İSTANBUL TEKNİK ÜNİVERSİTESİ ★ LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

ELGAMAL ALGORİTMASI VE ARNOLD DÖNÜŞÜMÜNE DAYALI BİYOMETRİK GÖRÜNTÜ KRİPTOLOJİSİ

YÜKSEK LİSANS TEZİ

RABİA ÜNLÜ

Matematik Mühendisliği Anabilim Dalı

Matematik Mühendisliği Programı

EYLÜL 2024

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

ELGAMAL ALGORİTMASI VE ARNOLD DÖNÜŞÜMÜNE DAYALI BİYOMETRİK GÖRÜNTÜ KRİPTOLOJİSİ

YÜKSEK LİSANS TEZİ

RABİA ÜNLÜ
(509201241)

Matematik Mühendisliği Anabilim Dalı

Matematik Mühendisliği Programı

Tez Danışmanı: Prof. Dr. Gülçin ÇİVİ BİLİR

EYLÜL 2024

ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL

BIOMETRIC IMAGE ENCRIPTION BASED ON ARNOLD TRANSFORM AND ELGAMAL ALGORITHM

M.Sc. THESIS

RABİA ÜNLÜ
(509201241)

Department of Mathematics Engineering

Mathematics Engineering Program

Thesis Advisor: Prof. Dr. Gülçin ÇİVİ BİLİR

SEPTEMBER 2024

1

İTÜ, Fen Bilimleri Enstitüsü'nün 509201241 numaralı Yüksek Lisans Öğrencisi RABİA ÜNLÜ, ilgili yönetmeliklerin belirlediği gerekli tüm şartları yerine getirdikten sonra hazırladığı "ELGAMAL ALGORİTMASI VE ARNOLD DÖNÜŞÜMÜNE DAYALI BİYOMETRİK GÖRÜNTÜ KRİPTOLOJİSİ" başlıklı tezini aşağıda imzaları olan jüri önünde başarı ile sunmuştur.

Tez Danışmanı : **Prof. Dr. Gülçin ÇİVİ BİLİR** .....
İstanbul Teknik Üniversitesi

Jüri Üyeleri : **Doç. Dr. Güler GÜRPINAR ARSAN** .....
İstanbul Teknik Üniversitesi

Prof. Dr. Gürsel YEŞİLOT .....
Yıldız Teknik Üniversitesi

Teslim Tarihi : 11 Temmuz 2024
Savunma Tarihi : 03 Eylül 2024

Anneme ve eşime,

ÖNSÖZ

Bu çalışmayı hazırlamamda bana rehberlik eden Prof. Dr. Gülçin ÇİVİ BİLİR'e, yaşamım boyunca her türlü desteği veren sevgili aileme ve eşime teşekkür ederim.

Temmuz 2024 Rabia ÜNLÜ

İÇİNDEKİLER

Sayfa

KISALTMALAR

RSA : Rivest-Shamir-Adleman
RGB : Red - Green – Blue
XOR : Exclusive Or
DES : Data Encryption Standard
NIST : National Institute of Standards and Technology
VPN : Virtual Private Network
IKE : Internet Key Exchange
TLS : Transport Layer Security

ÇİZELGE LİSTESİ

Sayfa

ŞEKİL LİSTESİ

Sayfa

ELGAMAL ALGORİTMASI VE ARNOLD DÖNÜŞÜMÜNE DAYALI BİYOMETRİK GÖRÜNTÜ KRİPTOLOJİSİ

ÖZET

Görüntü şifreleme, modern iletişim ve güvenlik alanlarında önemli bir rol oynayan kritik bir teknolojidir. Hassas verilerin, özellikle insan yüzlerinin güvenli bir şekilde saklanması, aktarılması ve doğrulanması, kişisel gizliliğin korunması ve yetkisiz erişime karşı savunmanın sağlanması açısından hayati öneme sahiptir. Bu çalışmada, kişisel verilerin güvenli bir şekilde paylaşılması, veri bütünlüğünün ve gizliliğinin sağlanması için etkili bir homomorfik şifreleme yöntemi olan ElGamal dönüşümü ile Arnold dönüşümünün entegrasyonundan oluşan bir görüntü şifreleme algoritması önerilmektedir.

Arnold Dönüşümü, eşit uzunluk ve genişlikteki piksel noktalarından oluşan bir görüntünün piksellerinin konumları üzerinde birden çok matris işlemi gerçekleştiren klasik kriptografik sistem temelli bir görüntü şifreleme algoritmasıdır. Bu dönüşüm, dijital görüntünün karıştırılıp tanınmaz bir hale getirilmesini sağlamaktadır. NxN boyutundaki görüntünün piksellerinin x ve y koordinatlarını değiştirerek bir karışıklık oluşturur ve orijinal görüntünün okunmasını zorlaştırır. Yeni x ve y koordinatları orijinal x ve y koordinatları üzerinde bazı matematiksel işlemler uygulanarak hesaplanır. Bu işlemler piksellerin yatay konumlarını değiştirerek görüntünün şifrelenmesini sağlar. Arnold dönüşümü, geniş çapta kullanılan önemli bir görüntü şifreleme tekniği olmasına rağmen, güvenlik zayıflıklarına sahiptir ve her boyuttaki görüntü verilerine uygulanması zordur. Bu zayıflıkları aşmak için mevcut çalışmalar Arnold dönüşümüne entegre edilebilecek çeşitli yaklaşımlar önermektedir.

ElGamal şifreleme sistemi ise 1985 yılında Taher Elgamal tarafından geliştirilen ve Diffie-Hellman anahtar değişimi prensibine dayanan bir genel anahtarlı kısmi homomorfik şifreleme algoritmasıdır. Asimetrik anahtarlı kriptografi kullanarak güvenli bir şifreleme yöntemi sunar. Özel anahtar şifreleme işlemi için gizli tutulurken genel anahtar çözme işlemi için kullanılır ve genel olarak erişilebilir. Şifreyi oluşturan kişinin şifrelemeyi inkar edemeyeceği bir doğruluk seviyesi sağlar. Bu nedenle sadece şifreleme değil aynı zamanda görüntü doğrulama için de uygundur.

Bu çalışmada, öncelikle simetrik, asimetrik ve hibrit görüntü şifreleme tekniklerinin kapsamlı bir incelemesi yapılmıştır. Daha sonra, literatüre giren görünü şifreleme çalışmaları dikkate alınarak, biyometrik görüntülerin, $2D$ Arnold dönüşümü kullanılarak karıştırılması ardından ElGamal algoritmasıyla şifrelenmesi ve tersine işlem ile orijinal görüntünün elde edilmesi problemi ele alınmıştır. Önerilen yaklaşımı özel kılan nokta, Arnold dönüşümü sonrası şifrelenmiş biometrik görüntünün deşifresinin ayrık logaritma hesaplamasının zorluğuna dayalı olmasıdır. Örneklerle açıklanan yaklaşım ile biyometrik görüntülerin depolanması, iletilmesi ve doğrulanması sırasında olabilecek saldırılara karşı etkili ve güvenli olan bir hibrit görüntü kriptolojisi hedef alınmıştır.

BIOMETRIC IMAGE ENCRIPTION BASED ON ARNOLD TRANSFORM AND ELGAMAL ALGORITHM

SUMMARY

In today's digital age, image encryption has become a cornerstone technology for safeguarding sensitive visual information. From securing personal photographs on social media platforms to encrypting biometric data such as human faces used in identification systems, encryption has found widespread application. Biometric image encryption is particularly important in securing facial recognition systems, fingerprint scanners, and retinal recognition technology used in security protocols and identity verification systems. By ensuring that the data cannot be accessed or modified by unauthorized users, encryption plays a critical role in modern communication systems, where privacy and data integrity are essential.

The rapid growth of communication networks, alongside the ever-increasing reliance on digital data, has necessitated more advanced and secure methods of encryption. This is especially true for biometric data, where privacy concerns and the potential for misuse are heightened. Unauthorized access to this kind of sensitive information could lead to identity theft, fraud, and breaches of privacy. Consequently, developing more efficient, robust, and secure encryption methods for biometric images has become a top priority for both academic research and industry applications.

The Arnold Transform, named after Russian mathematician Vladimir Arnold, is a mathematical technique commonly used for scrambling the pixel positions in an image. It takes a square image of dimensions N x N and applies a transformation to the pixel coordinates (x, y) such that the image becomes scrambled and unrecognizable. This transformation is reversible, meaning that applying the Arnold Transform multiple times will eventually restore the original image, a process referred to as periodicity. The primary goal of the Arnold Transform is to introduce confusion into the image's structure, making it difficult for unauthorized users to decipher the image without knowing the exact transformation parameters.

Despite its effectiveness in creating confusion, the Arnold Transform suffers from limitations. For instance, the periodicity of the transform can be a vulnerability if the number of transformations is known or can be easily calculated. Furthermore, the standard Arnold Transform is only applicable to square images of specific dimensions, making it less versatile for real-world applications where image sizes vary. Additionally, the Arnold Transform does not provide encryption on its own; it merely scrambles the image, leaving it vulnerable to certain attacks if used as the sole method of protection. For these reasons, researchers have proposed integrating the Arnold Transform with other encryption techniques to enhance security.

The ElGamal encryption algorithm, introduced by Taher Elgamal in 1985, is an asymmetric encryption method based on the principles of the Diffie-Hellman key exchange protocol. Unlike symmetric encryption, where the same key is used for both encryption and decryption, ElGamal uses a pair of keys: a public key for encryption and a private key for decryption. The strength of the ElGamal encryption lies in the computational difficulty of the discrete logarithm problem, which ensures

that even with access to the public key, it is extremely challenging to reverse-engineer the private key or decrypt the message without it.

One of the distinguishing features of ElGamal encryption is its partial homomorphic properties, which allow certain operations to be performed on the encrypted data without needing to decrypt it first. This makes it highly suitable for applications where data integrity must be verified without revealing the original content, such as in secure voting systems, financial transactions, and biometric data encryption.

In biometric image encryption, ElGamal's public-key framework ensures that the encrypted data can only be accessed or modified by authorized users with the correct private key, thereby protecting the confidentiality of the biometric information. This is particularly important in situations where the data must be stored or transmitted over unsecured channels, such as the internet or cloud storage platforms. ElGamal encryption also provides a layer of authentication, ensuring that any modifications to the encrypted data can be detected and traced back to the source.

In this study, we propose a hybrid encryption method that combines the Arnold Transform with the ElGamal algorithm to provide a robust encryption solution for biometric images. The Arnold Transform is used to scramble the image and introduce confusion, while the ElGamal algorithm is used to encrypt the scrambled image, ensuring that the data is both confused and encrypted. The combination of these two methods provides a dual layer of security: even if an attacker were able to reverse the Arnold Transform, they would still need to break the ElGamal encryption to access the original image.

The proposed approach addresses the limitations of the Arnold Transform by integrating it with the more secure ElGamal encryption. By using the Arnold Transform to scramble the image before encryption, we reduce the risk of attacks that target the periodicity of the transformation. Additionally, the ElGamal encryption ensures that the scrambled image remains secure during storage and transmission. The difficulty of the discrete logarithm problem, on which ElGamal encryption is based, further enhances the security of the proposed method.

Although the hybrid approach described in this study provides enhanced security for biometric image encryption, there are still challenges to be addressed. One of the main challenges is the computational complexity of the ElGamal encryption algorithm, particularly when dealing with large images or real-time applications. Future research could focus on optimizing the algorithm to reduce the time and resources required for encryption and decryption. Additionally, further studies could explore the integration of other cryptographic techniques, such as elliptic curve cryptography, to further enhance the security and efficiency of the proposed method.

Another area for future research is the application of the proposed hybrid method to other types of biometric data, such as fingerprints or retinal scans. Each type of biometric data has its own unique characteristics, and the encryption methods used may need to be adapted accordingly. Finally, the robustness of the proposed method could be tested against a wider range of attacks, including brute force attacks, side-channel attacks, and quantum computing threats.

Image encryption is a critical technology that plays a significant role in modern communication and security fields. The secure storage, transmission, and verification of sensitive data, particularly human faces, are vital for protecting personal privacy and defending against unauthorized access. In this study, an image encryption algorithm is proposed, which consists of the integration of ElGamal transform, an effective homomorphic encryption method, and Arnold transform to share personal data securely and ensure data integrity and confidentiality.

Arnold Transform is an image encryption algorithm based on a classical cryptographic system that performs multiple matrix operations on the positions of pixels of an image consisting of pixel points of equal length and width. This transformation ensures that the digital image is scrambled and rendered unrecognizable. It changes the x and y coordinates of the pixels of the NxN image, creates confusion and makes the original image difficult to read. The new x and y coordinates are calculated by applying certain mathematical operations on the original x and y coordinates. These operations change the horizontal positions of the pixels, enabling the encryption of the image. Although Arnold is an important widely used image encryption technology, it has security weaknesses and is difficult to apply to image data of any size. To overcome these weaknesses, current studies propose various approaches that can be integrated into the Arnold transformation.

ElGamal encryption system is a public key partial homomorphic encryption algorithm developed by Taher Elgamal in 1985 and based on the Diffie-Hellman key exchange principle. It offers a secure encryption method using asymmetric key cryptography. It offers a secure encryption method using asymmetric key cryptography. While the private key is kept secret for the encryption process, the public key is used for the decryption process and is publicly accessible. It ensures a level of accuracy that the person creating the encryption cannot deny it. Therefore, it is suitable not only for encryption but also for image verification.

In this study, firstly, a comprehensive review of symmetric, asymmetric, and hybrid image encryption techniques was given. Then, by taking into account the image encryption studies in the literature, the problem of encrypting biometric images applied 2D Arnold transform with the ElGamal algorithm and obtaining the original image by reverse processing is discussed. What makes the proposed approach special is that deciphering the encrypted biometric image after the Arnold transform is based on the difficulty of discrete logarithm calculation. With the approach explained with examples, a hybrid image cryptology that is effective and secure against attacks is proposed.

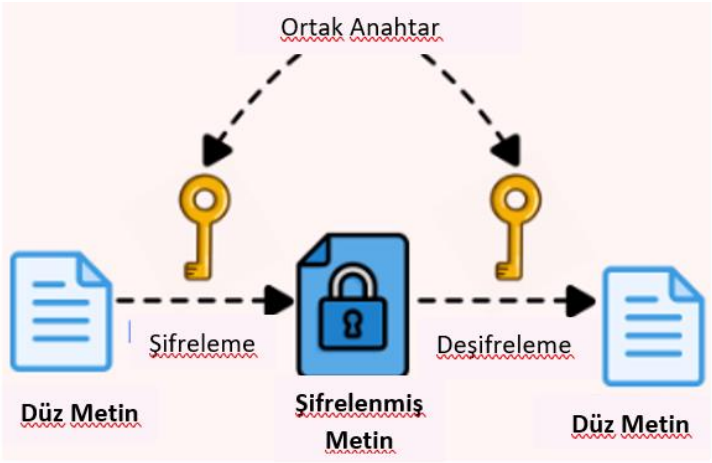
1. GİRİŞ

Günümüzde, bilgi teknololojilerinin yaygınlaşmasıyla birlikte büyük veri oranlarında artış meydana gelmiştir. Özellikle fotoğraf sayılarındaki artış, yeni medya araçlarının etkisiyle devasa boyutlara gelmiştir. Bu durum ise büyük verilerin muhafaza edilme ihtiyacını beraberinde getirmektedir. Teknolojinin ilerlemesiyle birlikte, büyük verinin muhafaza işlemleri artık bulut tabanlı teknoloji hizmetleri aracılığıyla gerçekleştirilebilmektedir. Global oyuncuların sunduğu bu teknoloji hizmetleri aracılığıyla farklı türdeki veriler sanal bir ortamda muhafaza edilebilmektdir. Bilhassa, yeni medya kanallarında oluşturulan içeriklerin sanal ortamda muhafaza edilmesi için bulut tabanlı teknolojiler büyük bir ihtiyacı karşılamaktadır. Modern bulut sistemleri, dağıtık depolama yapıları sayesinde geniş kapasite imkanları sunmakta ve çok fazla tercih edilmektedir. Bu sistemlerin yaygınlaşmasının nedenlerinden biri de yüksek performans ve yedekleme gibi özelliklerdir. Ancak, bulut depolama hizmetleri genellikle içeriklerin gizliliği ve güvenliğine yönelik olarak fotoğraflar gibi veriler üzerinde şifreleme işlemi gerçekleştirememektedir. Bu fotoğraflardan kimlik doğrulamalarda kullanılan yüz, avuç içi, iris gibi son derece önemli biyometrik verilere ulaşılabilmektedir. Bu veriler özellikle son yıllarda online bankacılık işlemleri, e-ticaret siteleri, sosyal medya platformları, e-devlet uygulamaları gibi günümüzde çok sık kullanılan hizmetlerde kimlik doğrulama için kullanılmaktadır. Bu verilerin yetkisiz kişiler tarafından ele geçirilmesi veya değiştirilmesi durumunda, bireylerin gizliliği ve güvenliği büyük ölçüde tehlikeye girebilir. Dolayısıyla, biyometrik verilerin güvenliğinin sağlanması büyük önem taşımaktadır. Artan güvenlik ihtiyaçları insanları şifreleme alanında çalışmalara yönlendirmiştir.

