajı çözmek olanaksız ve ya en azından elverişsiz olmalı.<br>3-Algoritmanın bilgisi ve cipher text 'in örnekleri anahtarı belirlemek için yetersiz olmalı. | **Güvenlik için ihtiyaçları:**<br>1-İki anahtardan biri elde gizli tutulmalıdır.<br>2- Eğer diğer bilgi mevcut olmazsa mesajı çözmek olanaksız ve ya en azından elverişsiz olmalı.<br>3-Anahtarlar ve ciphertextin örneklerinin biri ve algoritmanın bilgisi diğer anahtarı belirlemek için yetersiz olmalı. |

## 3.5. En Çok Kullanılan Açık Anahtarlı (Asimetrik) Şifreleme Modelleri

### 3.5.1 Diffie-Hellman Anahtar Değişimi

Diffie-Hellman 1976 de yayınladıkları "New Directions In Cryptography" adlı makale ile anahtar paylaşımı problemine ilk pratik çözümü yayınlamışlardır. Ayrık logaritma probleminin çözümünün zorluğuna dayanan bu sistem ile tarafların daha önce bir araya gelmesine gerek duyulmadan açık bir kanal üzerinden mesajlarını birbirlerine göndererek ortak anahtar oluşturmaları sağlanmıştır [21].

Bu algoritmanın amacı, iki kullanıcının bir anahtarı güvenli şekilde birbirlerine iletmeleri ve daha sonrasında da bu anahtar yardımı ile şifreli mesajları birbirlerine gönderebilmelerini sağlamaktır. Algoritma anahtar değişimi ile sınırlıdır [21]. A ve B kişileri aşağıdaki yolu izleyerek ortak bir anahtar yaratabilirler [19-21]:

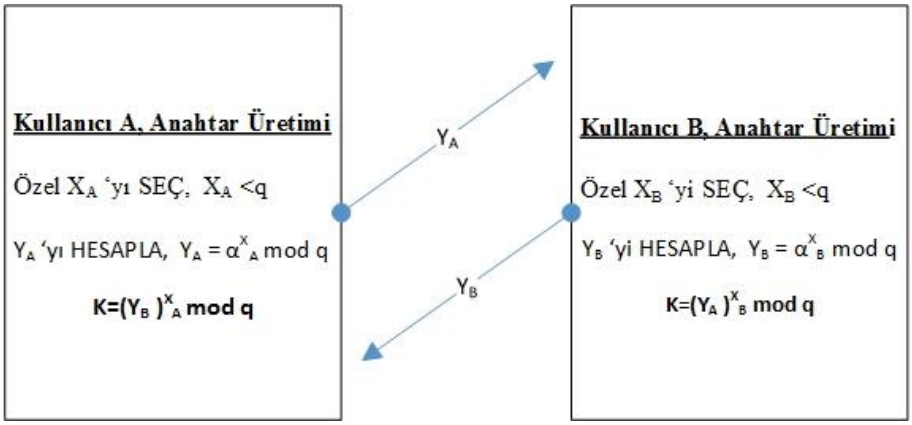
**1**. A, $0 \leq a \leq p-2$ eşitsizliğini sağlayan ve tesadüfî olan bir a sayısı seçer.
$C = g_a \pmod p$'yı hesaplar ve bunu B'ye gönderir.
**2**. B, $0 \leq b \leq p-2$ eşitsizliğini sağlayan ve tesadüfî olan bir b sayısı seçer.
$d = g_b \pmod p$'yı hesaplar ve bunu A'ya gönderir.
**3.** A, ortak anahtar K' yı şu şekilde hesaplar:
$K = d_a = (g_b)_a$
**4.** B, ortak anahtar K' yı şu şekilde hesaplar:
$K = c_b = (g_a)_b$

Böylece A ve B aralarında ortak bir anahtar olan K için anlaşmış olurlar.

| |
|---|
| **Global Açık Elemanlar** |
| q Asal Sayı |
| α α<q ve α primitif kök |
| **Kullanıcı A Anahtar Üretimi** |
| Özel $X_A$ 'yı seç $X_A$ <q |
| Açık $Y_A$ 'yı hesapla $Y_A = \alpha^{X}{}_A \bmod q$ |
| **Kullanıcı B Anahtar Üretimi** |
| Özel $X_B$ 'yi seç $X_B$ <q |
| Açık $Y_B$ 'yi hesapla $Y_B = \alpha^{X}{}_B \bmod q$ |
| **Kullanıcı A Tarafından Gizli Anahtarın Üretilmesi** |
| $K=(Y_B)^{X}{}_A \bmod q$ |
| **Kullanıcı B Tarafından Gizli Anahtarın Üretilmesi** |
| $K=(Y_A)^{X}{}_B \bmod q$ |

**Şekil 3.4** Diffie-Hellman Anahtar Değişim Algoritması[20].

Diffie-Hellman algoritmasının başka bir örnekle açıklayalım: bir grup kullanıcı varsayalım (LAN Ağındaki kullanıcılar gibi) her kullanıcı dayanıklı ve uzun birer $X_A$ özel anahtarını üretir ve $Y_A$ açık anahtarını hesaplar. Bu açık anahtarlar, q ve α değerleri ile birlikte herkese açık olan merkezi bir rehberde depolanır. Herhangi bir zamanda, B kullanıcısı A nın genel değerine erişebilir, gizli bir anahtar hesaplayabilir ve A kullanıcısına şifreli bir mesaj göndermek için kullanabilir. Merkezi rehber güvenli ise, bu iletişim iki tarafa da gizlilik sağlar ve kullanıcının gerçekliğini kanıtlar. Çünkü sadece A ve B anahtarı belirler, diğer kullanıcılar mesajı okuyamazlar. Alıcı A, sadece B kullanıcısının bu anahtarla bir mesaj oluşturabileceğini bilmektedir (Kimlik Doğrulama) [20].




**Şekil 3.5.** Diffie-Hellman Anahtar Değişimi

### 3.5.2 El-Gamal Şifreleme Sistemi

1985 de El Gamal; Diffie ve Hellman anahtar paylaşımını kullanarak daha farklı ve yeni bir açık anahtar kriptosistem önerdi. Her iki sistemin de güvenliği sonlu cisimlerde ayrık logaritma probleminin çözümünün zorluğuna dayanıyordu.

$Y=g^{x}$, (mod p) ; p E $Z^{+}$ , p asal sayı, koşuluyla verilen denklemde, x sayısının bulunmasına dayanır. El-Gamal tarafından 1985 yılında formalize edilen bu kriptosistemin ayrıntıları aşağıdaki gibidir:

A ve B kullanıcıları ortak bir anahtar oluşturmak istediklerinde:

$X^{a}$ ve $X^{b}$, A ve B kullanıcılarınca bilinen gizli ifadeler, p büyük asal bir sayı ve a(mod p)'nin ilkel elementi olsun.

Bu durumda:

$Y^{A} = a^{XA}$ (mod p) eşitliği A tarafından ve

$Y^{B} = a^{XB}$ (mod p) eşitliği B tarafından bulunarak, değiş tokuş edilir.

### 3.5.3 RSA Şifreleme Sistemi

RSA asimetrik şifreleme algoritması 1977 yılında R. Rivest, A. Shamir ve L.Adleman tarafından bulunmuştur. Daha sonra asimetrik şifreleme algoritmasına uyarlanarak geliştirilmiştir. Bu algoritma, asimetrik şifreleme sistemlerinde ve sayısal imza işlemlerinde güvenli bir şekilde kullanılır [18].

Günümüzde RSA şifreleme algoritması hala güvenilirliğini korumaktadır. Bunun en önemli nedeni modüler matematik üstüne kurulmuş, kripto analizi asal sayılara çarpanlara ayırmaya dayalı anlaşılması kolay, ama çözülmesi zor bir algoritma olmasıdır.

Öncelikle p ve q olmak üzere iki tane asal sayı üretilir. Bunların birbirleriyle çarpılmasıyla n=p*q 'dan n elde edilir. Bundan sonra n sayısından küçük ve (p–1) * (q–1) sayısıyla 1 dışında herhangi bir ortak böleni bulunmayan bir e sayısı seçilir.

Bundan sonraki aşamada (E* D=1) sayısının (p–1) * (q–1) çarpımına tam olarak bölünmesini sağlayan bir D sayısı bulunur. E ve D değerleri, sırasıyla, açık ve gizli anahtar olarak adlandırılırlar. Açık anahtarı n, E çifti, gizli anahtarı ise n, D çifti oluşturur. p ve q sayıları ya yok edilmeli, ya da gizli anahtar ile birlikte saklanmalıdır [18].

Gizli anahtar olan D sayısının n, E sayılarından elde edilmesi zor bir işlemdir. Eğer bir kişi n sayısını çarpanlarına ayırarak p ve q sayılarını elde edebilirse gizli anahtarı da rahatlıkla bulabilir.

Bu sebeple RSA sisteminin güvenliği çarpanlarına ayırma probleminin zorluğu temeline dayanır. Çarpanlarına ayırma işleminin kolay bir yönteminin bulunması, RSA algoritmasının kırılması anlamına gelir.

RSA sisteminin kırılması birkaç değişik şekilde yorumlanabilir. Sisteme en çok zarar verecek saldırı bir kripto analistin belli bir açık anahtara karsı gelen gizli anahtarı bulmasıdır. Bunu başarabilen bir hasım hem şifrelenen bütün mesajları okuyabilir, hem de imzaları taklit edebilir. Bunu yapmanın en akla gelen yolu n'nin asal çarpanlara ayrılması yani p ve q'nun hesaplanmasıdır. p, q ve açık üs e kullanılarak d kolaylıkla hesaplanabilir. Ancak buradaki zorluk n modülünün çarpanlarına ayrılmasıdır. RSA sisteminin güvenliği çok büyük sayıların asal çarpanlarına ayrılmasının zorluğu ihtimaline dayanır.

Büyük sayıların çarpanlarına ayrılmasının zorluğu ispatlanmış değildir. Son üç yüzyıl içerisinde Fermat ve Legendre gibi ünlü matematikçiler bu konuda çalışmalar yapmışlardır ve çalışmalarda hala devam etmektedir [19].

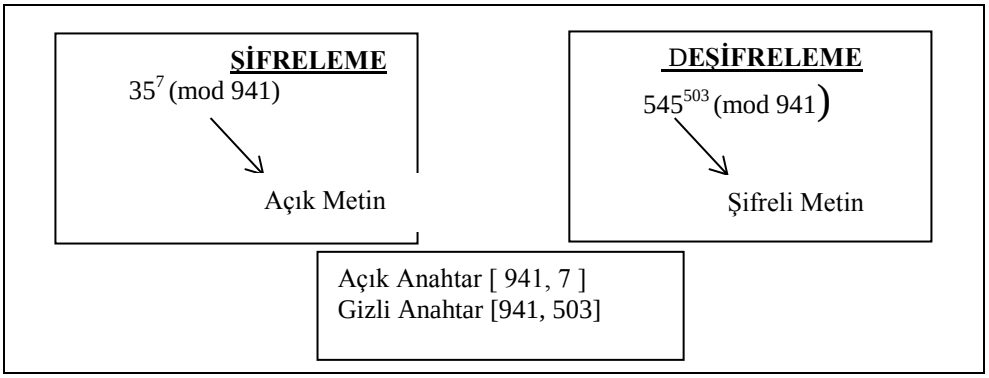
**ANAHTAR ÜRETİM ALGORİTMASI**

| | | |
|---|---|---|
| p ve q seçilir ( iki değerde asal sayı olmalıdır) | | |
| N= p x q ve | $\Phi(N) = (p-1) \times (q-1)$ hesaplanır. | |
| e tamsayı olarak seçilir. | OBEB ($\Phi$ (N), e ) =1 ve $1<e<\Phi(N)$ | |
| d değeri hesaplanır. | $d=e^{-1} \bmod \Phi(N)$ | |
| Açık Anahtar e, N | Gizli Anahtar d,N | |
| | **DEŞİFRELEME** | **ŞİFRELEME** |
| Açık Metin (M) | M<N olmalıdır | C |
| Şifreli Metin (C) | $C=M^e \pmod N$ | $M=C^d \pmod N$ |

**Şekil 3.6**. RSA Algoritması [20].

Şekil 3.6' da ki örneği açıklayalım. Bu örnek için anahtarların üretim adımları aşağıda verilmiştir.

1- İki asal sayı seçilir p=23 ve q=41.
2- n = p.q = 23x41 = 941 hesaplanır.
3- $\Phi(n) = (p-1)(q-1) = 880$ hesaplanır.
4- Öyle bir e seçki; e $\Phi$ (n)=880 ya asal olsun aynı zamanda (n) den küçük olsun. Bu durumda e=7 olur.
5- Öyle bir d belirleyelim ki d.e = 1 mod 880 ve d<880 olsun. Doğru değer d=503 'dir. Çünkü 503x7 = 3521 = 4 x 880 + 1.




**Şekil 3.7** RSA Algoritmasına örnek

Sonuçta anahtarlar; açık anahtar e ve N {7,941} ve özel anahtar d ve N {503,941} olur. Şekil 8 deki örnek, M=35 orijinal mesajı için anahtarların kullanımını örnekler. Şifreleme için 35 yedinci kuvveti alınır ve 64339296875 sayısı elde edilir. Bu sayının 941 tarafından bölünmesi sonucunda elde edilen kalan bulunur. Kalan olarak bulunan 545 sayısı bizim şifreli mesajımızdır. $C=M^e \pmod N$ formülünü uygulayarak C= 545 şifreli mesajımızı elde ettik. Elde edilen şifreli mesajımızı, deşifrelemek için $M=C^d \pmod N$ formülünü uygularız. $M= 545^{503} \bmod 941$ işleminin sonucunda orijinal mesajımız olan 19 sayısını elde ederiz [20].

RSA kriptosistemi, hem gizlilik hem de doğruluğunu ispatlama için kullanılabilen tek açık anahtar kriptosistemidir. RSA doğruluğunu ispatlama metodunu açıklarsak:

Y şifreli mesaji, orjinal mesaj X 'in, özel anahtarlar{d,n}'nin kullanılmasıyla elde edilir. Orjinal mesaj X, şifreli mesaj Y 'nin açık anahtarlar{e,n}kullanılması ile elde edilir. Aşağıdaki gibi;

$(X^d \bmod n)^e \bmod n = X$

X<n olduğu yerde $d.e=1 \bmod \varphi(n)$ 'dir. Denklemi şöyle yazabiliriz.

$(X^e \bmod n)^d \bmod n \equiv X$

A kullanıcısı p ve q 'nun asal olduğu yerde uygun bir n=p.q seçer. A göndericisi p-1 ve q-1 'in çarpımından $\varphi(n)$ 'yi hesaplar. Şimdi A göndericisi özel ve açık anahtarları olan d ve e 'yi seçer. Şöyle ki $d.e=1 \bmod \varphi(n)$ olur ve {e,n} açık anahtarı yayınlanır. A, M mesajını kendi özel anahtarı {d,n} ile şifreleyip Y 'yi elde. Şöyle ki;

$Y=X^d \bmod n$

Y mesajını B alıcısına gönderir. B alıcısı Y ve A'nın açık anahtarı KUa{e,n}'yi kullanarak X mesajını yeniden oluşturur. Şöyle ki;

$X = Y^e \bmod n$

$= (X^d)^e \bmod n$

$= X^{d.e} \bmod n$

$= X$

Gönderenin çift anahtarlarından $d.e = 1 \bmod \varphi(n)$ olduğunda dolayı d.e =1 oldu ve X'i elde ettik. A 'nın özel anahtarı yalnızca A da olduğu ve biz mesajı A 'nın açık anahtarı ile

deşifrelediğimiz için mesajı gönderenin A kullanıcısı olduğu doğrulanmış olur [20]. Başka bir örnekte incelersek [20]:

Seçilen p=41 ve q=59 ‘dan, n=(41)(59)=2419 veφ (n)=(40)(58)=2320 ‘yi hesaplayalım. e=169 seçildiğinde d=2169 olur. X = “FIRAT UNIVERSITESI” orijinal mesajını, her bloğu 4 karakterden oluşan bloklar halinde aşağıdaki gibi açıklayabiliriz.

1621 0212 0903 0011 0525 0003

1825 1620 1507 1801 1608 2500

İlk blok 1621 ’in, e=169 ‘a göre üssünü alırız. Çıkan sonucu n=2419’a böldüğümüzde kalan 1757 bize şifreli mesaj Y ‘yi verir. $1621^{169}$ (mod 2419) =1757 şifreleme işlemini yapmış oluruz. Aynı işlemi ilk bloktan başlayarak tüm bloklara uygularız. Böylece tüm X orjinal mesajı aşağıdaki gibi şifrelenir.

1757 0874 1272 1447 0241 1315

1843 2376 1931 1842 0788 1393

Şifreli mesaj Y ‘nin ilk bloğu 1757 ‘nin d=2169 ‘a göre üssünü alırız. Çıkan sonucu n=2419 ‘a böldüğümüzde kalan 1621 bize orijinal mesaj X’i verir.

$1757^{2169}$ (mod 2419) =1621 deşifreleme işlemini yapmış oluruz. Aynı işlemi ilk bloktan başlayarak tüm bloklara uygularız. Böylece tüm Y şifreli mesajı aşağıdaki gibi deşifrelenir.

1621 0212 0903 0011 0525 0003

1825 1620 1507 1801 1608 2500

## 3.6. En Çok Kullanılan Kapalı Anahtarlı (Simetrik) Şifreleme Modelleri

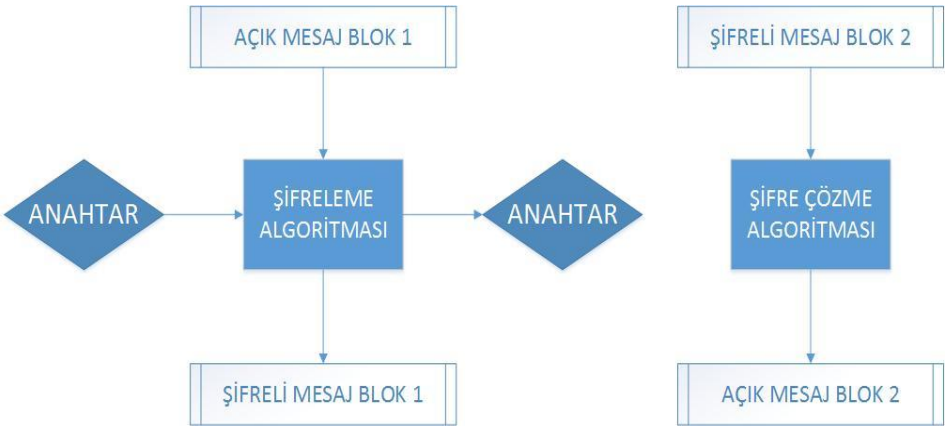
### 3.6.1. Simetrik Anahtarlı Veri Şifreleme Standardı

Gizli Anahtarlı kriptosistemler, Blok Şifreleme Sistemleri (Block Chiphers) ve Dizi veya Akan Şifreleme Sistemleri (Stream Ciphers) olarak ikiye ayrılırlar [2-3].

**1- Blok Şifreleme (Block Ciphers) Sistemleri**

Blok şifreleme sistemleride orijinal metin veya şifreli metin bloklarına ayrılmaktadır. Blok şifreleme algoritmalarına, SPN (Substitution-Permutation Network), DES ve AES örnek

gösterilebilmektedir. Blok şifreleme algoritmalarının gücü, Substitute (S) kutularına, döngü sayısına, anahtarların dış veya (exclusive OR - XOR) işlemine sokulmasına, blok uzunluğuna, anahtarın uzunluğuna ve özelliğine bağlıdır. Shannon, şifreleme algoritmasının gücü için "blok uzunluğunun en azından anahtar uzunluğuna eşit olması gerekir" demiştir. [24-25].




