T.C.
ERCİYES ÜNİVERSİTESİ
FEN BİLİMLER ENSTİTÜSÜ
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI

KRİPTOLOJİ YÖNTEMLERİNİN KARŞILAŞTIRILMASI

Hazırlayan
Zainab Hashim OBAID

Danışman
Doç. Dr. Bahriye AKAY

Yüksek Lisans Tezi

Nisan 2016
KAYSERİ

T.C.
ERCİYES ÜNİVERSİTESİ
FEN BİLİMLER ENSTİTÜSÜ
BİLGİSAYAR MÜHENDİSLİĞİANABİLİM DALI

KRİPTOLOJİ YÖNTEMLERİNİN KARŞILAŞTIRILMASI

(Yüksek Lisans Tezi)

Hazırlayan
Zainab Hashim OBAID

Danışman
Doç. Dr. BahriyeAKAY

Bu çalışma; Erciyes Üniversitesi Bilimsel Araştırma Projeleri Birimi Tarafından FYL-16-6463 kodlu proje ile desteklenmiştir

Nisan 2016
KAYSERİ

BİLİMSEL ETİĞE UYGUNLUK

Bu çalışmadaki tüm bilgilerin, akademik ve etik kurallara uygun bir şekilde elde edildiğini beyan ederim. Aynı zamanda bu kural ve davranışların gerektirdiği gibi, bu çalışmanın özünde olmayan tüm materyal ve sonuçları tam olarak aktardığımı ve referans gösterdiğimi belirtirim.

Zainab Hashim OBAID

YÖNERGEYE UYGUNLUK

Kriptoloji Yöntemlerinin Karşılaştırılması adlı Yüksek Lisans tezi, Erciyes Üniversitesi Lisansüstü Tez Önerisi ve Tez Yazma Yönergesi'ne uygun olarak hazırlanmıştır.

Tezi Hazırlayan
Zainab Hashim OBAID

Danışman
Doç. Dr. Bahriye AKAY

Bilgisayar Mühendisliği ABD Başkanı
Prof. Dr. Derviş KARABOĞA

Doç. Dr. Bahriye AKAY danışmanlığında **Zainab Hashim OBAID** tarafından hazırlanan "**Kriptoloji Yöntemlerinin Karşılaştırılması**" adlı bu çalışma, jürimiz tarafından Erciyes Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği Anabilim Dalında **Yüksek Lisans** tezi olarak kabul edilmiştir.

18 / 04 / 2016
(Tez savunma sınav tarihi yazılacaktır.)

JÜRİ:

Danışman	: Doç. Dr. Bahriye AKAY	.....
Üye	: Prof. Dr. Derviş KARABOĞA	.....
Üye	:Yrd. Doç. Dr. Gürkan YÜKSEK	.....

ONAY:
Bu tezin kabulü Enstitü Yönetim Kurulunun 27/04/2016 tarih ve 2016/20-33... sayılı kararı ile onaylanmıştır.

..27../..04../..2016..
Prof. Dr. Mehmet AKKURT
Enstitü Müdürü

ÖNSÖZ / TEŞEKKÜR

Çalışmalarım boyunca farklı bakış açıları ve bilimsel katkılarıyla beni aydınlatan, yakın ilgi ve yardımlarını esirgemeyen ve bu günlere gelmemde en büyük katkı sahibi tez danışmanım Doç. Dr. Bahriye AKAY'ya teşekkürü bir borç bilirim.

Ders dönemi boyunca engin bilgileri ile bizi aydınlatan rahmetli Yar. Doç. Dr. Necla ÖZKAYA hocamın öğütleri tez yazım süresi boyunca hep beni motive ettiği için bir kez de huzurunuzda kendisine teşekkür ediyorum. Yanı sıra Bilgisayar Mühendisliği Bölüm Başkanı Sayın Prof. Dr. Derviş KARABOĞA'ya da şükranlarımı sunarım.

Bu tez çalışmasına maddi destek veren Erciyes Üniversitesi Bilimsel Araştırma Projeleri Birimi'ne (Proje No: FYL-2016-6463) teşekkür ederim.

Ayrıca; çalışmalarım süresince sabır göstererek beni daima destekleyen eşime en içten teşekkürlerimi sunarım.

Zainab Hashim OBAID
Kayseri, Nisan 2016

KRİPTOLOJİ YÖNTEMLERİNİN KARŞILAŞTIRILMASI

Zainab Hashim OBAID

Erciyes Üniversitesi, Fen Bilimleri Enstitüsü
Yüksek Lisans Tezi, Nisan2016
Danışman: Doç. Dr. Bahriye AKAY

ÖZET

Kriptoloji, bilgilerin şifrelenmesi ve şifrelenmiş bilgilerin çözümlenmesi için kullanılan metotlarlailgilenir. Kriptoloji Yöntemleri Klasik ve Modern Yöntemler olmak üzere ikiye ayrılmaktadır. Bu çalısmadaKlasik algoritmalardan yaygın olarak kullanılan Sezar, Vigenere, Vernam, Hill, Playfair algoritmalarının genel yapısı hakkında bilgi verilerek sözü geçen algoritmaların metinler ve Base64 tabanlıgörüntü üzerindeki analizleri yapılacaktır.

Metin yadagörüntülerşifreleme algoritmalarına girdi olarak verilerek anahtar ile şifrelenmiş metin elde edilir. Kullanılan algoritmalar çalışma zamanı yani görüntü şifrelendiği zaman ve şifreyi çözdüğü zaman, zaman karmaşıklığı, işlemci karmaşıklığı, hafıza karmaşıklığı, kurulacak sisteme uygunluk, esneklik, güven oranıkriterlerine göre kıyaslanmıştır.Kıyaslama yapılacak ve performans değerleri üzerinden algoritmaların avantaj ve dezavantajları belirlenmiştir.

Anahtar Kelimeler:Şifreleme algoritmaları, performans analizi, kriptoloji,bilgi güvenliği, görüntü şifremle,Klasik şifreleme algoritmaları.

COMPARISON OF CRYPTOGRAPHY METHOD

Zainab Hashim OBAID

Erciyes University, Graduate School of Natural and Applied Sciences
M.Sc. Thesis, April2016
Supervisor: Assoc. Dr. Bahriye AKAY

ABSTRACT

Cryptography is the science which deals with methods and theories of encryption and decryption. Cryptography is divided into two parts: modern and classic methods.

In this study, we use knowledge of common classical encryptionalgorithms including Caesar, Vigenere, Vernam, Hill, Playfair algorithms. The overall structure of these algorithms that have been explained and an analysis have been performed on the images encoded with Base64. Images and text data are passed to the algorithms and encrypted data is obtained using crypto key.

The performances of the algorithms are analyzedbased on: time consumption for encryptiondecryption the memory consumption, the processorconsumption, time complexity, space complexity, adaption to the system, flexibility and confidence rate.

Once the performance of the algorithms obtained, the advantages and disadvantages of these algorithms are evaluated.

Keywords:cryptography, cryptography algorithms, performance analysis, information security, imagecryptography, classical cryptography algorithms.

İÇİNDEKİLER

KRİPTOLOJİ YÖNTEMLERİNİN KARŞILAŞTIRILMASI

4. BÖLÜM TARTISMALAR-

SONUÇ ve ÖNERİ

TABLOLAR LİSTESİ

ŞEKİLLER LİSTESİ

GİRİŞ

Günümüzde gelişen teknoloji karşısında bilginin aktarılması gizliliği ve güvenirliliği önem kazanmıştır. Her geçen gün internetteki veri akışı da bu teknolojik gelişmelere paralel olarak artmaktadır.

İnternet üzerinden iletilen veri sadece metin değil, aynı zamanda ses, resim ve diğer çoklu ortam bilgilerini de kapsamaktadır. Günlük hayatımızda görüntü çok geniş bir kullanım alanına sahip olduğunda güvenliklerini sağlamak da çok büyük önem kazanmıştır. Bu nedenle günümüzde çok çeşitli şifreleme yöntemleri geliştirilmiştir. Geliştirilen hızlı bilgisayarlar ve güçlü sistemler sayesinde veriler, oldukça hızlı ve iyi şifrelenerek iletilebilmektedir.

Verilerin hızlı bir şekilde iletilmesi kadar, kullanılan şifreleme algoritmalarının güvenirliğinin test edilmesi de çok büyük önem taşımaktadır. Bu çalışmada, mevcut şifreleme algoritmalarının genel yapısı ve performansları hakkında bilgi verilecek ve sonuçlar analiz edilecektir. Çalışma kapsamında klasik şifreleme algoritmalarına yer verilmiştir.

Klasik şifreleme algoritmalarından Sezar, Vigenere, Vernam, Hill, Playfai, Performansları bakımından karşılaştırılması ve performans sıralamaları yapılmıştır.

Bu çalışmanın amacı görüntülerin algoritmalarla şifrelenmesi ve algoritmaların görüntüyü şifrelerken veya şifreyi çözerken harcadıkları zamanı, işlemci tüketimi ve hafızada ne kadar yer kapsadığını hesaplayarak kurulacak sisteme uygunluğunu, esnekliğini, güven oranını, incelemektir. Diğer bir deyişle görüntü şifreleme ve çözme algoritmaların performanslarını ölçmek ve kıyaslayarak avantaj ve dezavantajlarının tespit edilmesi amaçlanmaktadır.

1. BÖLÜM

GENEL BİLGİLER ve LİTERATÜR ÇALIŞMASI

1.1. Kriptoloji

Şifre bilimi olan kriptoloji verilerin şifrelenmesi ve şifrelenmiş verilerin çözümlenmesi için tekniklerleilgilenir. Yunanca saklanmış gizli demek olan kriptos ve söz, bilgi, bilim gibi birçok anlamı olan logos kelimelerinden türetilmiştir [1].

Günümüzün rekabetçi sisteminde bilgi daha da önemli hale gelmektedir. Artan veri miktarı veya saklama ve iletmeyizorlaştırmakta ve yeni tekniklere ihtiyaç duyulmaktadır. Uydu ve internet üzerinden yapılan ileti aktarımında araya sızama ihtimali giderek artmaktadır. Gizliliği kritik olan özel, ticarî, politik ve askerî verilerin elektronik ortamlarda transferi ile kriptografinin önemi daha da artmıştır. Yaygınlaşan internet kullanımı ile bireysel, kurumsal ve ulusal verilerin büyük ölçekte risk altındadır. Sanal ortam ve internetin yaygınlaşması bilgiye ulaşımı ne kadar basitleştirdiyse de güvenlik tehdidini kat kat yükseltmiştir. Böyle bir ortamda bilgileri korumanın önemi de artmıştır. Bu yollardan biri de kriptoloji bilimine bağlı olan kriptografidir. Bu bilim dalı günümüzün birçok alanında kullanılmaktadır [6, 7, 8].

Günümüzde veri güvenliğinin önemli bir duruma geldiğine şahit oluyoruz.Örneğin bankamatikler, rezervasyon sistemleri, e-posta iletileri, telefon bankacılığı, normal bankacılık işlemleri, cep telefonları, uydu sistemleri, füze sistemleri, savaş uçakları, askeri bilgiler, sağlık sistemleri, mimari projeler, fatura sistemleri, şehir şebekeleri, e-ticaret, e-devlet, e-imza, üyelik istemleri gibi alanlarda yer verilmektedir. Birey ve kurumlara kolaylık yapmak adına elektronik ortamda aktarılan verilerin giderek tehlikenin hacmini arttırmaktadır. Bu güvenliği gerçekleştiren yollarından biri

kriptolojibilimi ile ilgilidir. Kriptografi, uydu, internet ve haberleşmede kimlik doğrulamayı ve gizliliği sağladığı için önemli bir yer almaktadır [9].

Özel veriler; örneğin kimlik bilgileri, banka hesap bilgileri, kredi kartı bilgileri, telefon numarası, adres, fotoğraflar gibi bilgilerin başkaları tarafından izinsiz kullanılması ve bu bilgilerin diğer birçok kişi ile paylaşılması günümüzde sık yaşanan sorunlardandır. Kurumsal bilgi ve belgelerin kurum dışına sızdırılması ya da çalınması prestij kaybına veya maddi anlamda büyük zararlara neden olmaktadır. Bazen gizli bilgilerin ele geçirilmesi ulusun güvenlik ve istikbalini tehlikeye sürükleyebilmektedir ki bunlarla mücadeleye geniş yer verilmelidir. Siber savaş olarak nitelendirilen bu durum ulusal bilgilerin güvenli bir şekilde korunması ihtiyacını arttırmış ve verilerin şifrelenerek sanal ortamda aktarılmasını gerekli kılmıştır [10,11].

Bunlarla birlikte ülkeler arasındaki veri hırsızlığı da gittikçe çoğalmaktadır. Siber dünya, ülkeler ve bölgeler arasındaki harita sınırlarının ortadan kalkmasını sağlamıştır. Bu şekilde gerçekte geçilmeyen ülke hudutları elektronik ağlar sayesinde veri çalma ile aşılmaktadır. Bu gibi durumlarda gizli olan ve herkesin erişimine açık olmayan bilgiler izinsiz müdahale sonucunda açığa çıkarılmıştır. Böylelikle devletlerin her an bir saldırıya maruz kalabileceklerini ve ulusların güvenliği sarsılabileceği görülmektedir. Bilgilerin şifrelenmiş halde saklanıp transfer edilmesi bilgi güvenliğini elde etmek için başlı başına yeterli olmamakla birlikte olası tehlikeleri daha aza indirgemek için gereklidir [12].

Yaşanan bu güven sorunu ancak kriptoloji bilimiyle çözülmektedir. Kişiler, sahtekârlık ve aldatılma kaygısı taşımadan elektronik ortamda tüm işlemlerini yapabilmelidir. Kriptolojinin en önemli ayağı ise ulusal bilgi ve verilerin korunması ve güvenirliğinin sürdürülmesi olduğu için bu alanda yapılacak çalışmalar oldukça önemlidir.

1.2. Kriptolojinin Tarihçesi

Eskiden kriptoloji daha çok askeri ve diplomatik haberleşme alanında güvenliği sağlamak adına kullanılmakta idi [4].

Günümüzde ise cep telefonu, bilgisayar sistemleri ve uydu sistemleri gibi oldukça geniş kullanım alanları mevcuttur. Literatürde bulunan en eski şifreleme algoritmaları

şifreleme metodunun yalnız alıcı ile gönderen arasında olduğu bilinen algoritmalardır. Gizli algoritma ile şifrelenmiş metnin çözümlenmesi için şifreleme algoritmasının tersine çevrilmesi yeterli olurdu. Bu tip algoritma ilk defa Roma İmparatorluğunda Sezar tarafından generallerle olan yazışmasında kullanılmıştır. Her bir karakteri belirli ölçüde kaydırarak yapılmaktaydı. Bu şifreleme işleminin öbür örnekleri de yer değiştirme ve alfabe değiştirme şifreleridir [4,5].

Ebced hesabı İbrani-Süryani, Grek ve Latin harf-sayı sistemidir. Daha sonrabu sistem Arapça için de kullanılmıştır. Ebced hesabında, her alfabe karakterine belli bir sayı tahsis edilir. Bir kelimeyi oluşturan harflerin toplam sayı değeri, vurgulanmak istenilen bir olayın tarihine denk getirilir. Bu şekilde ebced hesabıyla belirli bir tarihi anlatan kelimelere veya satırlara bakıldığında hiçbir şekilde bir sayı görmek mümkün olmaz, ama kelimelerin sayısal değerleri hesaplandığında bir sonuç elde edilir. Bu metod şifrelemede bir ilham kaynağı sayılmaktadır [5].