Şifreleme süreçlerinde, şifreleme algoritmaları ile anahtar yönetimi ve çözümleme algoritmaları kullanılmaktadır. Şifreleme yaklaşımları, anahtarın türüne ve kullanım amacına bağlı olarak simetrik ve asimetrik şifreleme olmak üzere ikiye ayrılmaktadır.

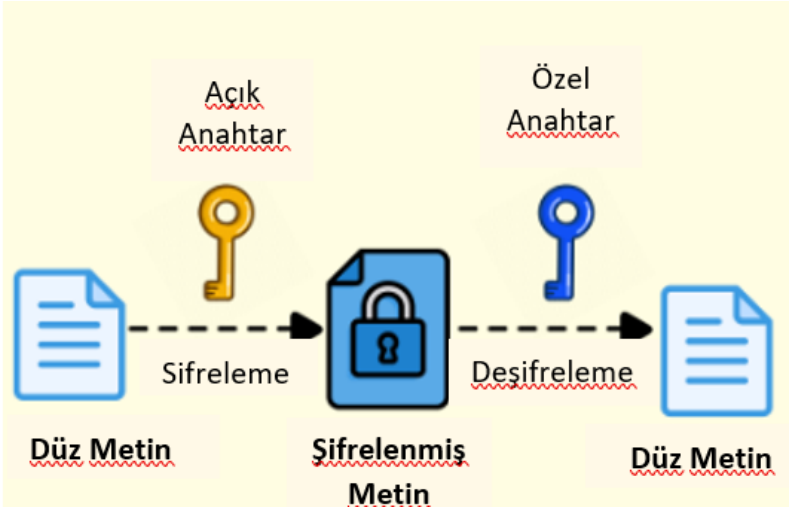
Simetrik şifreleme algoritmaları, verinin şifrelenmesi ve çözülmesi için aynı anahtarı kullanır. Bu anahtar, veriyi gönderen ve alıcılar arasında paylaşılır ve gizli tutulur. Şifrelenmiş veri ile anahtar, alıcıya iletildiğinde anahtar kullanılarak şifre çözme işlemi gerçekleştirilir (Şekil 1.1). Simetrik şifreleme algoritmaları, genellikle daha

hızlı çalışır ve uygulama prensipleri daha basittir. Ancak, anahtar dağıtımı gibi güvenlik sorunlarına neden olabilir.



Şekil 1.1. Simetrik Şifreleme.

Asimetrik şifreleme algoritmaları ise farklı anahtarlar kullanarak şifreleme ve çözme işlemlerini gerçekleştirir (Şekil 1.2). Bu algoritmaların kullanımı daha karmaşıktır ancak anahtar yönetimi daha güvenlidir. Anahtar boyutları genellikle simetrik şifreleme algoritmalarından daha uzundur. Anahtar dağıtımı gibi sorunları çözmek için asimetrik şifreleme algoritmaları tercih edilebilir.



Şekil 1.2. Asimetrik Şifreleme.

1.1 Tezin Amacı

Bu çalışmada, biyometrik görüntülerin şifrelenmesi, deşifre edilmesi ve doğrulanması için mevcut yaklaşımlara alternatif olarak etkin ve güvenliği artıracak bir yaklaşım modeli hedeflenmiştir. Bu amaçla, 1978 yılında Rivest, Adleman ve Dertouzos ile kriptoloji terminolojisine katılan ve pratik olarak teorik yapısı ile 2009 yılında Craig Gentry'nin doktora tezinin konusunu oluşturan homomorfik şifrelemeye odaklanılmıştır. Homomorfik şifrelemenin gizlilik ve güvenlik bakımından önemi bulut hizmetinin gerçek verileri bilmeden hesaplamalar yapabilmesinden kaynaklanmaktadır. Diğer taraftan, Arnold dönüşümünün görüntü şifrelemede kullanılan etkin yöntemlerden biri olması nedeni ile tezin hedefi, iki boyutlu Arnold (2D) dönüşümü ile karıştırılan biyometrik verilerin uygun bir homomorfik şifreleme yöntemiyle ardışık uygulanarak güvenliğin güçlendirilmesi olarak belirlenmiştir.

Önerilen yöntem, bir renkli görüntünün RGB bileşenlerinin piksel sırasını Arnold dönüşümü ile karıştırır. Ardından, asimetrik şifrelemenin en önemli şifreleme yöntemlerinden biri olan ElGamal kısmi homomorfik şifreleme algoritması uygulanır. Büyük asal sayıların kullanımı ve üstel hesaplamaların zorluğu üzerine kurulu olan bu yöntem, hem güvenlik hem de performans açısından etkili sonuçlar sunar**.**

1.2 Literatür Araştırması

Görüntü Şifreleme alanında yapılmış birçok çalışma mevcuttur. Başlangıçta, görüntü dosyalarının şifrelenmesi için metin şifreleme algoritmaları tercih edilmiştr. Ancak bu yaklaşımın kullanılmasıyla iki adet mühim problem ortaya çıkmıştır. Birincisi, görüntüler metinlere göre çok daha büyük yer kapladığı için görüntü şifreleme metin şifrelemeden daha yavaş gerçekleşmektedir. İkincisi, şifre çözme aşamasında, metinlerdeki kayıpların aksine, görüntülerde veri kayıpları yaşanabilir. Bu veri kayıpları, görüntünün yapısını bozan bilgilerin, örneğin bit ve renk bilgilerinin zarar görmesiyle oluşabilir. Şifre çözüldüğünde, veri kayıpları yaşanabilmekte ve fark edilemeyecek kadar ufak olabilirken, bazen de gözle görülebilir boyutta olabilir

Yen ve Guo (1999) tarafından yapılan çalışmada, veri kayıplarını önlemek amacıyla ayna yansıması algoritması önerilmiştir. Bu algoritma, pikseller permütasyon yöntemiyle karıştırılır [1].

Veri kaybının minimize edilmesi amacıyla Chang, Hwang ve Chen (2001) tarafından yapılan bir çalışmada, karmaşık bir şifreleme algoritması geliştirmiş ve herhangi bir veri kaybı olmadan şifreleme işlemi gerçekleştirmiştir. Önerilen algoritma, görüntüdeki piksel değerlerini değiştirmek yerine, piksellerin konumlarını değiştirerek şifreleme işlemini gerçekleştirir [2].

Bunların yanı sıra, görüntü şifrelemede kullanılan bir diğer algoritma olan Brie algoritması, piksel yerlerinin değiştirilmesi mantığına dayanmaktadır. Shujun ve Xuan (2002) tarafından yapılan çalışmada, bu algoritmanın saldırılara karşı yeterince güvenli olmadığı tespit edilmiş ve güvenliğin öncelikli olduğu durumlarda kullanılmaması gerektiği belirtilmiştir [3].

Arnold Cat Mapping kullanılarak görüntülerin piksel konumlarının karıştırılmasıyla gerçekleştirilen bir çalışmada, dijital renkli görüntülerin şifrelenmesi hedeflenmiştir. Arnold Cat Mapping, geleneksel yöntemler kullanarak görüntü piksellerinin değerlerini değiştirip gri tonlu karıştırma görüntüsü oluşturan bir algoritmadır. Bu yöntem, karıştırma görüntüsü üzerinde difüzyon mekanizmasıyla birleştirme-karıştırma sürecini kullanarak şifreleme işlemini yapar. Önerilen bu yöntem, Veginner Substitution Cipher ve Hill Cipher gibi klasik şifreleme yapılarını içermektedir. Bu çalışma kapsamında yapılan deneylerde, Hill Cipher yönteminin diğerlerine kıyasla daha güvenli ve hızlı olduğu belirlenmiştir (Hariyanto ve Rahim, 2016) [4].

Başka bir çalışmada ise, görüntü şifreleme teknikleri ile metin şifreleme arasındaki farklar üzerinde durulmuştur. Bu farkların en aza indirilmesi için görüntü üzerinde pozisyon permütasyonu teknikleri, değer dönüşümü teknikleri ve bu tekniklerin kombinasyonlarının uygulanması gerektiği vurgulanmıştır. Son yıllarda, Kaotik Haritalar yapısına dayalı görüntü şifrelemesinde permütasyon ve difüzyon işlemlerinin kullanıldığı belirtilmiştir (Sharma, Godara, Singh, Tech ve Sabo, 2012) [5].

Görüntü şifrelemede AES gibi algoritmaların da kullanılabileceği düşünülmüştür. Ghoradkar ve Shinde (2015) tarafından yapılan bir çalışmada, AES'in görüntü şifrelemede kullanılabileceği belirtilmiştir. Bu çalışmada, ayrıca AES şifreleme algoritmasının işleyiş aşamaları da ayrıntılı olarak ele alınmıştır [6].

Başka bir araştırmada, görüntülerin AES algoritması ile şifrelenmesine yer verilmiştir. Bu yöntemde, görüntü önce dizi formatına dönüştürülmüş ve ardından AES algoritması kullanılarak şifrelenmiştir. Ayrıca, RSA algoritmasının da görüntü

şifrelemede kullanılabileceği vurgulanmıştır. RSA, hem şifreleme hem de kimlik doğrulama sağlayan bir algoritmadır. Bu tür bir şifreleme sisteminde, şifreleme anahtarı herkese açıkken, şifre çözme anahtarı gizli tutulur (Anandakumar, 2015) [7].

Xingbin Liu, Dia Xiao ve Yanping Xiang (2018) tarafından, kuantum görüntü şifrelemesi için bit seviyesinde permütasyon stratejisi içeren bir yöntem önerilmiştir. Bu yöntemde, görüntü önce yeni bir kuantum temsil modeliyle şifrelenir ve ardından bitler üzerinde XOR (Exclusive Or) işlemi ile permütasyon uygulanır [8].

Satish A. ve diğerleri tarafında 2019 yılında, Arnold dönüşümünü dikkate alarak şifre çözme karmaşıklığını artırmaktır. İlk seviyede, $N \times N$ boyutundaki orijinal görüntü üzerinde Arnold piksel karıştırması yapılır ve pikseller iterasyon sayısına göre karıştırılır. İkinci seviyede, karıştırılmış görüntü 8×8 bloklara bölünür ve blok bazında Arnold karıştırma gerçekleştirilir. Şifreleme işlemi sırasında, karıştırılmış görüntü için önce ters Arnold blok karıştırma uygulanır sonra Arnold piksel karıştırma yapılır [9].

Wen-Wen Hu ve diğerleri tarafından 2020 yılında, modifiye edilmiş esnek kuantum görüntüsü temel alınarak yeni bir kuantum görüntü şifreleme algoritması önerilmiştir. Şifreleme işlemi, öncelikle kuantum görüntü bilgisini mekansal alanı bozmak için Arnold karıştırma işlemini gerçekleştirirler. Ardından, önerilen algoritmada, yazarlar öncelikle kuantum görüntü bilgisini mekansal alanda bozmak için Arnold karıştırma işlemi ve kuantum wavelet dönüşümlerini ardışık olarak kullanarak karışık kuantum görüntüsü elde etmektedir. Bu sayede frekans alanında bir dizi alt görüntü oluşturulmaktadır [2].

2. ŞİFRELEME YÖNTEMLERİ VE GÖRÜNTÜ ŞİFRELEME

Bu bölümde, öncelikle şifreleme bilimini oluşturan temel kavramlar kısaca verildikten sonra çalışmaya konu olan homomorfik şifreleme ve görüntü şifreleme tanıtılmaktadır.

2.1 Kriptografi ve Kriptoanaliz

Kriptoloji, bilgilerin matematiksel prensipler kullanılarak gizlenmesi ve açığa çıkarılmasıyla ilgilenen bir bilim dalıdır. İki ana dalı vardır: Kriptografi ve Kriptoanaliz. Bir şifreleme algoritmasının sahip olması gereken temel özellikler aşağıda belirtilmiştir [11].

- *Mahremiyet:* Bilgi, sadece yetkili şahıslar tarafından anlaşılabilir olmalıdır.
- *Bütünlük:* Bilgi, yetkilendirilmemiş kişiler tarafından değiştirilememelidir.
- *İnkâr Edilemezlik*: Bilgiyi oluşturan veya ileten kişi, daha sonradan bu eylemi inkâr edememelidir .
- *Erişilebilirlik:* Yetkilendirilmiş kişiler, ihtiyaç duydukları anda gereken bilgiye erişebilmelidir.
- *Kimlik Doğrulama:* Mesajı ileten ve alan taraflar, birbirlerinin kimliklerini doğrulamalıdır .

Kriptografi, verilerin açık durumdan korunan ve gizli hale getirilen işlemler bütününü içerir. Bu süreç, verilerin gizliliği, bütünlüğü ve güvenliği için önemlidir. Süreci gerçekleştiren kişilere kriptograf denir. Orijinal metin, yeterince açık bir formatta olduğunda düz metin veya açık metin olarak adlandırılır. Düz metin, çeşitli işlemlerden geçirilerek anlaşılmayacak bir formata dönüştürüldüğünde yeni bir şifreli metin elde edilir [4,5]. Kriptografinin ana hedefi bilgiyi gizli tutmaktır.

Kriptografide başlıca yöntemler; yerine koyma, yer değiştirme ve cebirsel yöntemler gibi klasik şifreleme yöntemleri yanında simetrik ve asimetrik anahtarlı şifreleme yöntemlerinin oluşturduğu modern şifreleme yöntemleridir.

Kriptoanaliz, kriptografik sistemleri analiz ederek ve zayıf noktalarını ortaya çıkararak şifrelenmiş bilgileri çözerek orijinal bilgiye ulaşmayı hedefler. Bu alandaki araştırmalar, güçlü şifreleme algoritmalarının geliştirilmesi ve güvenlik açıklarının

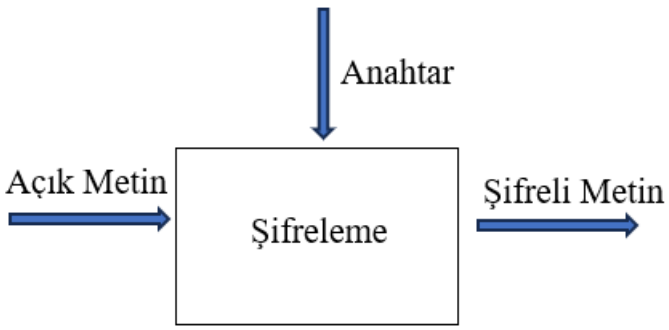
kapatılması için önemlidir. Frekans analizi, diferansiyel kriptoanaliz, lineer kriptoanaliz gibi çeşitli yöntemler, kriptoanalistlere şifreleme sistemlerini anlama ve zayıf noktalarını bulma konusunda yardımcı olur.