**Şekil 3.8.** Blok Şifreleme İşlemleri [25]

Blok şifreleme modelinde öne çıkan önemli değişkenler şifreleme anahtarı, algoritmadaki döngü sayısı ve algoritmaya gücünü veren S (Substitute) kutularıdır. Bu başlıkları kısaca açıklarsak [25]:

a) **Şifreleme Anahtarı:** Blok şifreleme algoritmalarında, anahtarın uzunluğu ya da bit sayısı en temel saldırı olan geniş anahtar arama saldırısına karşın güçlü olmalıdır. DES algoritması 56-bit anahtar kullanır. AES'te 128, 192, 256 bit anahtar seçenekleri mevcuttur. Ayrıca anahtarın rastlantısal olması gerekir.
b) **Algoritmadaki Döngü Sayısı:** Blok şifreleme algoritmalarında döngü sayısı çok önemlidir. Çünkü lineer(doğrusal) transformasyon ve yer değiştirmelerin bu seçilen değerle algoritmaya yeterli gücü vermesi gerekmektedir.

| ŞİFRELEME ALGORİTMASI | DÖNGÜ SAYISI |
|---|---|
| DES | 16 |
| IDEA | 8 |
| BLOWFISH | 16 |
| AES | 10 |

**Şekil 3.9.** Bazı Şifreleme Algoritmaları için Döngü Sayısı

c) **S Kutuları:** S kutuları algoritmadaki tek doğrusal olmayan (non-lineer) yapıdır. Bu haliyle algoritmaya gücünü vermektedir.

Bunlara bağlı olarak blok şifreleme algoritmasının örneklerinden biri olan DES (Data encryption standard)' i açıklarsak:

DES karıştırma ve yayılma şifreleme tekniğinden meydana gelmiştir. Karıştırma işlemi yer değiştirme ile yapılır. Özellikle verinin belirli bölgeleri orijinal veriden bazı bölgeler ile yer değiştirilir. Yer değiştirilen verinin belirlenmesi orijinal metne bağlıdır.

Yayılma permütasyon ile gerçekleşir. Farklı bölümlerin sırası yeniden düzenlenerek veri değiş tokuş edilir. Bu permütasyonlar, yer değiştirmeye benzer şekilde, anahtar ve orijinal metne bağlıdır. Yer değiştirmeler ve permütasyonlar DES algoritması tarafından belirlenirler [26].

DES, feistel şifreleme mimarisi temel alınarak 64 bitlik veri bloklarının şifrelenmesi ve şifrelerinin çözülmesi işlemleri için geliştirilip kullanılmaktadır. Girişte 64 bitlik anahtar kullanılır ve şifreleme yapılır. Bundan sonra ki aşamada, şifreleme işlemi için kullanılan anahtarın aynısı şifre çözme işlemi için kullanılır [19].

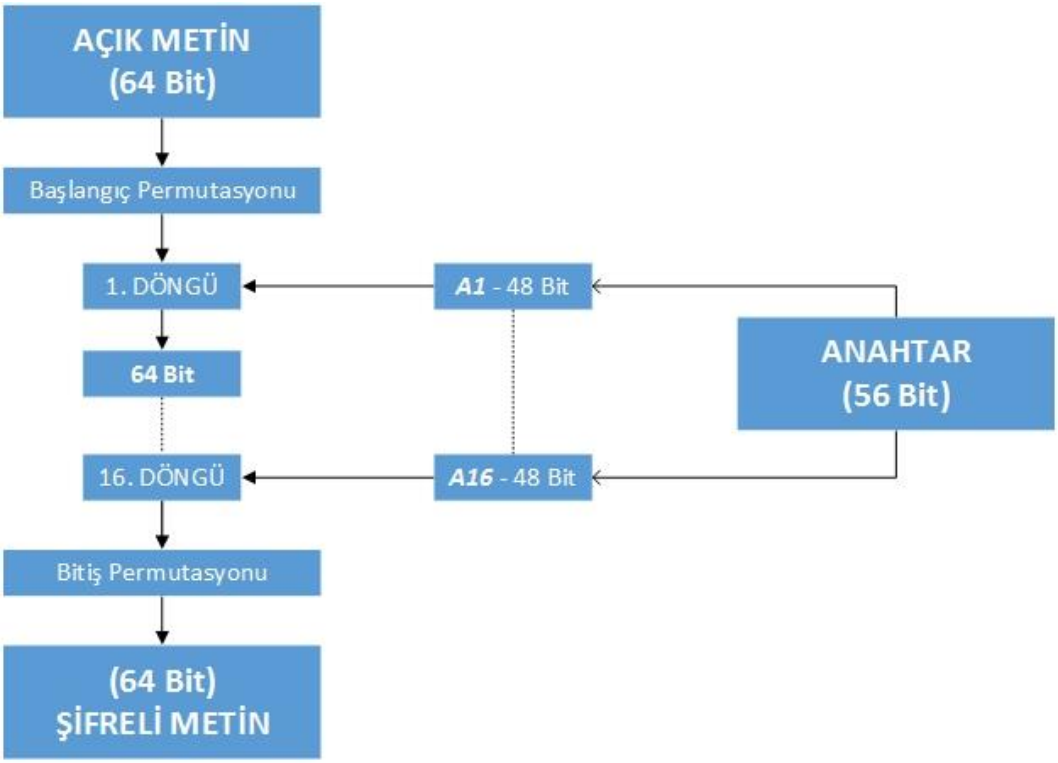

**Şekil 3.10.** DES (Data Encryption Standard) Algoritması Modeli

DES'te kullanılan feitsel mimarisini açıklarsak:

Açık metnin işlenmesini, her bir döngüde yer değiştirme ve permütasyon adımı takip eder. Her bir döngüdeki veri bloğu 2 eşit parçaya bölünür. Bu parçalar, L (left) ve R(right) diye gösterilir. Burada (L) sol taraftaki sütun için (R) ise sağ taraftaki sütun için kullanılır. Her bir döngüde, bloğun sağ tarafı (R) hiç değişmeden devam eder. Sol taraf (L), yer değiştirme işlemine sokulur. Her bir döngünün sonunda düzenlenmiş yeni (L) ve (R) taraflarının yeri değiştirilir [26].

DES işlemleri, başlangıç permütasyonu (IP) ile başlar. Belirli aşamalardan sonra ters başlangıç permütasyonu ($IP^{-1}$) uygulanarak biter. Şifrelemede 64 bitlik anahtarın her sekizinci biti permütasyon seçimi tablosu ile parite (eşlik) biti olarak ayrılır. Dolayısıyla şifreleme işlemi 56 bitlik anahtar kullanılarak gerçekleşir. Bu 56 bit anahtar üzerinden her bir döngü için ayrı ayrı 16 adet alt anahtar oluşturulur. Oluşturulan bu alt anahtarlar 16 döngü boyunca sisteme uygulanır [25].

DES şifreleme algoritmasının genel şeması şu şekildedir:




**Şekil 3.11.** DES Şifreleme Algoritmasının Blok Diyagramı [26].

Bu blok diyagramıma göre:

- 64 bitlik açık metin başlangıç permütasyonu olan IP'ye (Initial Permutation) maruz tutulur.
- 64 bitlik açık metin, eşit uzunluktaki sağ (R) ve sol (L) parçalara bölünür. Bölünen parçaların her biri 32 bit uzunluğundadır. İlk döngüde bu parçalar L0 ve R0 olarak kalırlar.
- f fonksiyonu döngü için oluşturulmuş alt anahtar ile işlem yapar.
- Bu işlemler 16 döngü boyunca tekrarlanır. 16 döngü sonunda sol yarı (L) ile sağ yarı (R) değiştirilir.
- Son adımda 64 bitlik açık metin üzerine başlangıç permütasyonun tersi ($IP^{-1}$) uygulanır.

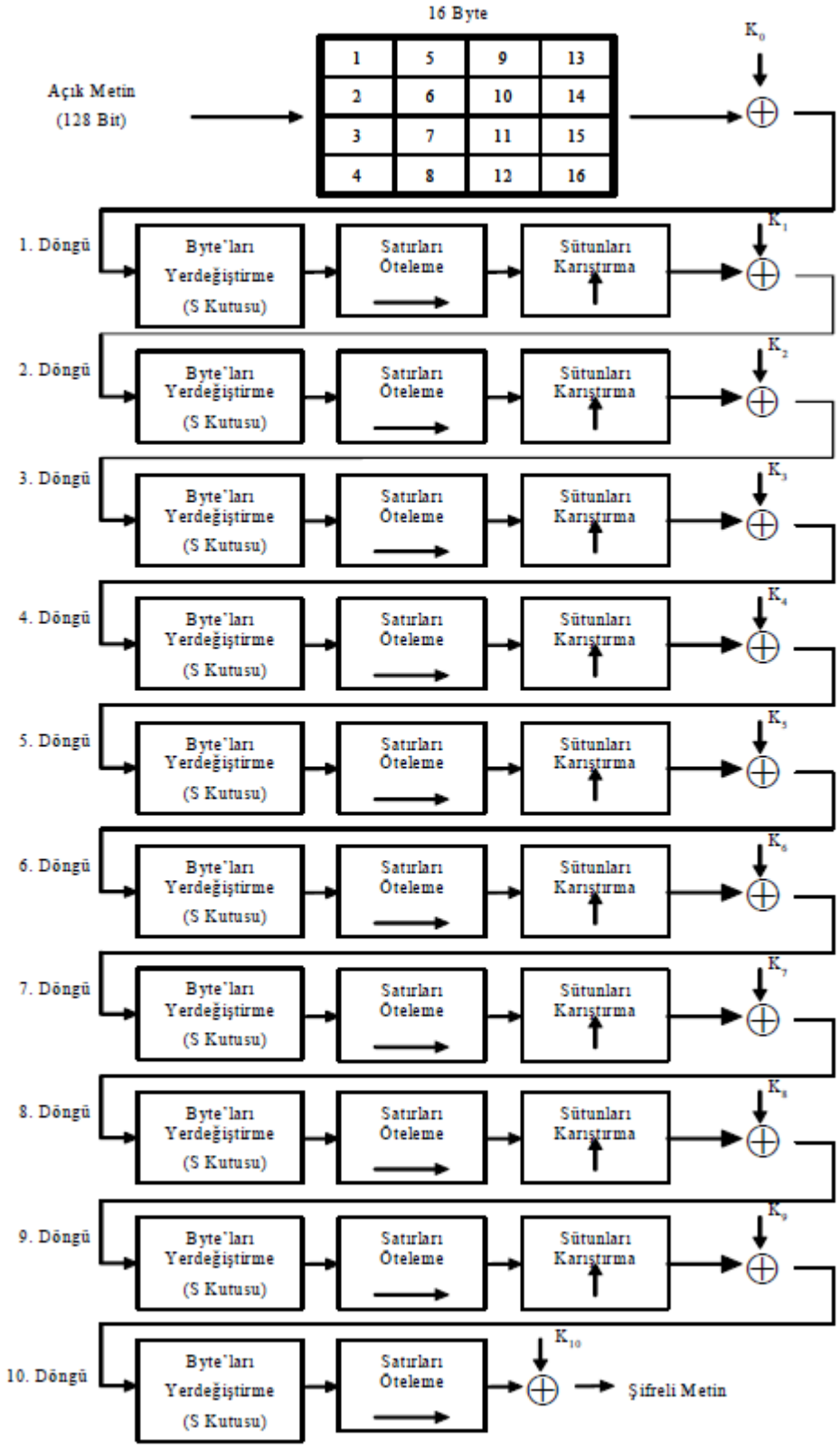
### 3.6.2. Gelişmiş Şifreleme Standardı (AES- Advanced Encryptıon Standard)

AES (Rijndael-Gelişmiş Şifreleme Standardı) algoritması, 128 bit veri bloklarını 128, 192, 256 bit anahtarlarla şifreleyen bir şifreleme algoritmasıdır. AES'in döngü sayısı anahtar genişliğine göre değişkenlik gösterir. AES'te 128 bit anahtarla 10 döngüde şifreleme yapılır. 192 ve 256 bit anahtarlarla sırasıyla 12 ve 14 döngüde şifreleme işlemi gerçekleşir. AES, kullandığı anahtar boyutuna göre tanımlanır. Bunlar; "AES-128", "AES-192", "AES-256" gibi adlandırılır. AES algoritmasının her döngüsü dört bileşenden oluşur [25].

Vincent Rijmen ve Joan Daemen tarafından gelistirilmis AES, DES sifreleme algoritmasının zayıf noktalarını karsı olusturulmustur. Bu algoritmada 128 bit veri bloğu 4×4 bayt'lık matrise dönüştürülür. Bu işlemden sonra her döngüde sırasıyla aşağıdaki işlemler yapılır [27]:

a. Bayt'ların yer değiştirmesi (Sub-Bytes dönüşümü)
b. Satırların ötelenmesi (Shift-Rows dönüşümü)
c. Sütunların karıştırılması (Mix-Columns dönüşümü)
d. Döngüye anahtar eklenmesi (AddRoundKey dönüşümü)

Birinci işlemde 16 byte değerinin her biri 8 bit girişli ve 8 bit çıkışlı S kutusuna yerleştirilir. S kutusu değerleri, Galois cisiminde ((Galois Field-GF), GF(28)), 8 bitlik polinom için ters alındıktan sonra doğrusal bir dönüşüme sokularak hesaplanır. İkinci işlemde 4×4 bayt matrisinde satırlar ötelenir ve üçüncü işlemde herhangi bir sütun için o sütundaki değerler kıyaslanır. Döngünün son kısmında o döngüye ait anahtar ile XOR işlemi meydana gelir [25-27].




**Şekil 3.12.** 128 bit Anahtarlı AES Şifreleme Algoritması Blok Diyagramı

### a-) AES'te Şifreleme İşlemi

**Ş**ekil 3.12' de, 1. döngüden önce 128 bitlik veri 4×4 bayt matrisine dönüştürülür. Bu matris, K0 master anahtarla XOR işlemine sokulur. Her döngüde dört temel işlem yapılır. Ancak son döngüde sütunların karıştırılması işlemi yapılmadan metin şifrelenir. Bu şekilde gerçekleşen AES şifreleme algoritması tekrarlayan bir şifreleme algoritmasıdır. AES algoritmasında giriş bloğu, çıkış bloğu ve durumun uzunluğu "Nb" ile gösterilir. 'Nb' değeri sabit ve uzunluğu 32 bittir. Bir kelime sayısını veren bu değer 4'e eşittir [25-26].

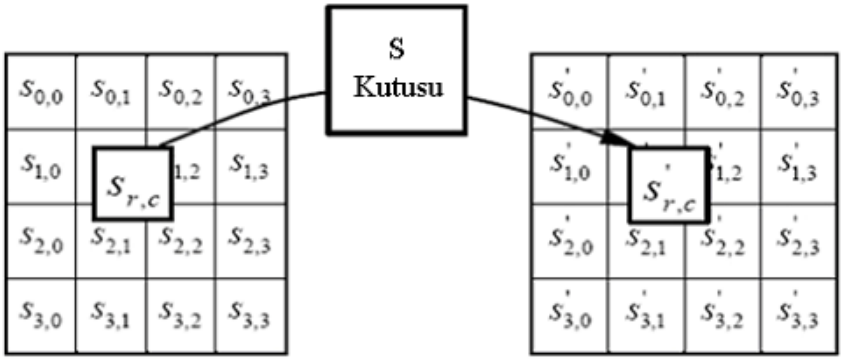
Algoritmanın anahtar uzunluğu "Nk" diye tanımlanır. Nk anahtarın sahip olduğu kelime sayısını gösterir. Bu sayı 4, 6 veya 8 değerlerini alabilir. AES'te tekrarlanan tur sayısı, "Nr", algoritmada kullanılan anahtar uzunluğuna bağlıdır [26].

**Tablo 3.3.** Tur Sayısının Anahtar Uzunluğu Göre Değişimi [2].

| | **AES (128 Bit)** | **AES (192 Bit)** | **AES (256 Bit)** |
|---|---|---|---|
| **ANAHTAR (Nk)** | 4 | 6 | 8 |
| **BLOK (Nb)** | 4 | 4 | 4 |
| **DÖNGÜ (Nr)** | 10 | 4 | 14 |

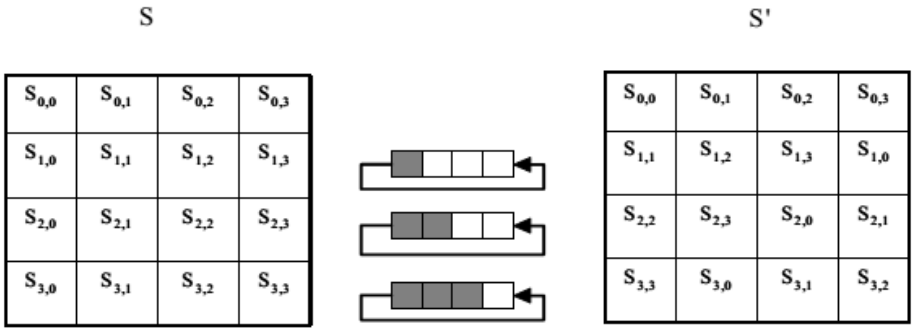
Bu tablonun her adımını açıklarsak:

1- **Bayt'ların Yer Değiştirmesi (SubBytes Dönüşümü):** Bir S kutusu ele alınarak işlemin her bayt değeri üzerinde bağımsız olarak çalışması sağlanır. Bu işlemde, durumun her bir bayt'ı, S kutusuna bir değiştirme tablosuyla gönderilir ve başka bir bayta dönüştürülür. Ayrıca, S kutusu tersine çevrilebilir bir kutudur [25].



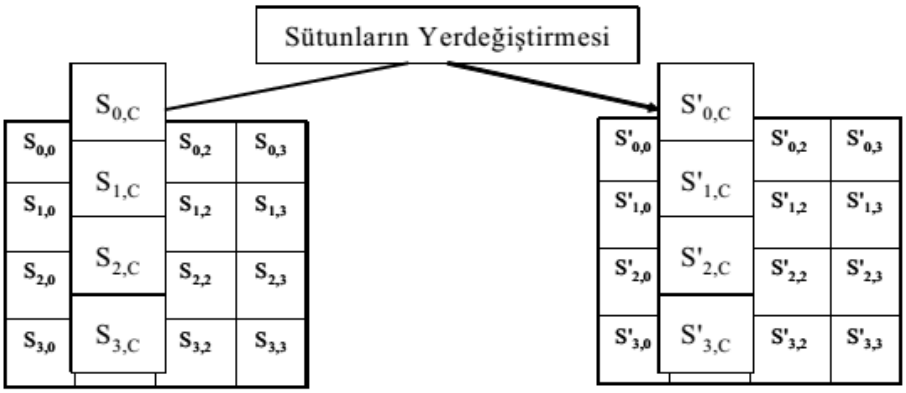

**Şekil 3.13.** Baytların Yer Değiştirmesi İşlemi [26].

**2- Satırların Ötelenmesi (ShiftRows Dönüşümü):** Bu işlemde veri bloğunun ilk satırında herhangi bir öteleme işlemi yapılmaz. Fakat 2. satırda sola doğru 1 kez, 3. satırda sola doğru 2 kez, 4. satırda sola doğru 3 kez öteleme yapılır. Şekille gösterirsek [25]:



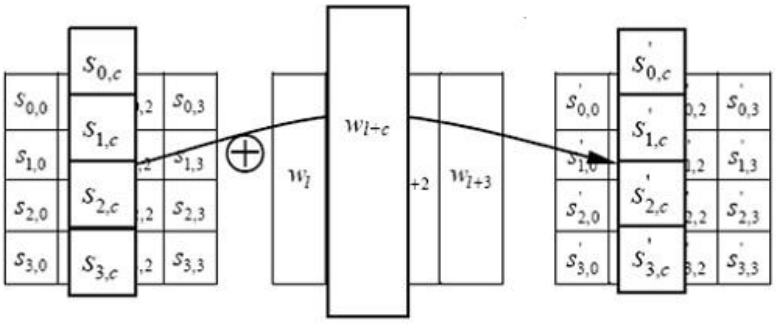

**Şekil 3.14.** Satırların Ötelenmesi İşlemi

**3- Sütunların Yer Değiştirmesi (MixColumns Dönüşümü):** Bu basamakta, işlem üzerinde sütun sütun işlem yapılır. Bu dönüşüm, 32 bit doğrusal bir dönüşümdür. Buradaki sütunlar GF($2^8$)'de polinomlar olarak düşünülür ve sabit bir a(x) polinomu ile mod $x^4+1$'e göre çarpılarak sonuca ulaşılır [27].