Spartalılar M.Ö. 400 yıllarında ilk kriptografik cihaz olan scytale'yi geliştirmişler. Bu gelişme de geçmişteki kripto sistemlerine örnektir [4]. Bu cihazla bir mesajı şifrelemek için uzun bir parşömen yada papirüs, silindirik bir sopa etrafına sarılmıştır. Gizlenecek mesajın kelimeleri dikey veher bir şerit turunda bir harf gelecek şekilde yazılmıştır. Daha sonra şerit açılmış ve böylece anlamsız harflerin oluşturduğu metin ortaya çıkmıştır. Şifrelenen bu mesajın çözülebilmesi için de şifreleme işleminde kullanılan silindirle aynı çapa sahip bir silindirin kullanılması gereklidir. Bu sistemdeki silindir çapı kripto anahtarı anlamına gelir [5].

1404-1472 yılları arasında yaşamış olan Leon Alberti, ilk kez 1466-1467 yılları arasında çoklu alfabe ile şifreleme gerçekleştirmiştir. Leon Alberti, Sezar şifreleme tekniğinebenzer olan harf kaydırma metodu kullanılmıştır ancak Sezar şifrelemeden ayrılan tarafı ise kaydırma miktarının sabit olmaması ve kullanan kişinin isteği doğrultusunda belirlenebilirliğidir. Harflerin kaydırılmasında Alberti Diski kullanılmıştır. İç çemberi sabit, dış çemberi hareketli bu disk yardımıyla, her harfin istenilen miktarda ötelenmiş hali görülebilir [3, 4].

Uzun yıllar boyunca kırılmayan çok alfabeli şifreleme metodu 1553 yılında İtalyalı Giovan Batista Belaso tarafından geliştirilmiştir. Bu şifre,1586 yılında Fransız bir

diplomat tarafından biraz daha geliştirilerek Vigenére Şifresi adını alan şifreleme tekniğinin orjinal halidir. Bu şifrede açık metindeki her bir harf ayrı bir şifrealfabeyle şifrelenir.1854-1863 yılları arasında matematikçi Charles Babbage ile birlikte Avusturya ordusunda görev yapan kriptograf Friedrich Kasiski tarafından uzun yıllar sonra kırılmıştır [3, 4, 5].

1790 yılında Thomas Jefferson, Jefferson Diski yöntemini geliştirmiştir. İngiliz alfabesine göre düzenlenmiş olan Jefferson Diski İngiliz alfabe sayısına göre 26 diskten oluşmaktadır. Alfabedeki tüm harfler her diskin üzerine rastgele bir şekilde dizilmiştir. Aynı özellikleri taşıyan Jefferson Diski, hem alıcı hem degöndericide bulunmak zorundadır [4, 5].

Wheatstone Playfair şifresi 20. yüzyılın başlarına dek güvenilirliğini sağlayabilmiş olup 5x5'lik bir matris kullanılmış ve bu şifre Charles Wheatstone ve Baron Lyon Playfair tarafından geliştirilmiştir [4, 5].

1883 yılında Hollandalı dilbilimci ve kriptograf Auguste Kerekhoffis tarafından yayınlanan La Cryptographie Militarie makalesinde bir şifreleme sisteminin taşıması gereken özelliklere yer vermiştir. Ona göre sistemin güvenilir ve pratik olması önemlidir. Makalesinde yer alan 1883'ten sonraki yıllarda tasarlanan sistemlere bir yol açmıştır. Kerckhoff Prensipleri aşağıda görülmektedir [4, 5]:

1. Sistem kullanışlı ve matematiksel bir realiteye dayanmalıdır.
2. Sistem gizli tutulmamalı. Her şeyherkesce kavranmalı..
3. Sistemde kullanılan anahtarlar gönderici ve alıcılar tarafından kimse erişmeden değiştirilebilmelidir.
4. Sistemin faaliyete geçebilmesinde çok fazla insana ihtiyaç duyulmamalıdır.
5. Sistemin uygulaması ve anlaşılması kolay olmalı ve şifreleme sisteminin güvenliği, şifreleme algoritmasını gizli tutmayadayanmamalıdır. Güvenlik; yalnızca anahtarı gizli tutmaya dayanmalıdır [4, 5].

Şifreleme işleminde matematiğin kullanımı 1917 yılında ABD'de başlamıştır. Vernam şifresi olarak bilinen ve Vigenére şifreleme sistemini temel alan yeni bir şifreleme yöntemi geliştirilmiştir [4, 5].

1918 yılında Almanlar tarafından kullanılan Enigma adlı verilerin kripto cihazı vasıtasıyla Polonyalı kriptoanalistlerce 1938 yılına kadarki tüm mesajları çözülmüştür. Bunun üzerine 1939 yılında Enigma'nın yapısı daha da karmaşıklaştırılmıştır [3, 4, 5,].

Hill şifresi 1929'da Hill tarafından keşfedilmiş olup çok alfabeli şifreleme metoduna başka bir örnek teşkil etmiştir.6x6'lık bloklarla Hill şifresinin uygulanabileceği bir makine Lester Hill ve ortağı geliştirmiştir, fakat bu makineye çok fazla talep olmamıştır [3, 4].

1970 yılında IBM laboratuarlarında Demon adıyla 64 bit anahtar kullanan yeni bir şifreleme sistemi geliştirilmiştir. Daha sonra adı Lucifer olarak değiştirilmiştir. Bu sistem 1975 yılında Amerika'da Birleşik Bilgi İşleme Standardı olarak seçilmiştir [3, 4].

Elektronik ortamlarda haberleşmenin geniş bir şekilde yaygınlaşmasıyla birlikte herkes tarafından kullanılabilecek bir kriptografi algoritmaya ihtiyacın olduğu Amerika Milli Standartlar Bürosu tarafından dile getirilmiş. Lucifer algoritması üzerinde yapılan bazı değişikliklerle bu ihtiyaç giderilmiş ve Veri Şifreleme Standardı (Data Encrypt System(DES)) ortaya çıkmıştır [3, 4, 13].

Sayılar teorisinin özellikleri 1976 yılında Diffie-Hellman anahtar değişim algoritması ile ortaya çıkarılmış. Diffie ve Hellman, anahtar karakterlerin belirlenmesi için her iki taraf alıcı ve göndericinin bir araya gelmelerinin zorunlu olmadığını ispatlamıştır. Diffie-Hellman anahtar değişim algoritması, sayılar teorisi ile şifreleme işleminin herkese açık bir şekilde yapılabileceğini gözler önüne sermiştir. Böylece kriptografide yeni bir kavram olan Açık Anahtarlı Kriptografi literatüre eklenmiştir [3, 6, 4, 5].

Açık anahtar kriptografisi temel olarak herkesin birbirini tanımadan gizli bir şekilde iletişim kurabilmesi demektir. Bu kavram şifrelemede bir dönüm noktası olmuştur [3, 6, 4, 5].

Açık anahtar kriptografi kavramından etkilenen Ronald Rivest, Adi Shamir ve Leonard Adleman, 1977'de RSA(Rivest-Shamir-Adleman) adını verdikleri yeni bir açık anahtar şifreleme yöntemi keşfetmişler. RSA, Diffie-Hellmananahtar değişim algoritmasının felsefesini içeren ilk şifreleme sistemidir. RSAsistem olarak matematiğin konusu olan

çarpanlara ayırma probleminedayanmaktadır. RSA'da hem açık hem de gizli anahtar kullanılmaktadır [3, 6, 4, 5].

Mikro nokta şifreleme metodu ilk kez Alman ajanları tarafından İkinci Dünya Savaşı'nda istihdam edilmiştir. Bu metot bir sayfalık metini şifreler ve fotoğraf yöntemiyle çapı bir milimetreye yakın bir nokta boyutuna küçültülür. Daha sonra mikro nokta görünüşte önemsiz bir mektubun son noktası üzerine gizlenir [3, 4, 5].

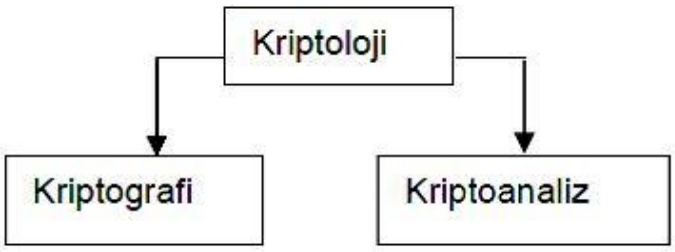
Mor ötesi boya ile yazı yazabilen spreyler ve kalemler 1960'lı yıllarda çıkmıştır. Bu sprey ve kalemlerle yazılan yazılar ancak mor ötesi bir ışıkta okunabilmektedir. Bu steganografide geniş kullanımı olan metotlardan biridir [3, 4].

Tüm bu teknolojik gelişmeler ne kadar çevik, verimli ve aktif bir şekilde kullanılırsa rakiplerin o kadar önüne geçilmiş olur. Şifreleme sistemlerindeki karmaşanın giderek artması günümüzde kuantum şifreleme metodunun gelişime katkısı olmuştur. Kuantum şifreleme verilerinaynı anda hem 1 hem de 0 biti ile ifade edilebilmesini sağlamaktadır [3, 14].

1987 yılında kriptoloji alanında ilk ders kitapları basılmıştır. Türkiye'de ise bu konuyla alakalı ilk bilimsel çalışmalar ODTÜ matematik bölümündeseçmeli ders olarak 1990 yılında gündeme gelmiş 1995'te ilk araştırma merkezi olarak TÜBİTAK bünyesinde bulunan Ulusal Elektronik ve Kriptoloji Araştırma Enstitüsü (UEKAE) birimi kurulmuştur. 2002 yılında ise ODTÜ Uygulamalı Matematik Enstitüsü bünyesinde Kriptografi Bölümü açılmıştır [5].

1.3. Kriptoloji Yöntemleri

Şekil1.1'de görüldüğü gibi kriptoloji bilimi iki alt bilim dalına ayrılmaktadır:



Şekil1.1. Kriptoloji bilimi alt bilim dalları[6].

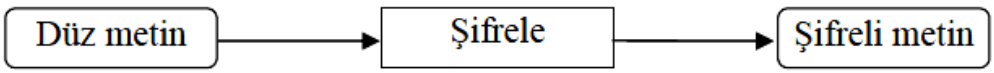
Şekil 1.1'de görülen kriptoloji bilimi alt bilim dalı Kriptografi, herkese görünebilen verileri algılanmaz biçime getirmek için kullanılan bir bilimidir. Kriptoanaliz ise şifrelenmiş bir metnin analiz aşamalarını konu alır ve metni açık hale getirmek için metotları kapsayan kriptoloji alt bilim dalıdır [6].

1.3.1. Kriptografi Nedir

Kriptografi; veri korunmasında gizlilik, kimlik denetimi, bütünlük gibi tanımların yapılabilmesi için çalışan bir takım matematiksel eylemlerdir. Bunlar verilerin mevcudiyetini, göndereni ve alıcısını korumak ve güvence altına almaktır [2].

Kriptografi (şifreleme), düz metinin içinde yer alan bilgileri, anahtar bilgisi olmaksızın okunması mümkün olmayan bir hale getirilerek verileri saldırılardan korur. Anahtar,belli bir metni şifreli hale getirmek için kullanılan sayı, sembol, harf, kelime veya belli bir veri parçasıdır. Bu anahtar aynı zamanda şifrelenmiş veriyi de çözmek için de kullanılır [2].

Şekil1.2 ile şifreleme işlemi sembolik olarak anlatılmaktadır. Eldeki metnin şifrelenmemiş ve algılanabilir haline düz ya da açık metin adı verilir. Düz metin ise bir takım işlemler neticesinde algılanmayacak duruma getirilerek şifreli metin adı verilir [2].Şifreleme eyleminde izlenecek şifreleme metoduna göre değişik biçimlerde yapılabilmektedir. Düz metinin geçirdiği değişimler neticesinde şifrelenmiş metin elde edilir. Şifreleme verilerin gizliliğini, bütünlüğünü, korunmasını garanti altına alır.



Şekil1.2. Şifreleme işlemi [2].

1.3.2. Kriptoanaliz Nedir

Kriptoanaliz şifrelenmiş verilerin çözümlenmesinde kullanılan teknikler bileşenidir. Şifreli belli bir verinin düz hale çevrilmesi eylemine şifre çözümleme ya da deşifre denir. İyi bir kriptoanaliz için kriptografi de iyi bilinmelidir. Bu şifrelerin üretilme şekli bilindiği takdirde şifreleri çözmek mümkündür [3,4, 5].

Şekil1.3 ile şifrelenmiş düz metnin çözümleme işlemi sembolik olarak anlatılmaktadır. Şifreyi çözmek için, uygun olan yöntem seçilerek düz metine ulaşılabilir:

Şekil1.3. Şifre çözme işlemi[3].

Kriptoanaliz ilk etapta şifre kırma çalışmaları ile ortaya çıkmıştır.Kriptoanaliz ile anahtar kelime bilinmeksizin şifrelenmiş metin çözülmeye çalışılır. Basit şifreleri çözmek için olası tüm alternatifleri denemek yerine matematik, istatistik ve dilbilim alanlarındaki bilgi kullanılarak şifrenin çözülmesi amaçlanır [3,4, 5].

1.4. Kriptografik Algoritmalarının Gücü

Gizli algoritmalarda çalışma biçimi gizliliğe dayalı olmakta ve günümüzün şartlarına pek uymamaktadır. Belli bir grup içerisinde kullanılan bir algoritma o grup üyelerinin biri gruptan çıktığında, yanlışlıkla ya da bilinçli olarak kullanılan algoritmayı deşifre etmesi durumunda gruptaki herkesin başka bir algoritmaya geçmesi gereklidir. Daha da kötüsü, gizli algoritmalar kalite kontrolüne ve standardizasyona olanak tanımamaktadır[20]. Yüksek güvenlik gerektirmeyen algoritmalarda, ciddi eksiklikleri bulunmasına rağmen gizli algoritma kullanılmaktadır. Literatürde bulunan en ilkel şifreleme algoritmaları sadece alıcı ile gönderen arasında bilinen ve birbirinin tersi olan gizli bir algoritmaya dayanmaktadır[20].