Kriptografi, bilgi güvenliği alanında üç temel unsurun korunmasını sağlar:

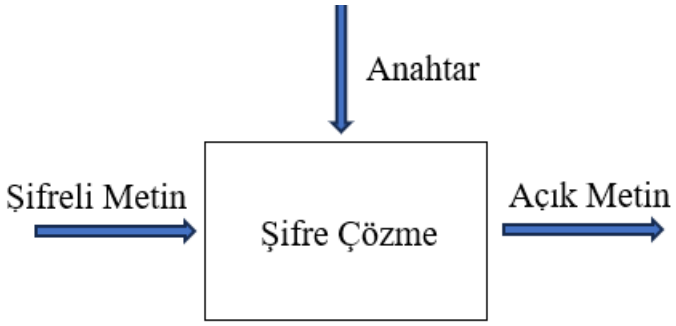
- *Veri Gizliliği*; bilginin kötü niyetli veya istenmeyen kişiler tarafından ele geçirilmesi durumunda anlaşılamaz hale gelmesi,
- *Veri Bütünlüğü*; iletilmek istenen kişiye tam ve eksiksiz bir şekilde ulaşması,
- *Veri Erişilebilirliği*; yetkililer tarafından gerektiğinde erişilebilir ve kullanılabilir durumda olması .

Kriptografi, bilgi güvenliğini sağlamak için önemli bir araç olarak kabul edilir. Bu araçlar aşağıdaki gibi sıralanabilir.

- *Açık metin (Düz metin):* İşlenmemiş, orijinal haliyle bilgi,
- *Şifreli metin*: Matematiksel işlemlerle dönüştürülmüş açık metin,
- *Şifreleme*: Açık metni şifreli metin haline getiren matematiksel işlemler topluluğu,
- *Şifre çözme (Deşifreleme):* Şifreli metni açık metne dönüştüren matematiksel işlemler dizisi,
- *Anahtar*: Şifreleme ve şifre çözme işlemlerinde kullanılan veri parçasıdır. Şifreleme işlemi gönderici tarafından gerçekleştirilirken, şifre çözme işlemi alıcı tarafından yapılır.



Şekil 2.1 : Şifreleme İşlemi.



Şekil 2.2 : Şifre Çözme İşlemi.

2.2 Kriptografinin Gelişimi

MÖ 450 yıllarında Spartalılara ait Skytale adı verilen şerit şeklinde deri yada parşömenin bir tahta silindire sarılarak okunduğu yöntemin kullanıldığı kitabeler kriptolojinin ilk örneğidir. Skytale yönteminde şifrelemek istediğimiz metin "İstanbul Teknik Üniversitesi" olsun. Bir bez parçası üzerine 5 harf yan yana sığacak şekilde düşünüldüğünde ortaya çıkacak görüntü şu şekilde olacaktır. Parşömen üzerine yazıldığında tahta silindire sarılmadığı müddetçe tek şerit halinde görünecek ve "İBKNSİSUNİİTLİVTATKEENEÜRS" olarak görünecektir (Şekil 2.3).

İ	S	T	A	N
B	U	L	T	E
K	N	İ	K	Ü
N	İ	V	E	R
S	İ	T	E	S
İ				

Şekil 2.3 : Skytale Şifreleme Örneği.

MÖ 3. yüzyılda, Kautilya tarafından kaleme alınan Artha-Aastra, çeşitli kriptoanaliz yöntemlerini içermekteydi. MÖ 100-50 yıllarında ise Jül Sezar, komutanlarıyla iletişim kurmak için kullandığı kriptografi yöntemi olarak bilinen Sezar şifrelemesini geliştirdi.

16. yüzyılda Vigenere'in bir şifre tasarladığı ve bu şifrenin, şifreleme anahtarı kullanan ilk şifre olduğu belirtilmektedir [13]. Vigenere'in şifrelerinden birinde, şifreleme anahtarı mesajın tamamını kapsayacak şekilde birden çok kez tekrarlanır ve sonra şifreli metin, mesaj karakterinin anahtar karakteriyle toplanması ve çıkan sayının kullanılan alfabedeki harf sayısı n olmak üzere mod n değeri ile bulunur.

9. yüzyılın başında, her şeyin elektrikli hale geldiği dönemde, Hebern adında bir kişi, Hebern rotor makinesi olarak adlandırılan bir elektro-mekanik cihaz tasarladı. Bu cihazda, gizli anahtar dönen bir disk içine gömülüdür. Anahtar, bir yerine koyma tablosunu kodlar ve klavyeden her tuşa basıldığında şifreli metin çıktısı üretilir. Bu aynı zamanda diski bir çentik döndürür ve bir sonraki düz metin karakteri için farklı bir tablo kullanılır. Bu da harf frekansları kullanılarak kırılmıştır.

Arthur Scherbius tarafından icat edilen Enigma makinesi, Birinci Dünya Savaşı'nın sonlarında ortaya çıktı ve İkinci Dünya Savaşı sırasında Alman kuvvetleri tarafından yaygın bir şekilde kullanıldı. Bu şifreleme cihazı, genellikle üç veya dört adet olan farklı hızlarda dönen rotorlardan oluşuyordu. Kullanıcılar metin girdikçe uygun şifrelenmiş harfleri üretiyordu. Bu şifreleme yönteminin anahtarı, rotorların başlangıç pozisyonlarındaydı. İkinci Dünya Savaşı'na kadar, kriptografiyle ilgili yapılan çalışmaların çoğu askeri amaçlar için yapılmıştı, genellikle gizli askeri bilgileri saklamak için kullanılıyordu. Ancak, savaştan sonra kriptografi ticari ilgi çekmeye başladı ve işletmeler rakiplerinden verilerini korumaya çalıştı.

1970'lerin başında IBM, müşterilerinin şifreleme taleplerini karşılamak için Horst-Feistel liderliğinde bir "crypto group" kurdu ve Lucifer adında bir şifre geliştirdi. 1973'te, ABD'deki Ulusal Standartlar Bürosu (NIST), bir blok şifre için ulusal bir standart olacak bir teklif talep etti. Lucifer, DES veya Veri Şifreleme Standardı olarak kabul edildi. 1997'de DES, hesaplamalı gücün artmasıyla kırıldı ve NIST, yeni bir blok şifre için teklif talep etti. 2000'de Rijndael algoritması, AES veya Gelişmiş Şifreleme Standardı olarak kabul edildi. AES, günümüzde simetrik şifreleme için yaygın olarak kullanılmaktadır. Simetrik şifrelemenin yanında 1976 – 1978 yıllarında asimetrik anahtarlı RSA, Diffie-Hellman ve Homomorfik şifrelemeler geliştirilmeye başlanmıştır. Bu kısım çalışmanın ilerleyen bölümlerde detaylı anlatılacaktır.

2.3 Homomorfik Şifreleme

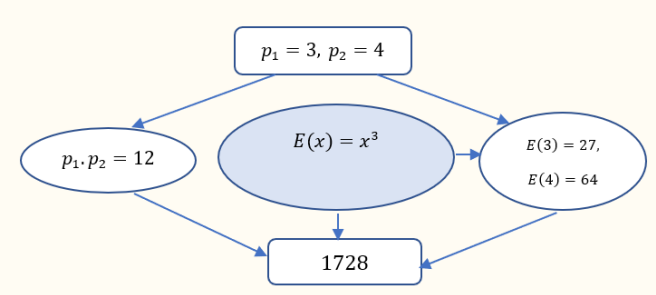
Matematikte, homomorfik, bir veri kümesinin diğerine dönüştürülmesini, her iki kümedeki öğeler arasındaki ilişkilerin korunmasını tarif eder. Terim, aynı yapı anlamına gelen Yunanca kelimelerden türetilmiştir. Homomorfik şifreleme şemasındaki veri, aynı yapıyı koruduğundan, şifreli veya şifresiz veri üzerinde gerçekleştirilen aynı matematiksel işlemler eşdeğer sonuçları sağlar. Bu yüzden, şifrelemenin güvenliğini tehlikeye atmadan şifreli veri üzerinde karmaşık matematiksel işlemlerin gerçekleştirilmesini sağlar. Bu açıdan geleneksel şifreleme yöntemlerinden farklıdır. Zira, matematiksel hesaplamaların doğrudan şifreli veri üzerinde gerçekleştirilmesine olanak tanır, bu da üçüncü tarafların kullanıcı verilerini işleme sürecini daha güvenli hale getirebilir. Homomorfik şifreleme, şifreli veriye sonsuz sayıda eklemeler yapılmasını sağlayan bir şifreleme algoritması oluşturmak için tasarlanmıştır.

Homomorfik şifreleme yaparken, şifreli veriye sonsuz sayıda ekleme veya çarpma işlemini olanaklı kılmak zordur. Bu nedenle, homomorfik şifreleme, nasıl tasarlandığına bağlı olarak farklı türde şifrelemelere ayrılabilir. Bir algoritma eklemeli homomorfikse, iki şifreli metni bir araya getirmek, iki açık metnin toplamını şifrelemekle aynı sonucu verir.

$$E(p_1 + p_2) = E(p_1) + E(p_2)$$

Burada p_i $(i = 1,2)$ düz metin karakterlerine karşı gelen sayıları, E şifreleme işlemini göstermektedir. Benzer şekilde, bir algoritma çarpmalı homomorfikse, aynı anahtarla iki şifreli metni çarpmak, açık metinlerin çarpımını gizli bir anahtarın gücüne yükseltmekle eşdeğerdir (Şekil 2.4).

$$E(p_1.p_2) = E(p_1).E(p_2)$$



Şekil 2.4 : Homomorfik Şifreleme İşleyişi.

Homomorfik şifreleme, hem eklemeli hem de çarpmalı olabilirken, aynı zamanda kısmen veya tamamen homomorfik olarak şifrelenmiş olabilir.

2.3.1 Matematiksel Altyapı

Bu bölümde şifrelemenin matematiksel alt yapısının anlaşılması için bazı temel tanım ve teoremler verilecektir.

Tanım 2.1. a ve b pozitif tam sayılar olsun. $a > 1$ koşuluyla, $1 \leq b < a$ aralığındaki tüm b tam sayıları için $\frac{a}{b}$ değerleri pozitif tam sayılar kümesi olan C 'nin elemanları olsun. Eğer $s(C) = 1$ ise, a sayısına asal sayı denir [14].

Teorem 2.1. $a, b, c \; ve \; d \in \mathbb{Z}^+$ olsun. $a^d \equiv b (mod\, c)$ ise $b \equiv a^d (mod\, c)$'dir [14].

Euler Phi Fonksiyonu (φ): Bir n pozitif tam sayısı verildiğinde, $\varphi(n), n$ ile aralarında asal olan ve 1 ile n arasında bulunan pozitif tam sayıların sayısını temsil eder.

$$\varphi(n) = \{\, k | 1 \leq k \leq n \,\}, EBOB(k, n) = 1$$

Örnek 2.1 : $n = 10$ için $\varphi(10)$ 'u hesaplayalım :

10 ile aralarında asal olan sayılar: $1, 3, 7, 9$. $\varphi(10) = 4$ elde edilir.

Teorem 2.2. $a \in \mathbb{Z}^+$ olsun. a tam sayısı, asal ise, $\varphi(a) = (a - 1)$ 'dir [15].

İspat : p'nin herhangi bir asal sayı olduğunu varsayalım. p'nin yalnızca 1 ve p olmak üzere iki böleni vardır. p'ye eşit veya p'den küçük olan $p - 1$ pozitif tam sayı vardır, bu nedenle $\varphi(\mathrm{p}) = \mathrm{p} - 1$. Dikkat edilirse, 1 ile p arasındaki tüm pozitif bölenler içinde p ile aralarında asal olmayan tek bölen p'nin kendisidir, çünkü $(p, p) = p$ ' ye eşittir.

Örnek 2.2 : $a = 7$ olsun. 1 ve 7 arasında 7 ile aralarında asal olan sayılar : $1, 2, 3, 4, 5, 6$. $\varphi(7) = 7 - 1 = 6$ 'dır.

Teorem 2.3. $a, b \in \mathbb{Z}^+$ olsun. a, b sayıları asal sayı ise

$$\varphi(a \; x \; b) = (a - 1) \times (b - 1) \text{ dir [15]}$$

İspat : $n = ab$ olsun. $\varphi(a\ x\ b)$ için n den n/a , n/b ve 1 i çıkaralım :

$$n - n/b - n/a + 1$$

Sonrasında $ab/n\ (= 1)$ ile çarpalım:

$$ab - a - b + 1 = a(b-1) - (b-1) = (a-1)x(b-1)$$

Örnek 2.3 : Asal sayılar: $a = 3$ ve $b = 5$ seçelim. İlk olarak, φ fonksiyonunu ayrı ayrı a ve b için hesaplayalım:

a = 3 için $\varphi(3) = 3 - 1 = 2$ olur.

$b = 5$ için $\varphi(5) = 5 - 1 = 4$ olur.

Şimdi, $a \times b$ ' yi hesaplarsak : $a \times b = 3 \times 5 = 15$

Son olarak, Teorem 2.3' e göre $\varphi(15)$ hesaplayalım ve teoremi doğrulayalım:

$\varphi(15) = (a-1) \times (b-1) = (3-1) \times (5-1) = 2 \times 4 = 8$

Ayrıca, 15 ile aralarında asal olan sayılar: $1, 2, 4, 7, 8, 11, 13, 14$.

Bu sayılar toplamda 8 tanedir, $\varphi(15) = 8$ olur.

Tanım 2.2. Bir küme olan G ve üzerinde tanımlı bir * işlemi düşünelim.

Eğer * işlemi aşağıdaki özellikleri sağlıyorsa, $(G, *)$ yapılanmasına "grup" denir:

(i) * işlemi birleşme özelliğini sağlar.

Her $a, b, c \in$ G için $(a * b) * c = a * (b * c)$.

(ii) Her $a \in G$ için bir $e \in G$ bulunur, öyle ki $a * e = e * a = a$.

Burada, e, G 'nin birim elemanı olarak adlandırılır.

(iii) Her $a \in G$ için bir $a^{-1} \in G$ bulunur,

$$a * a^{-1} = a^{-1} * a = e .$$

Burada a^{-1} , a 'nın ters elemanı olarak adlandırılır.

Eğer $(G, *)$ grubunda her $a, b \in G$ için $a * b = b * a$ ise, bu gruba "değişmeli" veya "abelyan" grup denir [16].

Örnek 2.4 : Bu örnek, toplama işlemi altında tam sayıların oluşturduğu bir gruptur.

Birleşme Özelliği: Her $a, b, c \in$ için $(a + b) + c = a + (b + c)$.

Birim Eleman: Her $a \in Z$ için $a + 0 = 0 + a = a$. Burada 0 birim elemandır.

Ters Eleman: Her $a \in Z$ için $\exists -a \in Z$ öyle ki, $a + (-a) = (-a) + a = 0$. Burada $-a, a$'nın ters elemanıdır.

Değişme Özelliği: Her $a, b \in Z$ için $a + b = b + a$. Bu nedenle, $(\mathbb{Z}, +)$ bir abelyan gruptur.

Tanım 2.3. G bir grup ve A, G nin boş olmayan bir altkümesi olsun. Eğer her $a, b \in A$ için $ab \in A$ ise o zaman A, G nin işlemine göre kapalı denir [16].

Tanım 2.4. G bir grup ve H, G nin boş olmayan bir altkümesi olsun. Eğer H, G nin işlemine göre kapalı ve bu işleme göre bir grup ise H' ya G nin altgrubu denir ve $H \leq G$ ya da $G \geq H$ ile gösterilir [16].

Tanım 2.5. G bir grup ve a ,...,$a \in G$ olsun. Eğer $G = \langle a_1,...,a_n \rangle$ ise $a_1,...,a_n$ elemanlarına G'nin üreteçleri denir [17].

Tanım 2.6. R, boş olmayan bir küme olsun. R üzerinde her $a, b \in R$ için

$$+ : (a, b) \rightarrow a + b, \quad \times : (a, b) \rightarrow a \times b$$

biçiminde tanımlı ve sırasıyla,

toplama ve çarpma denilen "+" ve "×" ikili işlemleri verilsin. Eğer

(i) $(R, +)$ bir abelyan grup ise,

(ii) çarpma birleşme özelliğini sağlarsa; yani, her $a, b, c \in R$ için

$$(a \times b) \times c = a \times (b \times c) \text{ ise}$$

ve

(iii) R üzerinde dağılma özellikleri sağlanırsa; yani, her $a, b, c \in R$ için

$$a \times (b + c) = a \times b + a \times c \text{ (sol dağılma özelliği)}$$

$$(a + b) \times c = a \times c + b \times c \text{ (sağ dağılma özelliği)}$$

ise $(R, +, \times)$ sıralı üçlüsüne bir halka denir [16].