**Şekil 3.15.** Sütunların Yer Değiştirmesi [27].

4- **Döngüye Anahtar Ekleme Dönüşümü (AddRoundKey Dönüşümü):** Anahtar ekleme dönüşümü, ana anahtar üzerinden üretilen alt anahtarla ilgili döngüdeki durumun XOR işlemine sokulmasıyla gerçekleşir. Başlangıçta seçilen anahtar boyutuna bağlı kalınarak işlemin ne kadar süreceği belirlenir. Anahtar boyutu 128 bit ise 10 döngü boyunca anahtar ekleme işlemi yapılır. Şekille açıklarsak [27]:

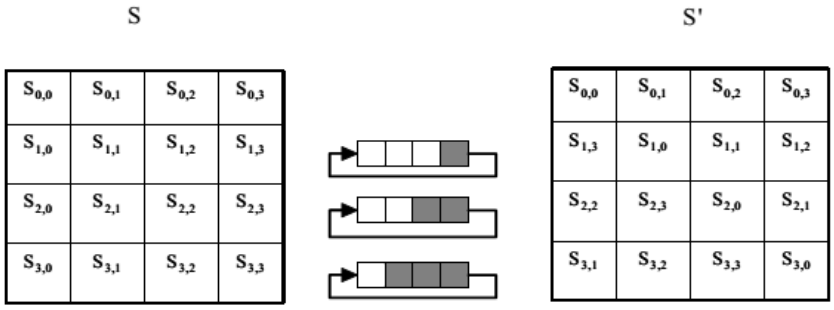


**Şekil 3.16.** Döngüye anahtar ekleme dönüşüm sistemi.

## b) AES Şifre Çözme İşlemi

Algoritmayı oluşturan fonksiyonların tersleri alınır. Bununla birlikte fonksiyonlar ters sırada işleme sokularak deşifreme işlemi yapılır. Bu işlem basamaklarını açıklarsak:

**1- Ters Satırları Öteleme (ShiftRows) Dönüşümü:** Bu işlemde, satır öteleme dönüşümünün tersi yapılır. Satır öteleme işlemindeki gibi ilk satır ötelenmez ve alttan üç satır için öteleme yapılır. Bu durumu şekille gösterirsek [25-26]:




**Şekil 3.17.** Ters Satırları Öteleme İşlemi

**2- Ters Bayt'ları Yer Değiştirme (SubBytes) Dönüşümü:** Bu basamakta dönüşüm, bayt'ları değiştirme işleminin tersi olup durumun her bir bayt'ı ters S kutusundaki değeri ile yer değiştirilerek gerçekleşir [26].

**Tablo 3.4.** Ters S Kutusu [26].

| | | y | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | a | b | c | d | e | f |
| x | 0 | 52 | 09 | 6a | d5 | 30 | 36 | a5 | 38 | bf | 40 | a3 | 9e | 81 | f3 | d7 | fb |
| | 1 | 7c | e3 | 39 | 82 | 9b | 2f | ff | 87 | 34 | 8e | 43 | 44 | c4 | de | e9 | cb |
| | 2 | 54 | 7b | 94 | 32 | a6 | c2 | 23 | 3d | ee | 4c | 95 | 0b | 42 | fa | c3 | 4e |
| | 3 | 08 | 2e | a1 | 66 | 28 | d9 | 24 | b2 | 76 | 5b | a2 | 49 | 6d | 8b | d1 | 25 |
| | 4 | 72 | f8 | f6 | 64 | 86 | 68 | 98 | 16 | d4 | a4 | 5c | cc | 5d | 65 | b6 | 92 |
| | 5 | 6c | 70 | 48 | 50 | fd | ed | b9 | da | 5e | 15 | 46 | 57 | a7 | 8d | 9d | 84 |
| | 6 | 90 | d8 | ab | 00 | 8c | bc | d3 | 0a | f7 | e4 | 58 | 05 | b8 | b3 | 45 | 06 |
| | 7 | d0 | 2c | 1e | 8f | ca | 3f | 0f | 02 | c1 | af | bd | 03 | 01 | 13 | 8a | 6b |
| | 8 | 3a | 91 | 11 | 41 | 4f | 67 | dc | ea | 97 | f2 | cf | ce | f0 | b4 | e6 | 73 |
| | 9 | 96 | ac | 74 | 22 | e7 | ad | 35 | 85 | e2 | f9 | 37 | e8 | 1c | 75 | df | 6e |
| | a | 47 | f1 | 1a | 71 | 1d | 29 | c5 | 89 | 6f | b7 | 62 | 0e | aa | 18 | be | 1b |
| | b | fc | 56 | 3e | 4b | c6 | d2 | 79 | 20 | 9a | db | c0 | fe | 78 | cd | 5a | f4 |
| | c | 1f | dd | a8 | 33 | 88 | 07 | c7 | 31 | b1 | 12 | 10 | 59 | 27 | 80 | ec | 5f |
| | d | 60 | 51 | 7f | a9 | 19 | b5 | 4a | 0d | 2d | e5 | 7a | 9f | 93 | c9 | 9c | ef |
| | e | a0 | e0 | 3b | 4d | ae | 2a | f5 | b0 | c8 | eb | bb | 3c | 83 | 53 | 99 | 61 |
| | f | 17 | 2b | 04 | 7e | ba | 77 | d6 | 26 | e1 | 69 | 14 | 63 | 55 | 21 | 0c | 7d |

3- **Ters Sütunları Karıştırma (Mix Columns) Dönüşümü:** Sütunları karıştırma işleminin tersi olan bu işlem, durum üzerinde sütun sütun yapılır. Sütun değerleri GF($2^8$)'de polinomlar olarak ele alınır ve işlem yapılır [25].

4- **Ters Alt Anahtar Ekleme (AddRoundKey) Dönüşümü:** Bu işlemde alt anahtar ekleme işleminin tersidir. Alt anahtar ekleme, veri bloğunun anahtar ile XOR işlemine sokulması anlamına gelir. Bu işlemin tersi anahtarın kendisi olacaktır [25-27].

# 4. STEGANOGRAFİ

## 4.1 Steganografinin Tanımı

Anlamı "*kaplanmış yazı*" (*covered writing*) olan Steganografi kelimes köken olarak "*στεγανος*" ve "*γραΦειν*"'den gelen ve Yunan alfabesinden oldukça eski bir bilgi gizleme sanatıdır [28-29].

Steganografi bilimi gizli ileti veya bilginin varlığını saklamayı amaçlamaktadır. İletilmek istenen ileti gizlenerek üçüncü kişilerin taşınan bu mesajın varlığından haberdar olmaları engellenir. Gizlenmesi istenilen veriler text veya görüntü formatında olabilirken hatta bir görüntü içerisine başka bir görüntünün gizlenmesi bile olasıdır [30-31].

Steganografi veri iletimini gizli bir şekilde sağlamaktadır Steganografi tekniğinin amacı ise gönderici ve alıcı arasındaki iletişimin üçüncü şahıslar tarafından farkedilmemesidir. Literatüre bakıldığında ise Steganografi çalışmaları 1983 yılında Simmons tarafından "Prisoner Problem"'in tanımlanması ile başlamaktadır. Bu problemde Alice ve Bob hapishanededir ve hapishaneden kaçmak için planlar yapmaktadırlar. Fakat bu planların Willie adındaki gardiyan tarafından fark ettirilmeden yapılması gerekmektedir. Eğer Willie bunu fark ederse kaçma planlanı başarısız olacaktır. Bu nedenle de çeşitli gizli haberleşme yöntemleri geliştirilmesi gerekmektedir [32].

Steganografi tekniğinde içine veri gizlenen ortama örtü verisi (cover-data) veya örtü nesnesi (cover-object), oluşan ortama da stego-metin (stego-text) veya stego-nesnesi (stego object) denmektedir [33]. Steganografi tekniği şifreleme bilimi olarak adlandırılan kriptolojiye yakın olmasına rağmen şifrelemeden faklıdır. Şifreleme mesajın içeriğinin korunması ile ilgilenirken steganografi mesajın varlığının gizlenmesi ile ilgilenmektedir. Kısaca veri gizleme tekniğinde amaç gönderici ve alıcı arsında ki mesajın fiziksel olarak varlığını gizlemek olarak bilinirken, şifreleme biliminde ise bu maç mesajın varlığından fiziksel olarak haberdar olunmasına rağmen mesajın içeriğinin gizlenmesidir. Bundan dolayıdır ki steganografi bir şifreleme yöntemi değil şifrelemeyi tamamlayıcı bir öğedir [34]. Bu tez çalışması sırasında geliştirilen uygulamalarda kullanılan yöntemlerde veri şifreleme ve veri gizleme teknikleri birlikte kullanılmış ve güvenlik düzeyi daha da artırılmıştır.

## 4.2 Steganografinin Tarihçesi

Steganografi Antik yunan ve Antik Çağlara kadar dayanan en eski veri gizleme yöntemlerinden biridir. Bu tarihsel gelişime göz atarsak,

- **MÖ 440,** Antik Yunan'da gönderilerin elçilerin saçlarının kazıtılarak deri üzerine mesajın yazılması ve elçinin gönderildiği yerde tekrar saçlarının kazınmasıyla birlikte iletinin okunması.
- **Antik Çağlarda,** Gönderilecek iletilerin gizlenmesi için her meyvenin birbirine göre pozisyonlarına göre farklı bir anlam ifade ettiği meyve sepetlerinin kullanılması.
- **1650,** Gaspar Schott tarafından müzik notaları ile verilerin kodlanması.
- **1914 - 1918,** I. Dünya Savaşı sırasında görünmez mürekkep ve Mikrodot'ların kullanılması.
- **1945,** II. Dünya Savaşı sırasında Semagram ve yine Mikrodot'ların kullanılması.

Steganografi, Antik yunan ve Heredot zamanına kadar uzanan oldukça eski bir veri gizleme yöntemidir. Yunan tarihçi Heredot, eserinde İran'da bulunan casusun, Pers istilasını Yunanistan'a nasıl ilettiğini kaydetmektedir. Yazıya göre, casus kölesinin saçını kazıtmış; istila uyarısını da kafa derisine kazıtmıştır. Daha sonra yapılacak olan, kölenin saçının yazıyı kapatacak kadar uzamasını beklemek ve bu köleyi Yunanistan'a göndermektir. Kölenin bilmesi gereken tek bilgi *"kafamı kazıyın"* olacaktır. Yine aynı çağda avcı kılığındaki bir ulağın, avladığı hayvanın karnına parşömen saklayarak Yunanistan'a girmesi anlatılmaktadır [35].

Antik dönemdeki bu basit uygulamalar steganografinin gizli iletişimdeki kullanımının insanlık kadar eski olduğunu bizlere göstermektedir [36].

Steganografi hakkında yazılan ilk kitap Johannes Trithemus (1462–1516) tarafından yazılmış olan *Steganographiæ* isimli kitaptır. 1600'lü yıllarda yaşamış olan Gaspar Schott (1608–1666) tarafından yazılmış olan *Schola Steganographica* (Schott, 1665) isimli kitapta ise müzik notalarının bilgi gizlemek için nasıl kullanıldığı anlatılmıştır.

Stenografi, savaşlarda da sıkça karşımıza çıkmaktadır. I. ve II. Dünya Savaşlarında kullanılan Mors kodları gibi uygulamalarda bunun örneklerini görmekteyiz [37]. Ancak, çarpıcı kullanımı ikinci dünya savaşında kendini göstermektedir. İkinci dünya savaşı esnasında, Alman casusların gizli bilgileri kimyevi bir madde ile beyaz bir mendile yazdıkları ortaya çıkartılmıştır. Casus, gizli mesaj içeren bu mendili daha önce belirlenen noktalarda çöpe atmakta; alıcı ise yine kimyevi maddeler kullanarak bu yazıyı okumaktadır [38].

İkinci dünya savaşı döneminde bir başka örneği ise: Almanlar "mikrofilm" teknolojisi kullanarak "mikro noktalar" (microdot) kullanmıştırlar. Bu yöntemde A4 büyüklüğündeki herhangi bir belge veya çizim bir dizi işlem sonrasında daktilo yazısında kullanılan bir nokta kadar küçültülmektedirler. Bu yöntem kullanılarak masum içerikli bir sayfa düz metindeki i ve j harflerinin noktalarına oldukça büyük miktarda veri saklamak mümkün olmuştur [39].

İkinci dünya savaşında kullanılan bir steganografi örneğini incelersek [40] :

"*Apparently neutrals protest is thoroughly discounted and ignored. Isman hard hit. Blockade issue affects pretex for embargo on by-products, ejecting suets and vegetable oils*."

Yukarıda verilen paragrafta her kelimenin ikinci harfleri yan yana getirildiğinde "*Pershing sails from NY June 1*." Mesajı ortaya çıkmaktadır.

### 4.3 Steganografinin Alt Alanları

Steganografi, Dilbilim Steganografi ve Teknik Steganografi olmak üzere kendi içerisinde ikiye ayrılmaktadır. Dilbilim Steganografi de taşıyıcının metin olduğu steganografinin alt dalı, Teknik Steganografi ise birçok konuyu kapsamaktadır [41].

#### 4.3.1 Dilbilim Steganografi

Taşıyıcı verinin text olduğu steganografi koludur. Burada değişiklik yapmanın bazı farklı yolları vardır. Bunlar; grafik kullanılarak yapılabilir, text'in yapısı değiştirilerek yapılabilir ya da amacı sadece veriyi saklamak olan yeni bir text yaratılabilir.

Dilbilim Steganografi'de kullanılan bazı yöntemler vardır onlar ise şunlardır:

Açık kodlar: Gizli mesaj, açıkça okunabilir fakat zararsız bir mesaj haline gelir. Bu işlem; maskeleme, boş şifreler ve grid (ızgara) ile yapılmaktadır.

Şemagramlar: Gizli mesaj, açık metinin ufak fakat gizli bir detayının içine gizlenmektedir. Bunun için grafiksel değişiklikler yapılmaktadır. Kullanılan yöntemler ise; farklı yazı tipleri kullanmak, eski daktilo yazılarını kullanmak, resimler içinde boşluklar kullanmak vb'dir.

### 4.3.2 Teknik Steganografi

Teknik Steganografi ise birçok konuyu içine almaktadır. Bunlarda şöyledir:

Görünmez mürekkep: Geleneksel haline gelmiş olan görünmez mürekkeple yazma yöntemidir.

Gizli yerler: Kimsenin göremeyeceği gizli yerlere saklama (çanta, dolap vb.)

Microdot'lar: Bilgiyi noktalar halinde sayfaya gizleme

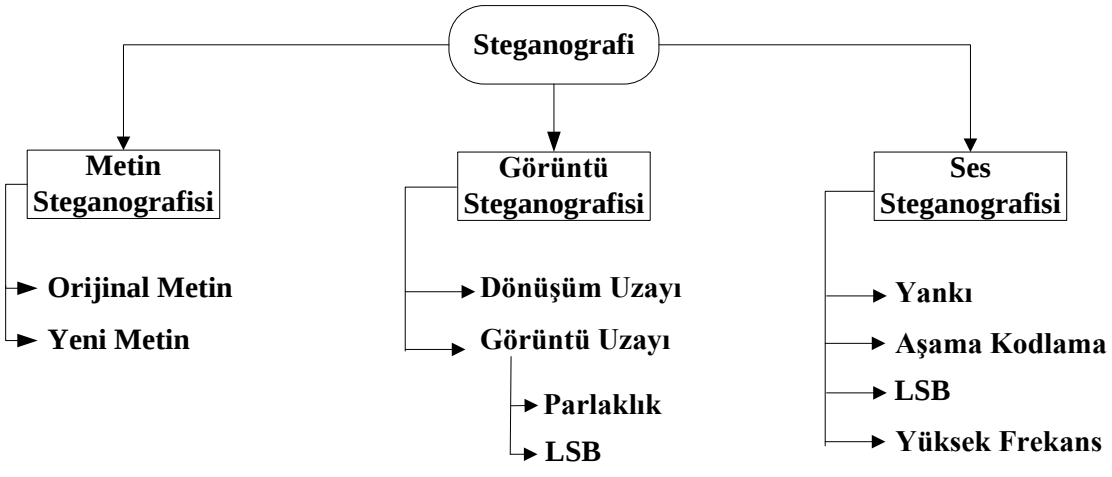
Bilgisayar tabanlı yöntemler: Text, ses, görüntü, resim dosyalarını kullanarak veri gizleme yöntemleridir.

## 4.4 Steganografinin Uygulama Alanları

Sayısal steganografi kullanım alanları açısından genel olarak üçe ayrılmaktadır.

Bunlar aşağıdaki gibidir [42]:

- Metin (text) steganografi
- Görüntü (image) steganografi
- Ses (audio) steganografi




**Şekil 4.1.** Sayısal steganografi yöntemlerinin sınıflandırılması

### 4.4.1. Metin (text) Steganografi

Metin steganografi, taşıyıcı ortamın metin olduğu steganografi dalıdır. Metin steganografide gizlenebilecek olan veri miktarı az ve sınırlıdır. Bunun sebebi taşıyıcı metinin içerisindeki kullanılmayan alanların miktarının sınırlı olmasıdır. Uygulanabilmesi için çeşitli yöntemler vardır. Bunlar [42-43]:

1- Açık Alan Yöntemleri (Open Space Methods)
2- Yazımsal Yöntemler (Syntactic Methods)
3- Anlamsal Yöntemler (Semantic Methods)

### 4.4.2. Görüntü Steganografi

Sayısal resimler dağıtımı en kolay ve internette hemen her sayfada karşılaşılabilecek dosyalardır. Kullanıldıkları formatlara göre farklılık göstermekle birlikte steganografi uygulamalarında en rağbet gören resim dosyalarıdır. Bu nedenle steganografi konusunda yapılan çalışmalar ve geliştirilen teknikler ağırlıklı olarak resim steganografi çerçevesinde yer almaktadır [45].

Görüntü dosyalarının içerisine bir metin gizlenebileceği gibi bir resim dosyasının içine bir başka resmi de gizlemek mümkündür. Gizli bilgiyi bir resme gömme (ya da gizleme) işleminde iki dosya söz konusudur. Kapak resim ya da örtü verisi (cover image) olarak adlandırılan ilk dosya, gizli bilgiyi saklayacak resim dosyasıdır. İkinci dosya ise gizlenecek bilgi olan mesajdır. Bu mesaj da stego olarak isimlendirilmektedir. Mesaj, açık metin (plain text), şifreli metin (chipher text), başka resimler veya bit dizisi içinde saklanabilecek başka bir şey olabilir. Gömme işlemi sonucunda kapak resim ve gömülü mesajın oluşturduğu dosyaya "stego resim" adı verilir [47].

### 4.4.3. Ses (Audio) Steganografi

İnsan işitme sistemi (Human auditory system-HAS) frekans aralığı yüzünden, ses sinyalleri içerisine bilgi gizleme oldukça çaba gerektirir. HAS 1/1.000'den daha büyük frekans aralığını fark edebilir. Aynı zamanda HAS nereden geldiği belli olmayan gürültülere de oldukça duyarlıdır [42].

# 5. VERİ ŞİFRELEME VE VERİ GİZLEME UYGULAMALARI

Bu bölümde tez sırasında geliştirilen iki adet uygulamaya yer verilmiştir. Geliştirilen uygulamalar uygulamanın yapıldığı alana katkı sağlamak için değil daha çok tez sırasında geliştirilen teknik ve yöntemlerin hayata geçirilmesi amacıyla düşünülmüş senaryolardır. Bu nedenle uygulamanın yapıldığı alanda tamamen çözüm üretmek veya tüm sorunları ortadan kaldırmak gibi bir amaç bulunmamaktadır. Bahsedildiği gibi gerçekleştirilen uygulamalar, geliştirilen metotların teoride kalmaması ve hayata geçirilmesi amacıyla yapılmıştır.

## 5.1. Radyografik Görüntüler Üzerinde Veri Şifreleme ve Veri Gizleme Uygulaması

Diş hekimliği alanında kullanılan ve OPT ismi verilen diş radyografileri hastalığın teşhis ve tedavisi için oldukça önemli bir role sahiptir. Diş rahatsızlığı nedeniyle kliniklere başvuran hastaların hemen hemen hepsine OPT resimleri çektirilir ve bu görüntüler üzerinden tanı ve teşhisler yapılarak tedavi başlatılır. Bu nedenle hastane ortamlarında bu kadar fazla radyografik resimlerin saklanması güç olmakla beraber hastanın dosyasında saklanan kimlik bilgileri, tahliller, formlar vs. gibi verilerin de saklanması hem yer hem zaman israfına yol açmaktadır. Sağlık kurumu veya hastanın herhangi bir belgeyi kaybetme durumunda birçok hastanın yapılan işlemleri bilmediği bunun da karışıklıklara neden olduğu tespit edilmiştir. Yine günümüzde Sağlık Bakanlığı'nın diş hekimliği alanında uygulanan sistemle tüm hasta bilgileri ortak bir veri tabanında tutulmakta buda güvenlik ve gizlilik açısından halen tartışılmaktadır.