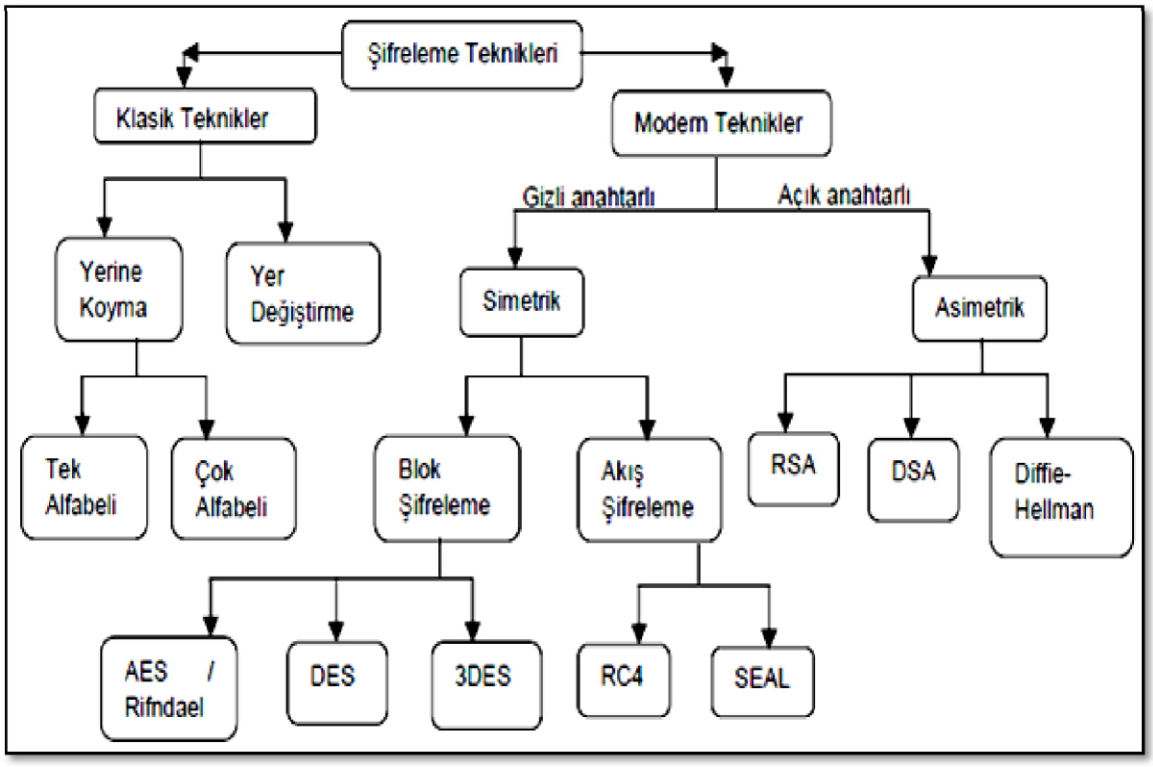
2. BÖLÜM YÖNTEM VE

MATERYAL

2.1. Kriptografi Algoritmalarının Sınıflandırılması

Gelişen teknoloji ile şifreleme için uygulanan yöntemler de değişkenlik sergilemiştir. Kriptografi algoritmaları Şekil 2.1'de görüldüğü gibi klasik ve modern olmak üzere iki ana kategoriye ayrılabilir [22,23].

1. Klasik teknikler.
2. Modern teknikler.



Şekil 2.1.Şifreleme algoritmalarının sınıflandırılması [23].

2.2.Şifreleme Algoritmalarının Üretilmesinde Temel Yöntemler

1. Yerine Koyma Yöntemleri (substitution) .

2. Yer Değiştirme Yöntemleri (transposition) .

3. Cebirsel Yöntemler.

Bu teknikler şifreleme işlemlerinde ayrı ayrı kullanılabileceği gibi bir arada da kullanılabilmektedir [16].

2.2.1. Yerine Koyma Yöntemi

Bu tür şifreleme metodunda metindeki asıl karakterler kimliklerini yitirir ancak aynı yerde kalırlar. Yani açık metindeki karakterlerin yerleri sabittir, karakterleri gizlemek için başka bir alfabenin karakterleri ya da sayılar kullanılır [16]. Bu metot maskelemede kullanılır [17].

Yerine koyma yöntemleri; basit yerine koyma, homofonik yerine koyma, çok alfabeli yerine koyma, polygram yerine koyma olarak dörde bölünür. Basit yerine koymada şifrelenmemiş ve şifreli metin harfleri arasında karşılıklı ilişki mevcuttur. Şifrelenmemiş metindeki her bir karakterin şifreli metinde tek bir karşılığı yer alır. Homofonik yerine koyma, basit yerine koyma ile benzerdir. Tek fark, homofonik yerine koymada bire çok ilişki vardır, şifrelenmemiş metindeki her bir harf şifreli metinde değişik harflerle şifrelenir. Çok alfabeli sistemler, şifrelenmemiş metindeki karakterlerin şifreli karşılıklarına ulaşmak için türlü haritalar (planlar) kullanır. Polygram şifrelemede genellikle harf grupları için keyfi(rastgele) yerine koyma işlemi uygulanmaktadır [18].

Aşağıdaki paragrafta yerine koyma yöntemi matematiksel olarak açıklanmaktadır:

Tablo2.1.Yer değiştirme yöntemi için uygulama örneği.

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

Örnek 2.1. Düz metin “ERCİYES”, M = E R C İ Y E S

Ek(M) = I M R B Z I N

Tablo 2.1 kullanılarak şifreli metin “IMRBZIN” olarak elde edilir.

2.2.2. Yer Değiştirme Yöntemi

Bu yöntemde şifrelenmemiş asıl karakterler bulundukları pozisyonu yitirirler ancak kimlikleri herhangi bir değişime maruz kalmaz. Mevcut karakterlerin yeri değişir ancak yerine başka bir karakter kullanılamaz ve metin karmaşık hale getirilerek gizlenir [16]. Yer değiştirme işlemleri gizleme tekniği için kullanılır [17].

Şifrelenmemiş metindeki harfler belli bir şekilde yeniden yerleştirilir. Klasik olarak yeniden dizme işlemi bazı geometrik şekiller vasıtasıyla yapılır. İki kademeden oluşur. İlk aşamada geometrik şeklin nasıl olacağına, ikinci aşamada ise harflerin şekil üzerine nasıl yerleştirileceğine karar verilmelidir [18].

Yer değiştirme yöntemi sütunlu yer değiştirme ve periyodik yer değiştirme olmak üzere ikiye ayrılır. Sütunlu yer değiştirme yönteminde geometrik şekil genelde iki boyutlu bir matris dizisi olmaktadır. Örneğin şifrelenmemiş metine ait harfler matrise satır satır yerleştirilirken, şifreli metne ait harfler sütun sütun seçilebilir. Sütunlu yer değiştirme yöntemi farklı dizi boyutları için de uygulanabilir. Periyodik yer değiştirme yönteminde bir periyot (her bir gruptaki harf sayısı) belirlenir.

Örnek 2.2. Şekil 2.2'de "bilgisayar" şifrelenmemiş metninin 3X4'lük bir matrise satır satır yerleştirilişi gösterilmiştir.

	1	2	3	4
1	B	İ	L	G
2	İ	S	A	Y
3	A	R		

Şekil 2.2. Yer değiştirme yöntemi için uygulama örneği matrisi[18].

Örnek 2.2'de geometrik şekil olarak 3X4'lük bir matris kullanılmıştır. Harf yerleşimi matris düzeninde görüldüğü gibi yapılmıştır. Sütunları kuralsız bir şekilde seçilir.

Sütunlar 3-1-4-2 sıralamasıyla seçileceği takdirde; şifreli metin: LABİAGYİSR olarak elde edilir.

2.2.3. Cebirsel Yöntemler

Cebirsel yöntemlerde ise karmaşık yer değiştirme, yerine koyma ya da her iki yöntemi matematiksel dönüşümler yardımıyla kullanan teknikler kullanılır [25]. Matematiksel dönüşüm işlemini asal sayılar, modüler aritmetik, eliptik eğriler, sonlu cisimler, çarpma, üs alma gibi işlemler yardımıyla yapar [19]. Hill algoritması bu türden algoritmalara örnek verilebilir.

2.3. Klasik Algoritmalar

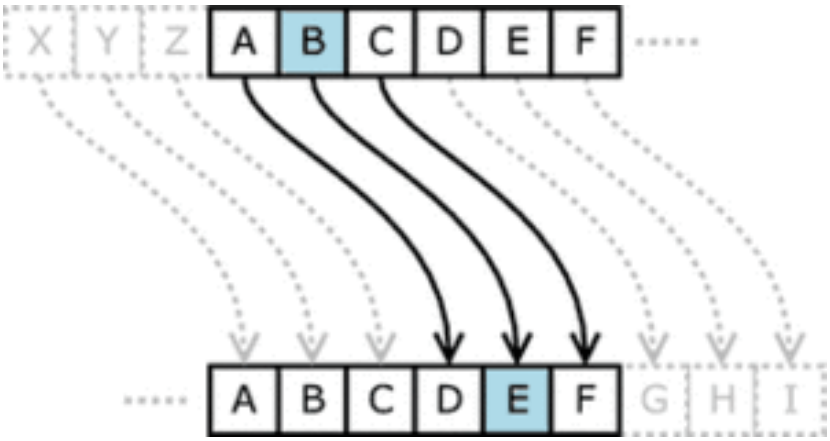
Klasik şifreleme yöntemi genellikle basit işlemlerle hesaplanabilecek algoritmalardan oluşmaktadır [24]. İlk klasik yöntemlerden biri olarak ENIGMA İkinci Dünya Savaşı döneminde kullanılmıştır. Klasik şifreleme yöntemleri geçmişte sadece askeri ve bazı ileri akademik alanlarda kullanılmıştır [23].

Klasik yöntemler, algoritması gizli olan şifreleme yöntemlerini kapsamaktadır. Bu yöntemde algoritmaların gizli tutulur olması nedeniyle güvenlik açısında düşük düzeydedir [2].

Sezar, Vigenére, Vernam,Playfair, Hill sistemleri klasik yöntemlerden bazılarıdır [22,23].

2.3.1. Sezar Şifreleme

MÖ.60-50 Julius Caesar alfabedeki harflerin yerini değiştirerek ortaya koymuş olduğu şifreleme yöntemi devletin haberleşme alanında kullanılmıştır. Şekil 2.3'tegüldüğü gibi alfabedeki harflerin sıra numaralarının belirli bir ölçüde ötelenmesi ile şifreli karşılıkları elde edilir. Tek alfabeli şifreleme yöntemidir. Bu yöntem öteleme şifrelemesi olarak da tanınmaktadır [24,27,28,29].



Şekil 2.3. Sezar şifreleme yöntemi [24].

Matematiksel bir biçimde anlatmak gerekirse; "$E_k(m) = (m + k) \bmod 26$" şeklindedir.

m, açık metindeki sıradaki harfin sıra numarasıdır.

k, harfleri öteleme ölçüsüdür.

mod26'daki 26 İngiliz Alfabesinde yer alan karakter sayısıdır.

Öteleme şifresi Türk alfabesinde kullanıldığı takdirde karakter sayısı 29 olması gereklidir [24]. Türk alfabesi örnek alınarak hazırlanmış bir uygulama örnek 2.3'te verilmektedir.

Tablo2.2. Sezar Şifreleme için uygulama örneği.

A	B	C	Ç	D	E	F	G	Ğ	H	I	İ	J	K	L	M	N	O	Ö	P	R	S	Ş	T	U	Ü	V	Y	Z
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28

Örnek 2.3.

Düz metin :ERCİYES → 5, 20, 2, 11, 27,5,21. (Tablo 2.2).

Anahtar= 3

Ek(5) = (5+ 3) mod29 = 8

Ek(20) = (20 + 3) mod29 = 23

Ek(2) = (2 + 3) mod29 = 5

Ek(11) = (11 + 3) mod29 = 14

Ek(27) = (27 + 3) mod29 = 1

Ek(5) = (5 + 3) mod29 = 8

Ek(21) = (21 + 3) mod29 = 24

Şifreli Metin: 8, 23, 5, 14, 1, 8, 24 → ĞTELBĞU olarak elde edilir.

Algoritmanın asıl metindeki her karakteri kendisinden sonra gelen üçüncü. harfle yer değiştirerek yeni bir metin elde edilmektedir. K keyfi bir değerdir. Yöntem açık olduğundan şifrelenmiş metnin daha sonradan aslına dönüştürülmesi mümkündür. Geçmişte bu teknik savaş haberleşmesini güven altına almak için kullanılmıştır.

2.3.2. Vigenere Şifreleme

Uzun yıllar boyunca bir türlü kırılması mümkün olmayan alfabeli şifre Giovan Batista isimli bir İtalyan tarafından 1553'te geliştirilmiştir. Bu şifre 1586 yılında Fransız diplomat tarafından daha da geliştirilerek Vigenere ismini almıştır. Bu yöntemde şifrelenmemiş metindeki her bir harf başka bir şifre alfabeyle şifrelenir. Şifre alfabenin seçimine anahtar kelimeye göre karar verilir. Anahtar kelimenin farklı seçilmesi, şifrelenmemiş metindeki aynı kelimeler için farklı şifreli metinler oluşmasını sağlar. Bu durum frekans analizinin uygulandığı tek alfabeli şifreleme yöntemindeki gibi (Sezar Şifreleme) iyi sonuca ulaşılmasını engeller. Uzun yıllar güvenilirliğini koruyabilen bu şifre 1854-1863 yılları arasında İngiliz matematikçi Charles Babbage ve Avusturya ordusunda görevli kriptograf Friedrich Kasiski tarafından kırılmıştır [3,4,27,28,29].

Vigenére şifreleme için alfabedeki harflerin yer aldığı bir tablo kullanılır. Şekil 2.4'te Türkçe alfabe için oluşturulmuş Vigenere tablosu görülmektedir. Asıl Vigenere şifreleme yönteminde sadece Vigenere Square olarak adlandırılan tablo kullanılır. Bu tablo şifreleme ve şifre çözme eylemlerinde sabit olarak (tablodaki harflerin yerleri değişmez) kullanılır [3,4,25,27,28,29].

Şifreleme eyleminde kullanılmak için seçilen anahtar kelime şifrelenmemiş metinin uzunluğu kadar kendisini tekrar eder. Anahtar kelimedeki harfler Vigenere tablosundaki en sol sütundan, şifrelenmemiş metindeki harfler ise en üstteki satırdan seçilir. Matris üzerinde çizilen dik doğruların birleştiği yerdeki harf şifreli metnin harfini temsil eder. Harfler eşleştirilir ve yer değiştirilir. Bu eşleştirme ve yer değiştirme işlemi her harf için yapılır[25,27, 28, 29]. Örnek 2.4'te Vigenere şifreleme yönteminin yapılması yer almaktadır:

Örnek 2.4.

Şifreleme işleminde Şekil 2.4 kullanılmıştır.

Anahtar:K R İ P T O K R İ P T O K R İ P TOK

Düz Metin*:* E R C İ Y E S Ü N İ V E R S İ T E S İ

Şifreli Metin*:* Ö İ K B S Ş E N F B R Ş D J Ş K Z H U

Vigenere şifreleme yönteminde anahtar kelimedeki harflerin sayısal karşılıkları, şifrelenmemiş metindeki harflerin öteleme ölçeğini gösterir [18].

Türk alfabesi için uygulanmış Örnek 2.4'te, şifrelenmemiş metindeki Eharfinin sayısal karşılığı 5, anahtar kelimedeki K harfinin sayısal karşılığı 13'tür. Eharfi 13 harf ötelendiğinde sayısal karşılığı18 olan Öharfi şifreli karşılığı olur. Matematiksel olarak açıklamak gerekirse [18].

Anahtar kelime= (a_1 , a_2, ……a_i) ve Düz metin = (d_1, d_2, ….., d_i) olmaküzere

$f_i(d) = (d + a_i)$ mod 29 olur.

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

Şekil 2.4. Tük alfabesi için oluşturulmuş Vigenere tablosu [18].

2.3.3.Vernam Şifreleme

Gilbert Vernam 1917'de bu yöntemi geliştirmiş olup telgraf bağlantılarında kullanılmıştır. Bu yöntem Vigenére yöntemine benzemektedir. Farkı ikili sayı sistemine (0 ve 1) yer vermesi ve şifrelenmemiş metinin XOR (exclusive-or) eyleminden geçmesidir.Veri rastgele belirlenmiş ve kendisini tekrarlatmayan anahtarlar vasıtasıyla şifrelenir. Şifreleme işleminde ikili sistemde yazılmış ASCII adı verilen Tablosunu kullanılır şekil 2.4'te ki gibi. Her harf için ASCII kodu ayrıdır. Rastgele belirlenen anahtar dizinin her bir karakterine karşılık gelen ASCII koduna şifrelenmemiş metinin her bir karakterinin ASCII kodu eklenerek (XOR) yeni şifreli karakter dizisi elde edilir.ASCIIkod tablosu şekil 2.5'teverilmektedir. XOR işleminin kuralına göre, her iki bit aynı ise sonuç olarak 0 biti, her iki bit ayrı ise sonuç olarak 1 biti üretilir[27,28,29].