Tanım 2.7. R ve S iki halka ve $\varphi : R \rightarrow S$ fonksiyonu verilsin. Eğer her $a, b \in R$ için

$$\varphi(a + b) = \varphi(a) + \varphi(b) \text{ ve } \varphi(a \times b) = \varphi(a) \times \varphi(b)$$

ise φ 'ye R' den S' ye bir halka homomorfizması denir [16].

Teorem (Fermat'ın Küçük Teoremi) : p bir asal sayı ve a tamsayısı $(0 < a < p)$ p ile tam bölünmeyen bir tam sayı ise

$$a^{p-1} = 1(mod\, p)'\text{dir.}$$

İspatı : a ve p aralarında asal yani $EBOB(a,p) = 1$ olduğunu ve $Z_P = \{0,1,2 \dots, p-1\}$ kümesinin p ile bölündüğünde kalanı temsil eden sayılar olduğunu kabul edelim.

Fermat'ın Küçük Teoremi'nin ispatı için Lagrange Teoremi'nden yararlanacağız. Lagrange Teoremi'ne göre, bir sonlu gruptaki herhangi bir altgrubun mertebesi, grubun mertebesinin bir bölenidir.

Z_p^* kümesi, Z_P kümesindeki 0 dışında kalan tüm elemanlardan oluşur ve bu küme bir çarpma grubudur.

Bu grubun mertebesi $p-1$ 'dir, çünkü asal p için 0 dışındaki her sayı p ile aralarında asaldır.

a, p ile aralarında asal olduğundan a , Z_p^* grubunun bir elemanıdır.

a'nın çarpma tablosunu düşünelim: $\{a, 2a, 3a, \dots, (p-1)a\}$

Bu elemanlar $mod\, p$' de farklıdır ve $\{1, 2, 3, \dots, (p-1)\}$ permütasyonuna karşılık gelir.

Bu permütasyonun çarpımını düşünelim:

$a\ x\ 2a\ x\ 3a\ x \dots x\ (p-1)a = (1\ x\ 2\ x\ 3\ x \dots\ x\ (p-1))\ (mod\, p)$

eşitliği $a^{p-1} \cdot (p-1)! \equiv (p-1)!\,(mod\, p)$ verir. Her iki taraftan $(p-1)!$ terimini çıkartabiliriz, çünkü $(p-1)!$ asal p' ye tam bölünmez. Sonuç olarak,

$$a^{p-1} = 1(mod\, p)$$

elde edilir.

2.3.2 Kısmi-Homomorfik Şifreleme

Kısmen homomorfik şifreleme, şifreli veriler üzerinde toplama veya çarpma gibi tek bir işlemi istenilen sayıda gerçekleştirebilir. Diffie-Hellman Anahtar Değişimi, RSA [18], ElGamal [4] ve Paillier [19] Kısmi-Homomorfik Şifreleme özelliği gösterirler. Bu bölümde, Homomorfik Şifreleme yönteminin temelini oluşturan RSA, ElGamal ve Paillier algoritmaları ile bu algoritmaların bazılarını destekleyen Diffie-Hellman anahtar değişim protokolü incelenmiştir.

2.3.2.1 Diffie-Hellman Anahtar Değişimi

Diffie-Hellman Anahtar Değişimi, iki tarafın güvenli bir şekilde ortak bir gizli anahtar oluşturmasını sağlayan bir kriptografik protokoldür. Bu protokol, 1976 yılında Whitfield Diffie ve Martin Hellman tarafından önerilmiş olup, modern kriptografide devrim yaratmıştır. Anahtar değişimi, özellikle güvenli iletişim kurmak isteyen tarafların, ortak bir şifreleme anahtarını güvenli olmayan bir kanal üzerinden paylaşmalarına olanak tanır.

Diffie-Hellman Anahtar Değişimi, büyük asal sayılar ve bu asal sayıların üslerinin kullanımıyla çalışır. Protokolün işleyişi şu adımlarla özetlenebilir:

1. *Ortak Parametrelerin Seçimi:*

Nur ve Can adını verdiğimiz iki taraf, p ve q olarak pozitif tam sayılar üzerinde karşılıklı anlaşırlar, burada p bir asal sayıdır ve q, p'nin bir üreteçidir. Üreteç q, p'den küçük pozitif tam sayı kuvvetleri için aynı sonucu üretmez. p'nin değeri büyük olabilir ancak, q'nun değeri genellikle küçüktür. Bu parametreler herkese açık olabilir ve güvenli bir şekilde paylaşılabilir.

2. *Anahtarların Seçimi* :

Nur ve Can, sadece kendilerine ait olacak şekilde rastgele birer özel anahtar seçerler. Nur'un özel anahtarı a ve Can'ın özel anahtarı b olsun. Bu anahtarlar gizli tutulur ve kimseyle paylaşılmaz.

3. *Ortak Anahtarların Hesaplanması* **:**

Nur, ortak anahtarı hesaplamak için $A = g^a \, mod \ p$ işlemini gerçekleştirir ve elde ettiği A değerini Can'a gönderir.

Can da benzer şekilde ortak anahtarını hesaplamak için $B = g^b \, mod \, p$ işlemini gerçekleştirir ve elde ettiği B değerini Nur'a gönderir.

4. *Gizli Ortak Anahtarın Oluşturulması* :

Nur, Can'dan aldığı B değerini kullanarak gizli ortak anahtarı

$$K_N = B^a \ mod \, p$$

şeklinde hesaplar.

Can da Nur'dan aldığı A değerini kullanarak gizli ortak anahtarı

$$K_C = A^b \, mod \, p$$

şeklinde hesaplar.

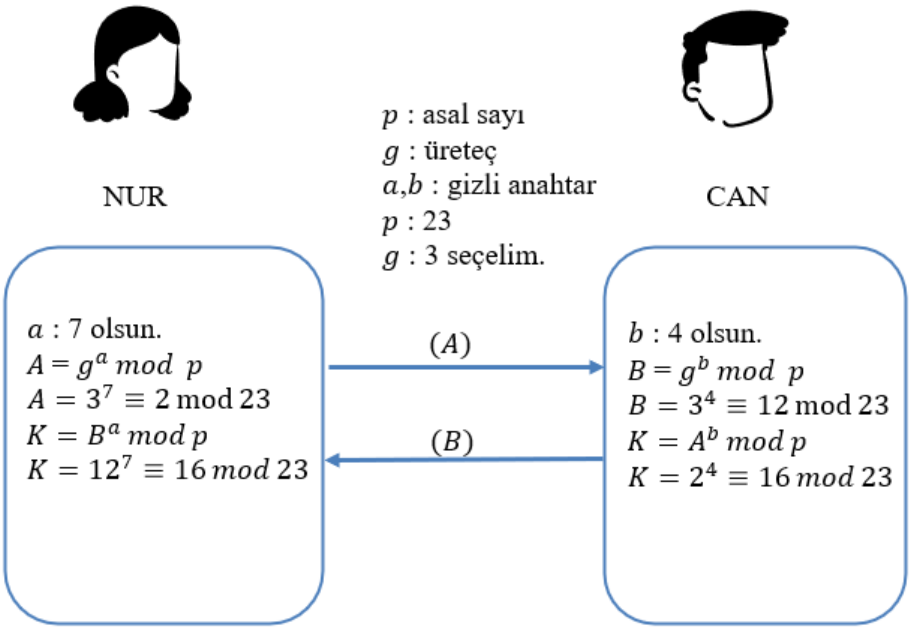
Matematiksel olarak, her iki taraf da aynı gizli ortak anahtarı elde eder. Can aynı gizli anahtara sahip olur.

$$K_N = (9^b \, mod \, p)^a \, mod \, p \; = g^{ba} \bmod p$$

$$K_C = (9^a \, mod \, p)^b \, mod \, p = g^{ab} \bmod p$$

Uygulama Alanları :

Diffie-Hellman Anahtar Değişimi, birçok kriptografik protokolün ve güvenlik mekanizmasının temelini oluşturur (Şekil 2.5). Örneğin, Transport Layer Security (TLS) ve Internet Key Exchange (IKE) gibi protokoller, güvenli iletişim kanallarının kurulmasında Diffie-Hellman anahtar değişimini kullanır. Bu yöntem, aynı zamanda VPN bağlantıları, e-posta güvenliği ve diğer birçok alanda da yaygın olarak kullanılmaktadır.



Şekil 2.5 : Diffie-Hellman Anahtar Değişimi.

Bu şifreleme yöntemi büyük asal sayılarla çalıştığından yüksek hesaplama gücü gerektirir ve düşük güçlü cihazlarda performans sorunları yaratabilir. Kuantum bilgisayarların yaygınlaşması durumunda kırılabilir, bu nedenle kuantum güvenli algoritmalar gereklidir. Diffie-Hellman protokolü ek güvenlik önlemleri ile desteklenmelidir.

2.3.2.2 RSA Algoritması

Diffie ve Hellman, asimetrik kriptoloji sistemlerinde, şifreleme işlemi kolay ancak tersine çevrilebilmesi zor olan tek yönlü fonksiyonlar önermiştir. Bu fonksiyonların kullanıldığı asimetrik şifreleme sisteminin ilk örneği RSA ise, 1978 yılında R. Rivest, A. Shamir ve L. Adleman tarafından geliştirilmiş ve "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems" isimli makalede yayınlanmıştır. RSA, geliştiricilerinin soyadlarının baş harflerinden oluşan adını almıştır ve 1983'te Amerika'da patentlenmiştir. RSA, hem şifreleme hem de dijital imza sağlayan bir kriptografik yapıdır ve güvenliği, büyük asal sayıların çarpanları olan p ve q gibi sayıların faktörlerinin bulunmasının zorluğuna dayanır. Bugüne kadar büyük sayıları faktörize ederek RSA'yı kırmaya yönelik yapılan girişimlerin başarılı olmamıştır [15].

RSA algoritmasının ana prensipleri şunlardır:

1. *Açık Anahtarlı Şifreleme:* RSA, her kullanıcının bir açık anahtar (E) ve bir gizli anahtar (D) olmak üzere iki anahtara sahip olduğu bir sistemdir. Açık anahtar, herkese açık bir şekilde paylaşılır ve mesajların şifrelenmesinde kullanılır. Gizli anahtar ise sadece kullanıcısı tarafından belirlenir, paylaşılmaz. Bu simetrik olmayan yapı sayesinde, mesajları sadece doğru gizli anahtara sahip olan kişi çözebilir.
2. *Dijital İmza:* RSA, dijital imza oluşturmak için de kullanılır. Bir gönderici, mesajını gizli anahtarıyla imzalar ve alıcılar, gönderici tarafından imzalanan mesajın doğruluğunu alıcının açık anahtarıyla doğrular. Bu süreç, mesajın gönderici tarafından gönderildiğini ve sonradan inkar edilemeyeceğini sağlar.
3. *Asal Sayılar ve Faktorizasyon:* RSA'nın güvenliği, iki büyük asal sayının çarpımını faktörize etmenin zorluğuna dayanır. Eğer p ve q gibi iki büyük asal sayı seçilirse, $n = p * q$ formülü ile bir modulus oluşturulur. Bu modulus ile n'in faktörlerini bilmeden, RSA mesaj şifreleme ve çözme işlemleri güvenli bir şekilde gerçekleştirilir.

RSA (Rivest-Shamir-Adleman) algoritması, geniş bir uygulama yelpazesine sahiptir. İlk olarak, internet üzerindeki güvenli iletişimde SSL/TLS protokollerinde kullanılır, bu da web tarayıcıları ve sunucular arasında güvenli veri iletişimini sağlar. Ayrıca dijital imzaların oluşturulması ve doğrulanması için kullanılır, bu sayede belgelerin kimlik doğrulaması yapılabilir. Finansal işlemlerde elektronik ödemelerin güvenliğini sağlamak amacıyla da kullanılır. Özellikle elektronik fon transferleri için de uygulamalara sahiptir. Finansal bilgilerin güvenli olması gereklidir ve RSA ile çekler elektronik olarak imzalanabilir. Bu tür işlemler için özel çek numaraları gibi ek önlemler alınabilir ki bu numaralı çekler sadece bir kez nakit edilebilir veya iletilip kullanılabilir [15]. Genel olarak, RSA'nın güvenlik, gizlilik ve kimlik doğrulama gerektiren her türlü elektronik sistemde kritik bir role sahiptir.

Örnek 2.5 :

1. iki asal sayı olarak $p = 53, q = 59$ seçelim.

2. $n = p\,x\,q$ ile $p = 3217$.

3. $\varphi(n) = (\rho - 1)(q - 1) = 52\,x\,58 = 3016$ (Teorem 2.3)

4. $1 < e < \varphi(n)$ ve $ebob(e, \varphi(n)) = 1$

olacak şekilde bir e tam sayısı seçelim: e = 3

5. $d.e = 1\,mod(\varphi(n))$ olacak şekilde d tam sayısı hesaplayalım:

$$d.e = 1\,mod(\varphi(n)) \text{ ise } d.e = 1 + k.\varphi(n)\,,\, d = \frac{(1 + k.\varphi(n))}{e}$$

$$d = (1 + 2.3016)/3 = 2011.$$

6. $p, q, k\ ve\ \varphi(n)$ değerleri gizlenecek, n ve e değerleri açık olacaktır.

Açık metin $\boldsymbol{m}$ ve şifreli metin $\boldsymbol{c}$ olsun. m sayısı aşağıda şekilde şifrelenir:

$m = 89$ alırsak , $m^e \ mod\, n = c \ => \ 89^3\ mod\ 3217 = 1394.$

7. $c^d mod\ n = m \ => \ (1394)^{2011}\ mod\ 3217 = 89.$

RSA algoritmasının bazı olumsuz yanları vardır. İlk olarak, anahtar yönetimi kritik öneme sahiptir; büyük anahtar uzunlukları gerektiğinde, anahtarların güvenli bir şekilde oluşturulması, saklanması ve paylaşılması zor olabilir.

İkinci olarak, RSA'nın güvenliği büyük asal sayıların faktörizasyon zorluğuna dayandığından, özellikle de anahtar uzunlukları arttıkça hesaplama gücü yoğunluğu gerektirebilir.

Üçüncü olarak, yüksek güvenlik seviyeleri için kullanılan büyük anahtar uzunlukları performansı olumsuz etkileyebilir, özellikle hızlı işlem gerektiren uygulamalarda.

Dördüncü olarak, bilgisayar gücü ve kriptoanaliz tekniklerinin ilerlemesi, RSA'nın gelecekteki güvenlik zorluklarıyla karşı karşıya kalabileceği anlamına gelmektedir.

Son olarak, açık anahtarların güvenli dağıtımı ve korunması gerekliliği, RSA'nın genel güvenliği üzerinde önemli bir etkiye sahiptir. Yüksek güvenlik gerektiren uygulamalarda dikkatle planlanmalı ve uygulanmalıdır.

2.3.2.3 Paillier Algoritması

Paillier şifreleme algoritması, 1999 yılında Pascal Paillier tarafından önerilen bir asimetrik şifreleme yöntemidir. Bu algoritma, homomorfik özellikleri sayesinde çeşitli kriptografik protokoller ve gizlilik odaklı hesaplamalarda kullanılır. Algoritmanın ana hatları şu şekildedir:

Anahtar Üretimi

1. *İki büyük asal sayının seçimi:*

p ve q olmak üzere, iki büyük asal sayı seçilir.

2. Modülüs hesaplanması:

$$n = p \times q$$

$$\lambda = EKOK(p-1, q-1)$$

3. Rastgele bir g sayısının seçimi:

g, n^2, ile aralarında asal olacak şekilde $g \in \mathbb{Z}_n^2$ seçilir.

4. Fonksiyon L tanımlanır:

$$L(u) = \frac{u-1}{n}$$

$$\mu = \left(L\left(g^{\lambda} \bmod n^2\right)\right)^{-1} \bmod n$$

5. Açık ve gizli anahtarların oluşturulması:

Açık anahtar : (n, g)

Gizli anahtar : (λ, μ)

Şifreleme

1. *Mesajın belirlenmesi:*

   Şifrelenecek mesaj m, $m \in \mathbb{Z}_n^2$ olmalıdır $(0 \leq m < n)$.