Bu çalışmada öncelikle hasta bilgileri, tedavide kullanılacak teşhis ve tanıların kriptografi yöntemiyle şifrelenerek güvenlik zafiyetinin giderilmesi ardından da şifrelenmiş verilerin OPT resimlerinin üzerine gömülerek gizlenmesi böylece saklama ve kaybolma problemlerinin ortadan kaldırılması amaçlanmıştır.

### 5.1.1 Verilerin Şifrelenmesi

Veri güvenliği ve gizliliğinin oldukça önemli bir konuma geldiği günümüzde özellikle kamunun yararlanıp takip edildiği sistemlerde çok hayati bir role sahiptir. Çünkü bahsedildiği gibi şu an Sağlık Bakanlığı diş hekimliği alanında hastaların takibini yapmakta hangi hastaya

hangi tedavi yapıldığı bilinmektedir. Ancak bu verilerin hemen hemen herkesin erişimine açık olması bilgi güvenliğini tehdit etmekte ve bilgilere erişme veya kullanma yetkisi olmayan 3.kişilerin eline geçmesi durumunda olumsuz sonuçlar doğuracaktır.

Bu nedenle uygulama da veri güvenliğini sağlamak, bilgilerin bir şekilde yetkisiz kişilerin eline geçmesi durumunda dahi kullanılmaması için Visual C# kodlarıyla hasta takip sisteminde kullanılacak kimlik, tanı, teşhis ve tedavi gibi veriler şifrelendi. Sistemde yer alacak tüm harfler, sayılar, semboller hemen hemen tüm karakterler Tablo 5.1'de görüldüğü gibi bir tablo içerisinde rasgele sıralanmış ve hepsine ASCI kodlarına benzer birer numara atanmıştır.

(Örneğin tabloda "F" karakterinin 28 id'sine denk geldiği görülüyor).

**Tablo 5.1.** Karakter Seti

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | n | **16** | / | **32** | h | **48** | P | **64** | ! | **80** | p |
| **1** | 9 | **17** | 0 | **33** | , | **49** | r | **65** | K | **81** | v |
| **2** | * | **18** | = | **34** | % | **50** | 6 | **66** | U | **82** | b |
| **3** | E | **19** | space | **35** | B | **51** | ) | **67** | \ | **83** | Y |
| **4** | d | **20** | m | **36** | ı | **52** | S | **68** | ^ | **84** | 1 |
| **5** | + | **21** | - | **37** | Ş | **53** | H | **69** | ü | **85** | " |
| **6** | O | **22** | z | **38** | ' | **54** | i | **70** | C | **86** | & |
| **7** | R | **23** | Ğ | **39** | > | **55** | Ü | **71** | A | **87** | Ö |
| **8** | f | **24** | t | **40** | ö | **56** | _ | **72** | G | **88** | u |
| **9** | ; | **25** | y | **41** | o | **57** | s | **73** | Ç | **89** | < |
| **10** | Z | **26** | 2 | **42** | T | **58** | . | **74** | : | **90** | İ |
| **11** | e | **27** | L | **43** | 5 | **59** | 7 | **75** | V | **91** | ş |
| **12** | k | **28** | F | **44** | 4 | **60** | D | **76** | ( | **92** | @ |
| **13** | ç | **29** | N | **45** | M | **61** | c | **77** | 8 | | |
| **14** | l | **30** | I | **46** | J | **62** | # | **78** | j | | |
| **15** | a | **31** | ? | **47** | ğ | **63** | 3 | **79** | g | | |

Şifrelemede ise şöyle bir yöntem kullanılmıştır; öncelikle şifrelenmek istenen karakter girildiğinde sistem tarafından Karakter Seti tablosundan rasgele atanmış id'si tespit edilmiş daha sonra mod 16 ya göre bölme işlemi uygulanmıştır. Burada elde edilen tam kısım ve kalan kısımlar yakalanmış bu değerler ise ikişer bit halinde bitişik yazılmış böylece bir karakter sistem tarafından 4 bit olarak şifrelenmiştir.

Örneğin "V" karakteri id

75 olarak atandığı için (75 ≡ mod 16 )

**Tam Kısım →04 Kalan Kısım →11 olarak bulunur.**

Böylece "V" karakterinin sistem tarafından şifrelenmiş hali **0411** olur.

Verilerin şifrelenmiş hali ise Şekil 5.1' de görülmektedir.


**Şekil 5.1.** Verilerin Şifrelenmesi İşlemi

### 5.1.2 Verilerin Görüntü İçerisine Gizlenmesi

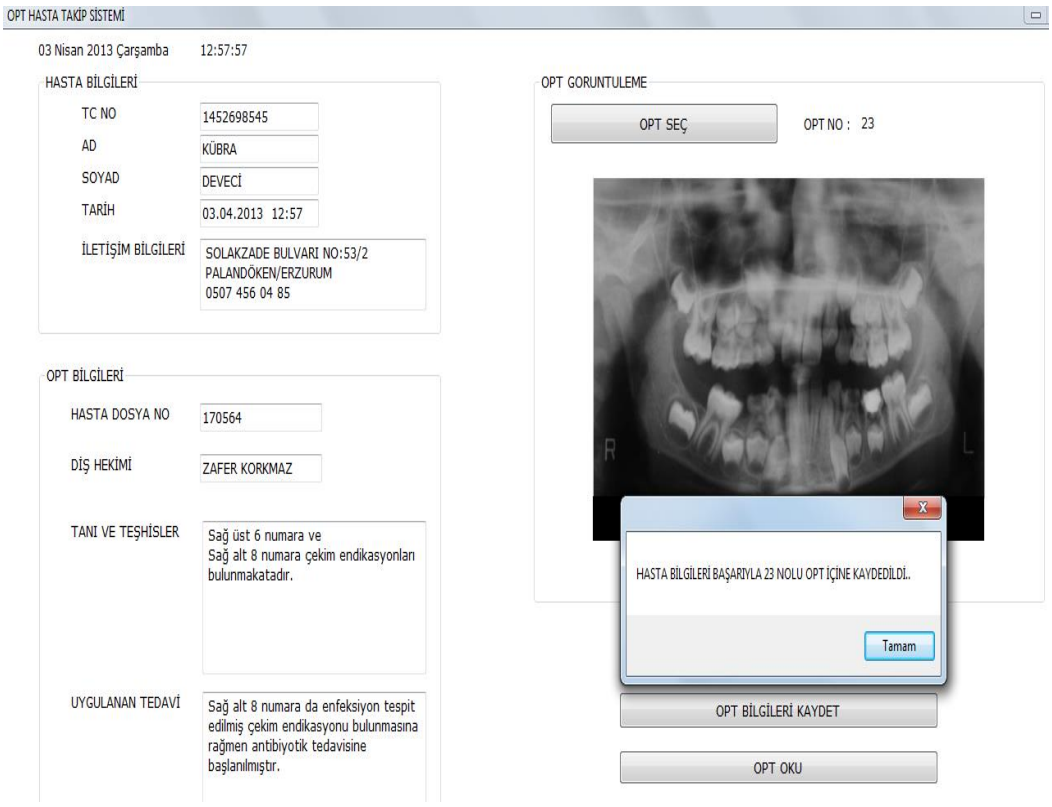
Diş hekimliği alanında kullanılan ve OPT olarak adlandırılan (Şekil 5.2.) görüntüler standart olarak 1000x1512 boyutundadır. Genellikle bu görüntülerin 845x1500'lük kısımları kullanılırken geri kalan özellikle de alt kısımları kullanılmaz ve bu piksellerin renk değeri 0'dır yani siyah renkle kaplıdır ve içerisinde herhangi bir bilgi yoktur. Bu alan verilerin kaydedilmesi ve tekrar şifrelerin çözülüp okunması için en elverişli bölümdür. Aksi takdirde OPT görüntüsünün rasgele herhangi bir yerine veri eklemek görüntünün yapısını bozabilir, veri kayıplarına yol açabilir. Bu da özellikle sağlık alanında hiç istenmeyen bir sonuçtur.

Verinin kaydedilecek yerinin belirlenmesi sadece görüntünün yapısını bozmamak için değil aynı zamanda kaydedilen verilerin tekrar geri getirilip okunması için de oldukça önemlidir. Çünkü hangi verinin nereye kaydedildiği bilinirse geri getirme işlemi de daha basit olacak formlarda yer alan bilgiler için her veri kendisine tahsis edilmiş kısımlardan çağırılacaktır.


**Şekil 5.2.** Verilerin gizleneceği bilgi taşımayan bölge

Bu uygulamada hasta kimlik bilgileri için satır aralığı 900-905 sütun 0-100, opt bilgileri için ise 907-950 sütun 0-100 pikseller arasındaki noktaların kullanılması planlanmıştır. Görüldüğü üzere OPT üzerinde ihtiyaçtan çok daha fazla veri gizlemeye müsait alan bulunmaktadır.

Veri güvenliği ve gizliliğini daha da artırmak için şifreleme yaparken oluşturulmuş olan anlamsız bilgileri anlaşılır hale getirmeden önceden belirlenmiş tamamı "0" renk koduna sahip pikseller arasına 1 piksel aralıklarla tam kısım ve kalan kısımla üretilmiş şifreli bilgi birer renk kodu kabul edilir. Şekilde görüldüğü gibi "Hasta Bilgileri Kaydet" butonuyla bu prensibe göre bilgiler önce şifrelenir ardından da şifreli hali OPT içine kaydedilir (Şekil 5.3.).

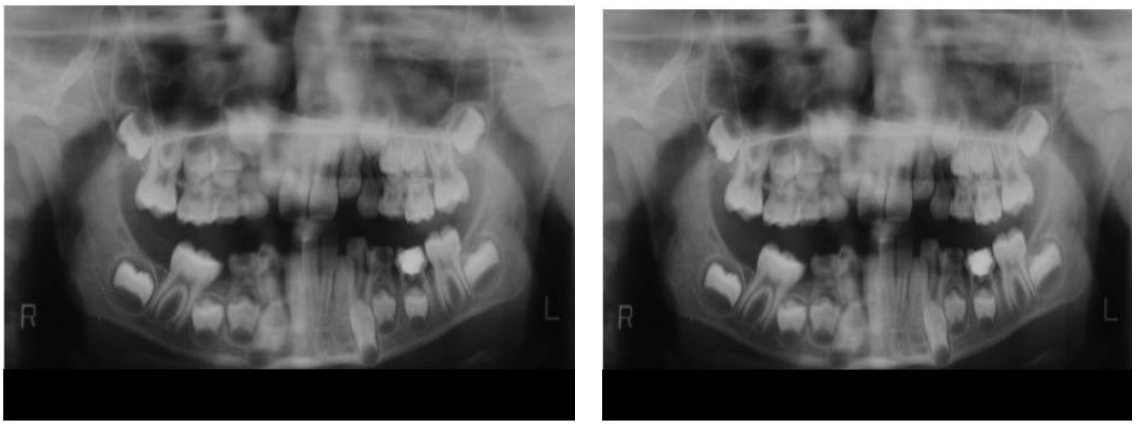



**Şekil 5.3.** Hasta Takip Sistemi Arayüzü

Veri gizleme işlemi ardından önceden tamamı 0(siyah) olan renk kodların bir kısmı aşağıdaki gibi olacaktır.

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 0 | 01 | 0 | 14 | 0 | 03 | 0 | 11 | 0 | 0 |

.

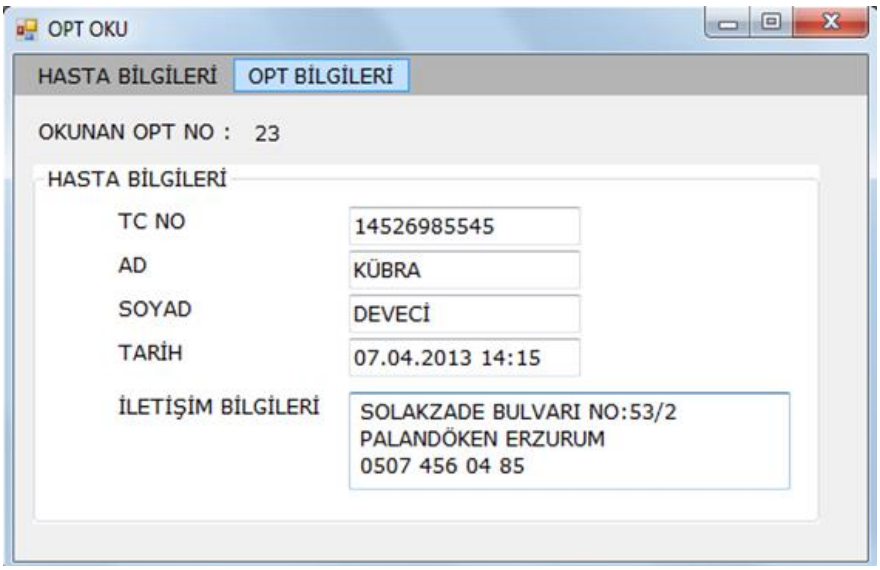


**Şekil 5.4.** Veri gizlenmiş resimler

Şekil 5.4.'te görüldüğü gibi resimler gri tonlu olduğu ayrıca 1 piksel aralıklarla renk kodları değiştirilip veriler gizlendiği için iki resim arasında gözle görülür bir fark bulunmamaktadır.

### 5.1.3. Verilerin Görüntü İçerisinden Okunması

Şifrelenmiş ve görüntü içerisine gizlenmiş verilerin sisteme tekrar getirilebilmesi için öncelikle şifreli verilerin geri alınması ardından da decript edilerek şifrelerin çözülmesi ve sisteme yüklenmesi gerekmektedir. Hangi bilginin nerede olduğu bilindiğine göre burada ki işlem nispeten daha basit olacaktır. Öncelikle başlangıç ve bitiş değerleri kaydedilmiş verinin muhafaza edildiği piksellerin koordinatlarını karşılayacak şekilde döngüler kurulmalı ardından şifreleme yaparken anlatılan sistematik bu kez tersten yapılarak 0 ve 92 arasında bir sayı elde edilecek bu sayınında karakter setinde hangi karakterin id'si olduğu bulunmuş olacaktır. Böylece karakter çözülmüş ve okunmuş olarak Şekilde olduğu gibi sisteme geri yüklenecektir (Şekil 5.5.).




**Şekil 5.5.** OPT Oku Modülü

### 5.1.4 Uygulamadan Elde Edilen Sonuç ve Öneriler

Bu uygulamada veri şifreleme ve veri gizleme teknikleri incelenerek sağlık alanında oldukça fazla kullanılan OPT görüntüleri üzerinde hasta kişisel bilgilerini, muayene bilgilerini ve isteğe bağlı birçok bilgiyi alıp öncelikle bunları şifreleyen böylece veri güvenliği ve gizliliğini arttıran daha sonra da bu bilgileri görüntünün yapısına herhangi bir zarar vermeden içine gizleyen ve istendiğinde tekrar bilgileri okuyup şifresini çözerek anlaşılır hale getiren bir yazılım geliştirilmiştir. Genelde birbirinin alternatifi olarak kabul edilen şifreleme ve gizleme teknikleri bu çalışmada birbirinin tamamlayıcısı olarak kullanılmış ayrıca Kriptoloji ve

steganografi günümüzde popüler kavramlar olmasına rağmen tıbbi görüntüleme alanında yeterince kullanılmadığı görülmektedir. Visual C# kodlarıyla geliştirilen program sayesinde bu kavramların tıbbi alanlar da kullanılabileceği bir yazılım önerilmiştir.

## 5.2 Kimlik Sahteciliğine Karşı Geliştirilen Uygulama

Günümüzde kimlik sahteciliği yaparak birçok suç işlenmekte bu nedenle çok sayıda masum insanda mağdur olmaktadır. Bu sahteciliği önlemek adına geliştirilen bu uygulama kişilerin TC kimlik numarasında yer alan sayıların sırasıyla şifrelenmesi ardından bu şifreli verilerin kimlik vb. üzerinde bulunan fotoğraf içerisine gizlenmesi son olarak şifreli verilerin geliştirilen yazılım aracılığıyla resim içerisinden okunması ve şifrelerin deşifre edilerek kimlik üzerinde bulunan TC kimlik numarasıyla kıyaslanarak doğrulanması prensibiyle çalışmaktadır.

### 5.2.1 Verilerin Şifrelenmesi

Verilerin Şifrelenmesi işlemi için birisi özgün olmak üzere iki adet farklı şifreleme algoritması kullanılmaktadır. Bu algoritmalar uygulama içerisinde 1.Şifreleme Basamağı ve 2.Şifreleme Basamağı adlarıyla isimlendirilmişlerdir. İlerleyen bölümde detaylı olarak anlatılan şifreleme algoritmaları şu aşamada kısaca açıklanırsa 1.şifreleme basamağı, bu uygulama için geliştirilmiş özel bir şifreleme algoritmasıdır. Bu işlemin ardından şifreli 1.algoritmayla şifrelenmiş veriler 2.şifreleme basamağında RSA şifreleme algoritmasıyla bir kez daha şifrelenmesiyle veri şifreleme işlemi tamamlanır.

### 5.2.2. Şifreleme Algoritması

1. Şifreleme basamağı algoritmik olarak 3 aşama şeklinde tasarlanmıştır.

**1.Adım: Yeniden Sıralama**

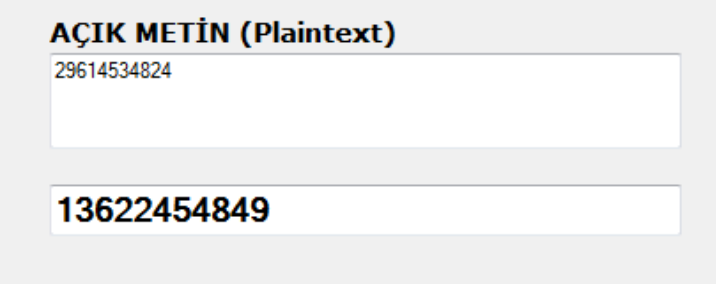
Giriş kısmında bahsedilen şifreleme algoritması tasarlanırken ilk olarak şifrelenecek ve 11 basamaktan oluşan TC kimlik numarasının her bir hanesi birer birer ele alınmıştır. Bu sayıların şifrelenmesi işleminde aşağıdaki resimde görülen ve Karakter Seti adı verdiğimiz tablodan yararlanılmıştır. Her bir karakterin ascii koduna benzer bir sayısal değer taşıdığı bu Karakter Setinin ascii koduna karşılık olarak tercih edilmesinin sebebi ascii kod tablosunda

ascii değerlerin belirli bir mantığa göre hareket etmesi Karakter Setinin ise tamamen random olarak oluşmuş olmasıdır.

**Tablo 5.1.** Karakter Seti

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | n | **16** | / | **32** | h | **48** | P | **64** | ! | **80** | p |
| **1** | 9 | **17** | 0 | **33** | , | **49** | r | **65** | K | **81** | v |
| **2** | * | **18** | = | **34** | % | **50** | 6 | **66** | U | **82** | b |
| **3** | E | **19** | space | **35** | B | **51** | ) | **67** | \ | **83** | Y |
| **4** | d | **20** | m | **36** | ı | **52** | S | **68** | ^ | **84** | 1 |
| **5** | + | **21** | - | **37** | Ş | **53** | H | **69** | ü | **85** | " |
| **6** | O | **22** | z | **38** | ' | **54** | i | **70** | C | **86** | & |
| **7** | R | **23** | Ğ | **39** | > | **55** | Ü | **71** | A | **87** | Ö |
| **8** | f | **24** | t | **40** | ö | **56** | _ | **72** | G | **88** | u |
| **9** | ; | **25** | y | **41** | o | **57** | s | **73** | Ç | **89** | < |
| **10** | Z | **26** | 2 | **42** | T | **58** | . | **74** | : | **90** | İ |
| **11** | e | **27** | L | **43** | 5 | **59** | 7 | **75** | V | **91** | ş |
| **12** | k | **28** | F | **44** | 4 | **60** | D | **76** | ( | **92** | @ |
| **13** | ç | **29** | N | **45** | M | **61** | c | **77** | 8 | | |
| **14** | l | **30** | I | **46** | J | **62** | # | **78** | j | | |
| **15** | a | **31** | ? | **47** | ğ | **63** | 3 | **79** | g | | |

Gerçekleştirilen uygulamada kullanılacak TC kimlik numarası 29614534824 şeklinde belirlenmiş standart 11 haneli bir TC kimlik numarasıdır. Giriş kısmında belirtildiği gibi şifreleme işleminde kullanılacak yöntem basamakların birer birer ele alınarak önce sayıların yeniden sıralanması ardından gerekli işlemler yapıldıktan sonra görsel içerisine gizlenmesidir.