Şifrelenmemiş metin $M_1M_2\ldots$,anahtar dizi $K_1K_2\ldots$,ve şifreli metin $C_1C_2\ldots$,ile temsil edildiğinde, şifreli metin $C_i=(M_i XOR\ K_i)$ile elde edilir [18, 26]. Örnek 2.5'teVernam Şifreleşmesi için yapılmış bir örnek görülmektedir.

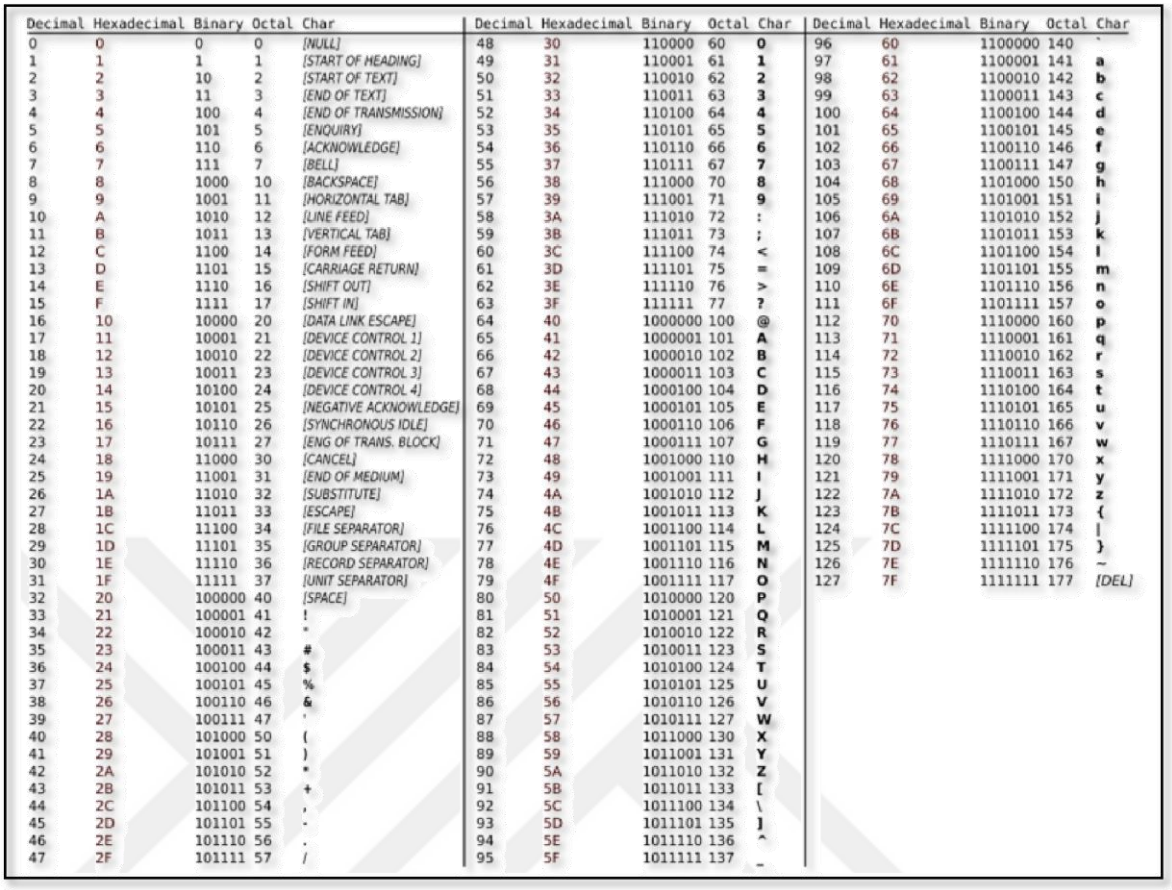
Örnek 2.5.

Düz Metin karakteri = w, ASCCI kodu 1110111

Anahtar dizi karakteri = A, ASCCI kodu 1000001

Şifreli Metin karakteri = 6, ASCCI kodu 0110110

w ve A harflerinin ASCIIkodları ikilik sisteme göre toplanır ve şifreli karakterin ASCIIkodu elde edilir [18].



Decimal	Hexadecimal	Binary	Octal	Char
0	0	0	0	[NULL]
1	1	1	1	[START OF HEADING]
2	2	10	2	[START OF TEXT]
3	3	11	3	[END OF TEXT]
4	4	100	4	[END OF TRANSMISSION]
5	5	101	5	[ENQUIRY]
6	6	110	6	[ACKNOWLEDGE]
7	7	111	7	[BELL]
8	8	1000	10	[BACKSPACE]
9	9	1001	11	[HORIZONTAL TAB]
10	A	1010	12	[LINE FEED]
11	B	1011	13	[VERTICAL TAB]
12	C	1100	14	[FORM FEED]
13	D	1101	15	[CARRIAGE RETURN]
14	E	1110	16	[SHIFT OUT]
15	F	1111	17	[SHIFT IN]
16	10	10000	20	[DATA LINK ESCAPE]
17	11	10001	21	[DEVICE CONTROL 1]
18	12	10010	22	[DEVICE CONTROL 2]
19	13	10011	23	[DEVICE CONTROL 3]
20	14	10100	24	[DEVICE CONTROL 4]
21	15	10101	25	[NEGATIVE ACKNOWLEDGE]
22	16	10110	26	[SYNCHRONOUS IDLE]
23	17	10111	27	[ENG OF TRANS. BLOCK]
24	18	11000	30	[CANCEL]
25	19	11001	31	[END OF MEDIUM]
26	1A	11010	32	[SUBSTITUTE]
27	1B	11011	33	[ESCAPE]
28	1C	11100	34	[FILE SEPARATOR]
29	1D	11101	35	[GROUP SEPARATOR]
30	1E	11110	36	[RECORD SEPARATOR]
31	1F	11111	37	[UNIT SEPARATOR]
32	20	100000	40	[SPACE]
33	21	100001	41	!
34	22	100010	42	"
35	23	100011	43	#
36	24	100100	44	$
37	25	100101	45	%
38	26	100110	46	&
39	27	100111	47	'
40	28	101000	50	(
41	29	101001	51	)
42	2A	101010	52	*
43	2B	101011	53	+
44	2C	101100	54	,
45	2D	101101	55	-
46	2E	101110	56	.
47	2F	101111	57	/
48	30	110000	60	0
49	31	110001	61	1
50	32	110010	62	2
51	33	110011	63	3
52	34	110100	64	4
53	35	110101	65	5
54	36	110110	66	6
55	37	110111	67	7
56	38	111000	70	8
57	39	111001	71	9
58	3A	111010	72	:
59	3B	111011	73	;
60	3C	111100	74	<
61	3D	111101	75	=
62	3E	111110	76	>
63	3F	111111	77	?
64	40	1000000	100	@
65	41	1000001	101	A
66	42	1000010	102	B
67	43	1000011	103	C
68	44	1000100	104	D
69	45	1000101	105	E
70	46	1000110	106	F
71	47	1000111	107	G
72	48	1001000	110	H
73	49	1001001	111	I
74	4A	1001010	112	J
75	4B	1001011	113	K
76	4C	1001100	114	L
77	4D	1001101	115	M
78	4E	1001110	116	N
79	4F	1001111	117	O
80	50	1010000	120	P
81	51	1010001	121	Q
82	52	1010010	122	R
83	53	1010011	123	S
84	54	1010100	124	T
85	55	1010101	125	U
86	56	1010110	126	V
87	57	1010111	127	W
88	58	1011000	130	X
89	59	1011001	131	Y
90	5A	1011010	132	Z
91	5B	1011011	133	[
92	5C	1011100	134	\
93	5D	1011101	135	]
94	5E	1011110	136	^
95	5F	1011111	137	_
96	60	1100000	140	`
97	61	1100001	141	a
98	62	1100010	142	b
99	63	1100011	143	c
100	64	1100100	144	d
101	65	1100101	145	e
102	66	1100110	146	f
103	67	1100111	147	g
104	68	1101000	150	h
105	69	1101001	151	i
106	6A	1101010	152	j
107	6B	1101011	153	k
108	6C	1101100	154	l
109	6D	1101101	155	m
110	6E	1101110	156	n
111	6F	1101111	157	o
112	70	1110000	160	p
113	71	1110001	161	q
114	72	1110010	162	r
115	73	1110011	163	s
116	74	1110100	164	t
117	75	1110101	165	u
118	76	1110110	166	v
119	77	1110111	167	w
120	78	1111000	170	x
121	79	1111001	171	y
122	7A	1111010	172	z
123	7B	1111011	173	{
124	7C	1111100	174	|
125	7D	1111101	175	}
126	7E	1111110	176	~
127	7F	1111111	177	[DEL]

Şekil 2.5. ASCIIkodtablosu [18].

2.3.4. Hill Şifreleme

Bu yöntem bir blok şifrelemedir. Şifrelenmemiş metni bitişik ve aynı uzunluktaki bloklara bölme, her bloğu şifreleyerek şifreli metin bloklarına çevirme ve bu şifreli blokları şifreli metin çıktısı olarak gruplara ayırmaktır. Bu yöntem 1929'da Lester Hill tarafından keşfedilip yaygınlaşmıştır.

Hill şifrelemede şifreleme anahtarı olarak bir katsayılar matrisi (K) kullanılır. Katsayılar matrisinin elamanları ile şifrelenmemiş metindeki karakterlerin sayısal karşılıklarından oluşturulan matrisin elamanları çarpılır[27,28,29].

Şifrelenmemiş metin uzunluğu 2 birim olan karakter gruplarına bölünür. Şifrelenmemiş metinin uzunluğu D = 2, şifrelenmemiş metin M_1M_2, şifreli metin C_1C_2olmak üzere katsayılar matrisinin eleman sayısı 4'tür ve 2X2'lik bir matristen oluşur [23,18].

Alfabedeki harf sayısı n kabul edildiğinde Hill şifreleme tekniği, matematiksel bir şekilde anlatmak gerekirse;

$$\begin{bmatrix} C_1 \\ C_2 \end{bmatrix} = \begin{bmatrix} K_{11} & K_{12} \\ K_{21} & K_{22} \end{bmatrix} * \begin{bmatrix} M_1 \\ M_2 \end{bmatrix} \text{mod n}$$

$C_1 = (k_{11}m_1 + k_{12}m_2)$ mod n ve $C_2 = (k_{21}m_1 + k_{22}m_2)$ mod n, olarak hesaplanır

Örnek 3.6'daİngiliz Alfabesi kullanılarak uygulanmış Hill şifreleme örneğine yer verilmektedir

Örmek 3.6.

Düz Metin = BILGISAYAR → BI (1,8), LG (11,6), IS (8,18), AY(0,24), AR(0,17).

Anahtar= HILL (7, 8, 11, 11)

$$K = \begin{bmatrix} 7 & 8 \\ 11 & 11 \end{bmatrix}$$

BIiçin; $C_1 = (7*1 + 8*8) \text{ mod } 26 = 19$, $C_2 = (11*1 + 11*8) \text{ mod } 26 = 21$

$C_1C_2 = (19, 22) \rightarrow$TV

LG için; $C_1 = (7*11 + 8*6) \text{ mod } 26 = 21$,$C_2 = (11*11 + 11*6) \text{ mod } 26 = 5$

$C_1C_2 = (21, 5) \rightarrow$VF

İSiçin; $C_1 = (7*8 + 8*18) \text{ mod } 26 = 18$, $C_2 = (11*8 + 11*18) \text{ mod } 26 = 0$

$C_1C_2 = (18, 0) \rightarrow$SA

AYiçin; $C_1 = (7*0 + 8*24) \text{ mod } 26 = 10$, $C_2 = (11*0 + 11*24) \text{ mod } 26 = 4$

$C_1C_2 = (10, 4) \rightarrow$KE

ARiçin; $C_1 = (7*0 + 8*17) \text{ mod } 26 = 6$, $C_2 = (11*0 + 11*17) \text{ mod } 26 = 5$

$C_1C_2 = (6, 5) \rightarrow$GF

Şifreli Metin = TVVFSAKEGF olarak elde edilir.

2.3.5. Playfair Şifreleme

Bu yöntem adını İngiliz bilim adamı Lyon Playfair'den almıştır. Yerine koyma yönteminin bir türüdür. Lyon Playfair tarafından geliştirilen bu yöntem 1854 yılında Wheatstone tarafından keşfedilmiştir.

Bu yöntem Birinci Dünya Savaşı'nda İngilizler tarafından kullanılmıştır. 5X5'lik matris düzeni ile şifreleme işlemini gerçekleştirir.

Tablo2.3'te. Playfair için anahtar diziyi gösteren matris verilmiştir. Şifrelenmemiş metindeki her harf ikilisi (m_1m_2), takip eden kurallarla şifrelenir [18,27,28,29]:

1. $m_1$ve m_2 aynı satırdaysa, $m_1$ve m_2'nin yer aldıkları sütununsağında olduğu kabul edilen ilk sütunda sırasıyla $m_1$ve m_2'ninsağındaki karakterler $c_1$ve c_2'yi (şifreli haller) verir.

2. $m_1$ve $m_2$aynı sütundaysa, $m_1$ve m_2'nin yer aldıkları satırınaltındaki ilk satırda sırasıyla m_1 *ve* m_2'nin altındaki karakterler $c_1$vec_2'yi (şifreli haller) verir.

3. $m_1$ve m_2 farklı satır ve sütundaysa, m_1 ve m_2'nin yer aldıkları yermatrisin köşesi kabul edilir. c_1, m_1'in olduğu satırın öbür köşesindeki karakter olur. c_2, m_2'nin olduğu satırın diğer köşesindeki karakter olur.

4. Eğer $m_1 = m_2$ olursa, şifrelenmemiş metinde kullanılmayan bir harf m$_1$vem_2'nin arasına yerleştirilir. Böylece aynı iki harfin yan yana gelmesi önlenmiş olur.

5. Eğer şifrelenmemiş metin ikili harf gruplarına ayrıldığında tekli harf kalırsa şifrelenmemiş metinde kullanılmamış bir harf yanına eklenir. Örnek 2.7'de playfair şifreleme yöntemine ait bir uygulama görülmektedir:

Örnek 2.7.Tablo 2.3 kullanılarak gerçekleştirilmiştir.

Anahtar: TASHFER

Şifrelenecek Metin: PLAYFAIRCIPHER

PL – AY – FA – IR – CI – PH – ER

Şifrelenmiş Hali: QK – HW – ST – OI – RL – QS – RB

Tablo2.3. Playfair için anahtar diziyi gösteren matris.

T	A	S	H	F
E	R	B	C	D
G	I/J	K	L	M
N	O	P	Q	U
V	W	X	Y	Z

2.4. Bir Şifreleme Algoritmasının Taşıması Gereken Temel Özellikleri

1. Gizlilik: veriler konuyla alakalı kimseler tarafından algılanabilmelidir.

2. Bütünlük: verilerin değişime maruz kalıp kalmadıklarını açıklar. Veriler yetkili şahıslar dışında kimse tarafından değiştirilmemeli.

3. İnkar edememe: Bilgiyi üreten ya da ileten kişinin daha sonrasındabunu inkar edememesi durumudur.

4. Kimlik doğrulama: Bilginin iletiminde karşılıklı iki tarafta da kimlikdoğrulama kullanılırsa üçüncü bir kişinin bilgiye ulaşmasınıengellemek için kimlik doğrulama bilgilerine ulaşmaması gereklidir.