2. *Rastgele bir r sayısının seçimi:*

   r, n ile aralarında asal bir sayı olacak şekilde $r \in Z_n$

3. *Şifreli metin c'nin hesaplanması:*

$$c = g^m . r^n \bmod n^2$$

Şifre Çözme

1. *Şifreli metin ccc'nin alınması:*

   Şifreli metin c, $c \in Z_n (0 \leq c < n^2)$

2. *Mesaj m'nin hesaplanması:*

$$m = L\left(c^\lambda \bmod n^2\right) . \mu \bmod n$$

Paillier algoritması, homomorfik şifreleme yeteneği sayesinde güvenli hesaplamalar için güçlü bir araçtır ve çeşitli kriptografik uygulamalarda kullanılır.

2.3.2.4 ElGamal Algoritması

ElGamal algoritması, sonlu bir alan üzerinde ayrık logaritma problemine dayanan bir Açık Anahtar Şifreleme Sistemi olup, 1985 yılında Taher ElGamal tarafından önerilmiştir. Bir grup G ve bu grubun bir elemanı g (bu genellikle grubun bir üreteci olarak seçilir) ve bir başka elemanı h verildiğinde, $h = g^x$denklemini sağlayan x tam sayısını bulma problemine "ayrık logaritma problemi" denir. Burada x, g tabanında h'nin ayrık logaritmasıdır.

$$x = log_g(h)$$

Ayrık logaritma problemi, büyük sayılarla çalışıldığında çözülmesi zor bir problemdir. Bunun nedeni, bilinen en iyi algoritmaların bile büyük modüllerde bu problemi çözmesinin oldukça zaman alıcı olmasıdır. Birçok kriptografik sistemin güvenliğini sağlar. Bu problem üzerinde etkin algoritmalar geliştirilmediği sürece, bu sistemlerin

güvenli kalacağı öngörülmektedir. Elgamal algoritmasının detayları son bölümde anlatılacaktır.

2.3.3 Tam Homomorfik Şifreleme Algoritmaları

Kısmi homomorfik şifrelemeden farkı, hem toplama hem de çarpma gibi çeşitli matematiksel işlemleri şifrelenmiş veriler üzerinde gerçekleştirebilme yeteneğine sahip bir algoritma olmasıdır. Böylece, herhangi bir aritmetik işlemi şifrelenmiş veriler üzerinde gerçekleştirmeyi mümkün kılar. Şifrelenmiş veriler üzerinde her türden hesaplamayı yapabilme yeteneği sayesinde geniş bir uygulama yelpazesine sahiptir. Bu özellik, veri gizliliğinin kritik olduğu bulut bilişim, veri analitiği ve sağlık hizmetleri gibi alanlarda kullanılabilir.

Bu bölümde Gentry'nin Şeması, BGV, BFV, CKKS tam homomorfik şifreleme algoritmalarından bahsedilecektir.

2.3.3.1 Gentry'nin Şeması

Craig Gentry, 2009 yılında tam homomorfik şifrelemenin ilk pratik uygulanabilirliğini gösteren üç ana bileşenli bir şema önerdi:

1. Bootstrapping: Gentry'nin şeması, sürekli olarak artan gürültüyü yönetmek için "bootstrapping" adı verilen bir teknik kullanır. Bootstrapping, şifreli bir metni kendi şifresini çözmeden yeniden şifreleyerek gürültüyü azaltır.

2. Ideal Lattices: Şema, ideal kafeslere dayanan şifreleme teknikleri kullanır. Bu matematiksel yapı, homomorfik özelliklerin güvenli bir şekilde uygulanmasını sağlar.

3. Gürültü Yönetimi: Her homomorfik işlem, şifrelenmiş metne gürültü ekler. Gentry'nin şeması, bu gürültüyü yönetmek ve kontrol altında tutmak için çeşitli teknikler kullanır.

Gentry'nin çalışması, tam homomorfik şifreleme alanında çığır açmış ve bu alanda birçok yeni araştırmanın temelini oluşturmuştur.

2.3.3.2 BFV (Brakerski/Fan-Vercauteren) Şeması

BFV şeması, Brakerski ve Vaikuntanathan'ın çalışmaları ile Fan ve Vercauteren'in katkıları sonucu geliştirilmiş bir tam homomorfik şifreleme şemasıdır. BFV'nin özellikleri şunlardır:

1. *Modüler Şifreleme*: BFV, tam sayılar üzerinde homomorfik işlemleri destekler.
2. *Güçlü Güvenlik*: RLWE problemine dayanır ve yüksek güvenlik sağlar.
3. *Pratik Uygulamalar*: BFV, veri gizliliği ve güvenliği gerektiren birçok pratik uygulamada kullanılabilir.

2.3.3.3 BGV (Brakerski-Gentry-Vaikuntanathan) Şeması

BGV şeması, 2011 yılında Brakerski, Gentry ve Vaikuntanathan tarafından önerilen bir tam homomorfik şifreleme şemasıdır. Bu şema, ideal kafesler yerine "learning with errors" (LWE) veya "ring learning with errors" (RLWE) problemine dayanır. BGV'nin özellikleri şunlardır:

1. *Verimli İşlemler*: BGV, özellikle çarpma işlemleri için verimlidir ve bu sayede daha pratik bir kullanım sunar.
2. *Modüler Redüksiyon*: Şema, modüler redüksiyon tekniklerini kullanarak gürültü yönetimini optimize eder.
3. *Parametre Ayarlamaları*: BGV, çeşitli parametre ayarlamaları ve optimizasyonlarla performansı artırır.

2.3.3.4 CKKS (Cheon-Kim-Kim-Song) Şeması

CKKS şeması, 2017 yılında Cheon, Kim, Kim ve Song tarafından önerilen bir homomorfik şifreleme şemasıdır. Bu şema, özellikle sayısal hesaplamalar için tasarlanmıştır ve aşağıdaki özelliklere sahiptir:

1. *Yaklaşık Şifreleme*: CKKS, gerçek sayıları yaklaşık olarak şifreleyebilir. Bu, makine öğrenimi ve veri analitiği gibi uygulamalarda büyük bir avantaj sağlar.
2. *Verimli İşlemler*: CKKS, toplama ve çarpma gibi temel işlemleri verimli bir şekilde destekler.
3. *Düşük Gürültü:* Şema, düşük gürültü seviyeleri ile çalışarak daha uzun hesaplama zincirlerine izin verir.

2.4 Görüntü Şifreleme

Görüntü şifreleme, dijital görüntülerin yetkisiz erişimden korunması amacıyla dönüştürülmesi işlemidir. Bilgi güvenliğinin giderek önem kazandığı günümüzde, görüntülerin güvenli bir şekilde iletilmesi ve depolanması, sağlık, askeri, endüstriyel ve kişisel gizlilik gibi pek çok alanda kritik bir gereklilik haline gelmiştir. Geleneksel şifreleme algoritmaları metin verilerini korumak için etkili olsa da görüntü verilerinin yapısal özellikleri ve büyük boyutları, bu algoritmaların doğrudan uygulanmasını zorlaştırmaktadır. Bu nedenle, görüntülerin özel şifreleme yöntemleriyle korunması gerekmektedir. Görüntü şifreleme teknikleri, piksel permütasyonu, kaotik haritalar, ve homomorfik şifreleme gibi çeşitli yöntemleri içermekte olup, bu teknikler aracılığıyla görüntülerin bütünlüğü ve gizliliği güvence altına alınmaktadır.

Dijital Görüntü :

Dijital görüntü, bir sahnenin veya nesnenin sayısal formatta temsil edilen görsel bir kaydıdır. Dijital görüntüler, bilgisayarlar tarafından işlenebilir, saklanabilir ve iletilebilir. Bu görüntüler, iki boyutlu bir düzlemdeki noktaların (piksel) düzenlenmesiyle oluşturulur. Her piksel, belirli bir renk veya gri ton değeri ile ilişkilidir ve bir matris içinde düzenlenir.

Dijital görüntüler, piksellerin bir matris (2D) halinde düzenlenmesiyle temsil edilir. Bu matrisin her bir elemanı, görüntünün bir pikseline karşılık gelir. Piksellerin değerleri, gri tonlamalı görüntülerde tek bir yoğunluk değeri (0-255 arasında bir değer) veya renkli görüntülerde üç renk bileşeni (genellikle kırmızı, yeşil ve mavi - RGB) şeklinde ifade edilir.

Gri tonlamalı bir görüntü, her pikselin 0 ile 255 arasında bir yoğunluk değeri aldığı bir matris olarak temsil edilir. Burada 0 siyahı, 255 ise beyazı temsil eder.

Renkli bir görüntü ise genellikle üç ayrı matris kullanılarak temsil edilir: Kırmızı (R), Yeşil (G) ve Mavi (B) bileşenleri için ayrı ayrı matrisler. Her pikselin üç değeri vardır: R, G ve B bileşenleri. Bu çalışmada görüntü şifreleme algoritması olarak Arnold dönüşümü kullanılacak olup detayı ilerleyen bölümlerde verilecektir.

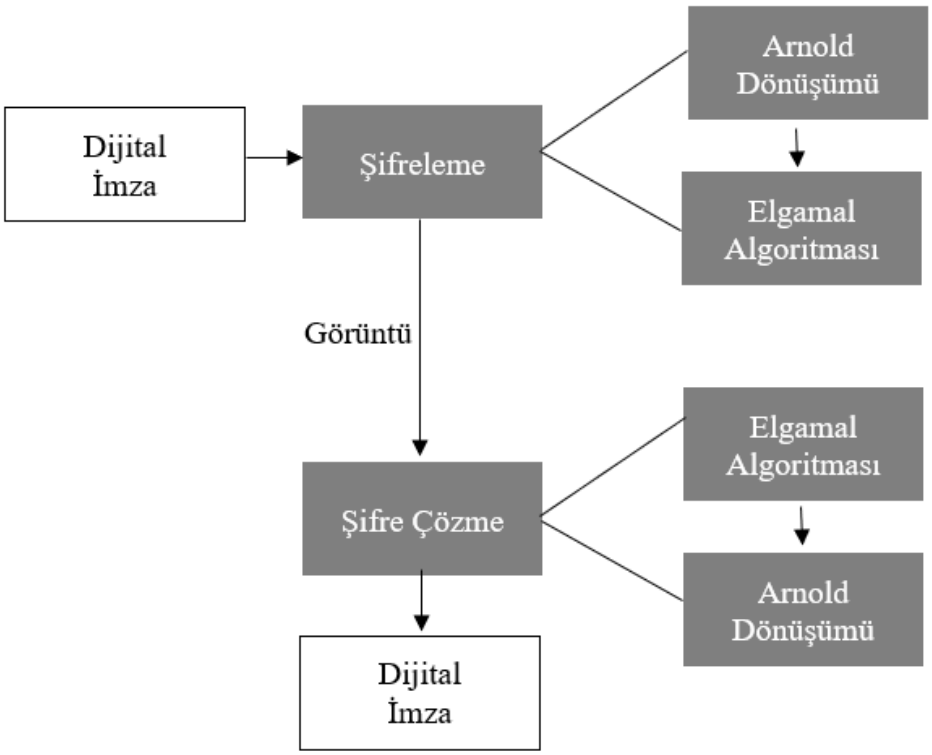
3. ÖNERİLEN YÖNTEM VE MATEMATİKSEL ANALİZ

Bu bölümde, genel olarak görüntü şifreleme de kullanılan yöntemlerden kısaca bahsedildikten sonra hedef problemin çözümünde kullanılan Arnold dönüşümü tanıtılmaktadır. Daha sonra, Arnold dönüşümü uygulanmış biometrik görüntülerin Elgamal yöntemiyle şifrelenmesi ve böylece Arnold dönüşümü ile karıştırılan verinin belirli bir periyodda orijinal hale getirilmesi mümkün olduğundan önerilen entegrasyon ile güvenliğin artırılması hedeflenmiştir.

3.1 Genel Çerçeve

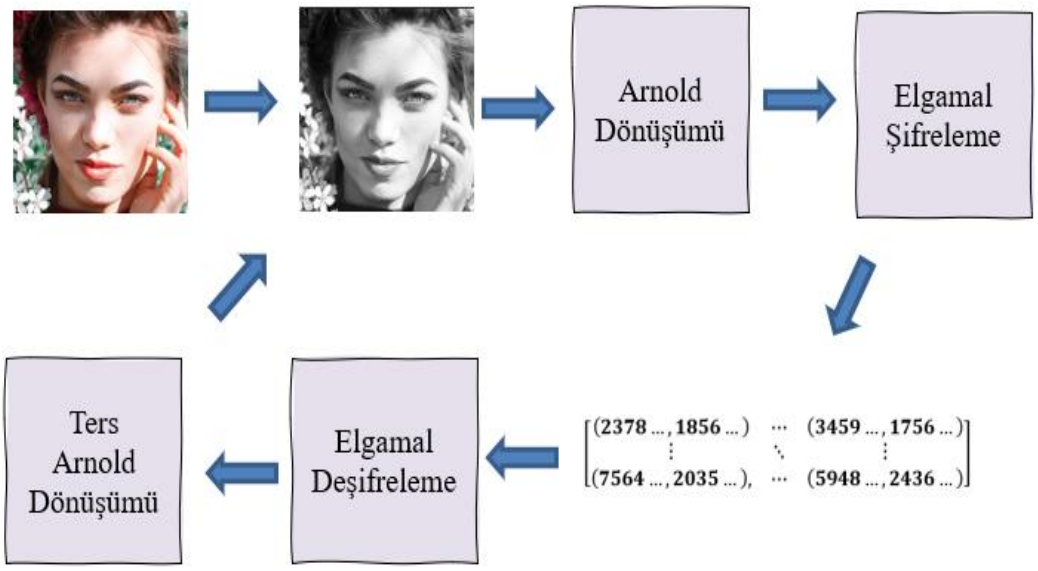
Genel çerçeve iki ana bölüme ayrılmıştır:

- Görüntünün pixellerinin karıştırılması ve her bir pixel değerinin şifrelemesi. Görüntü pixellerinin karışıtırılması için iki boyutlu Arnold dönüşmü (2D) ,
- Şifreleme için ElGamal Algoritmasının uygulanması (Şekil 3.1).



Şekil 3.1 : Önerilen yöntemin Şeması.

Hedeflenen problemde, bir görüntünün RGB (Red-Green-Blue) bileşenlerinin piksel sırası, Arnold dönüşümüyle değiştirilir. Böylece orijinal görüntüyü karıştırarak ilk şifreleme sağlanır ancak bu algoritma yalnızca piksel yerini değiştirdiği için saldırgan, açık metin ve şifreli metnin piksellerini karşılaştırarak doğrudan karıştırma kuralını keşfedebilir ve böylece sayıyı çalabilir. Değiştirilen piksel konumlarını da Elgamal algoritması ile tekrar şifreleyerek saldırılara karşı çok daha güçlü bir yapı kurulmaktadır (Şekil 3.2.).



Şekil 3.2 : Hedef Problem.