**Şekil 5.6.** Rasgele sıralı TC numarası

29614534824 olarak belirlenmiş TC kimlik numarasının yazılım tarafından random olarak yeniden sıralanmış hali yukarıda ki ekran çıktısında da görüldüğü gibi 13622454849 şeklinde

oluşturulmuştur. Bu aşamadan sonra şifreleme ve ardından deşifre edilme işlemlerinde kullanılacak olan sayısal ifade 13622454849 olacaktır.

**Tablo 5.2** Verilerin Karakter Setindeki Değerleri

| SAYI | KARAKTER SETİ |
|---|---|
| 1 | 84 |
| 3 | 64 |
| 6 | 50 |
| 2 | 26 |
| 2 | 26 |
| 4 | 44 |
| 5 | 43 |
| 4 | 44 |
| 8 | 78 |
| 4 | 44 |
| 9 | 93 |

**2.Adım: Karakter Seti**

Tablo 5.2 ’de yeniden sıralanmış sayı dizisinin birebir karşılıkları Karakter Setinden alınarak oluşturulmuştur. Böylece bu aşamadan sonra uygulamada kullanılacak olan TC kimlik numarasının içerisinde yer alan rakamların yeniden sıralanışı ve karşılıkları belirlendikten sonra sistem tarafından her bir sayının anlamı tabloda belirlendiği gibidir.

Kısaca TC kimlik numarasının artık sistem için bir rakamlar topluluğu değil bunların birer birer parçalandığı ve yeni değerler atandığı söylenilebilir. Bu aşamada ascii kod karşılıklarının tercih edilmemesinin sebebi ascii kod tablosunun belli bir sistematiğe göre Karakter Setinin ise tamamen rasgelelilik üzerine tasarlanmış olmasıdır. Örneğin 0 karakterinin ascii karşılığı 48 olup 9’a kadar devam eden rakamlar dizisinin ascii karşılık değerleri de ardışık olarak devam etmesi şifre güvenliği açısından bu aşamada bir handikap olarak görülmektedir. Karakter Setinde yer alan sayıların kod olarak karşılıkları belirtildiği üzere tamamen random olarak oluşturulması aralarında hiçbir yakınlık ve sistematiklik bulunmamasından dolayı tercih edilmiştir.

**3.Adım: Kontrol Biti**

Geliştirilen şifreleme algoritması ile ilk iki adımda şifrelenecek ve 11 haneden oluşan TC kimlik numarasını oluşturan sayılar birer birer ele alınmış ve 11 basamak yeniden sıralanmıştır. Ardından her bir sayının, uygulama için geliştirilmiş Karakter Setinde karşılık gelen değerleri bulunmuştur. Bu aşamada uygulanacak olan 3.Adım güvenlik geliştirmesi ise Kontrol Biti adımı olarak isimlendirilmiştir. Kontrol Biti adımında, her bir sayının Karakter Setindeki bulunmuş olan karşılıkları ele alınmıştır. Bu değerler ikili kod (binary) sayı sistemine dönüştürülmüştür. 1 ve 0 değerlerinden oluşan Sekiz bitlik bu ifade içerisinden 1 değerlerinin sayısı bulunarak kontrol biti ifadesinin karşılığı oluşmuştur. Kontrol bitinin sistemde ki önemi ve gerekliliği görsel içerisinde şifreli olarak gizlenmiş verinin deşifre edilmesi kısmında daha detaylı olarak açıklanacak olsa da deşifre edilen verinin doğru veri olup olmadığının kontrolörüdür denilebilir. Kontrol biti olarak binary ifadeden elde edilen değer kullanarak karakter seti değerinden başka bir sayıya dönüşümü yapılır.

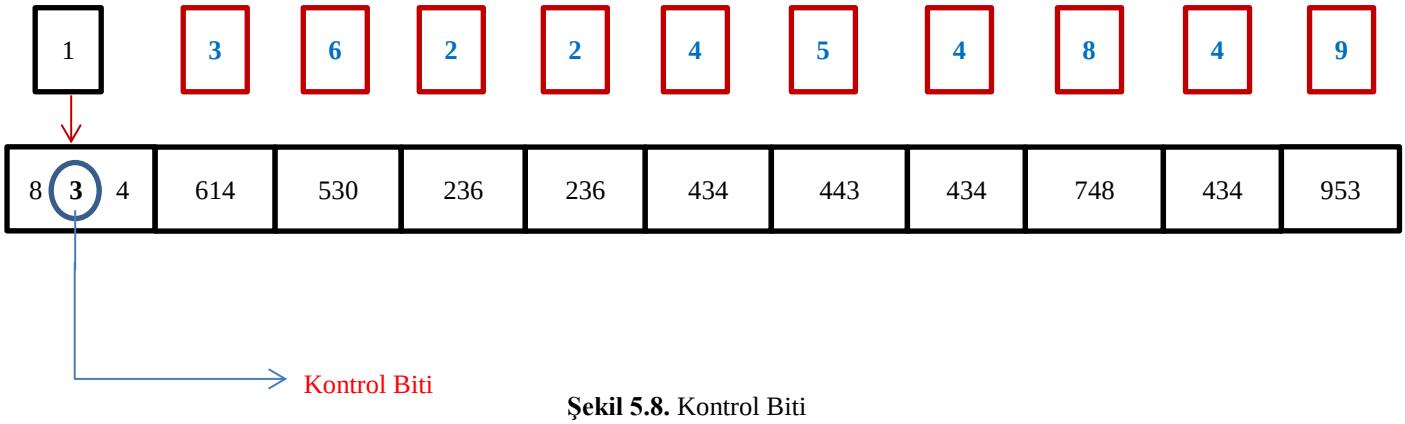
Alt kısımda bulunan şekil 5.7.'de, uygulama için kullanılacağı belirtilen TC kimlik numarasının yeniden sıralanmış hali ve içerdiği rakamlar **1.Sütun** da yeşil renk tonuyla, her bir sayının ascii kod benzeri bir yapıyla oluşturulduğu Karakter Setindeki karşılıkları **2.Sütun** da siyah renk tonuyla, Karakter Seti değerlerinin ikili kod sistemine dönüştürülmesiyle bulunan değer **3.Sütun** da mavi renk tonuyla ve son olarak ikili kod karşılıklarından elde edilen Kontrol biti **4.Sütun** da kırmızı renk tonuyla gösterilmiştir.

| 1.SÜTUN | 2.SÜTUN | 3.SÜTUN | 4.SÜTUN |
|---|---|---|---|
| 1 | 84 | 01010100 | 3 |
| 3 | 64 | 01000000 | 1 |
| 6 | 50 | 00110010 | 3 |
| 2 | 26 | 00011010 | 3 |
| 2 | 26 | 00011010 | 3 |
| 4 | 44 | 00101100 | 3 |
| 5 | 43 | 00101011 | 4 |
| 4 | 44 | 00101100 | 3 |
| 8 | 78 | 01001110 | 4 |
| 4 | 44 | 00101100 | 3 |
| 9 | 93 | 01011101 | 5 |

**Şekil 5.7.** Sayı Dönüşümleri

Kontrol bitlerinin belirlenmesi işleminden sonra bu değerler kullanılarak yeni sayıların üretilmesi işlemi yapılır. Örneğin şifreleme algoritmasının 1.Adımında TC kimlik numarasında yer alan sayıların yeniden sıralanmasıyla elde edilen sayı dizisinin ardından 2.Adımda dizinin ilk sayısı olan 1 rakamının Karakter Setinde yer alan karşılığının 84 olduğu görülmektedir. 3.Adımda ise kontrol biti denetimi yapılmaktadır. Böylece ilk etapta 84 değerinin ikili kod sistemine göre dönüşümü yapılır ve 8 bit olarak elde edilen değerden 1 bitleri sayılarak kontrol biti elde edilir. Bu işlemin ardından 1 sayısının kontrol biti 3 olarak belirlenmiş olur. Bu ifadenin anlamlı bir sayı haline getirilmesi işlemi ise kontrol bitlerinin, orijinal sayıların Karakter Setindeki karşılıklarının arasına yazılması ile yapılır. 1 sayısı için bu yapılırsa Karakter Setinde ki karşılığı olan 84 değeri ve kontrol biti olan 3 değeri kullanılarak 834 sayısı üretilir. Kontrol biti olan 3 ifadesinin sayının onlar basamağına yazıldığına dikkat edilmelidir. Böylece 3 basamaklı yeni bir sayı elde edilmiş olur. Şifreleme algoritmasının en başından itibaren incelenirse 1 sayısı önce 84 ardından 834 sayısına dönüşmüştür. Bu aşamadan sonra 1 sayısının sistem için karşılığı 834 sayısıdır denilebilir. TC

kimlik numarasında yer alan tüm sayıların yeniden sıralanmış ve kontrol bitlerinin hesaplanmasından sonra üretilmiş halleri şekil 5.8.' de yer almaktadır.




**Şekil 5.8.** Kontrol Biti

Sahte kimlik tespiti yazılımı uygulamasının ilk modülü olan verilerin şifrelenmesi kısmının ilk basamağı olan ve 3 adımdan oluşan 1.şifreleme basamağı, tek basamaklı rakamlardan tamamen birbirinden bağımsız ve aralarında hiçbir sistematiğin bulunmadığı 3 basamaklı yeni sayıların üretilmesiyle tamamlanmıştır. 2.basamakta ise şifreli bu sayıların RSA şifreleme algoritması ile yeniden şifreleme işlemi yer almaktadır. Böylece sistemin çok kaliteli bir güvenlik düzeyi sunması hedeflenmektedir.

## 5.3. RSA Şifreleme Algoritması

### 5.3.1. Anahtar Üretme Algoritması

İki büyük asal sayının üretilmesi ve buradan yola çıkarak uygulanan matematiksel işlemlerin ardından gizli ve açık anahtarların edinilmesine dayanan RSA şifreleme algoritmasının kullanıldığı bu uygulamada öncelikle random olarak P ve Q sayıları yazılım tarafından oluşturulur. Ardından Gizli (Private) ve Açık (Public) anahtarlar şu algoritma ile elde edilir.

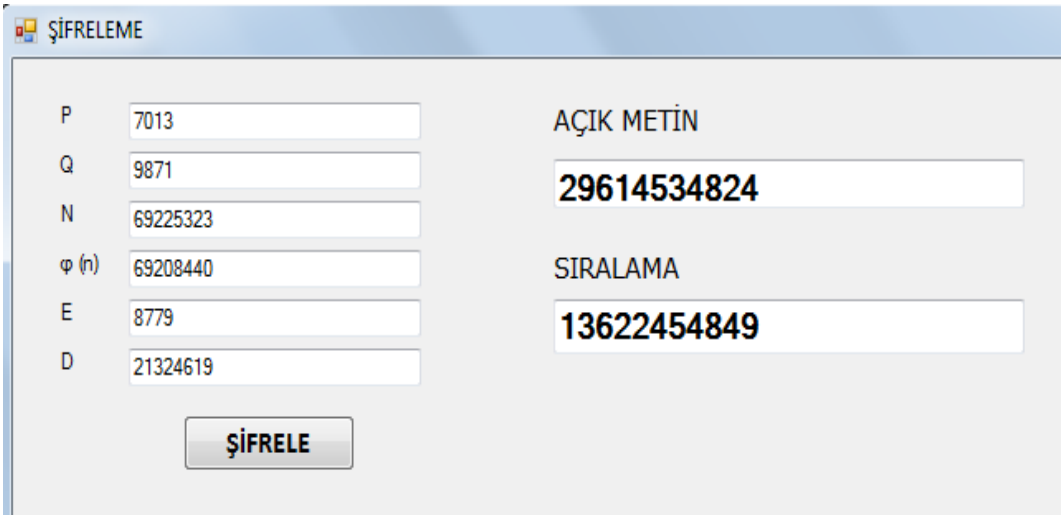
1) **P** değeri **7013** asal sayısı, **Q** değeri ise **9871** asal sayısı olarak random olarak yazılım tarafından üretilmiştir.

**2)** P ve Q sayılarının çarpma işlemine tabi tutulmasıyla **N** anahtarı elde edilir. Geliştirilen uygulamada 7013 * 9871 işleminin gerçekleştirilmesiyle N anahtarı olarak **69225323** sayısı oluşmuştur.

**3)** P ve Q sayılarının sayısal olarak 1 eksik değerlerinin çarpılması ile **φ (n)** değeri hesaplanır. φ (n) değerinin hesaplanması bu aşamadan sonra geliştirilecek olan E ve D anahtarlarının üretilmesinde önemli bir yer kapsamaktadır. Geliştirilen uygulamada (7013 -1 ) * (9871 –1) işleminin gerçekleştirilmesiyle φ (n) değerinin **6908440** olarak hesaplandığı görülmektedir.

**4)** **E** anahtarı belirlenirken ise izlenen yol şöyledir; E anahtarının değeri 1<E<N formatına uygun olarak seçilmeli ve seçilen değer E ve φ (n) değerleri için aralarında asal olan bir sayısal ifade olmalıdır. Değerlerin şifrelenmesi kısmında kullanılacak olan E anahtarı sistem tarafından aşağıdaki ekran çıktısında olduğu gibi **8779** olarak belirlenmiştir.

**5)** Son olarak değerlerin deşifrelenmesi olayında kullanılacak olan **D** anahtarının üretilmesi işleminde ise E*D≡1(mod φ (n)) formülüne uygun olarak anahtar üretimi yapılır. Daha önce oluşturulan anahtarlar bu formülde yerine koyulduğunda 8779*D=1(mod 6908440) işleminin sonucu olarak **21324619** olarak üretilmiştir.




**Şekil 5.9.** Anahtar Üretimi Arayüzü

Sistem tarafından RSA şifreleme algoritması ilke ve prensiplerine uygun olarak geliştirilen yazılım aracılığıyla mesajların şifrelenmesi ve ardından deşifre edilmesi işlemlerinde kullanılacak olan anahtarlar şekil 5.9. 'da görüldüğü gibi üretilmiştir. Ayrıca güvenliği bir kademe daha yukarıya taşımak içinde şifrelenecek metinin öncelikle random bir sırayla yeniden yazılması ve bu sıralamaya göre oluşan yeni metnin şifrelenmesi yöntemi tercih edilmiştir. Tabi ki veriler deşifre edilirken de önce metin orijinal haline dönüşümü yapılacak ardından verilerin şifreleri çözülerek ve orijinal metne ulaşılacaktır.

### 5.3.2. Verilerin RSA ile Şifrelenmesi

RSA algoritması prensiplerine göre anahtarların üretilmesinin ardından verilerin şifrelenmesi ve bu şifrelerin çözülmesi işlemlerinde kullanılacak olan açık ve gizli anahtarlar belirlenmiştir. Bu işlemin ardından ilk olarak verilerin şifrelenmesi işlemi yapılabilir. Aşağıda bulunan ekran çıktısından da görüleceği gibi sayılar 1.şifreleme basamağında olduğu gibi yine birer birer ele alınmıştır. Ele alınan bu sayılar 1.şifreleme basamağında şifrelenmiş halleriyle 2.şifreleme basamağında RSA yöntemi ile tekrar şifrelenmektedir.

Örneğin, TC kimlik numarasının yeniden sıralanmasından ardından 6.sırada yer alan 4 sayısının şifrelenme işlemi ayrıntılı olarak incelenmek istenirse öncelikle 1. şifreleme basamağında nasıl şifrelendiğine göz atılmalıdır. 1.şifreleme basamağında bulunan ve 3 adımda şifrelenen 4 sayısı önce 2.adımda yer alan Karakter Seti değerine bakılmalı ardından bu değerinde 44 olduğu anlaşılmaktadır. 44 sayısının ikili koda dönüştürülmesi ve 3.adımda açıklanan işlemin uygulanmasının ardından kontrol biti olarak 3 değerinin hesaplandığı görülebilir. Böylece 1.şifreleme basamağında 4 sayısının şifreli hali 434 olarak bulunur. 2.şifreleme basamağında yer alan RSA yöntemi ile şifrelemede kullanıcı tarafından şifrelenecek ifadenin 4 sayısı olduğu düşünülse de sistemin altında çalışan yapıda artık 4 değeri için 434 ifadesi yer almaktadır.

Bu nedenle 6. sırada yer alan 4 sayısının yani 1.şifreleme basamağına göre 434 değerinin RSA şifreleme algoritmasıyla 2.şifreleme basamağında şifrelenmesi işlemi ise şu şekildedir;

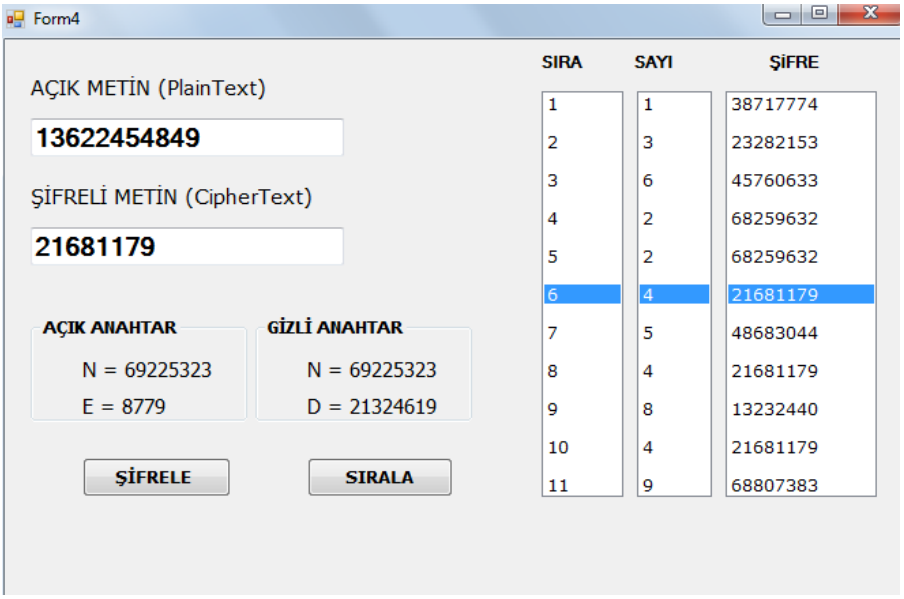
Anahtar üretme algortiması bölümünde ayrıntılı şekilde ele alındığı gibi ilk şifreleme basamağında şifrelenmiş verilerin 2.kez daha güçlü ve uluslararası literatür de kabul görmüş

bir şifreleme standardı olan RSA yöntemiyle şifrelenmesi ile verilerin güvenliğinin daha da üst bir noktaya taşınmasının hedeflendiği bu basamakta ilk adım anahtarların üretilmesi işlemidir. Aşağıda ki ekran çıktısından da görüldüğü gibi Açık ve Gizli anahtarlar ilk adımda belirlenmiştir.

RSA şifreleme algoritmasında $C = M^{e} \pmod{N}$ formülü kullanılmaktadır.

Bu formüle göre veriler şifrelenmek istenirse M değeri şifrelenecek mesajı, E ve N değerleri de açık anahtarları ifade etmektedir. 6. sırada yer alan 4 sayısının yani yukarı kısımlarda açıklandığı gibi daha doğru ifade ile 434 sayısının şifrelenmiş hali;

$434^{8779} \pmod{69225323}$ işlemi uygulandığında **21681179** sonucuna ulaşıldığı şekil 5.10'da görülmektedir.




**Şekil 5.10.** Şifreleme Arayüzü

TC kimlik numarasında yer alan ve sırası random olarak yeniden sıralanmış verilerin RSA şifreleme algoritması ile şifrelenmesi yukarıda ki görselde görüldüğü üzere 11 aşamalık bir döngü kullanılarak tekrarlanmış formülde belirtilen değerlerin yerlerine anahtar ve veriler yerleştirilerek 1.şifreleme basamağında şifrelenmiş verilerin 2.şifreleme basamağında RSA şifreleme standardı ile bir kez daha şifrelenmesi işlemi yapılmıştır.

**Tablo 5.3.** Şifreleme Basamakları

| ORİJİNAL | RANDOM SIRA | 1.ŞİFRELEME | 2.ŞİFRELEME (RSA) |
|---|---|---|---|
| **2** | 1 | 834 | 38717774 |
| **9** | 3 | 614 | 23282153 |
| **6** | 6 | 530 | 45760633 |
| **1** | 2 | 236 | 68259632 |
| **4** | 2 | 236 | 68259632 |
| **5** | 4 | 434 | 21681179 |
| **3** | 5 | 443 | 48683044 |
| **4** | 4 | 434 | 21681179 |
| **8** | 8 | 748 | 13232440 |
| **2** | 4 | 434 | 21681179 |
| **4** | 9 | 953 | 68807383 |

Tablo 5.3 ’de ise ilk iki sütunda sayıların orijinal ve yeniden rasgele sıralanmış hali, 3. sütunda bu sayıların 1.şifreleme basamağına göre şifrelenmiş hali görülmektedir. Son sütunda ise ilk şifreleme basamağıyla şifrelenmiş verilerin RSA şifreleme algoritmasıyla tekrar şifrelenmiş değerleri görülmektedir. Bu aşamayla birlikte verilerin şifrelenmesi işlemi tamamlanmıştır.