5. Erişilebilirlik/Süreklilik: Yetkili kişilerin istenilenzaman verilereulaşabilmesidir

Bu temel özelliklere göre, kriptografi verilerin gizlenmesinin yanı sıra veribütünlüğü, kimlik kanıtlama ve inkar edememe konuları ile de ilgilidir[15, 14].

2.5. Görüntü Şifreleme Sistemlerinin Özellikleri

Görüntü şifreleme yöntemlerini incelemek için ilk olarak metin verilerle görüntü verileri arasındaki uygulama ayrımlarını analiz etmek gerekir. Temel olarak görüntü verileriyle metin verileri arasındaki farklar şu şekildedir [21].

1. Belli bir metnin şifrelendikten sonra deşifre edilmesi durumunda veri kaybı olmamalıdır. Ancak bir görüntü şifrelenip çözüldükten sonra da belli bir veri kaybı söz konusu olabilmektedir. Görüntüler belli miktardaki veri kaybına daha az duyarlıdır.

2. Metin verileri kelimelerden oluşmaktadır. Bu sebeple metin verileri hemen akıcı ya da blok şifrelerle şifrelenebilmektedir. Ancak sayısal görüntü verileri çoğunlukla iki boyutlu dizilerle ifade edilmektedirler

3. Şifrelenecek metnin boyutu, tek bir karakterden tüm birsayfaya kadar değişebilmektedir. Bir görüntü için gerekli olan alan çoğunlukla çok büyük olduğu içingörüntüyüdirekt şifrelemek verimli değildir.

Büyük boyuttaki görüntülerin, gönderme süresini ya da kapsayacağı alanı indirgemek için görüntü sıkıştırma tekniklerinden yararlanılır.İyi bir veri koruma sistemi, sadece

metin formatındakisaklı verileri değil aynızamanda görüntü verilerini de çok iyi bir düzeyde saklayabilir.

Genel olarakbir bilgi güvenliği sisteminde üç temel özellik ön plana çıkmaktadır:

1. Gizlilik: Mesaj yetkililer dışında kimse tarafından okunamamalıdır.

2. Bütünlük: Mesaj yetkili dışında hiç kimse tarafından değiştirilmemeli ve sabote edilmemelidir .

3. Kullanılabilirlik: Mesajlar, yetkili kişilerce tam olarak erişilebilir formatta olmalıdır.Hatasız bir görüntü şifreleme sistemi, güvenlik sistemlerinde esnek olmanınyanında aynı zamanında yüksek performansa da sahip olmalıdır.

Bu nedenle yukarıda zikredilen özelliklere ek olarak görüntü güvenliğinde aşağıdaki özelliklere de yer verilmelidir [21].

1. Şifreleme sistemi, hesap işlemleri açısından çok güvenli olmalıdır.Yetkisiolamayan kişiler şifreli görüntüleri okuyamamalıdır.

2. Şifreleme ve deşifre algoritmaları, sistemin performansınıdüşürmeyecek kadar hızlı olmalıdır. Şifreleme ve deşifre algoritmalarıher türlü bilgisayara sahip kullanıcılarca uygulanabilecek kadar pratik olmalıdır.

3. Güvenlik mekanizması mümkün olduğunca geniş bir kulanım alanına sahip olmalıdır.

4. Güvenlik mekanizması esnek olmalıdır.

5. Şifrelenmiş görüntü verilerinde çok fazla büyüme olmamalıdır

2.6. Base64 Yöntemi

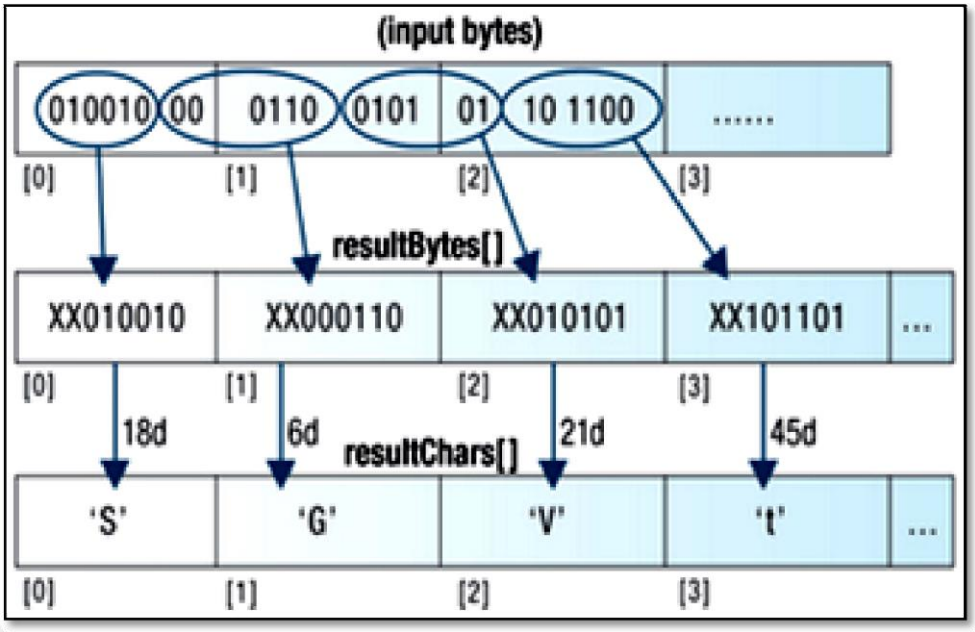
Base64 kodlamaları genelde binary verilerin ASCII kodlarına iletilmesini sağlayan kodlardır. 8 Bitlik verileri 64 Bitlik veriler şekline segmente ederek aslında 8 Bit olmayan değişik katmanlarda veri iletişimini sağlarken veri kaybını önlemek için kullanılmaktadır.Şekil 2.6'daBase64 tabanına çevrim gösterilmektedir[30].

Şekil 2.6. Base64 ile çevirinmiş görüntü.

Kodlama işlemi yapılırken 3 Byte'lik olan verileri 6'şar Byte'lik 4'lü gruplar şeklinde ayrılır. 2^6=64 olmasından dolayıher bir 6 bit'lik grup 0 ile 63 arasında bir değerle karşılaştırılır.

Yani girilen bir metnin Decimal değerine dönüştürülmektedir[30]. Ve sonrasında ikili sisteme çevrilmektedir. Elde edilen sonuç 6 Bitlik parçalara bölünür. Bölünmüş Bit grupları ikilik sisteme göre değerlendirecektir.

Şekil 2.7'de görüldüğü gibi. Sonra tekrar decimal değerlereçevrilir. Base64 tablosunda (Şekil 2.8) karşılık gelen verilere göre bilgiler sırayla yazılır[30].



Şekil 2.7. Base64 işlemi anlatımı[30].

Value	Char	Value	Char	Value	Char	Value	Char
0	A	16	Q	32	g	48	w
1	B	17	R	33	h	49	x
2	C	18	S	34	i	50	y
3	D	19	T	35	j	51	z
4	E	20	U	36	k	52	0
5	F	21	V	37	l	53	1
6	G	22	W	38	m	54	2
7	H	23	X	39	n	55	3
8	I	24	Y	40	o	56	4
9	J	25	Z	41	p	57	5
10	K	26	a	42	q	58	6
11	L	27	b	43	r	59	7
12	M	28	c	44	s	60	8
13	N	29	d	45	t	61	9
14	O	30	e	46	u	62	+
15	P	31	f	47	v	63	/

Şekil 2.8. Base64 karakter seti tablosunu [30].

2.7. Klasik Algoritmalarla Görüntü Şifreleme

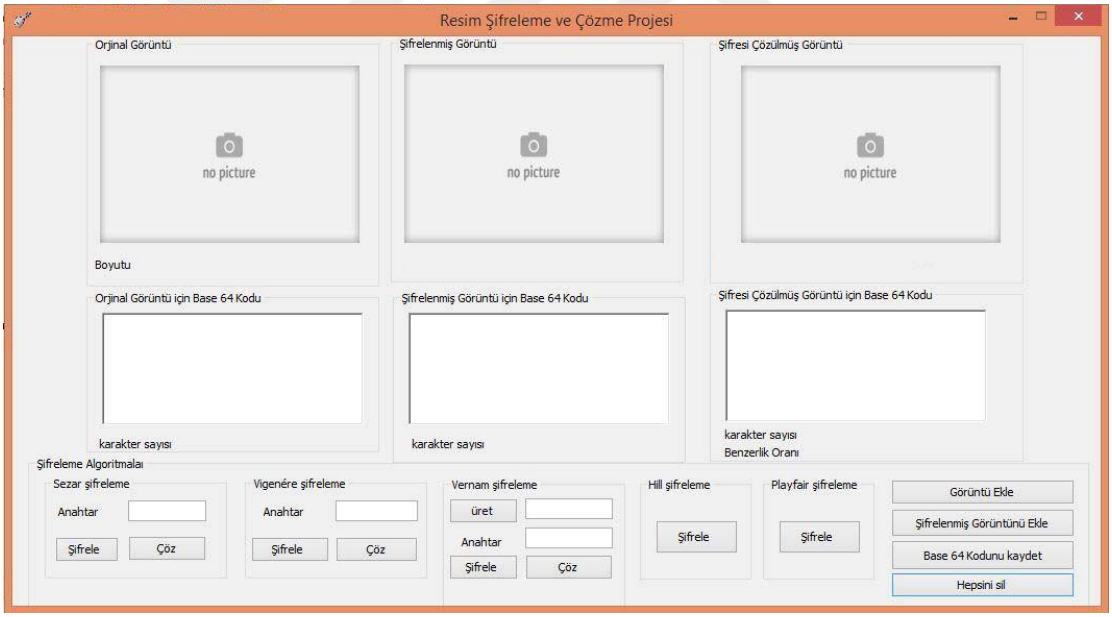
Metin şifrelenirkenkullanılan dildeki harf sayısına göre şifreleme işlemi yapılır. Görünü şifrelemede ise büyük/ küçük harfler ve nümerik rakamlar ve bazı operatörlerdenoluşan 64 karakterlik bir alfabe kullanılmaktadır (A-Z, a-z) (0-9)(+/).

Görüntüler Base64 vasıtasıyla bir karakter dizisi oluşturulur ve algoritma için bir düz metin girdisi olarak kullanılır.Daha sonra anahtar kullanarak işlemler gerçekleştirilir ve şifrelenmiş metini elde etmiş oluruz. Aynı zamanda da görüntünün şifrelenmiş olduğu anlamına gelir.

Klasik algoritmalarından en yaygın olan ve sık sık kullanılan Sezar, Vigenere, Vernam, Hill, Playfair algoritmaların genel yapısı hakkında bilgi verilerek sözü geçen algoritmaların metin kullanılarak çalışmasını ve görüntü şifrelenebilmesi için yapılması gereken değişikliklerden de söz edilecektir.

Bu bölümde şifreleme algoritmaların (klasik algoritmalar) çalışmasını gerçekleştiren program yapısı ve kullanımına açıklık getirilecektir.

2.8. Sistem Tasarımı



Şekil 2.9.Programın genel görüntüsü.

Belirlenmiş algoritmalarla sistemin amacı şu şekilde özetlenebilir:

1. Farklı boyuttaki görüntülerin şifrelenmesi ve şifrelerinin çözülmesini göstermek.

2. Görüntü şifreleme ve şifre çözme işleminde harcanan zaman miktarı, tüketilen hafıza ve işlemci miktarını göstermek.

3. Algoritmaların görüntüyü şifreleme performanslarını ölçmek ve karşılaştırmak.

Gerçeklenen sistemlere ait genel arayüzü uygulamanın şekil 2.9'da verilmektedir. Hazırlanmasında Visual C#.net 2015 kullanılmıştır.
Görüntü Şifreleme ve Çözme Programın ara yüzünü oluşturan elementlerin görevleri aşağıda verilmektedir:

1. **Görüntü Ekle**: jpg, png vb. Formattakişifrelenecek görüntünün belirlenmesi.

2. **Şifrelenmiş Görüntü Ekle**: daha önceden şifrelenmiş ve Base64 kodu olarak kaydedilmiş olan görüntünün deşifre edilmesi.

3. **Base64 kodunuKaydet**: orijinal görüntünün şifrelenmiş halinin görüntü veya Base64 kodu formatında kaydedilmesi.

4. **Hepsini Sil:**programın ara yüzündeki tüm verilerin silinmesi.

5. **Şifrele:**şifreleme algoritması belirlendikten sonra orijinal görüntüyü Base64 koduna çevirime ve şifreleme.

6. **Çöz:**şifreleme algoritması belirlendikten sonra şifrelenmiş görüntününBase64 kodundan orijinal görüntüyeçevrilmesi.

7. **Anahtar:**algoritmalar için gereken anahtarın girilmesi.

8. **Orijinal Görüntü:**seçilen görüntünün gösterim alanı.

9. **Şifrelenmiş Görüntü:**şifrelenmiş görüntünün gösterim alanı.

10. **Şifresi Çözülmüş Görüntü:** deşifre olmuş görüntünün gösterim alanı.

11. **Orijinal Görüntü İçin Base64 Kodu:**orijinal görüntüye ait Base64 Kodu.

12. **Şifrelenmiş Görüntü İçin Base64 Kodu:**şifrelenmiş görüntünün Base64.

13. **Şifresi Çözülmüş Görüntü İçin Base64 Kodu:**şifresi çözülmüş görüntünün Base64 Kodu.

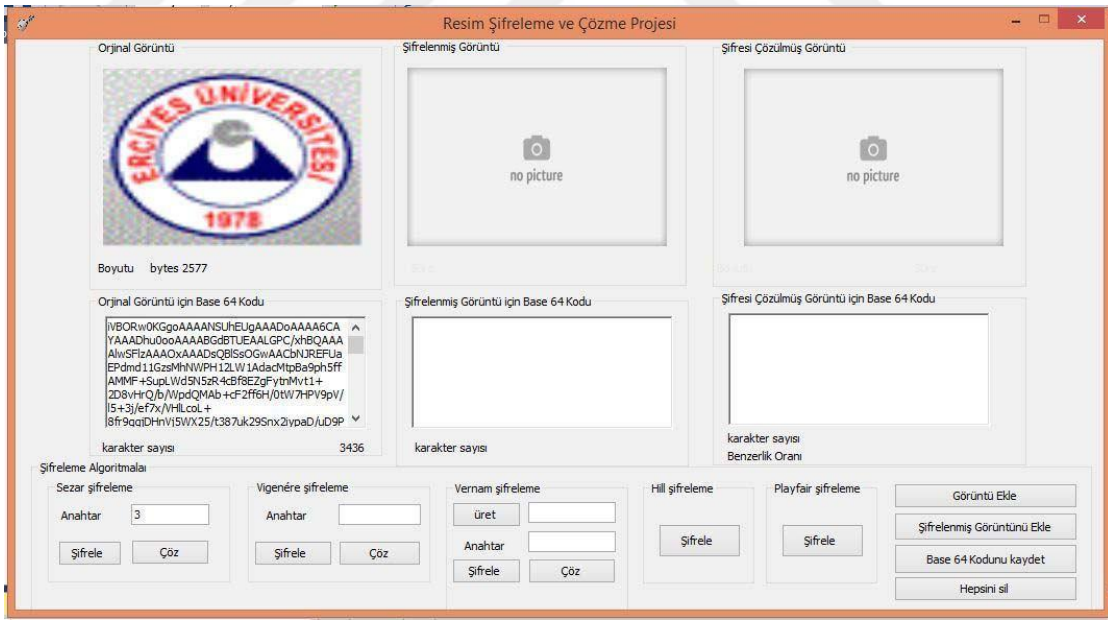
14. **Şifreleme Algoritmalar:**programda kullanılacak olan algoritmalarSezar, Vigenere, Vernam, Hill ve Playfair algoritmalarıdır. Bu algoritmaların tümünde de anahtar, şifreleme ve çözmeişlemleri yer almaktadır. Hill ve Playfair'e ait bazı farklılıklar ileride verilecektir.