3.1.1 Arnold Dönüşümü

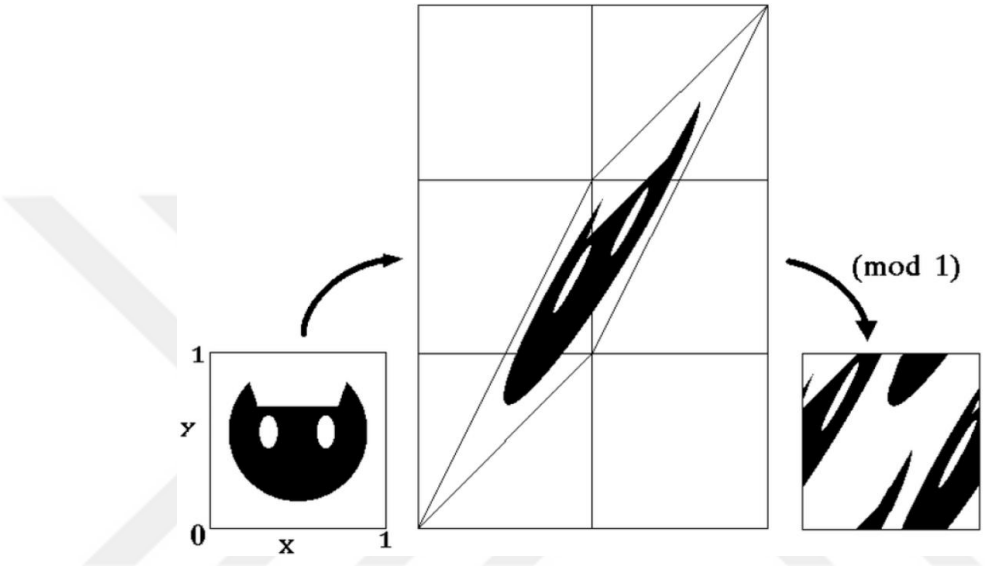
Herhangi bir dijital görüntü

$$Z = F(x, y),\ (x, y) \in R.$$

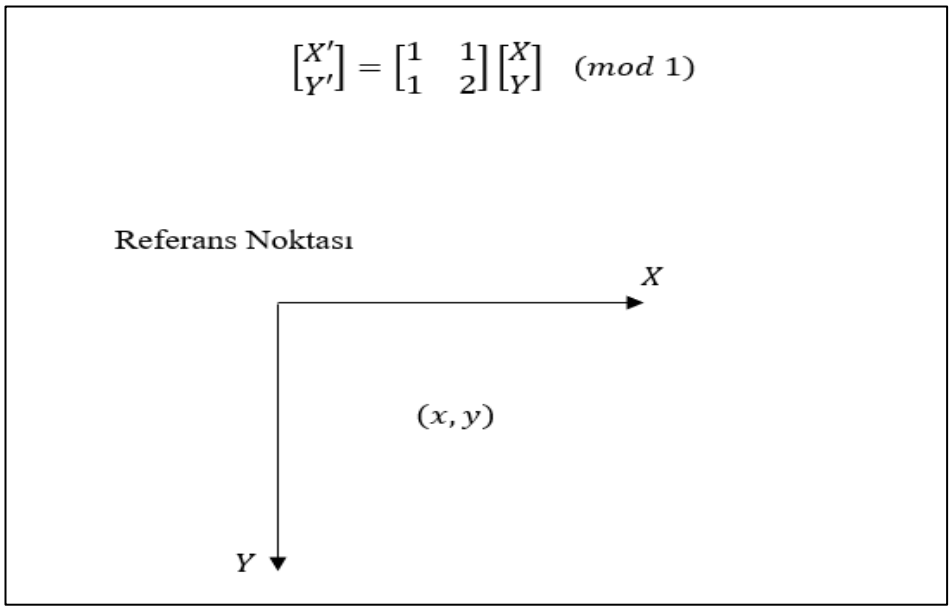
gibi iki değişkenli bir fonksiyon olarak kabul edilebilir. Böylece, ele alınan görüntü, iki boyutlu bir piksel matrisi olarak alınır ve görüntünün şifrelenmesi aslında matrisin şifrelenmesi haline indirgenmiş olur. Rus matematikçi Vladimir Igorevich Arnold, Arnold dönüşümünü önermiştir. Arnold dönüşümü, kaotik bir sistemin basit bir örneğidir. Bu dönüşüm, bir görüntünün piksellerini belirli bir kurala göre yeniden düzenleyerek, başlangıçtaki düzenli yapıyı bozup kaotik bir yapı oluşturur. Ancak, yeterince uzun bir süre boyunca (belirli bir periyot boyunca) uygulandığında, pikseller başlangıçtaki düzenli yapısına geri döner.

Dönüşümün periyodik olması, yani belirli bir sayıda yinelemeden sonra orijinal durumuna geri dönmesi, ergodik teori ile ilgilidir. Bu durum, dönüşümün tüm olası durumları eşit olasılıkla ziyaret etmesini ve belirli bir süre sonunda başlangıç durumuna dönmesini sağlar.

Vladimir Arnold bu dönüşümü ilk olarak kedi görüntülerinde yaptığı için, bu dönüşüm aynı zamanda Cat Mapping olarak da bilinir (Şekil 3.3). Cat Mapping dönüşümü, her pikselin yerini değiştirerek görüntüyü şifreleme amacına ulaşmaktadır [16].



Şekil 3.3 : Arnold Kedi Dönüşümü.



Şekil 3.4 : Görüntü Koordinatlarının Dağılımı.

NxN piksel değerinde bir görüntü için Arnold Dönüşümü (2) ‘de gösterilmektedir. $(x, y) \in \{1,2,\ldots,N-1\}$

$$\begin{bmatrix} X' \\ Y' \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} X \\ Y \end{bmatrix} \quad (mod\ N) \tag{2}$$

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 2 \end{bmatrix}^n \begin{bmatrix} x \\ y \end{bmatrix} \quad (mod\ N) \tag{3}$$

(3)’de görüldüğü gibi, matris işleminin n kez tekrarlanması durumunda görüntünün ilk halini elde etmek mümkündür Ancak, görüntü matrisinin boyutları değiştikçe yineleme sayısı da değişmektedir ve bu sayılar belirli bir düzene uymamaktadır.

Çizelge 3.1.’de NxN boyutlu bir matrisin dönüşüm uygulanarak kaç adımda orijinal haline döneceğini ifade edilmektedir.

Çizelge 3.1 : Arnold Dönüşüm Periyodu.

N	2	3	4	5	6	7	8	9	10	11
T_N	3	4	3	10	12	8	6	12	30	5
N	25	32	48	50	56	64	100	128	256	480
T_N	50	24	12	150	24	48	150	96	192	120

Bu dönüşümde, $(x, y, x', y') \in \{1,2,\ldots,N-1\}$’dır. N görüntü matrisinin boyutudur. (x, y) orijinal görüntünün piksel konumudur ve (x', y') karıştırılmış görüntünün piksel konumudur.

Dijital bir görüntü iki boyutlu bir matris olarak düşünülebilir. Arnold dönüşümü, orijinal noktanın (x, y) piksel değerini dönüştürülmüş noktanın (x', y') konumuna taşır ve görüntü netlikten bulanıklığa geçer. Arnold dönüşümünden sonra görüntünün piksel bitleri yeniden düzenlenir, böylece görüntü karıştırılır ve bu da görüntünün şifreleme etkisini gerçekleştirir.

Örnek 3.1 :

$5x5$ boyutunda bir görüntü için gerçekçi piksel değerleriyle Arnold dönüşümünün nasıl uygulandığını gösterelim.

Ele alınan görüntüye karşı gelen matris Şekil 3.5’de verilen matris olsun.

$$\begin{bmatrix} 12 & 45 & 78 & 123 & 90 \\ 34 & 67 & 89 & 145 & 134 \\ 56 & 89 & 123 & 167 & 156 \\ 78 & 111 & 145 & 189 & 178 \\ 90 & 134 & 167 & 200 & 189 \end{bmatrix}$$

Şekil 3.5 : 5 x 5 Boyutlu Görüntü Matrisi.

Öncelikle piksel konumlarını yeniden düzenleyelim:

$$\begin{bmatrix} X' \\ Y' \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} X \\ Y \end{bmatrix} \quad (mod\ 5) \qquad (4)$$

Her bir pikselin yeni konumunu bulmak için (4) ile verilen matris çarpımını kullanalım. Böylece, orijinal matris koordinatları ve yeniden düzenlenmiş karşı gelen koordinatlar Çizelge 3.2'de verildiği gibi elde edilir.

Çizelge 3.2 : Konum Koordinatlarının Değişim Çizelgesi

Orijinal Koordinat	(0,0)	(0,1)	(0,2)	(0,3)	(0,4)	(1,0)	(1,1)	(1,2)	(1,3)	(1,4)
Yeni Koordinat	(0,0)	(1,2)	(2,4)	(3,1)	(4,3)	(1,1)	(2,3)	(3,0)	(4,2)	(0,4)
Orijinal Koordinat	(2,0)	(2,1)	(2,2)	(2,3)	(2,4)	(3,0)	(3,1)	(3,2)	(3,3)	(3,4)
Yeni Koordinat	(2,2)	(3,4)	(4,1)	(0,3)	(1,0)	(3,3)	(4,0)	(0,2)	(1,4)	(2,1)
Orijinal Koordinat	(3,0)	(3,1)	(3,2)	(3,3)	(3,4)	(4,0)	(4,1)	(4,2)	(4,3)	(4,4)
Yeni Koordinat	(3,3)	(4,0)	(0,2)	(1,4)	(2,1)	(4,4)	(0,1)	(1,3)	(2,0)	(3,2)

Böylece Arnold dönüşümü uygulanan yeni görüntü matrisi elde edilmiş olur (Şekil 3.6).

$$\begin{bmatrix} 12 & 134 & 78 & 145 & 90 \\ 200 & 34 & 156 & 67 & 89 \\ 134 & 189 & 12 & 45 & 178 \\ 167 & 145 & 89 & 123 & 111 \\ 90 & 167 & 89 & 145 & 123 \end{bmatrix}$$

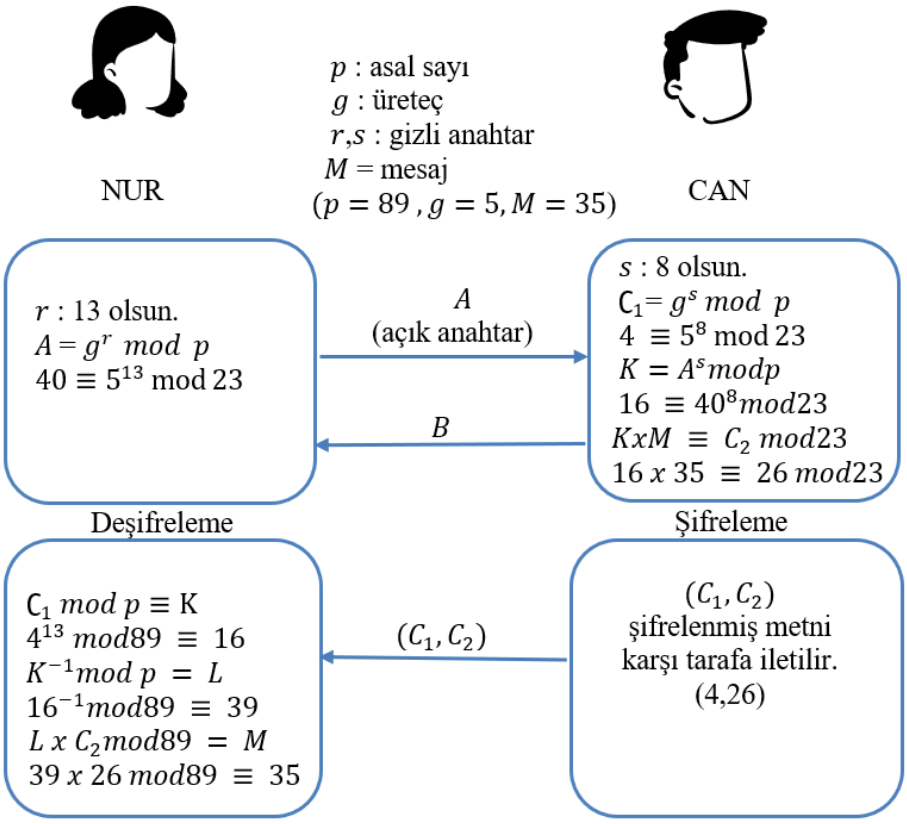
Şekil 3.6 : 5 x 5 Boyutlu Arnold Dönüşümlü Görüntü Matrisi.

3.1.2 ElGamal Şifreleme

Arnold dönüşümü döngüsel olduğundan ve anahtar alanı çok küçük ve kapsamlı saldırılara karşı savunmasız olduğundan, Arnold dönüşümünü tek başına kullanmak yeterli değildir. Bu nedenle, Arnold'un anahtar alanını genişletmek ve algoritmanın genel güvenliğini artırmak için ElGamal algoritması kullanılmaktadır. Bu çalışma, Arnold dönüşümü ile üst üste gelen geleneksel kriptografik yöntemleri kullanarak, Arnold şifrelemesinin üzerine ayrık logaritma tabanlı ElGamal Şifreleme Algoritması'nı kullanarak ikincil şifreleme sağlar ve biyometrik görüntülerin gizliliğini daha iyi korur. Şifreleme ve Şifre Çözme bölümlerini içeren Şekil 3.1'i takip ederek, Arnold Dönüşümünün sonucunu şifrelemek için ElGamal Şifreleme Algoritması'nı kullanıyoruz.

ElGamal algoritması, sonlu bir alan üzerinde ayrık logaritma problemine dayanan bir Açık Anahtar Şifreleme Sistemi olup, 1985 yılında ElGamal tarafından önerilmiştir [11]. Bu, verinin gizliliğini ve bütünlüğünü sağlayan güvenli bir şifreleme yöntemidir. Ayrık logaritmaların hesaplanmasının zorluğuna dayanan bu yöntem, kaba kuvvet saldırılarına karşı dayanıklıdır. Açık Anahtar Şifreleme Sistemleri, Özel Anahtar Şifreleme Sistemlerine (örn. DES, AES, vb.) göre bazı avantajlara sahiptir; örneğin, şifreleme ve şifre çözme işlemlerini birbirinden ayırabilir, böylece birden fazla kullanıcı tarafından şifrelenen mesajlar yalnızca bir kullanıcı tarafından çözülebilir veya bir kullanıcı tarafından şifrelenen mesajlar birden fazla kullanıcı tarafından çözülebilir. Bu nedenle, Açık Anahtar Şifreleme Sistemi sadece şifreleme için değil, aynı zamanda imza doğrulama için de uygundur. Bu model, Açık Anahtar kriptografik algoritmasının seçilmesini daha uygun hale getiren bir imza doğrulama bileşeni içermektedir.

Örnek 3.2 : Şekil 3.6 gönderici ve alıcı arasındaki anahtar değişimini ve elgamal ile M mesajının şifreleme ve deşifreleme aşamalarını göstermektedir.



Şekil 3.7 : ElGamal Şifreleme Algoritması.

3.2 Metodoloji

Önerilen çalışmada, öncelikle, (3) ile tanımlanan 2 boyutlu Arnold dönüşümünde kullanılan anahtar matris yerine rastgele seçilen a , b sayıları ve t tur sayısı ile

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} ab+1 & a \\ b & 1 \end{bmatrix}^t \begin{bmatrix} x \\ y \end{bmatrix} \ (mod \ N) \qquad (3)$$

dönüşümü hedef dönüşüm formülü olarak belirlenmiştir. Burada, rastgele sayı seçiminde *Phyton Random* kütüphanesi kullanılmıştır.

Ele alınan biometrik verinin piksel piksel alınmasıyla oluşturulan kare matrisin her koordinat konumunun (3) denklemi ile değiştirildikten sonra tüm piksel değerleri Elgamal algoritması ile tekrar şifrelenir. Burada, gizli anahtar ve açık anahtar seçimlerinde *Crypto.Util* kütüphanesi kullanılmıştır. Şifreleme adımları için Pyhton kodu yazılmıştır. Bu fonksiyondaki algoritma yapısı Şekil 3.7'da ifade edildiği gibi mesaj(M) yerine gri tonlamalı piksel değeri kullanılarak yapılmıştır.

Deşifreleme adımları için Elgamal algoritması ile şifrelenmiş görüntü matris piksel değerleri python deşifreleme fonksiyonu ile çözülür. Çözülen piksel değerlerinin yer aldığı matrise ters Arnold dönüşümü uygulanır.

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 1 & -a \\ -b & ab+1 \end{bmatrix}^{t} \begin{bmatrix} x' \\ y' \end{bmatrix} \pmod{N} \quad (4)$$

3.2.1 Arnold Dönüşümü Uygulaması

Arnold dönüşüm uygulamasında kullanılan algoritmanın adımları aşağıdaki gibidir:

1. Parametre Seçimi:

a, b ve dönüşüm tur sayısı rounds gibi parametreler rastgele seçilir. Bu parametreler dönüşümün kaotikliğini ve güvenliğini sağlar.

2. Görüntü Yükleme ve Hazırlık:

Şifreleme veya deşifreleme işlemi için uygun formatta bir görüntü yüklenir. Görüntü genellikle gri tonlamaya dönüştürülür ve pikselleri matris olarak temsil edilir.