## 5.4. Verilerin Gizlenmesi

Sayıların iki basamaktan oluşan şifreleme işleminin tamamlanmasının ardından uygulamamın verilerin şifrelenmesi aşaması tamamlanmış ve ikinci modül olan verilerin gizlenmesi kısmına geçilmiştir. Bu aşamada şifreli verilerin Sahte Kimlik Tespiti uygulamasında örnek olarak kullanılan vesikalık renkli resimler içerisine gizleme işleminin yapılma kısmına değinilecektir.

Öncelikle bu aşamada kullanılacak olan yazılımın en önemli üstünlüğü lokasyon yani yer belirleme işleminin de rasgele yapılmasıdır. Daha ayrıntılı açıklanacak olursa zaten sıralaması değiştirilmiş sayıların resim içerisinde gizlenecek noktaların adreslerinin de rasgele yapılması yani hangi verinin hangi adrese gizlendiğinin sistemden başka hiç kimsenin bilmemesidir. Doğal olarak her gizleme işlemi yapıldığında yine bu noktalar rasgele olarak

belirlenecek her defasında değişecektir. Verilerin şifrelenmesi kısmında detaylı olarak anlatıldığı gibi biri güçlü bir şifreleme algoritması olan RSA ile olmak üzere iki kez şifrelenen veriler, bu aşamada da resim içerisinde rasgele noktalara gizlenip tekrar bu adreslerden okunup deşifre edilebilmeleri sistemin güvenirliğini oldukça üst bir noktaya çıkarması sistemin en büyük üstünlüğü ve avantajı olarak göze çarpmaktadır.

Bu işlemlerin yapılma aşamaları ele alınırsa öncelikle kimliklerin de üzerinde yer alan vesikalık fotoğraf standardı 177x236 olarak kabul edilmiştir. Geliştirilen uygulama sahte kimlik tespitinde kullanılacak bir çalışma olduğu için veri gizleme aşamasında kimlik vb. belge üzerinde yer alan ve vesikalık fotoğraf standart ölçülerine sahip resimlerden yararlanılmıştır. 175x234 boyutunda ki bu resimlerde ki veri bütünlüğünü bozmamak adına resim üzerine iki satır ve iki sütun eklenerek bir çerçeve çizilmiş ve aşağıda gösteriliği gibi resmin yeni boyutu 177x236 ölçülerine ulaşmıştır.

175 x 234 177 x 236

**Şekil 5.11.** Veri gizleme işleminde kullanılan bölüm

Veri gizleme işlemi yapılmadan önce uygulamada kullanılacak olan ve üst kısımda ölçüleri açıklanarak yer verilen örnek vesikalık resmin orijinal piksel değerleri şekil 5.12. ’de görülmektedir. Resmin bir bölümü ele alınarak sadece bu kısmın piksel değerlerinin

verilmiştir. Ancak bundan sonra ki işlemlerin ardından da aynı kısım seçilerek piksel değerleri gösterilerek piksel değişimlerinin rahatça gözlemlenmesi ve kıyaslama yapılabilmesi imkanı sağlanmıştır.

| | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 18 | 17 | 21 | 16 | 10 | 16 | 14 | 14 | 13 | 16 | 14 | 15 | 13 | 13 |
| 13 | 14 | 16 | 15 | 20 | 15 | 13 | 13 | 14 | 14 | 12 | 14 | 13 | 12 |
| 13 | 14 | 15 | 15 | 18 | 16 | 13 | 12 | 13 | 14 | 13 | 12 | 15 | 10 |
| 12 | 15 | 17 | 15 | 14 | 13 | 13 | 14 | 14 | 11 | 11 | 11 | 15 | 14 |
| 11 | 14 | 15 | 15 | 12 | 15 | 14 | 12 | 13 | 13 | 15 | 16 | 9 | 11 |
| 14 | 15 | 13 | 14 | 14 | 16 | 13 | 14 | 14 | 17 | 13 | 14 | 13 | 11 |
| 17 | 16 | 14 | 14 | 13 | 15 | 13 | 13 | 13 | 14 | 15 | 12 | 11 | 16 |
| 13 | 19 | 13 | 11 | 11 | 14 | 14 | 12 | 13 | 12 | 14 | 9 | 10 | 10 |
| 14 | 14 | 12 | 13 | 12 | 12 | 13 | 13 | 13 | 17 | 11 | 10 | 11 | 7 |
| 15 | 15 | 14 | 13 | 13 | 13 | 11 | 16 | 14 | 11 | 14 | 15 | 15 | 14 |
| 11 | 12 | 14 | 11 | 11 | 13 | 16 | 15 | 15 | 14 | 11 | 16 | 13 | 14 |
| 13 | 15 | 14 | 12 | 14 | 14 | 14 | 16 | 17 | 20 | 14 | 14 | 13 | 13 |
| 18 | 15 | 16 | 15 | 16 | 18 | 20 | 18 | 20 | 23 | 20 | 15 | 13 | 14 |
| 19 | 16 | 15 | 15 | 16 | 18 | 17 | 19 | 19 | 22 | 18 | 15 | 12 | 11 |
| 20 | 13 | 14 | 16 | 15 | 15 | 13 | 16 | 18 | 19 | 17 | 12 | 12 | 12 |
| 19 | 17 | 18 | 17 | 12 | 12 | 9 | 15 | 17 | 14 | 14 | 14 | 14 | 12 |
| 18 | 17 | 16 | 14 | 11 | 14 | 11 | 15 | 13 | 15 | 11 | 12 | 12 | 10 |
| 17 | 19 | 16 | 14 | 16 | 16 | 15 | 16 | 14 | 15 | 14 | 15 | 10 | 11 |
| 18 | 14 | 17 | 16 | 17 | 16 | 15 | 17 | 17 | 18 | 16 | 16 | 13 | 14 |
| 18 | 15 | 16 | 17 | 17 | 17 | 13 | 15 | 15 | 15 | 16 | 15 | 15 | 15 |

**Şekil 5.12.** Orijinal Piksel Değerleri

**1.Adım:** 175x234 boyutunda standart ölçülere sahip vesikalık fotoğraf üzerinde iki satır ve iki sütundan oluşan çerçeve benzeri bir yapı eklenmesi işlemidir. 177x236 ölçülerine ulaşan bu yeni resim üzerinde yapılacak olan ilk işlem ise ilave edilen satır ve sütunlara yani bundan sonra ki kısımlarda da çerçeve adı verilecek olan yapıya renk değerlerinin atanmasıdır. 0 – 255 arasında renk değerlerinin atanarak siyah ve beyaz aralığındaki renk tonlarının oluşturduğu çerçevenin tüm kenarlarının piksel değeri şekil 5.13. ve şekil 5.14.'te ki ekran çıktılarında görülmektedir.

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 196 | 233 | 109 | 18 | 41 | 248 | 254 | 170 | 28 | 94 |
| 233 | 18 | 17 | 21 | 16 | 10 | 16 | 14 | 14 | 13 |
| 109 | 13 | 14 | 16 | 15 | 20 | 15 | 13 | 13 | 14 |
| 18 | 13 | 14 | 15 | 15 | 18 | 16 | 13 | 12 | 13 |
| 41 | 12 | 15 | 17 | 15 | 14 | 13 | 13 | 14 | 14 |
| 248 | 11 | 14 | 15 | 15 | 12 | 15 | 14 | 12 | 13 |
| 254 | 14 | 15 | 13 | 14 | 14 | 16 | 13 | 14 | 14 |
| 170 | 17 | 16 | 14 | 14 | 13 | 15 | 13 | 13 | 13 |
| 28 | 13 | 19 | 13 | 11 | 11 | 14 | 14 | 12 | 13 |
| 94 | 14 | 14 | 12 | 13 | 12 | 12 | 13 | 13 | 13 |
| 197 | 15 | 15 | 14 | 13 | 13 | 13 | 11 | 16 | 14 |
| 213 | 11 | 12 | 14 | 11 | 11 | 13 | 16 | 15 | 15 |
| 20 | 13 | 15 | 14 | 12 | 14 | 14 | 14 | 16 | 17 |
| 60 | 18 | 15 | 16 | 15 | 16 | 18 | 20 | 18 | 20 |
| 20 | 19 | 16 | 15 | 15 | 16 | 18 | 17 | 19 | 19 |
| 204 | 20 | 13 | 14 | 16 | 15 | 15 | 13 | 16 | 18 |

**Şekil 5.13.** Sol ve Üst Çerçeve

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| 5 | 6 | 10 | 9 | 10 | 10 | 13 | 15 | 17 | 16 | 16 | 151 |
| 0 | 1 | 5 | 9 | 10 | 12 | 14 | 15 | 16 | 16 | 17 | 10 |
| 0 | 0 | 4 | 9 | 11 | 13 | 15 | 15 | 16 | 16 | 18 | 24 |
| 1 | 5 | 7 | 8 | 11 | 14 | 16 | 17 | 16 | 16 | 19 | 31 |
| 1 | 4 | 7 | 14 | 18 | 23 | 21 | 18 | 16 | 19 | 24 | 130 |
| 5 | 6 | 9 | 15 | 19 | 21 | 19 | 16 | 13 | 17 | 21 | 201 |
| 7 | 8 | 11 | 14 | 18 | 19 | 16 | 13 | 9 | 13 | 18 | 180 |
| 6 | 7 | 10 | 14 | 16 | 17 | 12 | 9 | 6 | 10 | 15 | 189 |
| 3 | 4 | 8 | 13 | 15 | 15 | 10 | 6 | 3 | 9 | 14 | 192 |
| 2 | 4 | 8 | 13 | 14 | 13 | 8 | 5 | 3 | 10 | 16 | 17 |
| 3 | 6 | 10 | 12 | 14 | 12 | 7 | 5 | 4 | 12 | 19 | 35 |
| 5 | 8 | 13 | 12 | 13 | 12 | 7 | 5 | 5 | 14 | 21 | 89 |
| 2 | 6 | 10 | 17 | 11 | 7 | 7 | 9 | 8 | 15 | 23 | 194 |
| 1 | 5 | 9 | 14 | 9 | 5 | 5 | 8 | 8 | 15 | 22 | 237 |
| 1 | 4 | 7 | 10 | 5 | 2 | 4 | 8 | 8 | 15 | 22 | 167 |
| 0 | 2 | 6 | 7 | 2 | 0 | 3 | 8 | 8 | 15 | 22 | 0 |
| 0 | 2 | 5 | 5 | 1 | 0 | 4 | 9 | 9 | 16 | 22 | 43 |
| 2 | 3 | 5 | 6 | 2 | 2 | 6 | 10 | 10 | 16 | 22 | 130 |
| 3 | 3 | 5 | 8 | 5 | 4 | 8 | 12 | 11 | 17 | 23 | 210 |
| 2 | 4 | 6 | 9 | 5 | 6 | 10 | 12 | 12 | 16 | 23 | 160 |
| 4 | 9 | 12 | 15 | 14 | 18 | 21 | 21 | 15 | 15 | 21 | 121 |
| 4 | 10 | 11 | 14 | 14 | 16 | 21 | 19 | 14 | 16 | 20 | 121 |
| 5 | 10 | 12 | 13 | 13 | 16 | 21 | 19 | 15 | 17 | 21 | 140 |
| 6 | 10 | 11 | 13 | 12 | 15 | 21 | 19 | 15 | 18 | 22 | 193 |
| 186 | 103 | 18 | 221 | 41 | 97 | 31 | 243 | 224 | 122 | 67 | 160 |

**Şekil 5.14.** Sağ ve Alt Çerçeve

**2.Adım:** Eklenen çerçeveye 0-255 aralığında renk değerleri atanması işleminin ardından çerçevenin kullanılması adımı gerçekleştirilir. Bu adımda çerçevenin üst kenarında bulunan

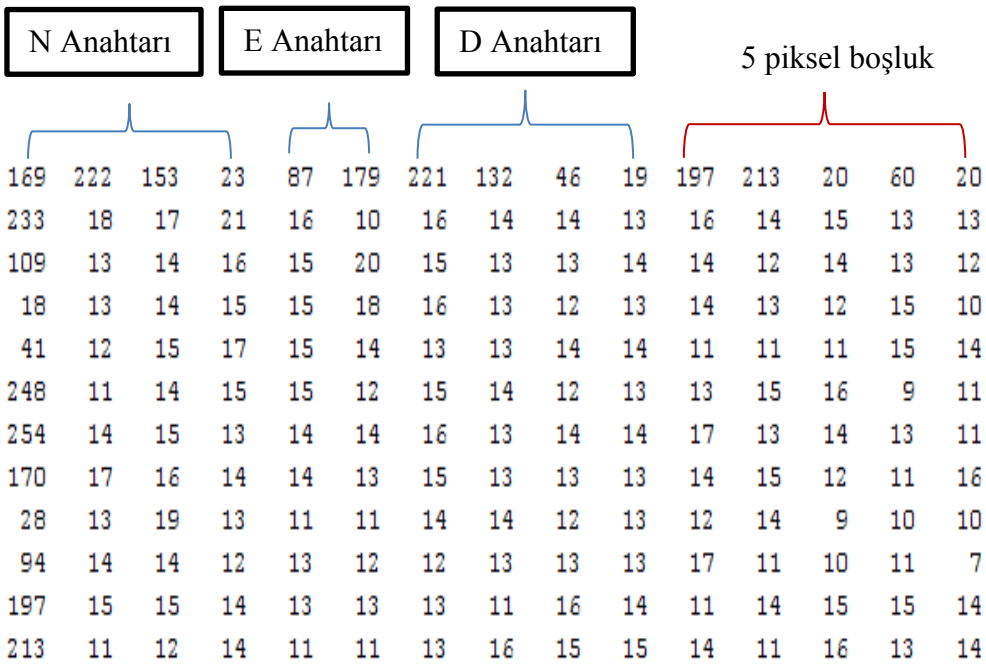
piksellerin bir bölümü anahtarların ve TC kimlik numarasında yer alan sayıların şifrelenmiş hallerinin saklanacağı daha doğru bir ifade ile gizleneceği noktaların adreslerinin tutulacağı pikseller olarak belirlenmiştir. Bu noktaların tasarımı ise şöyle yapılmıştır;

- İlk 10 piksel [0-10] aralığı N, E, D Anahtarlarının değerlerinin gizlendiği pikseller olarak atanmıştır.
- Daha sonra 0-10 aralığında sistem tarafından bir sayı belirlenerek piksellerin atlandığı yani aralık bırakılması ile bir boşluk oluşturulması; bu uygulamada bu değer **5** olarak atanmıştır.
- Ardından 11 adet şifreli sayı gizleneceği için, 11 sayının apsisinin ve ordinatının bulunduğu noktanın yer aldığı 22 adet adresleme noktası atanmıştır.

**3.Adım:** İlk olarak ilk 10 pikselin kullanıldığı anahtarların gizlenmesi işlemi yapılmalıdır. Çünkü anahtarlar şifreleri verilerin gizlenmesinin ardından okunup şifrelerinin çözülmesi işleminde oldukça önemlidir. Sistem ilk aşamalarda belirtildiği gibi anahtarlamayı her defasında random değerler ile yaptığı için anahtar değerlerinin saklanmaması durumunda deşifre edilen verilerin orijinal değerler olmama ihtimali oldukça yüksektir. Buda sistemin yanlış çalışmasına yol açar.

Atama işleminin yapılması anahtarların gizlenmesi işleminde analiz edilecek olunursa; N=69225323, E=8779 ve D=21324619 olarak anahtarlar daha önceki bölümlerde anlatılığı gibi belirlenmiştir.

Atama işlemi orijinal pikseller ile atanacak değerlerin ikişerli gruplar halinde değiştirilmesi ilkesine dayanmaktadır. Burada dikkat edilmesi gereken nokta orijinal piksel iki basamaklı ise doğrudan değiştirilmesi, üç basamaklı ise birler ve onlar basamağının değiştirilmesidir. Değişimin ardından oluşabilecek 255 değerinin aşılması durumunda ise bir problemle karşılaşılma ihtimalinin olmasından dolayı kontroller yapılmalı ve bu durumda da yüzler basamağında ki değerle oynanarak bu sorun giderilmelidir.




**Şekil 5.15.** Anahtarların Gizlenmesi

Şekil 5.15. 'te yer alan ekran çıktısında görüldüğü üzere çerçevenin üst kenarı diye adlandırılan satırda yer alan noktaların piksel değerleri verilmiştir. Buna göre ilk 10 pikselin anahtar değerlerinin gizlenmesi için kullanıldığı görülmektedir. İlk etapta tüm çerçevenin 0-255 aralığında ki değerler ile doldurularak siyah-beyaz aralığında ki renk tonlarıyla oluşturulması işlemi yapılmıştır. Ardından anahtarlar yerleştirilmiş ve 2.adımda bahsedildiği gibi 0-10 arasında değer alabilecek rastegele bir değer boyutunda boşluk bırakılmış daha dorğu bir ifade ile piksellere bir atama yapılmamış 0-255 aralığında rasgele doldurulan değerlerden bu adreslere atanan random değerler aynen korunmuştur.

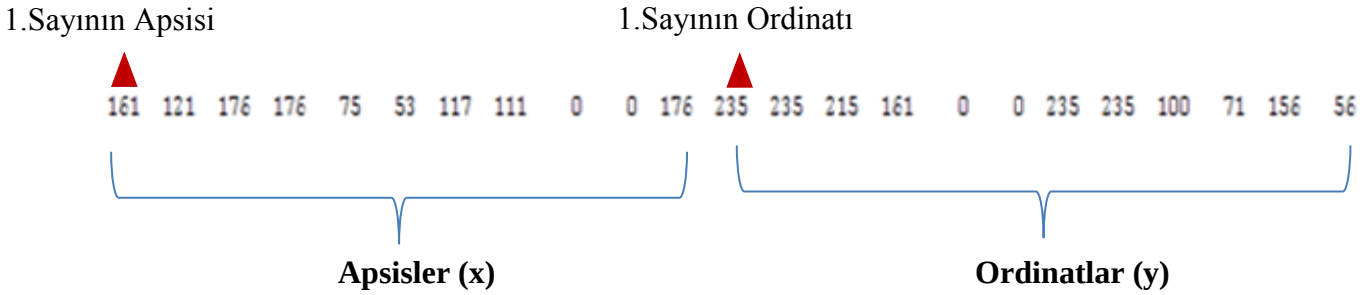
**Tablo 5.4.** Anahtarların Gizlendikleri Adres ve Değerleri

| **N ANAHTARI** | | | **E ANAHTARI** | | | **D ANAHTARI** | |
|---|---|---|---|---|---|---|---|
| *ADRES* | *DEĞER* | | *ADRES* | *DEĞER* | | *ADRES* | *DEĞER* |
| (1,0) | **1**69 | | (5,0) | 87 | | (7,0) | **2**21 |
| (2,0) | **2**22 | | (6,0) | **1**79 | | (8,0) | **1**32 |
| (3,0) | **1**53 | | | | | (9,0) | 46 |
| (4,0) | 23 | | | | | (10,0) | 19 |
| **69225323** | | | **8779** | | | **21324619** | |

Tablo 6.3 ’de ise ilk 10 pikselde gizlendiği ifade edilen N,E ve D anahtarlarının gizlendiği noktaların adresleri yani x ve y ekseni şeklinde ifadeleri ve bu adreslerde ki değerleri yer almaktadır. Anahtarların tam değerleri en alt satırda verilirken anahtarların pikseller üzerinde gizlenmiş hali ise renkli tonlarla gösterilmiştir.