15. **Karakter Sayısı:** görüntünün için Base64 Kod karakter sayısı gösterimi.

16. **Benzerlik Oranı:**deşifre işlemi tamamlandıktan sonra ise Base64 Kodunun orijinal görüntünün bu kod ile karşılaştırması ve şifrelenmiş ve orijinal görüntünün benzerlik oranının hesaplanması.

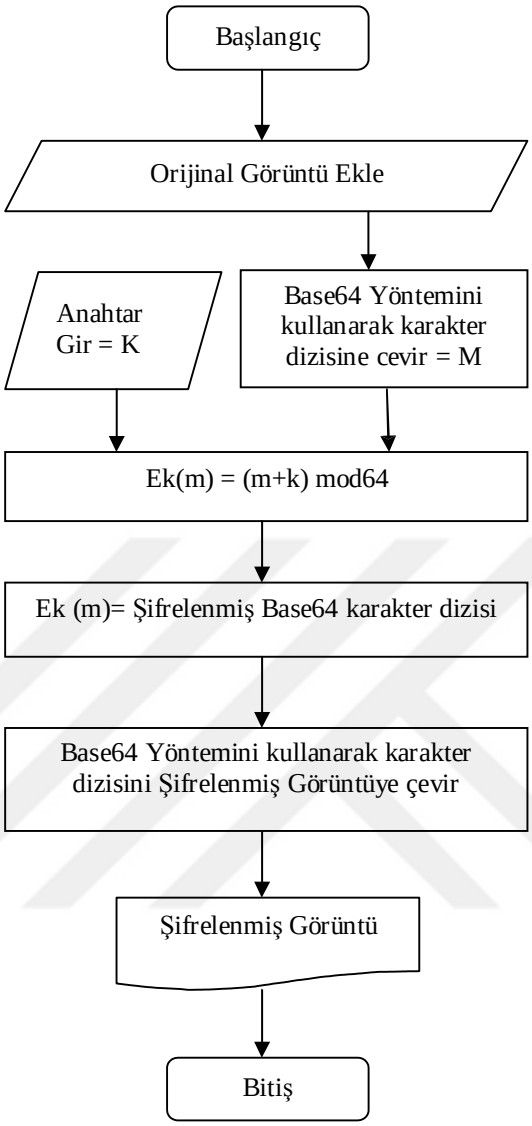
2.8.1. Sezar Şifreleme

Alfabedeki harflerin sıra numaralarının belirli bir ölçüde ötelenmesi ile şifreli karşılıkları elde edilir.



Şekil 2.10. Sezar algoritmasıyla orijinal görüntü için Base64 kodu işlemi.

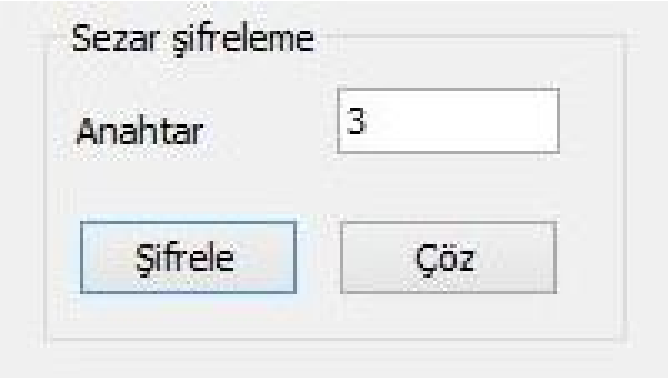
Şekil 2.11'deki gibi görüntü şifrelemek için Sezar Algoritmasının adımları şu şekilde olacaktır.



Şekil 2.11.Görüntü şifrelemek için Sezar Algoritmasının adımları.

Görüntü Şifreleme ve Çözme programı kullanılarak görüntü programa yüklenir ve Base64 koduna çevrilir. Sezar Algoritması ile şifreleme işlemi başlamış olacaktır. Öte yandan Base64 karakter sayısı görüntünün boyutuyla orantılı olacaktır. Şekil 2.10'da görüldüğü gibi Erciyes logogörüntüsü boyutu 2.5KB ve karakter sayısı 3436'dır.

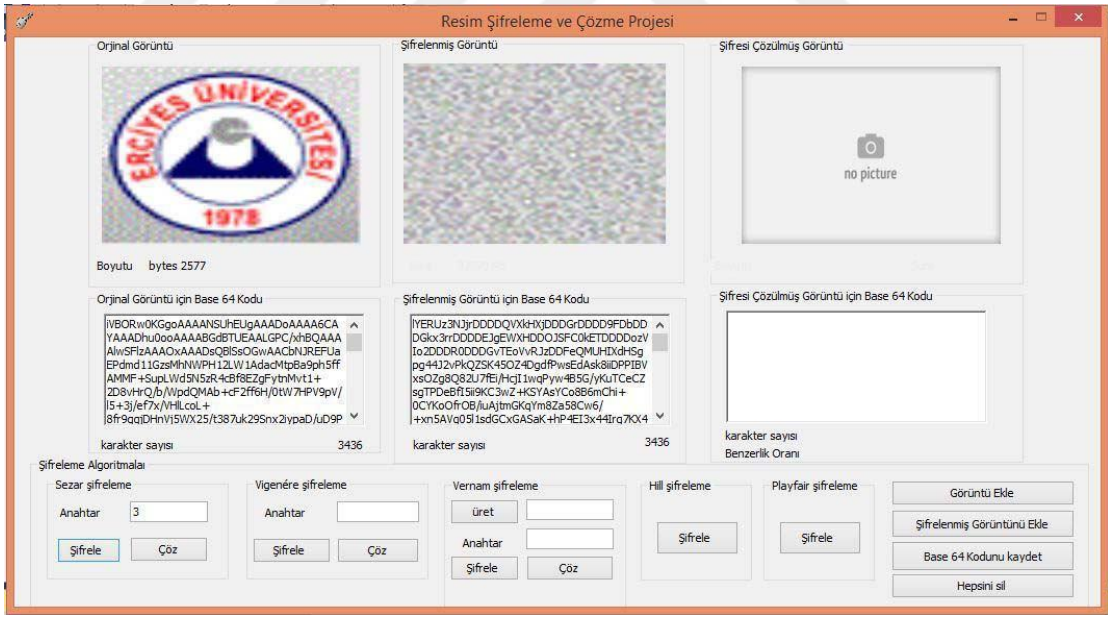
Orijinal görüntü programa yüklendikten sonra Şekil 2.10'da ki gibi ve Base64 yöntemiyle otomatik olarak karaktere çevrilir. Daha sonra anahtar belirlenir şekil 2.12'de yer verildiği gibi anahtar 3 olarak kullanılmıştır.



Şekil 2.12. Sezar algoritması için anahtar girme alanı.

Şifrelemeişlemi başlatıldıktan sonra Şekil 2.11'deki adımlar kullanılarak orijinal görüntünün Base64 kod çıktısının her karakteri girilen anahtar sayısına göre ötelenerek Base64 tablosuna denk gelen sayının karşısındaki karakter alınmaktadır. Denklem (1) kullanılarak görüntü şifrelenir. Şekil 2.13'te bu işlemin uygulamada yapılması gösterilmektedir.

Ek(m) = (m+k) mod 64….(1)



Şekil 2.13. Sezar algoritmasıyla görüntü şifreleme aşması.

Şifrelenmiş görüntüyü çözmek için de denklem (2) kullanılarak görüntünün şifresi çözülür ve orijinal görüntü tekrardan elde edilir. Şekil 2.14.

Ek(m) = (m-k) mod64…(2)

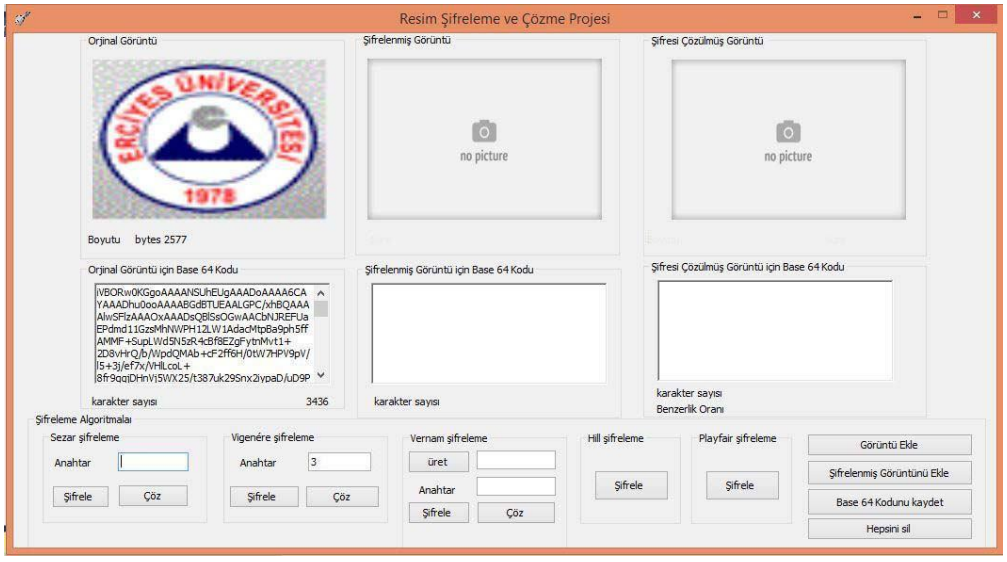
Sezar algoritmasında karakter silme veya ekleme olmadığı için benzerlik oranının %100 olduğu görülmektedir.

Şekil 2.14. Sezar algoritmasıyla şifre çözme aşaması.

2.8.2.Vigenere Şifreleme

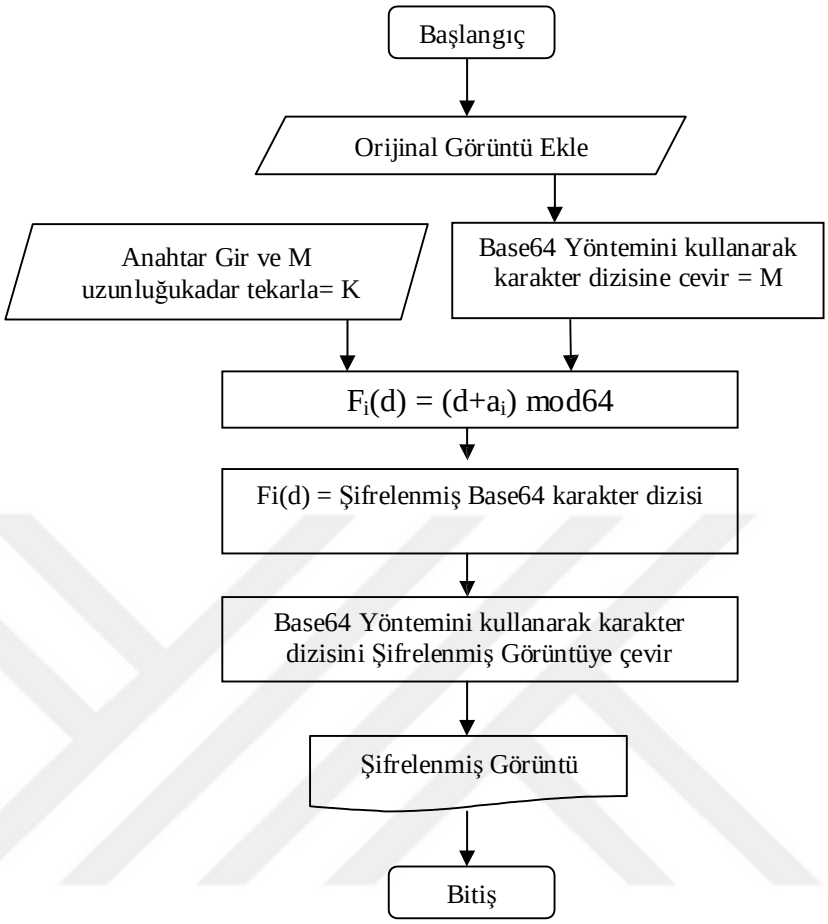
Bu yöntemde şifrelenmemiş metindeki her bir harf başka bir şifre alfabeyle şifrelenir. Şifre alfabenin seçimine anahtar kelimeye göre karar verilir.

Şifreleme eyleminde kullanılmak için seçilen anahtar kelime şifrelenmemiş metinin uzunluğu kadar kendisini tekrar eder.



Şekil 2.15. Vigenere algoritmasıyla orijinal görüntü ekleme aşaması.

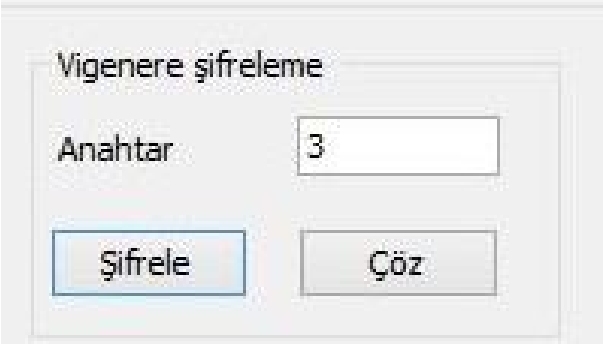
Vigenere algoritması ile görüntü şifreleme adımları Şekil 2.16'da verilmektedir.



Şekil 2.16.Vigenere algoritmasının adımları.

Görüntü Şifreleme ve Çözme programı kullanılarak görüntü programa çağırılıp Base64 yöntemiyle de karaktere çevirilince.Vigenere Algoritması ile şifreleme işlemi başlamış olacaktır. Öte yandan Base64 karakter sayısı görüntünün boyutuyla orantılı olacaktır.

Orijinal görüntü programa yüklendikten sonra Şekil 2.15'teki gibiBase64 koduna çevrilir. Şekil 2.17'de yer verildiği gibianahtar belirlenir anahtar 3 girildikten sonra anahtar Baes64 kodunun uzunluğu kadar anahtar tekrarlanır.

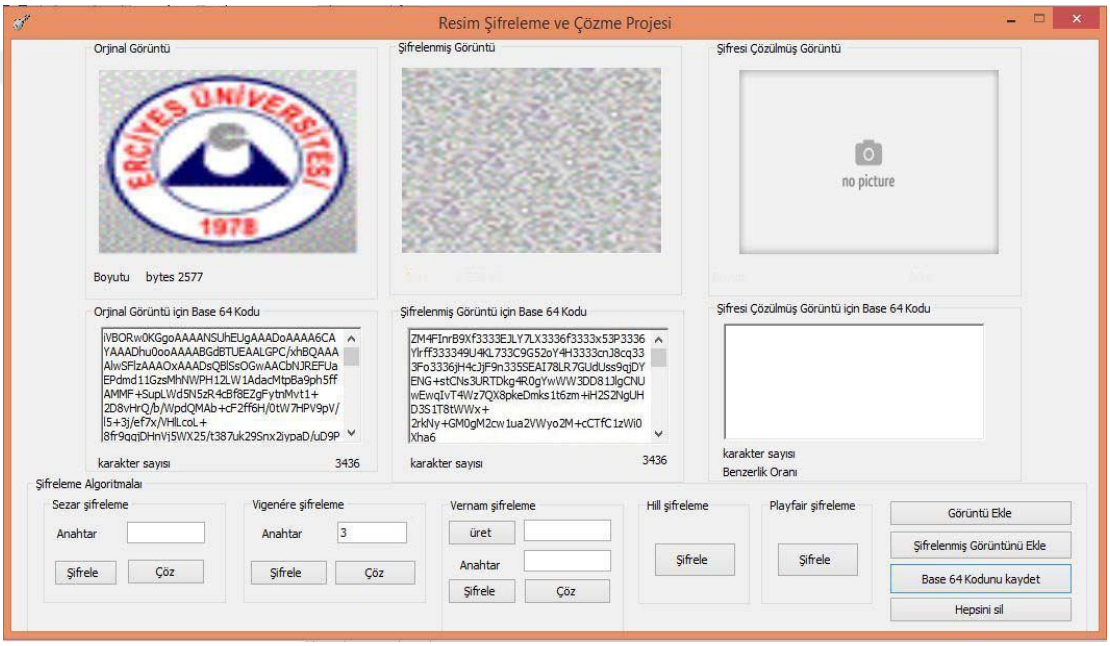


Şekil 2.17. Vigerere algoritması için anahtar girme alanı.