3. *Arnold Dönüşüm Haritalarının Oluşturulması:*

Mapping ve inverseMapping adında iki fonksiyon tanımlanır. Mapping fonksiyonu, piksellerin yeni pozisyonlarını belirlemek için kullanılır. InverseMapping fonksiyonu ise dönüşümün tersini yapmak için pikselleri orijinal pozisyonlarına geri taşır.

4. *Şifreleme veya Deşifreleme Adımları:*

Şifreleme:

applyTransformTo fonksiyonu kullanılarak, belirli sayıda Arnold dönüşümü uygulanır. Dönüşüm sırasında, pikseller mapping fonksiyonunun belirlediği yeni pozisyonlara taşınır. Sonuç olarak, görüntü tamamen karıştırılarak şifrelenmiş hal elde edilir.

Deşifreleme:

applyInverseTransformTo fonksiyonu kullanılarak, belirli sayıda ters Arnold dönüşümü uygulanır. Her turda, pikseller inverseMapping fonksiyonunun belirlediği orijinal pozisyonlarına geri taşınır. Bu işlem sonucunda, orijinal görüntü elde edilir.

Sonuçların İşlenmesi ve Kaydedilmesi:

Şifreleme veya deşifreleme işlemi sonrasında elde edilen görüntüler uygun formatta işlenir ve kaydedilir.

Arnold için uygulama Python'da yazılmış olup, görüntü işleme için ***PIL*** (Python Imaging Library) ve sayısal hesaplamalar için ***NUMPY***, rastgele sayı üretmek için ***RANDOM*** kütüphaneleri kullanılmıştır (Şekil 3.8).

```
import numpy as np
from PIL import Image
import random
```

Şekil 3.8 : Kullanılan Python Kütüphaneleri.

Arnold dönüşümü için gerekli olan **a, b** ve dönüşüm tur sayısı **rounds** parametreleri belirlenir (Şekil 3.9).

```
def __init__(self, a:int, b:int, turSayısı:int):
    self.__a = a
    self.__b = b
    self.__turSayısı = turSayısı
```

Şekil 3.9 : a, b, Tur Sayısı parametre fonksiyonu.

Dönüşümün uygulanacağı görüntü belirtilen dosya yolundan yüklenir ve gri tonlamaya dönüştürülür (Şekil 3.10).

```
lena = np.array(Image.open(görüntü_yolu).convert("L"))
```

Şekil 3.10 : Renkli fotoğrafın gri tonlarına dönüşmesi.

Görüntü piksellerinin yeni pozisyonlarını belirlemek için haritalama ve tersHaritalama fonksiyonları tanımlanır (Şekil 3.11).

```
def haritalama(self, s:np.shape):
    x, y = np.meshgrid(range(s[0]), range(s[1]), indexing="ij")
    xharita = (self.__a*self.__b*x + x + self.__a*y) % s[0]
    yharita = (self.__b*x + y) % s[0]
    return xharita, yharita
```

Şekil 3.11 : Haritalama Fonksiyonu.

Dönüşüm fonksiyonu kullanılarak, görüntüye belirli sayıda Arnold dönüşümü uygulanır. Dönüştürülmüş görüntü kaydedilir (Şekil 3.12).

```
def Dönüşüm(self, görüntü:np.ndarray):
    xm, ym = self.haritalama(görüntü.shape)
    img = görüntü
    for r in range(self.__turSayısı):
        img = img[xm, ym]
    return img
```

Şekil 3.12 : Dönüşüm Fonksiyonu.

```
# Arnold Dönüşümü
arnold = Arnold(a, b, turSayısı)
karıştırılanGörüntü = arnold.Dönüşüm(lena)
# Karıştırılan görüntüyü kaydet
im = Image.fromarray(karıştırılanGörüntü).convert("L")
im.save("karıştırılanGörüntü.jpg")
```

Şekil 3.13 : Arnold Dönüşümü ve Dönüştürülen Görüntünün Kaydedilmesi.

Ters Haritalama fonksiyonu, pikselleri orijinal pozisyonlarına geri döndüren haritayı oluşturur. Bu harita, şifreleme sırasında yapılan dönüşümün tersini yapar (Şekil 3.14).

```
def tersHaritalama(self, s:np.shape):
    x, y = np.meshgrid(range(s[0]), range(s[1]), indexing="ij")
    xharita = (x - self.__a*y) % s[0]
    yharita = (-self.__b*x + self.__a*self.__b*y + y) % s[0]
    return xharita, yharita
```

Şekil 3.14 : Ters Haritalama Fonksiyonu.

Ters Dönüşüm fonksiyonu, belirli sayıda (tur Sayısı) ters Arnold dönüşümü uygular. Her turda, pikseller Ters Haritalama fonksiyonunun belirlediği orijinal pozisyonlarına geri taşınır. Bu işlem, görüntünün orijinal haline dönmesini sağlar (Şekil 3.15).

```
def TersDönüşüm(self, görüntü:np.ndarray):
    xm, ym = self.tersHaritalama(görüntü.shape)
    img = görüntü
    for r in range(self.__turSayısı):
        img = img[xm, ym]
    return img
```

Şekil 3.15 : Haritalama ve tersHaritalama Fonksiyonu.

Yapılan bu kodlamalar sonucunda elde edilen uygulama çalıştırıldığında elde edilen görüntüler aşağıda örneklerle verilmiştir.

3.2.2 ElGamal Şifreleme Uygulaması

Elgamal şifrelemesi Python'da yazılmış olup,

Numpy (np): numerik hesaplamalar ve dizi işlemleri için kullanılır.

PIL (Pillow): Görüntü işlemleri ve görüntü dosyalarıyla çalışma işlemleri için kullanılır.

random: Rastgele sayılar üretmek için kullanılır.

Crypto.Util.number: Büyük asal sayılar üretmek ve bazı sayısal işlemleri gerçekleştirmek için kullanılır.

Sys: Komut satırı argümanlarını işlemek ve diğer sistem düzeyinde işlemler için kullanılır (Şekil 3.16).

```
import numpy as np
from PIL import Image
import random
from Crypto.Util import number
import sys
```

Şekil 3.16 : Kullanılan Python Kütüphaneleri

Elgamal_parametreleri_oluştur() fonksiyonu 1024 bitlik rastgele p sayısı üretir. p sayısını kullanarak q, g elde edilir. Random x ve y tamsayıları üretilir (Şekil 3.17).

```
def elgamal_parametreleri_oluştur(bits=1024):
    p = number.getPrime(bits)
    q = (p - 1) // 2
    g = number.getPrime(bits-1) % p
    x = random.randint(1, q)
    y = pow(g, x, p)
    return (p, q, g, y), (p, q, g, y, x)
```

Şekil 3.17 : Elgamal Parametrelerini Oluşturan Fonksiyon.

Şifrele() fonksiyonu, piksel değerini ElGamal şifreleme algoritmasını kullanarak şifreler. r rastgele bir sayı seçilir. c_1 ve c_2 şifreli metin bileşenleri hesaplanır (Şekil 3.18).

```
def şifrele(mesaj, pub_key):
    m, q, g, y = pub_key
    r = random.randint(1, q)
    c1 = pow(g, r, m)
    c2 = (mesaj * pow(y, r, m)) % m
    return c1, c2
```

Şekil 3.18 : Şifreleme İşlemini Yapan Fonksiyon.

Deşifre() fonksiyonu piksel değerini ElGamal şifreleme algoritmasını kullanarak çözer.s ve s_tersi hesaplanır. Orijinal piksel değeri elde edilir (Şekil 3.19).

```
def deşifre(şifre, priv_key):
    c1, c2 = şifre
    m, q, g, y, x = priv_key
    s = pow(c1, x, m)
    s_tersi = mod_tersi(s, m)
    return (c2 * s_tersi) % m
```

Şekil 3.19 : Deşifreleme İşlemini Yapan Fonksiyon.

EK A ve EK B 'de rastgele seçilen p, g, y ve bu sayılarla hesaplanan $C1, C2$ değerleri orijinal pikselleri şifrelemek için kullanılarak şifrelenmiş değerleri verilmektedir.

Örnek 3.3 :

Arnold Parametreleri ; a = 21, b = 50 ve tur sayısı = 97 olsun. Şekil 3.20 ile verilen 640 x 640 piksel boyutlu görüntüyü önerilen algoritma ile şifreleyelim.

Şekil 3.20 : Orijinal Görüntü.

1. Arnold Dönüşüm Aşaması:

Verilen renkli görüntüye PIL kütüphanesi ile gri tonlama yapılır (Şekil 3.21). Her piksel değeri (0,255) arası bir değer alır gri tonlama yapılan değerlere Arnold dönüşümü uygulanır ve karıştırılmış görüntü elde edilir (Şekil 3.22).

Her piksel, bir renk veya gri ton değeri içerir. Renkli bir görüntü genellikle üç kanal (kırmızı, yeşil, mavi - RGB) içerirken, gri tonlamalı bir görüntü sadece parlaklık değeri içerir. Görüntüyü gri tonlamalı hale getirmek, her pikselin parlaklık değerini 0(beyaz) ile 255(siyah) arasında bir değere dönüştürmeyi içerir. Bu dönüşüm, renkli bir

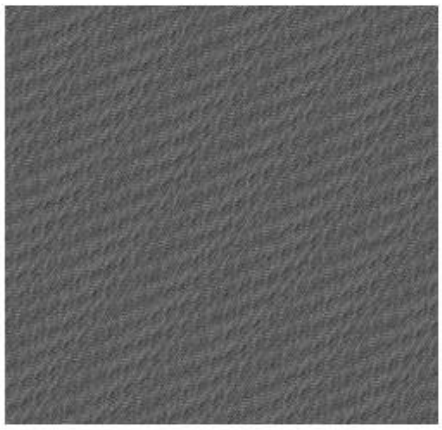
görüntünün üç kanalındaki değerlerin belirli bir ağırlıkla birleştirilmesiyle yapılır. Gri tonlama dönüşümünün matematiksel ifadesi genellikle şu şekildedir:

$$Y = 0.299\ x\ R + 0.587\ x\ G + 0.114\ x\ B$$

Burada R, G, ve B sırasıyla kırmızı, yeşil ve mavi kanalların değerleridir. Bu ağırlıklar, insan gözünün farklı renk kanallarına duyarlılığını yansıtır.

Gri tonlama dönüşümünde kullanılan bu katsayılar $(0.299, 0.587, 0.114)$, BT.601 standardı olarak bilinen bir televizyon yayın standardına dayanır. Bu standart, 1950'lerde ve 1960'larda televizyon mühendisleri tarafından geliştirilen ve televizyon yayınlarında kullanılan renk uzayının bir parçasıdır. BT.601 standardı, insan gözünün renk duyarlılığını dikkate alarak bu katsayıları belirlemiştir [18].

Şekil 3.21 : Gri Tonlamalı Görüntü.



Şekil 3.22 : Arnold Dönüşümlü Görüntü.

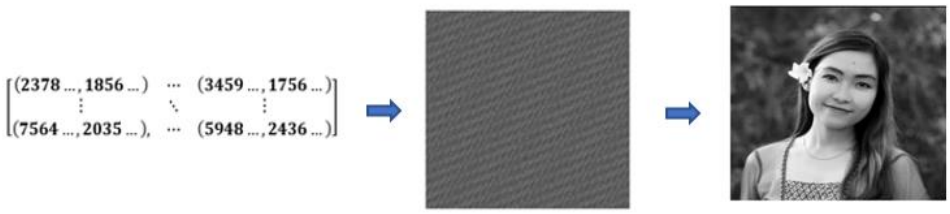
2. ElGamal Şifreleme Aşaması:

Şekil 3.21 görüntüsündeki gri tonlama işleminden sonra Şekil 3.22'de verilen Arnold dönüşümü ile elde edilen görüntünün her piksel değeri ElGamal ile şifrelenir. Şekil 3.22 ile verilen görüntünün ElGamal şifrelenmesi ile elde edilen (C_1, C_2) çifti Şekil 3.18 ile gösterilen fonksiyon ile elde edilmiştir. Her piksel değeri için toplam 409.600 (640 x 640) tane (C_1, C_2) çifti elde edilir.

Tersine olarak (C_1, C_2) çiftleri , Şekil 3.19 ile verilen deşifreleme fonksiyonu kullanılarak orjinal piksel değeri elde edilmiştir. Aşağıda M ile gösterilen orijinal piksel değeridir. C_2 ve C_1 'in tersinin çarpılmasıyla elde edilmektedir.

$$M = {C_1}^{-1}.C_2$$

409.600 tane şifreli piksel için deşifreleme işleminden sonra elde edilen M değerleri Şekil 3.22 ile verilen karıştırılmış görüntüyü verir. Daha sonra, Şekil 3.14 ve Şekil 3.15 de verilen metotlar kullanılarak Şekil 3.21 ile verilen gri tonlamalı görüntü elde edilir (Şekil 3.23).



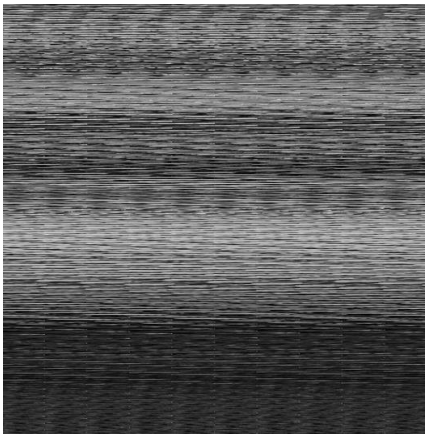
Şekil 3.23 : Şifrelenmiş Görüntünün Deşifre Edilmesi**.**

Örnek 3.4:

Şekil 3.24'de $640\ x\ 640$ piksel parametreleri ile gri tonlama uygulandığında Şekil 3.25 elde edilir. Bu görüntüye $a = 100$, $b = 62$, tur sayısı $= 96$ parametreleriyle Arnold dönüşümü uygulandığında Şekil 3.26 elde edilir.

Şekil 3.24 : Orijinal Görüntü.

Şekil 3.25 : Gri Tonlamalı Görüntü.

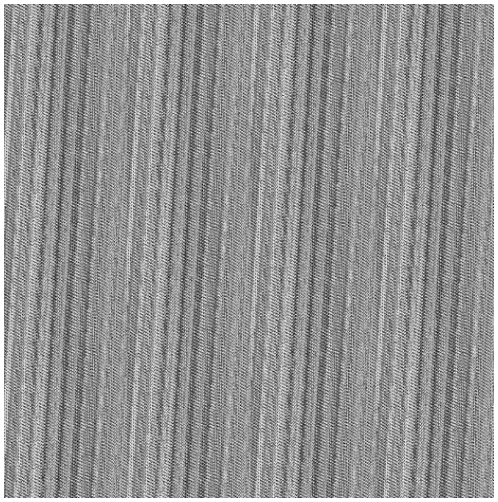


Şekil 3.26 : Arnold Dönüşümlü Görüntü.

Gri tonlamalı görüntüye ElGamal yönteminin uygulanması ile 409.600 tane (C_1, C_2) çifti oluşur. Tersine işlem olarak (C_1, C_2) çiftlerine deşifreleme fonksiyonu uygulanarak Şekil 3.26 elde edilir. Daha sonra Şekil 3.14 ve Şekil 3.15 'de verilen hesaplamalar kullanıldığında Şekil 3.25'de verilen gri tonlamalı görüntü meydana gelir.

Örnek 3.5:

Şekil 3.27 'de verilen Arnold dönüşümlü görüntüye $a = 79, b = 50, rounds = 4$ parametreleriyle Ters Arnold dönüşümü uygulandığında Şekil 3.28'de verilen gri tonlamalı görüntü elde edilmektedir.



Şekil 3.27 : Arnold Dönüşümlü Görüntü.

Şekil 3.28 : Gri Tonlamalı Görüntü.

Şekil 3.27'de verilen Arnold dönüşümlü görüntüye ElGamal yönteminin uygulanması ile elde edilen (C_1, C_2) çifti tersine işlem olarak Şekil 3.14 ve Şekil 3.15 'de verilen hesaplamalar kullanılarak Şekil 3.28'de verilen gri tonlamalı görüntü elde edilmiştir.