**4.Adım:** Eklenen çerçevenin üst satırında yer alan ve anahtarların gizlenmesi için kullanılan ilk 10 pikselin ardından önceki adımlarda anlatılan random bir değerin de eklenmesiyle toplamda 15 piksel kullanılmıştır. Verilerin şifrelenmesi bölümünde detaylı olarak anlatılmış olan 11 sayının şifrelenmiş halleri ise sistemde tutulmaktadır. Anahtarların gizlenmesi işleminin tamamlanmasının ardından 11 sayının şifreli hallerinin gizlenmesi işlemi yapılmalıdır. Uygulamanın en başında bahsedildiği gibi sayıların gizleneceği noktaların belirli olmaması sistem tarafından çerçeve üzerinde rasgele noktalara atanması ve tekrar deşifre edilebilir olması uygulamanın önemli bir avantajıdır. Buradan yola çıkarak şifreli sayıların gizlenmesi işleminde ise ilk olarak sayıların gizleneceği noktaların adresleri bilinmeli ve sistem tarafından tutulmalıdır. Gizleme işleminin yapılması gereken 11 adet sayı olduğuna göre bu sayıların adresleri için 11 adet x ekseninde nokta ve 11 adet y ekseninde nokta bulunmalıdır. Toplamda ise bütününe adres adı verilirse 22 adet nokta adres olarak atanmalıdır. Adresleme satırı adı verilen çerçevenin üst satırı yine bu atama işlemi için kullanılarak toplamda 37 piksel anahtar değerleri, güvenlik değeri ( random değer) ve adresleme noktaları için kullanılmıştır.

11 adet şifreli TC kimlik numarasını oluşturulan sayıların resim içerisine gizlenmesi işleminde öncelikle sayıların gizleneceği noktaların adresleri tanımlanmalıdır. Daha önce de belirtildiği gibi lokasyon işlemi de rasgele olduğu için adresler sistem tarafından rasgele atanır. Çerçevenin eklenmesinin ardından 177x236 ölçülerine oluşan fotoğraf üzerinde ki çerçeve üzerinde gizlenecek olan sayılar için adresleme noktaları olarak sistem tarafından üretilen resim boyutuna göre 0-236 aralığında ki değerler kullanılmıştır.




**Şekil 5.16.** Adresleme Satırı

Yukarıda görüldüğü gibi sayıların gizleneceği noktaların adreslenmesi şekilde ki gibi yapılmıştır. 11 adet şifrelenmiş sayı için, 0-238 aralığında 11 apsis noktası yine 11 adet ordinat noktası sistem tarafından rasgele olarak atanmıştır.


**Şekil 5.17** Verilerin Resim Üzerinde Gizlendiği Noktalar

Ayrıntılı olarak nokta ve koordinatların gösterildiği şekil 5.17.' de anahtarların değerlerinin, 11 sayının gizlendiği noktaların adreslerinin ve 11 adet şifreli sayının resim üzerinde gizlendiği noktalar gösterilmiştir. Buna göre;

- [0-10] aralığında bulunan pikseller üzerinde **Gold** renk tonunda ve **A** harfiyle gösterilen bölümde N, E ve D Anahtarlılarının Değerleri,
- [11-15] aralığında bulunan pikseller üzerinde **Indigo** renk tonunda ve **B** harfiyle gösterilen bölümde random olarak belirlenen sabit piksel değerlerinin muhafaza edildiği boşluk değeri,
- [16-26] aralığında bulunan pikseller üzerinde **Maroon** renk tonunda ve **C** harfiyle gösterilen bölümde 11 adet sayının adreslerinin apsis değerleri,
- [27-37] aralığında bulunan pikseller üzerinde **Aqua** renk tonunda ve **D** harfiyle gösterilen bölümde ise 11 adet sayının adreslerinin ordinat değerleri yer almaktadır.

**Tablo 5.5.** Karakter Analizi

| KARAKTER SIRASI | SAYI | RENK KARŞILIĞI | PİKSEL APSİS | PİKSEL ORDİNAT |
|---|---|---|---|---|
| 1.KARAKTER | **1** | White | 161 162 163 164 | 235 235 235 235 |
| 2.KARAKTER | **3** | Blue | 121 122 123 124 | 235 235 235 235 |
| 3.KARAKTER | **6** | Cyan | 176 176 176 176 | 215 216 217 218 |
| 4.KARAKTER | **2** | Magenta | 176 176 176 176 | 161 162 163 164 |
| 5.KARAKTER | **2** | Yellow | 075 076 077 078 | 000 000 000 000 |
| 6.KARAKTER | **4** | Brown | 053 054 055 056 | 000 000 000 000 |
| 7.KARAKTER | **5** | Red | 117 118 119 120 | 235 235 235 235 |
| 8.KARAKTER | **4** | Green | 111 112 113 114 | 235 235 235 235 |
| 9.KARAKTER | **8** | Black | 000 000 000 000 | 100 101 102 103 |
| 10.KARAKTER | **4** | Purple | 000 000 000 000 | 071 072 073 074 |
| 11.KARAKTER | **9** | Silver | 176 176 176 176 | 156 157 158 159 |

Resim üzerinde 11 adet şifreli sayının gizlendiği noktaların işaret edildiği piksellerin adresleri ise tabloda ayrıntılı olarak gösterilmektedir (Tablo 5.5.). Unutulmamalıdır ki adresleme satırında (22 piksel) tutulan değerler sayıların değerleri değil, gizleme işleminin yapıldığı noktaların adresleridir. Tabloda belirtilen adreslerde ise bu kez sayıların değerleri yer

almaktadır. Her bir şifreli sayı 8 karakterden oluştuğu için bu bölümde de 3.adımda bahsedilmiş olan anahtarların değerlerinin gizlenmesi işleminde kullanılmış olan yöntem uygulanmıştır. Bir sonraki adım olan değerlerin gizlenmesi işleminde de yine bu metoda değinilmiştir.

**5.Adım:** Bu adımda ise anahtar değerlerinin, sayıların adreslerinin tutulması işlemin ardından nihayet şifreli sayıların resim üzerine eklenmiş olan çerçeve üzerine gizlenmesi işlemi yapılmıştır. Bu aşamada ise dikkat edilmesi gereken iki ilke vardır. Birincisi, adresleme satırı denilen ve 37 pikselden oluşan bu bölüme veri girişinin yapılmamasıdır. Bu nedenle adresleme yapılırken x ekseninde bulunan 37 piksel harici bırakılmış onun dışında ise çerçevenin her bölümüne adresleme yapılmasına imkân verilmiştir. İkinci nokta ise şifreli sayılar toplamda 8 karakterden oluştuğu ve 3.adımda olduğu gibi ikişerli gruplar halinde piksel değerlerinin değişimi yapıldığı için 4 hücre kullanılmasıdır.

Verilerin gizlenmesi işleminde örnek olarak 1.Karakter olan 1 sayısının gizlenmesi işlemi yapılırsa ilk olarak işe verinin saklanacağı adresin belirlenmesinden başlanmalıdır. Apsis ve ordinat değerlerinin tutulduğu 22 hücrelik adresleme satırına ulaşılmalıdır. Örnek olarak ele 1.karakter alındığı için adresleme satırının ilk hücresinde yer alan ilk karakterin apsisi olan (16,0) noktası, ordinatı için ise (27,0) piksellerinin ayrıldığı bilinmektedir. Böylece bu adresleme algoritması ile 1.karakter için resim üzerinde ayrılan adresin (161,235) noktasının olduğu anlaşılmaktadır.

Veri gizleme işleminde dikkat edilmesi gereken iki ilkeden ikincisi bu bölümde dikkate alınarak tespit edilen adresten sonra 4 hücre daha işleme alınarak gizleme işlemi yapılmalıdır.

Veri gizleme işlemi TC kimlik numarasının sıralamasının yeniden yapıldıktan sonra 1.karakteri olan 1 sayısının veri şifreleme ve resim içerisine gizlenmesi işlemi daha da ayrıntılı olarak analiz edilirse;

1 Karakteri önce kullanılan 1.şifreleme algoritması ile 834 şeklinde bir değer almış ve bu değer de 2.şifreleme basamağı olan RSA şifreleme algoritması ile 38717774 sayısal değerine ulaşmıştır. Artık sistem için 1 sayısının karşılığı 38717774 sayısıdır denilebilir. Bu sayının gizlenmesi ise,

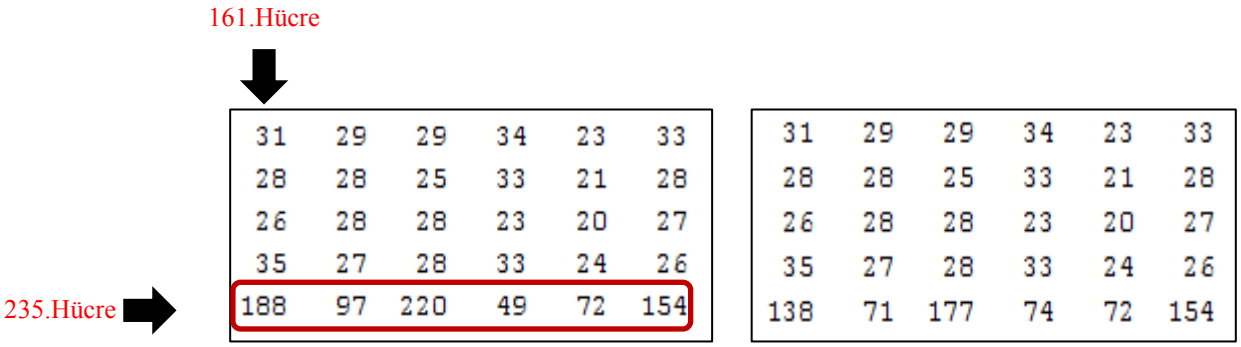
**38-71-77-74**

(161,235) Noktası -> Orijinal piksel değeri **188** iken son iki piksel değiştirilerek 1**38**

(162,235) Noktası -> Orijinal piksel değeri **97** iken **71**

(163,235) Noktası -> Orijinal piksel değeri **220** iken son iki hane değiştirildiğinde 277 olacağından ve bu değer **255** üst sınırını aşacağı için değer, kullanılan algoritma ile önce 100 eksiltilir ve 1**77** sayısı elde edilmiş olur.

(164,235) Noktası -> Orijinal piksel değeri **49** iken **74** olarak belirlenmiştir.




**Şekil 5.18.** Piksel Değişimleri

Oluşturulan 8 haneli şifreli veriler kullanılan algoritma ile orijinal pikseller içerisindeki renk değerlerinin son iki hanesindeki değerleriyle değiştirilerek toplamda 4 piksel kullanılarak 1 karakteri sayısal resim içerisine gizlenmesi işlemidir. Veri gizlemek için ise ilk karakterin piksel koordinatlarını belirten (161,235) noktasından 4 adet x ekseninde ilerleyen ve (161,235) – (164,235) aralıklarına veri gizleme işlemini gerçekleştiren bir algoritma kullanılmıştır.

## 5.5 Verilerin Deşifre Edilmesi

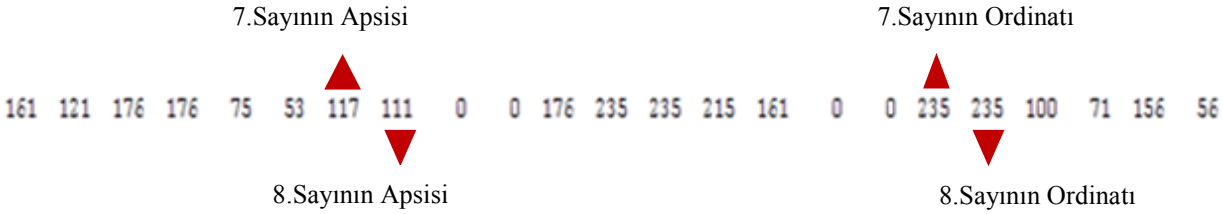
Sayısal resim içerisine gizlenmiş şifreli verilerin önce görüntü içerisinden okunması ardından deşifre edilmesi ve TC kimlik numarasına çevrilmesi için veri gizleme işleminde ihtiyaç duyulan bilgilerden daha fazlası kullanılır. Öncelikle yine adresleme için kullanılan 22 piksel (11 apsis, 11 ordinat) ve ilk 10 pikselde tutulan anahtar değerleri bu kısımda belki de ihtiyaç duyulan en önemli veriler olacaktır. Çünkü şifrelenmiş değerler hangi pikseller içerisinde gizlendiği adresleme verileri üzerinde tutulmaktadır. Adres bilgilerini kullanarak şifreli veriler elde edildikten sonra ise gerekli olan bilgi RSA şifreleme algoritması ile şifrelenmiş verilerin deşifre edilmesi için gizli anahtarların (N ve d) bilinmesidir. Veri gizleme aşamasında toplamda 5 adımda gerçekleştirilen işlemler bu bölümde 2 temel adımla deşifre edilmektedir.

### 1.Adım: Verilerin Okunması

Deşifre edilme işleminin ilk adımı olarak verilerin görüntü içerisinde gizlenmiş oldukları adreslerin bulunarak bu şifreli verilerin okunması işlemi yapılmalıdır. Bunun içinde yapılması gereken ilk iş veri gizleme basamağında olduğu gibi adresleme satırına ulaşılmasıdır.

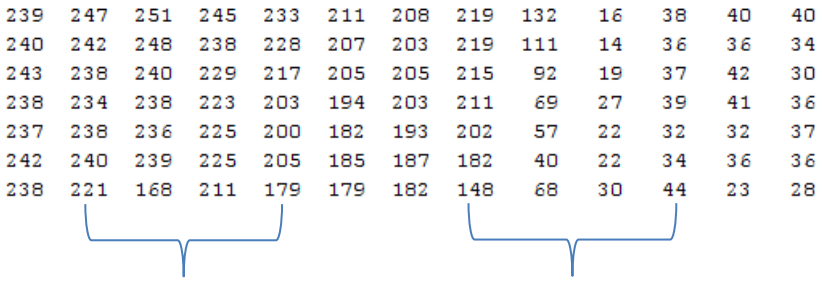
Bu aşamada da örnek olarak 11 haneli TC kimlik numarasının 7.karakteri olan 5 sayısının ve 8.karakteri olan 4 sayısının deşifre edilmesi işlemlerini ayrıntılı olarak ele alınmıştır.

Aşağıda ki şekilde gösterilen ve ilk 22 pikselde tutulan adresleme satırına ulaşıldıktan sonra 7. ve 8. sayıların apsis ve ordinatları tespit edilmelidir. İlk 15 pikselin anahtar değerleri için tutulduğu ve adreslemenin 16.pikselden itibaren başlandığı bilindiğine göre 7.sayının apsisi 32, 8.sayının apsisi ise 33.piksel olduğu bulunur. Ordinat değerleri ise sırasıyla 43 ve 44.piksellerdir. Böylece 7.sayının tam adresi 32 ve 43.adresleme pikselleri üzerinden (117,235) noktası olduğu, 8.sayının ise 33 ve 43.pikseller kullanılarak (111,235) noktasının olduğu bulunmuş olur.




**Şekil 5.19.** Gizli verilerin adresleri

Sayıların gizlendiği adreslerin yerleri tespit edildikten sonra veri gizleme basamağında uygulanan yöntemin benzeri bir işlem uygulanır. Bu kez veri değişimi yerine doğrudan verilerin alınması işlemi gerçekleştirilir. 7.Karakterin (111,235) noktasında tutulduğu bilindiğine göre bu noktaya ulaşıldıktan sonra yapılması gereken önceki bölümlerde anlatıldığı gibi 4 hücre daha noktayı yürüterek x ekseninde bulunan 111, 112, 113, 114 piksellerine ulaşmaktır. Aynı şekilde 8.karakter de 117, 118, 119, 120 piksellerinde bulunmaktadır.




**Şekil 5.20.** Verilerin deşifre edilmesi

7. ve 8. sayıların gizlendiği hücreler yukarıda ki şekilde gösterildiği gibi tespit edilmiştir. Yakalanan bu adreslere aşağıda ki tabloda da ayrıntılı olarak yer verilmiştir. Yukarıda ki ekran çıktısı ve tablo birlikte incelendiğinde ve piksellerin adresleri irdelendiğinde hem veri gizleme işleminin ilke ve prensipleri hem adresleme mantığı hem de veri okuma işlemi daha iyi anlaşılmaktadır.

| 7.KARAKTER | | 8.KARAKTER | |
|---|---|---|---|
| (117,235) | 148 | (111,235) | 221 |
| (118,235) | 68 | (112,235) | 168 |
| (119,235) | 30 | (113,235) | 211 |
| (120,235) | 44 | (114,235) | 179 |

**Şekil 5.21.** Karakter Değerlerinin Okunması

Veri gizleme işleminde yapıldığı gibi bu aşamada da şekil 35'te görüldüğü gibi ulaşılan piksellerin son iki basamağı alınarak bir bütün oluşturulmuş ve şifreli veriler elde edilmiştir. 11 karakterden oluşan sayı dizisinde ki;

7.Karakter [117-120,235] adresinden alınan veriler ile **48683044,**

8.Karakter [111-114,235] adresinden alınan veriler ile **21681179** değerinin olduğu anlaşılmıştır.

**2.Adım: Verilerin Deşifre Edilmesi**

Verilerin okunması adımında şifreli verilerin gizlenmesi aşamasında yapılan işlemlerin tersten yapıldığı şeklinde bir tespit yapılırsa bu yanlış bir değerlendirme olmaz. Hatta verilerin deşifre edilmesi yani şifreli verilerin şifrelerinin çözülmesi adımı da bu kez şifreleme işleminin tersten yapılması işlemidir denilebilir. Önceki bölümler hatırlanırsa verilerin şifrelenmesi basamağında 2 aşamalı bir şifreleme yöntemi uygulanmış, önce 1. şifreleme basamağı adı verilen bölümde sayıların karakter seti değerleri ve bu değerin ikili koda dönüşümü ile kontrol biti adı verilen özel bir değer elde edilmiş bu iki değer kullanılarak 3 basamaklı yeni bir sayı elde edilmişti. Ardından bu sayı 2. şifreleme basamağı denilen RSA şifreleme algoritması ile bir kez daha şifrelenmiş ve yepyeni bir değere ulaşılmış böylece verilerin şifrelenmesi basamağı tamamlanmıştır.

Bir önceki adım olan verilerin okunması adımıyla elde edilen şifreli verilerin deşifre edilmesi işlemi de şifreleme basamağında ki iki aşamadan gerçekleşmekte ve bahsedildiği üzere bu kez aşamalar tersten uygulanmalıdır. Yani ilk olarak şifreli veriler önce 2.şifreleme basamağında ki RSA şifreleme algoritması ile deşifre edilmelidir. RSA algoritması ile deşifre işleminin yapılması için şifreleme bölümünde detaylı olarak ele alınmış olan N ve D

anahtarlarına ihtiyaç duyulmaktadır. Bu nedenle verilerin gizlenmesi bölümünde sayılar ile birlikte anahtarlar da resim içerisine gizlenmiştir. Aksi durumda hatalı deşifrelere sebebiyet verilebilir.

İlk 10 pikselde yer alan anahtarların da okunması işlemi yapıldığında N anahtarının 69225323, D anahtarının ise 21324619 değerine sahip olduğu tespit edilir. 1.adımda örnek olarak ele alınan 7. ve 8. karakterlerin şifreli hallerine ulaşıldığı için aynı sayıları bu adımda da kullanarak orijinal sayılara dönüşümünü görmek sistemin mantığını çözmek adında önemli olabilir.

RSA şifreleme algoritması ilkelerine göre şifreli verilerin deşifre edilmesi için $M= C^{d} \pmod N$ formülü kullanılmaktadır. Buna göre;

1.Adımda bulunan 7.karakterin değeri 48683044, 8. karakterin değeri ise 21681179 ifadesidir. Veri okuma ile tespit edilmiş anahtar ve sayı değerleri formülde yerine yazılırsa

7.Karakter; $48683044^{21324619} \pmod{69225323} =$ **443** değerine,

8.Karakter; $21681179^{21324619} \pmod{69225323} =$ **434** değerine ulaşılır.

Şifreleme basamağı tersten uygulandığına göre böylece 2.şifreleme basamağında yani RSA şifreleme algoritması uygulanmış şifre çözülmüştür. Bu aşamadan sonra yapılması gereken işlem ise 1.şifreleme basamağında ki şifrelemenin çözülmesidir. Bu bölümde şifreleme işleminden sonra oluşan 3 basamaklı sayının onlar basamağında yer alan sayıların kontrol biti olarak kullanıldığı bilinmektedir. Kontrol bitinin sınanması işlemi ise işte tam da bu adımdadır. Onlar basamağında bulunan kontrol biti değeri sayıdan çıkarıldığında 7.Karakter için 43 değeri elde edilir. 43 sayısının ise ikili koda dönüşümü yapılırsa 00101011 şeklinde binary ifadeye ulaşılır ve bu ifade de yer alan 1 değerleri sayıldığında **4** rakamı elde edilir ve böylece doğru sayının resim içerisinden okunup deşifre edildiği test edilmiş olur. Yani kontrol biti sayesinde **43** sayısının bizim için doğru sayı olduğu kanaatine ulaşılmış olunur. Son olarak 43 sayısının aşağıda bulunan Karakter Setindeki karşılığına bakıldığında **5** sayısına ulaşıldığı görülür.