Denklem(3) kullanılarakBase64 tablosu kullanılarak orijinal görüntünün Base64 kodundan ilk karakter alınıp onun tablodaki denk geldiği sıra numarası ile anahtardaki ilk karakterin sıra numarası toplanıp mod64 işlemi yapılarak şifreleme işlemi gerçekleştirilmiş olur.

$f_i(d) = (d+a_i) \bmod 64$…..(3)

Bu işlemleri Base64 dizisindeki karakter sayısı uzunluğu kadar tekrarlanır Şekil 2.18'deverildiği gibi şifrelenmiş görüntü ve şifrelenmiş Base64 kodu programda üretilir.



Şekil 2.18. Vigenere algoritmasıyla görüntü şifreleme aşaması.

Şifrelenmiş görüntüyü çözmek için de denklem (4) kullanılarak orijinal görüntü tekrardan elde edilmiş olacaktır.Şekil 2.19.

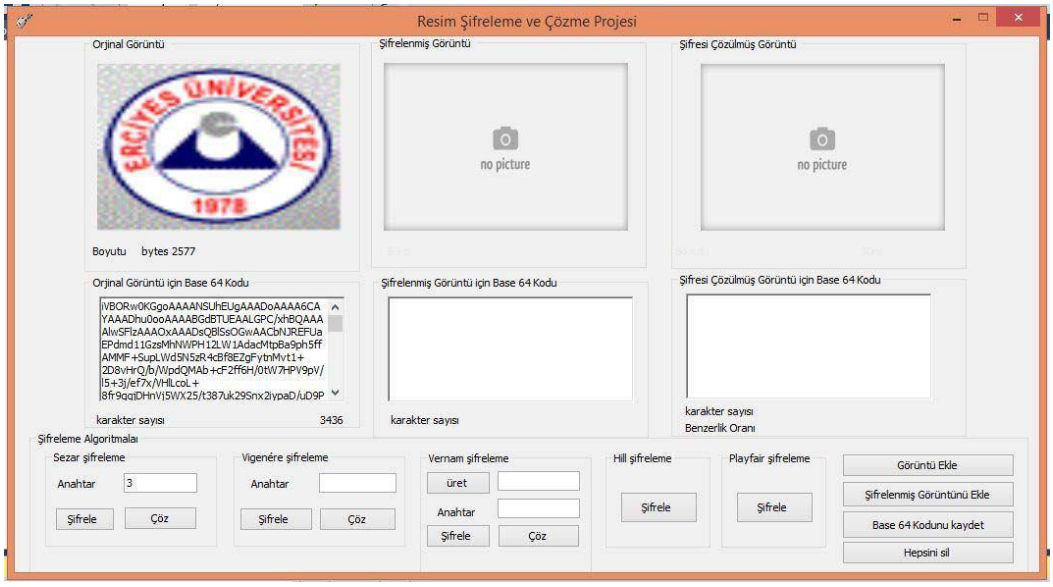
fi(d) = (d-ai) mod64… (4).

Vigenerealgoritmasında karakter silme veya ekleme olmadığı için de benzerlik oranı %100 görülmektedir.

Şekil 2.19.Vigenere algoritmasıyla şifre çözme aşaması.

2.8.3. Vernam Şifreleme

İkili sayı sistemine (0 ve 1) yer vermesi ve şifrelenmemiş metinin XOR (exclusive-or) eyleminden geçmesidir. Veri rastgele belirlenmiş ve kendisini tekrarlatmayan anahtarlar vasıtasıyla şifrelenir.



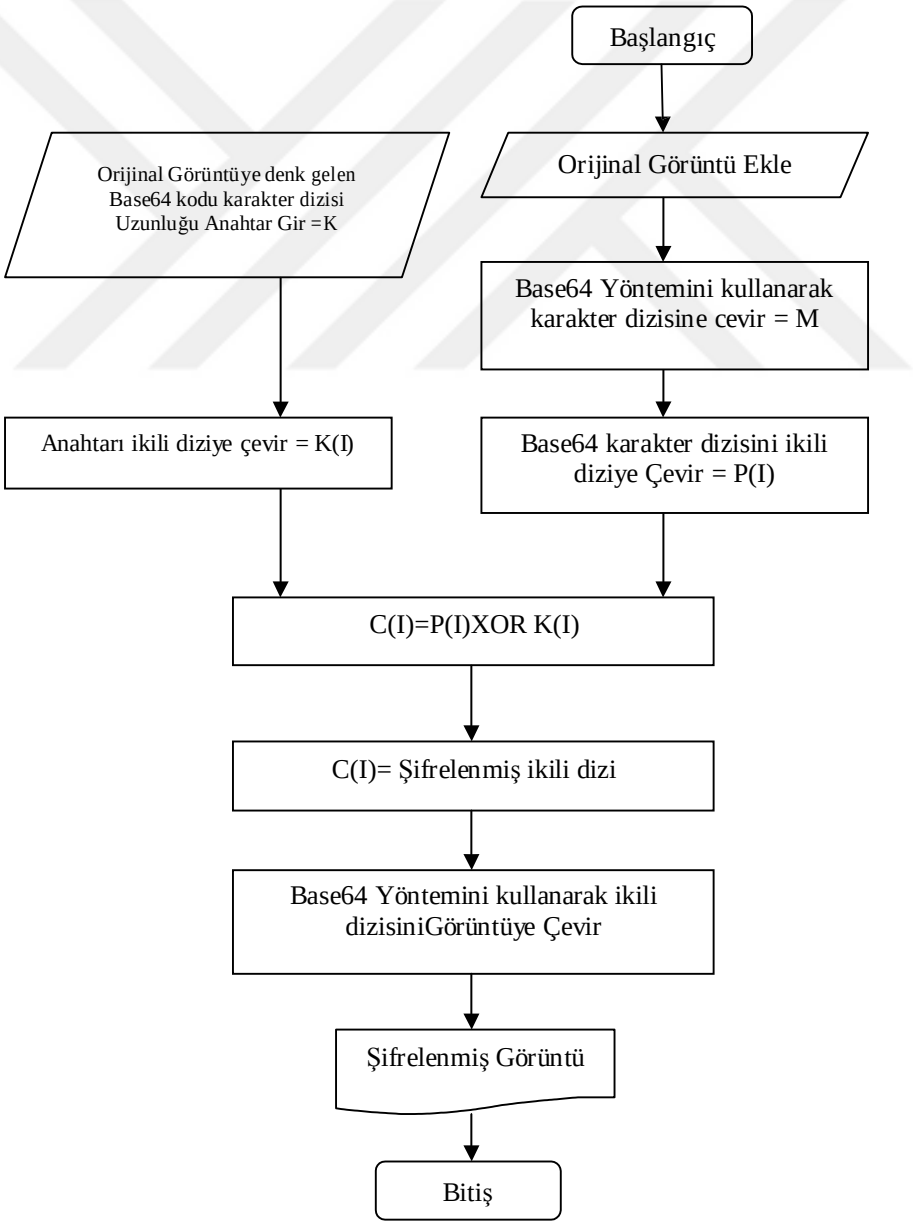
Şekil 2.20. Vernam algoritmasıyla orijinal görüntü için Base64 kodu işlemi.

Bu algoritmayla görüntü şifrelemek için Şekil 2.20'de görüldüğü gibi görüntü programın ara yüzüne çağrılarak Base64 kodu üretilir.

Vernam algoritmasında anahtar uzunluğu orijinal metin ile eşit olmalıdır yani görüntünün paralel Base64 kodu ile eşit uzunlukta olmalıdır. Başka bir deyişle karakter sayıları eşit olmalıdır.

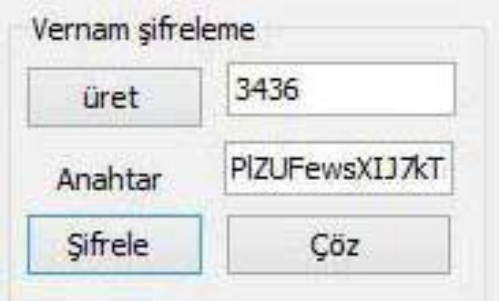
Görüntü yüklenip Base64 koduna çevrildikten sonra ve gereken uzunluktaki anahtar üretilince bu veriler ASCII Kodu tablosu ile(Şekil 2.5) ikili sisteme (0-1) çevrilir Base64 ve anahtar karakterleri arasında XOR işlemi vasıtasıyla gerçekleştirilir.

Şekil 2.21'deki gibi görüntü şifrelemek için Vernam Algoritmasının adımları şu şekilde olacaktır.



Şekil 2.21.Görüntü şifrelemek için Vernam Algoritmasının adımları.

Base64 kodu uzunluğuna eşit uzunlukta bir anahtar zor bulunur çünkü bir görüntüde 1000 karakter varsa 1000 karakterlik bir anahtar bulmak zor olduğu için görüntüye denk olan Base64 kodunun karakter sayısını yazarak üretme işleminden sonra gereken uzunluktaki bir anahtar otomatik üretilmiş olacaktır (Şekil 2.22) şifre çözülürken aynı anahtar kullanılacağından dolayı kaydedilmesi gereklidir.

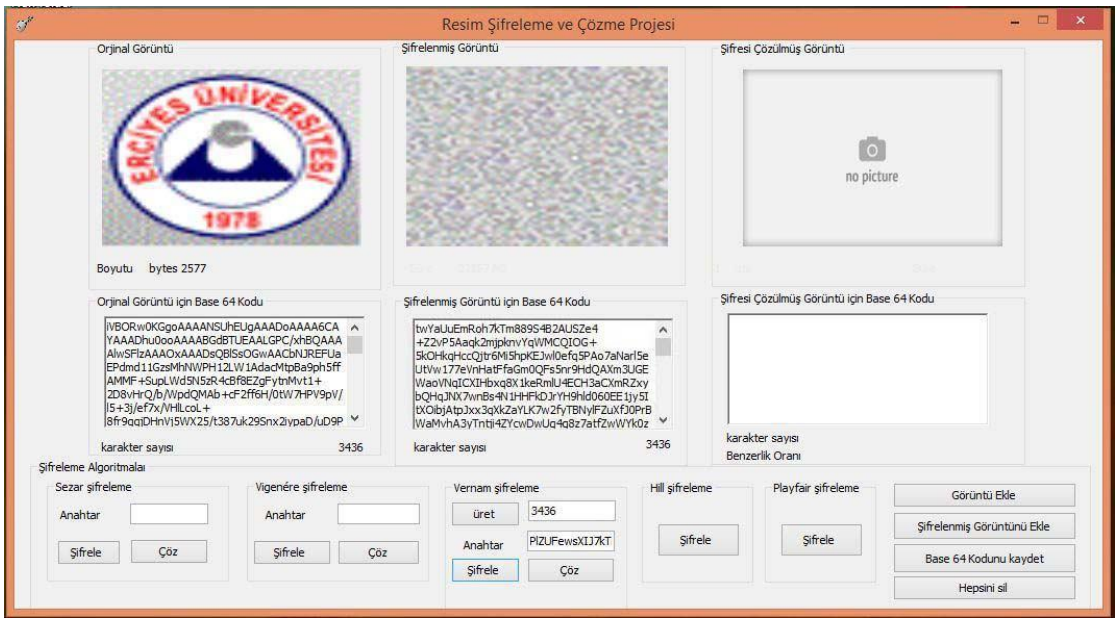


Şekil 2.22. Vernam algoritması için anahtar girme alanı.

Yukarıdaki şekilde görüldüğü gibi orijinal görüntünün Base64 kodundaki karakter sayısı 3436. Olduğunda programın üretmiş olduğu anahtardaki karakter sayısı da 3436 olmalıdır.

Denklem (5) kullanılarak görüntünün Baes64 kodu ve şifrelenmiş görüntü elde edilmiş olacaktır(Şekil 2.23).

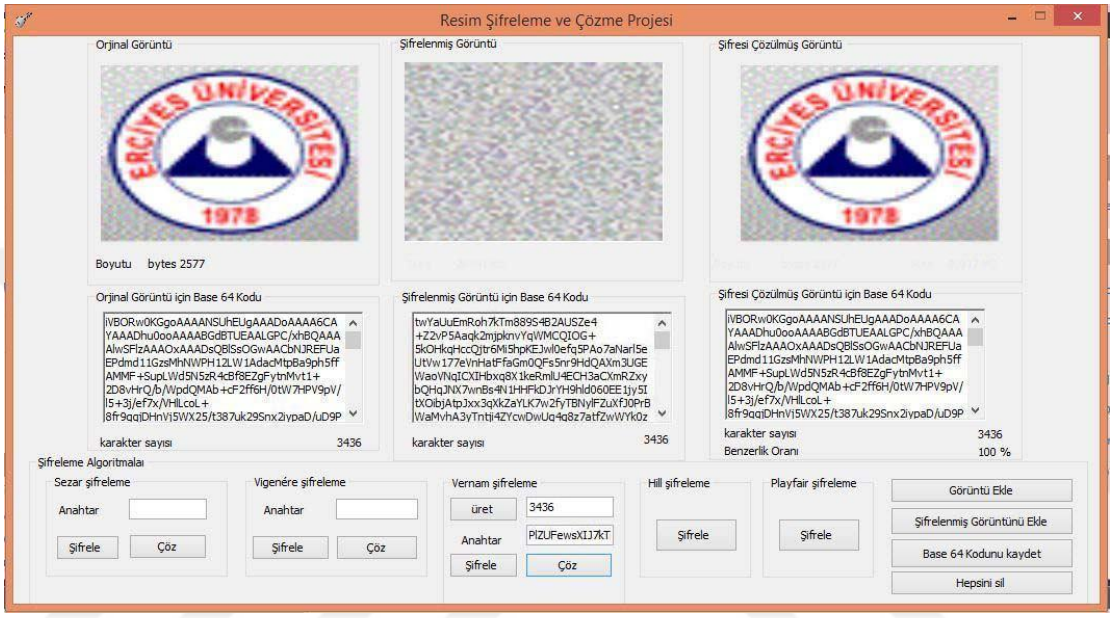
$C_i = P(i)$ XOR $K(i)$…..(5)



Şekil 2.23. Vernam algoritmasıyla görüntü şifreleme aşaması.

Orijinal görüntü elde edilmek istendiği zaman görüntü şifreleme işleminde kullanılan anahtarın aynısı kullanılarak şifrelenmiş Base64 kodu ve anahtar karakterleri arasında denklem (6) kullanılarak orijinal Base64 kodu ve orijinal görüntü elde edilir (Şekil 2.24).

Pi = Ci XOR K(i)…(6)



Şekil 2.24. Vernam algoritmasıyla şifre çözme aşaması.