4. SONUÇ VE ÖNERİLER

Bu çalışmada, dijital görüntülerin güvenliğini artırmak amacıyla çeşitli şifreleme ve şifre çözme algoritmalarının uygulanabilirliği araştırılmıştır. Çalışmada, Kaggle'dan [23] edinilen yaklaşık 100 insan fotoğrafı kullanılarak bu algoritmalar test edilmiştir. Görüntüler, JPEG veya JPG formatlarında olup, her biri şifreleme ve şifre çözme işlemlerine tabi tutulmuştur.

Kodlama süreci Python dilinde gerçekleştirilmiştir. Python'un tercih edilmesinin temel nedenleri, dilin güçlü kütüphaneleri ve geniş topluluk desteği sayesinde hızlı geliştirme ve genişletilebilirlik sağlamasıdır. Ayrıca, Python'un bilimsel hesaplamalar ve veri analizi için sunduğu zengin araçlar, algoritmaların etkili bir şekilde uygulanmasını ve test edilmesini mümkün kılmıştır.

Algoritmaların çalıştırıldığı bilgisayarın teknik özellikleri ise şu şekildedir: Cihaz adı Monster, işlemci Intel(R) Core(TM) i7-10510U CPU @ 1.80GHz 2.30 GHz ve RAM kapasitesi 16,0 GB'dir. Sistem, 64 bit işletim sistemi ve x64 tabanlı bir işlemciye sahiptir. Bu donanım konfigürasyonu, şifreleme ve şifre çözme işlemlerinin etkin ve verimli bir şekilde gerçekleştirilmesini sağlamıştır.

Yapılan kontroller sonucunda, şifreleme ve şifre çözme işlemlerinin görüntü üzerinde herhangi bir bozulma veya piksel değerlerinde değişiklik oluşturmadığı belirlenmiştir. Bu bulgu, kullanıcıların görüntülerini bozulma ve değişiklik yaşamadan güvenli bir şekilde depolayabilmelerini mümkün kılmaktadır.

Sonuç olarak, bu çalışma simetrik, asimetrik ve hibrit görüntü şifreleme tekniklerinin kapsamlı bir incelemesini sağlamıştır. Bu temel üzerine odaklanılan konu, biyometrik görüntülerin 2D Arnold dönüşümü kullanılarak karıştırılması ve ardından bunların ElGamal algoritmasıyla şifrelenmesi sorunuydu. Önerilen yaklaşımın temel özelliği, Arnold dönüşümünden sonra şifrelenmiş biyometrik görüntünün şifresinin çözülmesinin, ayrık logaritmaların hesaplama zorluğuna dayanmasıdır. Sunulan örnekler aracılığıyla hibrit görüntü kriptolojisi yaklaşımının biyometrik görüntülerin saklanması, iletilmesi ve doğrulanması aşamalarında olası saldırılara karşı hem etkili hem de güvenli olduğu ortaya konmuştur.

KAYNAKLAR

[1] **Yen, J.-C., & Guo, J.-I.** (2000). A new image encryption algorithm and its VLSI architecture**.** *Proceedings of the IEEE International Symposium on Circuits and Systems*, 2000, vol. 4, pp. 975-978. Department of Electronics Engineering, National Lien-Ho Institute of Technology, Miaoli, Taiwan, Republic of China.

[2] **Chang, C.-C., Hwang, M.-S., ve Chen, T.-S.** (2001). A new encryption algorithm for image cryptosystems. Journal of Systems and Software, 58(2), 83-91.

[3] **Shujun, L., ve Xuan, Z.** (2002). On the security of an image encryption method.

[4] **Hariyanto, E., & Rahim, R.** (2017). Arnold's Cat Map Algorithm in Digital Image Encryption**.** *Faculty of Computer Science, Universitas Pembangunan Panca Budi*, Medan, Sumatera Utara, Indonesia.

[5] **Sharma, P., Godara, M., Singh, R., Tech, S. M., ve Sabo, T.** (2012). Digital Image encryption techniques: A Review. International Journal of Computing ve Business Research, 2229-6166. http://researchmanuscripts.com/isociety20 12/46.pd

[6] **Ghoradkar, S., & Shinde, A.** (2015). Review on Image Encryption and Decryption using AES Algorithm**.** *International Journal of Computer Applications (0975 – 8887), National Conference on Emerging Trends in Advanced Communication Technologies (NCETACT-2015),* 11. D.Y. Patil College of Engineering, Akurdi, Pune.

[7] **Anandakumar, S**. (2015). **Image Cryptography Using RSA Algorithm in Network Security**. *International Journal of Computer Science and Engineering Technology (IJCSET)*, 5(9), 326-330. Bharathidasan University, Tiruchirappalli.

[8] **Liu, X., Xiao, D., ve Xiang, Y.** (2018). Quantum image encryption using intra and inter bit permutation based on logistic map. IEEE Access, 7, 6937-6946.

[9] **Satish A., Erapu Vara Prasad, Tejasvi R., Swapna P., & Vijayarajan R.** (2019). Image Scrambling through Two Level Arnold Transform. Presented at the Alliance International Conference on Artificial Intelligence and Machine Learning, April (AICAAM), p. 329.

[10] **Hu, W.-W., Zhou, R.-G., Luo, J., Jiang, S.-X., & Luo, G.-F.** (2020). Quantum image encryption algorithm based on Arnold scrambling and wavelet transforms. *Quantum Information Processing*, *19*(3), 82.

[11] **ElGamal, T.** (1985). A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms. *IEEE Transactions on Information Theory*, *31*(4), 469-472.

[12] **Aktan, E.** (2018). Büyük Veri: Uygulama Alanları, Analitiği ve Güvenlik Boyutu. Bilgi Yönetimi Dergisi, 1-2

[13] **Hoşcoşkun, R. E.** (2020). Homomorfik şifreleme yöntemi üzerine bir inceleme. (Yüksek Lisans Tezi). Trakya Üniversitesi Fen Bilimleri Enstitüsü, Bilgisayar Mühendisliği Anabilim Dalı, Edirne.

[14] **Çelik, B., Cangül, İ. N., Çelik, N., Bizim, O., Öztürk, M.** (2010*). Temel Matematik* (5. Baskı). Bursa: Dora

[15] **Akkaş, S., Hacısalihoğlu, H. H., Özel, Z., Sabuncuoğlu, A.** (1998). *Soyut Matematik* (3. Baskı). Ankara: Gazi Üniversitesi.

[16] **Demir, M. K.** (2014). Esnek Kümeler ve Esnek Cebirsel Yapılar (Master's thesis, Fen Bilimleri Enstitüsü).

[17] **Asar, A. O., Arıkan, A., Arıkan A**. (2012). *Cebir (2. Baskı). Ankara: Sincan Matbaası*.

[18] **Rivest, R. L., Shamir, A., & Adleman, L.** (1978). A method for obtaining digital signatures and public-key cryptosystems. Communications of the ACM, 21(2), 120-126.

[19] **Florian Bourse, Michele Minelli, Matthias Minihold, and Pascal Paillier.** (2018). Fast Homomorphic Evaluation of Deep Discretized Neural Networks. In Advances in Cryptology - CRYPTO 2018 - 38th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2018, Proceedings, Part III. 483-512

[20] **Milanov, E.** (2009). *The RSA Algorithm.* Retrieved: 3 Haziran 2009, from https://sites.math.washington.edu/~morrow/336_09/papers/Yevgeny.pdf

[21] **Saratov Group.** (2012). Classic Arnold's Cat and Other Maps on a Torus [Illustration]. Saratov, Russia: Saratov Group. Available online at http://www.sgtnd.narod.ru/science/cat/classic/eng/classic.htm

[22] **Fischer, W.** (2008). *Digital Video and Audio Broadcasting Technology: A Practical Engineering Guide* (2nd ed., pp. 81–85). Springer.

[23] **Gupta, A.** (2020). Human Faces. Kaggle. Retrieved July 19, 2024 from

https://www.kaggle.com/datasets/ashwingupta3012/human-faces

EKLER :

EK A : Rastgele seçilmiş 10 tane piksel değeri için oluşan $p, g, y, (C1, C2)$ değerleri.

$\boldsymbol{p}$	10315991261750896951063389047999399253240915000672156190776276513762550403636019311510473664526821623299179021559932669036360918729617379303644090432963733
$\boldsymbol{g}$	51222485243415053689266375384685370293671145792944818048182141018840928196039404506760697011201302106668658820095265990750824095696619547214366403241245 41,
$\boldsymbol{y}$	47167662944446359901921299345822380054841844729366276218595022991815799703595064782377110571124032928131704778366189249687934126730897065100964025821429 63
$(\boldsymbol{C_1}, \boldsymbol{C_2})^1$	(25459982752896709426555137084623708470234564045076712953534664652147527622930392254117348129301148389435528705591211659682689254604549167102465537763169 34, 38951968942005401519595723480335218095090496212032076269220216004420857307208659258923811811286930537222102412776746572070865482668461486206326145549325 87)
$(\boldsymbol{C_1}, \boldsymbol{C_2})^2$	(45234548893924135285675829387788104772695318237318924901365315706734302446542888716515013010226786827686068535835916937978757620092942527397538011500824 64, 46192375759586872391437242751221755064008453937441941147147625462884253004512087355893824748760448682399414086618117541829045749531661131226363426223260 01)
$(\boldsymbol{C_1}, \boldsymbol{C_2})^3$	(45234548893924135285675829387788104772695318237318924901365315706734302446542888716515013010226786827686068535835916937978757620092942527397538011500824 64, 46192375759586872391437242751221755064008453937441941147147625462884253004512087355893824748760448682399414086618117541829045749531661131226363426223260 01)
$(\boldsymbol{C_1}, \boldsymbol{C_2})^4$	(45234548893924135285675829387788104772695318237318924901365315706734302446542888716515013010226786827686068535835916937978757620092942527397538011500824 64, 46192375759586872391437242751221755064008453937441941147147625462884253004512087355893824748760448682399414086618117541829045749531661131226363426223260 01)

(devamı)

$(C_1, C_2)^5$	(576418327974517154134786261554852399360348737955378996408093315398469584346213322550419832598159155388976771057396760303639535781815378598016257984036776239675030363953578181537859801625798403677624, 396873297506301419325689656318783151684711153081219754762526091965946150155921723445366052435170625821686761954477336636146225440172881197233726214953346323)
$(C_1, C_2)^6$	(260878123799439626161957516619743075500006370523236820457423167656402990834516289300875273157545538589988449623619099457702409544236109657355558759565188596745740746774159839980416508547238993557386634804697063475553735510571188129809458848622229322862938402)
$(C_1, C_2)^7$	(283110532135724022633459894289779233801053571090946801446622717687215769168960439227880079478089088838412412986227934807325597641466194869845606001983192849274384857587245193473195011068621669738433978648531133682537792319887595157285831861593328736435075437747462921899813827365653377476292175795986433774602)
$(C_1, C_2)^8$	(882975070518569286079357047440759202364206688327552643064026658784533840841523947446719003153813017653236975823068040343780934082402668348472581058554971849764475898915963970842727798927485579427797419722790695457865174126771446826338135596187010166320766657513504757796192670670560966846173798233330165567772)
$(C_1, C_2)^9$	(834219800091640550675208413650439081545940919850600232377833369184947178388648629639077181037206746209535138141245453620027807622279920563165139371730985, 414704476743427099901523550659165085931336877770775426180125598355912626960825075091028098603525963486918974743697562088089757986840942840099650594631395 7)
$(C_1, C_2)^{10}$	(255123822197247508296424163095617966399129468214597176030168138788909164976081160068853269840828369315799452624154295129237461833502578042362446341142392 5, 536402592232306032774894374890078272083361636279619957404187889429055605019851445193027414258572537952401618781589895860204357635438423005914833295365415 2)

EK B : Arnold Dönüşümlü 9 Görüntü Piksel değerilerinin Elgamal ile Şifrelenmiş Değerleri.

Orijinal Piksel	**47**
Şifreli Piksel	(819054824820102159332327439939585375609552891561079 810650500652984344910944831919561277791524168106157 9 656105349454828255785086489969327392819295835941982, 102984908139002453171762939169001848859059224876109 9 013765263809805818176799139647294756435667682604484 1 481645586437847140662095820268855044622267939921211)
Orijinal Piksel	**65**
Şifreli Piksel	(417894292269692788946887984094745896591522050360461 385664226508841714414219283208022531114751599424333 9 527534489458149371286749680372752014511888131040746, 169005130059025181030698850848505088925305976694229 3 821649733494742826710451948305312776370361239733018 2 407024505805265772275027768893646446687522734614 98)
Orijinal Piksel	**64**
Şifreli Piksel	(301901549282961090140175546204106242941064381904942 292499249138140624814496619149394959505090126281191 6 741956258482774168610041430143992186876938734418259, 522567614713191313084549120226773916691382423645908 2 714416662060478616448691105693730837735716333272256 4 421109368869752024734774580918375049025956281391 04)
Orijinal Piksel	**92**
Şifreli Piksel	(813070276711325941972094652326958778709765107374611 394558447168475058591700180809814479057966038511224 9 999382448890889441240748794169446430324791411268142, 931533039398251579736334466208449377814097266982235 9 636362268945972574085637087920059384121139639082877 5 132880632397715053981645698460348775675272544495 77)

(devamı)

Orijinal Piksel	**176**
Şifreli Piksel	(120456831922463290250264641675244808673115341515796 97964815488952715140834737791100955188965382704601108512689119039802105592408238793389152791888537877 54, 30267187265258746966953008408225452768855870886352761959771111107311963399637637571536050197696910975693599532944693332110396730865193156926655138557078 94)
Orijinal Piksel	**217**
Şifreli Piksel	(11815442082862547875480908367492663172885903807548080586055563036370487323410414513863861022237051361853191399016739747540098351121394834705953283443610591, 50507100779159160074401122601562103917689211059942036317456102476300498291179520714149625430892976197732335791954460483440035910598477611637845735134809 07)
Orijinal Piksel	**233**
Şifreli Piksel	(70966336212299904436840930068363949451630350057107813838045691210765687226966565387067198256797709913924658342610140199673016640756306069779088753746841 7, 58380778624557170765675623955021763159817858554744851544467808698573542554759365007760936351729114580375248797859518880504028678398067092052649690847082 01)
Orijinal Piksel	**51**
Şifreli Piksel	(88890529820780469034362906731209828158560807849532328303708308716916606060050455138978953799260321102947530298142369861202427935478097421260268926698532 42, 32668768339862395807320216126283724392553474445526251872753762361402833156726676877956153705313852273581155434834212744828579750571865377147554002804498 62)
Orijinal Piksel	**59**
Şifreli Piksel	(52579610215415101300289737788220439411607597513535990482322952966817869696702711580861700136949280467459275813280095136841535508973728340651844875280880 14, 89187166568179733622814413950588310570813193062230908745647203564868621041852485537701095637404742124029657056932958630900571262837093842000065889162887 42)

ÖZGEÇMİŞ

Adı SOYADI **:** Rabia ÜNLÜ

ÖĞRENİM DURUMU:

- **Lisans** **:** 2019, İstanbul Teknik Üniversitesi, Fen Edebiyat Fakültesi, Matematik Mühendisliği
- **Yüksek Lisans** **:** 2024, İstanbul Teknik Üniversitesi, Fen Edebiyat Fakültesi, Matematik Mühendisliği

MESLEKİ DENEYİM :

- 2019 yılında İstanbul Teknik Üniversitesi Matematik Mühendisliği lisans eğitimini tamamladı.
- 2019 yılından beri özel sektörde yazılımcı olarak çalışmaktadır.

SUNUM VE YAYINLAR :

- **Ünlü, R.** (2024). Biometric Image Encryption Based on Arnold Transform and Elgamal Algorithm. 8th International Conference on Mathematics, July 9-11, 2024, Istanbul, Turkey.
- **Ünlü, R.,** & Çivi, G**.** (2024). Biometric Image Encryption Based on Arnold Transform and Elgamal Algorithm. *Proceedings of the 8th International Conference on Mathematics (ICOM 2024)*, July 9-11, 2024, Istanbul, Turkey.