Adresleme satırında yer alan tüm sayıların örnekte olduğu gibi öncelikle resim içerisinde gizlendikleri noktaların koordinatları belirlenmeli ardından şifreli veriler okunmalıdır. Bu işlemin ardından 2 şifreleme basamağı bu kez tersten işleme alınarak şifreler çözülmeli ardından kontrol biti aracılığıyla test edilerek orijinal sayılar deşifre edilmelidir.

### 5.6. Uygulamadan Elde Edilen Sonuç

TC kimlik numarasında yer alan sayıların, birisi oldukça güçlü bir şifreleme algoritması standardı olan RSA şifreleme algoritması diğeri de bu uygulama için özel olarak geliştirilmiş başka bir şifreleme yöntemi ile iki kez art arda şifrelenmesi yazılımın güvenilirliği açısından oldukça önemli bir yer tutmaktadır.

Şifreli verilerin resim içerisine gizlenirken ise dikkat edilen çok önemli iki nokta vardır. Önerilen uygulamanın iki adet üstünlüğü dikkat çekmektedir. Birincisi veri gizleme işlemi yapılırken resmin yapısının ve bütünlüğünün hiçbir şekilde bozulmamasıdır. Yazılımın ikinci avantajı olarak ise veri gizleme işlemi sırasında şifrelenmiş ve her biri 8 karakterden oluşmuş sayıların hiçbirinin sabit bir noktaya gizlenmemesi lokasyon işleminin de tamamen rasgele olması yani hangi verinin hangi noktaya gizlendiğinin bilinmemesidir. Geliştirilen bu yöntem güvenlik açısından uygulamayı bir üst noktaya çıkarmış ve güvenlik açısından benzer uygulamalara kıyasla oldukça önemli bir avantaj sağlamıştır.

# 6. SONUÇ ve ÖNERİLER

Teknolojinin gelişmesiyle birlikte dijital ortamların kullanımı her geçen gün artmış ve günümüzde çok yüksek boyutlara ulaşmıştır. Artık her kullanıcının kendi ihtiyacına göre bir şekilde kullandığı bu platformlar üzerinde kullanım sayısına bağlı olarak gün geçtikçe güvenlik açıkları meydana gelmiştir.

Veri ve bilgi güvenliğinin korunmasına paralel olarak yeni yöntem, metot ve algoritmalar geliştirilerek uygulamalar üzerinde sınanmasının hedeflendiği bu tez çalışmasında veri şifreleme ve veri gizleme tekniklerinden yararlanılmıştır. Veri gizleme ve şifreleme tekniklerinin, Kriptosistemlerin anlatıldığı çalışmada ana hatlarıyla bu konular irdelenmiş temel ve alt bileşenleri açıklanmıştır. Bilgi güvenliği, söz konusu olduğunda akla gelen önemli bilim dalları kriptoloji ve steganografi şimdiye kadar yapılan çalışmalarda genellikle birbirlerine alternatif olarak görülmüşlerdir. Çalışmalarda uygulama yapılan alana göre genelde bir tanesi seçilmiş ve işlenmiştir. Oysa bu tez çalışmasının ana amacı veri gizleme ve şifreleme tekniklerinin birlikte kullanımıdır. Tez çalışmasında yer alan iki uygulamada da bu tekniklerden değişik yöntem ve metotlar kullanılarak birlikte yararlanılmıştır. Böylece güvenliğin daha da artırılması amaçlanmış ve yapılan çalışmalar sonucunda bu hedefe ulaşıldığı görülmüştür.

Geliştirilen ilk uygulama olan OPT adı verilen Radyografik resimler içerisine veri gizlenmesi uygulanmasında önce veriler yeni geliştirilmiş özgün bir şifreleme algoritmasıyla şifrelenmiş ardından piksel değişim metoduyla OPT resimler içerisine gizlenmiştir. Ardından okuma ve deşifre işleminin yapılmasıyla birlikte verilere tekrar ulaşılmış böylece hasta bilgileri, teşhis tanı ve tedavi bilgileri OPT'ler içerisine gömülerek kâğıt israfı, dosya kalabalıklığı, evrak kaybolması gibi olumsuzluklar ortadan kaldırılmıştır.

Tez çalışması sırasında yapılan ikinci uygulamada ise kimlik sahteciliğini tespit eden bir uygulama geliştirilmiştir. Günümüzde teknolojinin gelişmesi insan hayatını kolaylaştırdığı gibi kötü amaçlar için kullanıldığında ise topluma birçok zarar verebilmektedir. Yine teknolojinin ilerlemesine paralel olarak sahtecilik ve dolandırıcılık suçlarının da emniyet kayıtlarına göre gün geçtikçe artmakta bu nedenle birçok masum insan mağdur olmaktadır. Geliştirilen uygulamada kullanılan yöntem ise TC kimlik numarasının şifrelenerek kimlik üzerinde bulunan resim içerisine gizlenmesi ardından deşifre edilerek elde edilen verinin kişinin gerçek TC kimlik numarasıyla eşleşip eşleşmediğinin kontrolünü yaparak doğrulama

işlemin gerçekleştirilmesidir. Yapılan uygulama da kullanılan şifreleme ve gizleme teknikleri özgün teknikler olup kriptografi ve steganografi tekniklerinin birlikte kullanılması sistemin güvenirliği açısından oldukça iyi sonuçlar elde edilmesini sağlamıştır. Uygulamanın en önemli üstünlükleri ise şifreleme basamağında iki adet farklı şifreleme algoritmasının kullanılmasıdır. Birisinin özgün olup bu uygulama için özel olarak geliştirilmiş olması, diğer şifreleme algoritmasının ise güçlü bir şifreleme standardı olan RSA şifreleme algoritması kullanılması şifreleme işleminin oldukça güçlü olmasını sağlamış ve güvenlik bakımından oldukça avantaj sağlamıştır. İkinci adım olan şifreli verilerin gizlenmesi basamağında ise bilinen veri gizleme yöntemlerine karşılık olarak yeni bir yöntem önerilmiş ve veri gizleme işleminde yer belirlenmesi rasgele olarak yapılmıştır. Bu aşamanın sisteme iki şifreleme algoritmasından sonra üçüncü bir şifreleme sistemi olarak yansıdığı uygulama sonucunda görülmüştür. Böylece bu yöntem de sistemin güvenirliği bir üst noktaya taşımıştır.

Tez çalışmasında geliştirilen yeni ve özgün veri şifreleme ve gizleme teknikleri tez aşaması sırasında geliştirilen uygulamalar içerisinde kullanılsa da sadece bu uygulamalarla kısıtlı kalınamayacağı çeşitli projelerde de bu tekniklerden yararlanılabileceği göz ardı edilmemelidir. Veri gizleme ve şifreleme tekniklerine ihtiyacın duyulacağı her uygulama da bu tez içerisinde önerilen yöntem ve algoritmalar kullanılabilir ve tekniklerde günümüzde oldukça önemli hale gelen veri şifreleme ve gizleme konularına yeni bir bakış açısı getirecektir.

**KAYNAKLAR**


[1] **Oppliger, R**.: *Contemporary Cryptology* , Artec House , Inc., ISBN: 9781580536431, Norwood, MA, USA**,** (2005), 1-3.

[2] **Başkök, M**.: ” AES Şifreleme Algoritmasının Modellenmesi ”, *Yüksek Lisans Tezi*, Gazi Üniversitesi Fen Bilimleri Enstitüsü, Ankara, Türkiye, (2007).

[3] **Tuncal, T**.: ” Bilgisayar Güvenliği Üzerine Bir Araştırma ve Şifreleme- Deşifreleme Üzerine Uygulama ”, *Yüksek Lisans Tezi*, Maltepe Üniversitesi Fen Bilimleri Enstitüsü, İstanbul, Türkiye, (2008).

[4] **Ersin, G**.: “TÜBİTAK UEKAE Açık Anahtar Altyapısı Eğitim Kitabı”, Mayıs, (2006).

[5] **Buluş, H. N**.: ” Temel Şifreleme Algoritmaları ve Kriptanalizlerin İncelenmesi ”, *Yüksek Lisans Tezi*, Trakya Üniversitesi Fen Bilimleri Enstitüsü, Edirne, Türkiye, (2006).

[6] **Weis, S**.: ” Theory and Practice of Cryptography ”, *Google Tech Talks,* (2007). http://www.youtube.com/watch?v=IzVCrSrZIX8 (23.12.2013).

[7] **Koblitz**, N. : *A Course in Number Theory and Cryptography,* Springer Verley, Newyork, USA, (2006).

[8] **Bahçetepe**, H.: ” Modüler Çarpma Algoritmalarının İncelenmesi ve Kriptolojide Uygulamaları ”, *Yüksek Lisans Tezi*, İstanbul Üniversitesi Fen Bilimleri Enstitüsü, İstanbul, Türkiye, (2006).

[9] **Sarıtaş, H**.: ” Dijital İmza Uygulamasının Eliptik Eğri Şifreleme Yöntemi Kullanılarak Gerçekleştirilmesi ”, *Yüksek Lisans Tezi*, Marmara Üniversitesi Fen Bilimleri Enstitüsü, İstanbul, Türkiye, (2010).

[10] **Turgut, S**. : “ Kuantum Bilgisayarlar”, *Bilim ve Teknik, Mayıs*, (2003), 50-52.

[11] **Denton, B**. : “ Evaluation of Cryptographic Construction Properties and Security Requirements of Modern Secure Hashing Algorithms”, *Master Thesis,* The Department of Electrical and Computer Engineering of Alabama University, Huntsville, Alabama, USA (2011).

[12] **Moldovyan, A. ; Moldovyan, N**. : *Innovative Cryptography,* 2 nd Edition; Course Technology, ISBN: 9781584506546, Boston, MA, USA, (2006) , 11-13.

[13] **Koltuksuz, A**. : ” Elektronik Ticarette Güvenlik, Özgürlük Denetimi, Doğruluk-Bütünlük ve Sayısal İmza ”, *4. Türkiye İnternet Konferansı*, İstanbul , Türkiye, (1998).

[14] İnternet: Pro-G ve Oracle, “Bilişim Güvenliği Sürüm 1.1”, http://www.prog.com.tr/whitepapers/bilisim-guvenligi-v1.pdf, (2003).

[15] **Soğukpınar**, _. , “Veri ve Ağ Güvenliği Ders Notları”, ***GYTE Bilgisayar Mühendisli****gi*, 44-101 (2005).

[16] **Denton, B**. : “ Evaluation of Cryptographic Construction Properties and Security Requirements of Modern Secure Hashing Algorithms”, *Master Thesis*, The Department of Electrical and Computer Engineering of Alabama University, Huntsville, Alabama, USA (2011).

[17] **Rivest, R..; Shamir, A.; Adleman, L**.: “A Method for Obtaining Digital Signatures and Public-Key Cryptosystems,”, *Communications of the ACM*, v. 21, n. 2, (Feb, 1978), pp. 120-126.

[18] **Ulutürk, A**., ’’ Gelişmiş Şifreleme Standardı’’, Yüksek lisans Tezi, Gazi Üniversitesi Fen Bilimleri Enstitüsü, Ankara, Türkiye (2010).

[19] **Yerlikaya, T. , Bulus, E. , Arda, D**. , “Asimetrik Kriptosistemler ve Uygulamaları”, *II. Mühendislik Bilimleri Genç Arastırmacılar Kongresi*, İstanbul, 24-31 (2005).

[20] **Çağlar, E**., ‘’Açık Anahtarlı Kriptografi ve Ağ Güvenlik Uygulamaları’’, Yüksek Lisans Tezi, Çanakkale Onsekiz Mart Üniveristesi, Fen Bilimleri Enstitüsü,Çanakkale, (2004).

[21] **Diffie W.; Hellman ,M. E**.: “New Directions in Cryptography ” , *IEEE Transactions on Information Theory, IT-22*, No.6, (1976) , 644-654.

[22] **Tsunoo Y.; Tsujihara E.; Minematsu K.; Miyauchi H**.:”Cryptanalysis of Block Ciphers Implemented on Computers with Cache”, ISITA, (2002).

[23] İnternet : Cyber-Warrior “Kriptoloji-Kriptografi Seminerleri Sonuç Bildirgesi”, http://www.kriptoloji.net/sonucbildirgesi.pdf (24.12.2013).

[24] **Schneider, B**.: ”Applied Cryptography Second Edition”, *John Wiley& Sons, Inc.*, New York, (1996).

[25] **Sakallı, T.; Buluş, E.; Şahin, A**.: "Modern Blok Şifreleme Algoritmalarının Gücünün İncelenmesi ", II. Mühendislik Bilimleri Genç Araştırmacılar Kongresi-MBGAK 2005, İstanbul,Türkiye, (2005).

[26] **Özduran, V**.: " Birleşik Şifreleme ve Turbo Kodlama Sistemleri ", *Yüksek Lisans Tezi*, İstanbul Üniversitesi Fen Bilimleri Enstitüsü, İstanbul, Türkiye, (2008).

[27] **Smith, D**.: " Multivariate Cryptography ", *Ph.D. Thesis*, The Department of Mathematics of Indiana University, USA, (2010).

[28] **Petitcolas F.A.P., Anderson R.J., Kuhn M.G**.1999. "Information Hiding–A Survey", Proceedings of the IEEE, Special Issue on Protection of Multimedia Content, 87(7):1062-1078.

[29] **Murray A.H., Burchfiled R.W** (eds.).1933. "The Oxford English Dictionary: Being a Corrected Re-issue", Oxford, England: Clarendon Pres.

[30] **Memon N., Wong, P**.1998."Protecting digital media content", Communications of the ACM, vol 41, no. 7 , pp. 34–43.

[31] **Wang H., Wang S**.1984."Cyber Warfare: Steganography vs. Steganalysis", Communications of the ACM, vol. 47, no. 10, October 2004.

[32] **Simmons G**.1984, "The Prisoners' Problem and the Subliminal Channel", CRYPTO83 Advances in Cryptology, pp. 51-67.

[33] **Kharrazi M., Sencar H.T**.1998. Memon N, "Image Steganography: Concepts and Practice", WSPC/Lecture Notes Series, April 22, 2004. Anderson R.J, Petitcolas F.A.P., "On the Limits of Steganography", IEEE Journal of Selected Areas in Communications, 16(4):474-481, Special Issue on Copyright & Privacy Protection. ISSN 0733–8716.

[34] **Anderson R.J**.1996. ed., Information Hiding: First International Workshop, vol 1174 of Lecture Notes in Computer Science, Isaac Newton Institute, Cambridge, England, Springer-Verlag, Berlin, Germany, ISBN 3-540-61996.

[35] **Newman B**.1940. "Secrets of German Espionage", London: Robert Hale Ltd.

[36] **Tacticus A**.1990. "How to Survive Under Siege / Aineias the Tactician", Oxford, England: Clarendon Pres, pp. 84-90 and 183-193, Clarendon Ancient History Series.

[37] **Katzenbeisser S., Petitcolas F.A.P**., "Information Hiding Techniques for Steganography and Digital Watermarking", Artech House, INC. 685 Canton Street Norwood, MA 02062, 2000.

[38] **Kahn D**., "The Codebreakers", Macmillan, New York, 1967.

[39] **Zim H.S**.1948. "Codes and Secret Writing", William Morrow, New York.

[40] **Johnson N.F, Jajodia** S.1998. "Exploring steganography: Seeing the Unseen", Computer, 31, no 2:26-34.

[41] **Johnson N.F., Rude T**.2001. "Introduction to Steganography Hidden Information", Regional Computer Forensic Group (RCFG) and Mid-Atlantic Chapter of High-Technology Crime Investigation Association (HTCIA) George Mason University Computer Forensics Symposium (GMU 2001), Fairfax, VA, August 13-17.

[42] **Şahin, A**.'' Görüntü Steganografi de Kullanılan Yeni Metodlar ve Bu Metodların Güvenilirlikleri''. Doktora Tezi., Trakya Üniversitesi, Bilgisayar Mühendisliği Anabilim Dalı, Edirne, 2007.

[43] **Popa R**., "An Analysis of Steganographic Techiques", Ph.D Thesis, 1998.

[44] **Fridrich J., Du R., Meng L**., "Steganalysis of LSB Encoding in Color Images", Proceedings IEEE International Conference on Multimedia and Expo, New York City, NY, July 30–August 2, 2000.

[45] **Bender W., Gruhl D., Morimoto N., Lu A**., "Techniques for data hiding", IBM Systems Journal, vol. 35, NOS 3&4, 1996.

[46] **Johnson N. F., Jajodia S**., "Stegananalysis of Images Created Using Current Steganography Software", Second Information Hiding Workshop held in Portland, Oregon, USA, April 15-17, 1998. Proceedings LNCS 1525, 273-289, Springer-Verlag, 1998.

[47] **Johnson N.F., Duric Z., Jajodia S**., "Information Hiding: Steganography and Watermarking - Attacks and Countermeasures", Kluwer Academic Publishers, ISBN: 0-79237-204-2, 2000.

[48] **Sellars D**., "An Introduction to Steganography", Online book, 1999.

[49] **Kim Y.S., Kim Y.M., Choi J.Y., Baik D.K**., "Information Hiding System StegoWaveK for Improving Capacity", International Symposium, ISPA 2003 Aizu-Wakamatsu, Japan, July 2-4, Proceedings, Springer Berlin/ Heidelberg, ISSN 0302-9743, vol. 2745/2003, 2003.

[50] **Gruhl D., Lu A., Bender W**., "Echo Hiding", in Proceeding of First International Workshop, Springer, Cambridge, UK, May-June, 1996.

[51] Kelly, T., *The Myth Of The Skytale*, , Cryptologia V XXII No 3, Pp 44–260,(1998), History of Steganografik and Cryptography, http://www.petitcolas.net/fabien/steganografik/history.html.

[52] **Sarıoglu, S., Tunçkanat, M**., 2002, Güvenli Internet Haberlesmesi için Bir Yazılım: TurkSteg, Olympos Security, http://www.teknoturk.org/ docking/yazilar/tt000106–yazi.html.

[53] **Potdar V.M., Chang E**., "Grey Level Modification Steganography for Secret Communication", Industrial Informatics, INDIN '04. 2004 2nd IEEE International Conference, pp. 223-228, ISBN: 0-7803-8513-6, 24-26 June 2004.

[54] **Morkel T., Eloff J.H.P., Olivier M.S**., "An Overview of Image Steganography", in Proceedings of the Fifth Annual Information Security South Africa Conference (ISSA2005), Sandton, South Africa, June/July 2005 (Published electronically).

[55] **Cox I.J., Kilian J, Leighton T., Shamoon T**., "A Secure, Robust Watermark for Multimedia", Proc. First Int'l Workshop Information Hiding, Lecture Notes in Computer Science No. 1, 174, Springer-Verlag, Berlin, 1996, pp. 185-206.

[56] **Alwan R.H., Kadhim F.J., Al-Taani A.T**., "Data Embedding Based on Better Use of Bits in Image Pixels", International Journal of Signal Processing Volum 2, Number 2, ISSN 1304-4494, 2005.

[57] **Kessler G.C**., "Steganography: Hiding Data within Data", http://www.garykessler.net/library/steganography.html, September 2001.

# ÖZGEÇMİŞ

1986 yılında Elazığ'da doğdu. İlk ve orta eğitimini Mustafa Kemal İlköğretim Okulunda, Lise eğitimini ise Mehmet Akif ERSOY (YDA) Lisesinde tamamladı. Elazığ Fırat Üniversitesi, Teknik Eğitim Fakültesi, Elektronik - Bilgisayar Eğitimi Bölümü, Bilgisayar Öğretmenliği Programını 2011 yılında derece ile mezun olarak lisans eğitimini tamamladı. 2012 yılında ise Fırat Üniversitesi, Fen Bilimleri Enstitüsü, Elektronik – Bilgisayar Eğitimi Bölümü, Telekomünikasyon Anabilim Dalında yüksek lisans sınavını kazanarak lisansüstü eğitimine başladı. 2012 yılında Kafkas Üniversitesi Rektörlük biriminde öğretim elemanı olarak göreve başladı ve Kafkas Üniversitesi Uzaktan Eğitim Uygulama ve Araştırma Merkezinde görevlendirildi. Halen Kafkas Üniversitesinde ki görevini sürdürmektedir. Yabancı dili İngilizcedir.