Şekil 2.24'te görüldüğü gibi orijinal görüntü ile %100 oranında (benzerlik) elde edilir. Bu algoritma Sezar ve Vigenere algoritmalarına göre daha güçlüdür ancak anahtarın orijinal görüntü için elde edilmiş olan Base64 kodu karakter sayısına eşit olması gerektiği için uygun bulunmamaktadır.

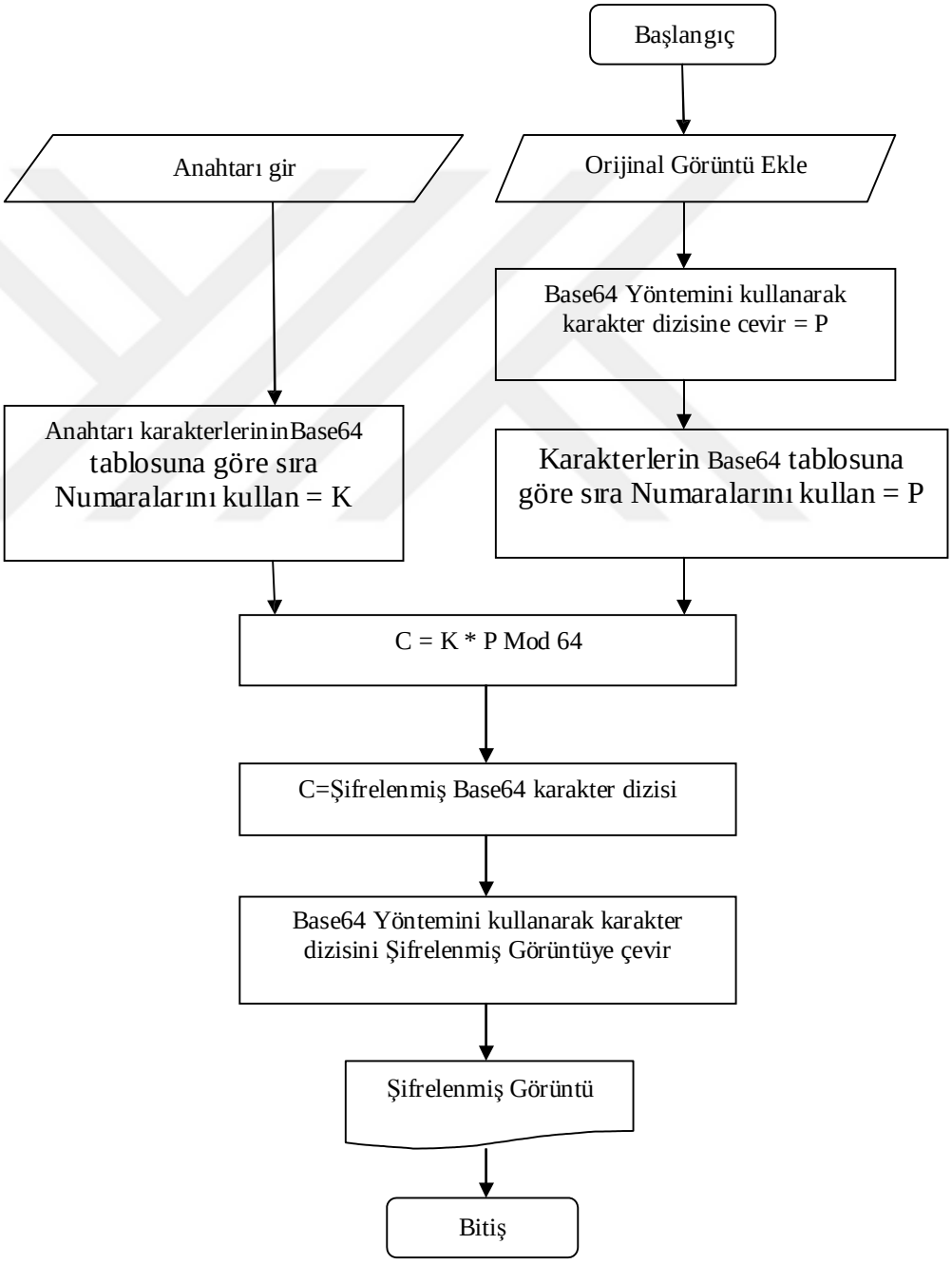
2.8.4. Hill Şifreleme

Bu yöntem bir blok şifrelemedir. Şifrelenmemiş metni bitişik ve aynı uzunluktaki bloklara bölme, her bloğu şifreleyerek şifreli metin bloklarına çevirme ve bu şifreli blokları şifreli metin çıktısı olarak gruplara ayırmaktır.

Bu algoritma sayısal değerlerle çalışmaktadır. Orijinal görüntünün Base64 koduna çevrildikten sonra karakterlerin Base64 tablosuna göre sayısal değerlerini alacaktır, aynı durum anahtar için de

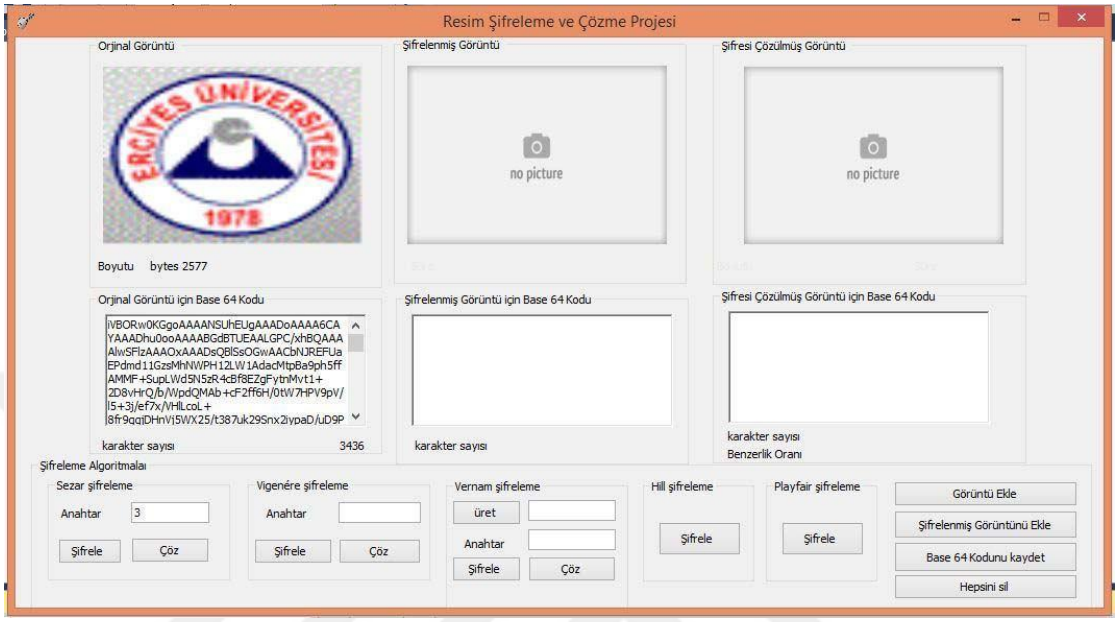
Daha sonra Base64 kodu dizisi gruplara ayrılacaktır. Bu gruplar: ikili, üçlü veya daha fazla elemandan oluşacaktır, kaç grup veya her grupta kaç eleman bulunması anahtar karakter sayısına göre belirlenecektir. Anahtar dizisinde sütunların sayısı Base64 kodu dizi satırlarıyla eşit durumda olmalıdır, Şekil 2.25'te belirtildiği şekilde denklem (7) kullanılarak şifremle işlemi yapılacaktır.

C = K * P mod 64… (7)



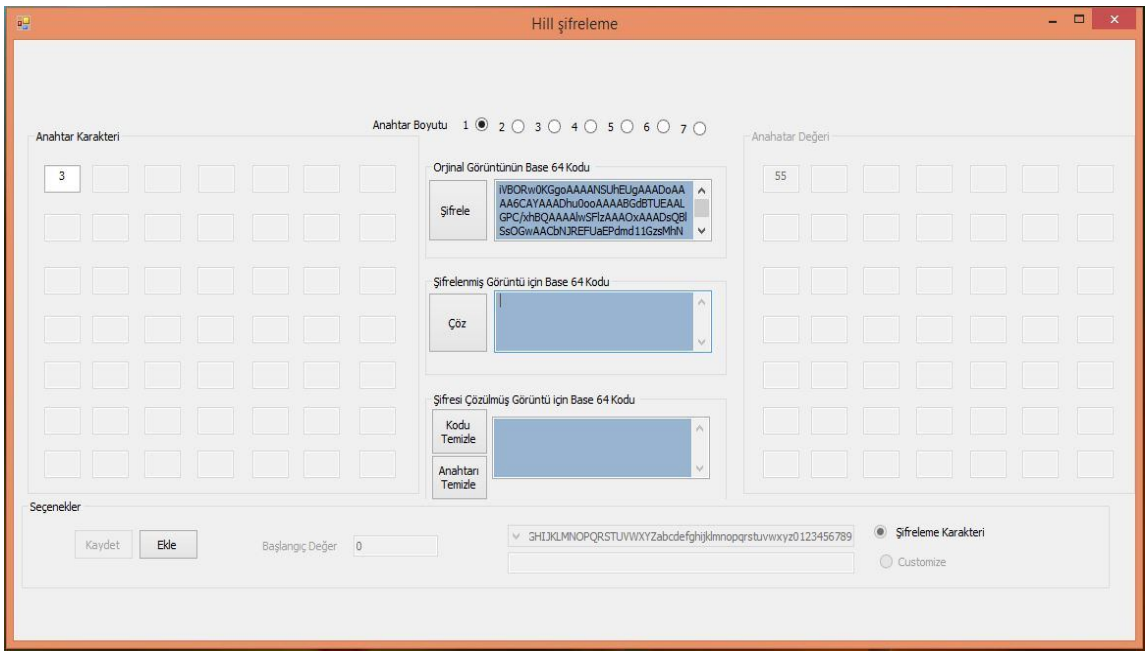
Şekil 2.25.Görüntü şifrelemek için Hill algoritmasının adımları.

Uygulamada bu algoritmayla görüntü şifrelemek için görüntü programın ara yüzüne çağrılarak görüntünün Base64 kodu otomatik olarak üretilecektir (Şekil 2.26).



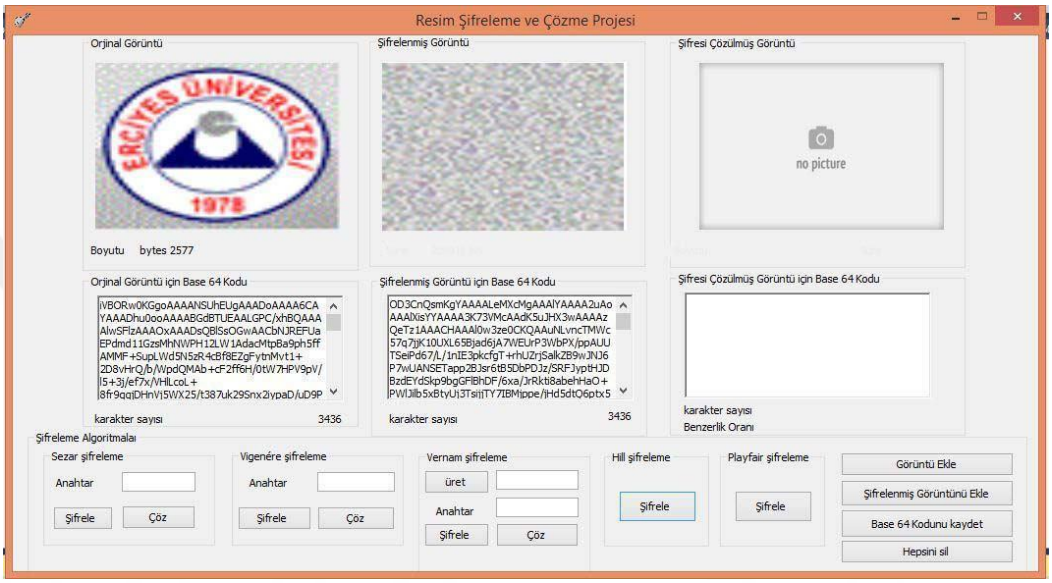
Şekil 2.26. Hill algoritmasıyla orijinal görüntü için Base64 kodu işlemi.

Hill Algoritması arayüzünde yer alan şifrelemeişlemiyle açılan yeni pencere bu algoritmanın detaylarını içermektedir (Şekil 2.27).



Şekil 2.27. Hill şifreleme penceresi.

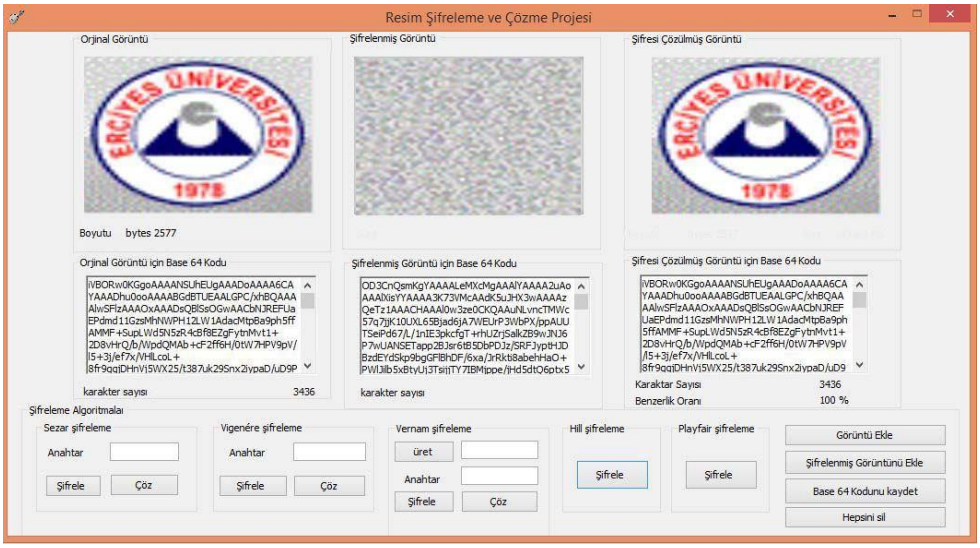
Anahtarı girdikten sonra ve daha önceden oluşturulmuş olan Base64 dizisi ve anahtar dizisi arasında çarpma işlemi gerçekleştirilip elde edilen sonuçta her eleman için mod64 yapılarak yeni bir dizi oluşturulur. Bu dizide yer alan sayısal değerlere denk delen Base64 tablosundan karakterlerle değiştirilir. Elde edilen karakter dizisi şifrelenmiş Base64 kodu yani şifrelenmiş görüntü olmaktadır (Şekil 2.28).



Şekil 2.28. Hill algoritmasıyla görüntü şifreleme aşaması.

Şifreleme işleminde kullanılan anahtarın tersialınır vedenklem (8) kullanılarakgörüntü deşifre edilmiş olur(Şekil 2.29).

$P = K^{-1} * C \bmod 64$ …..(8)



Şekil 2.29. Hill algoritmasıyla şifre çözme aşaması.

Şekil 2.29’da görüldüğü gibi, şifresi çözülmüş görüntünün benzerlik oranı %100’dür.Ayrıca bu algoritmanın anahtar kelimesinde benzer karakterler bulunması sakınca oluşturmamaktadır.

Bunların yanı sıra bu algoritmanın dezavantajı kullanılan bazı anahtarların tersi bulunmamaktadır. Bu nedenle bazen birden fazla anahtar denenmek zorunda kalınabilir.

2.8.5. Playfair Şifreleme

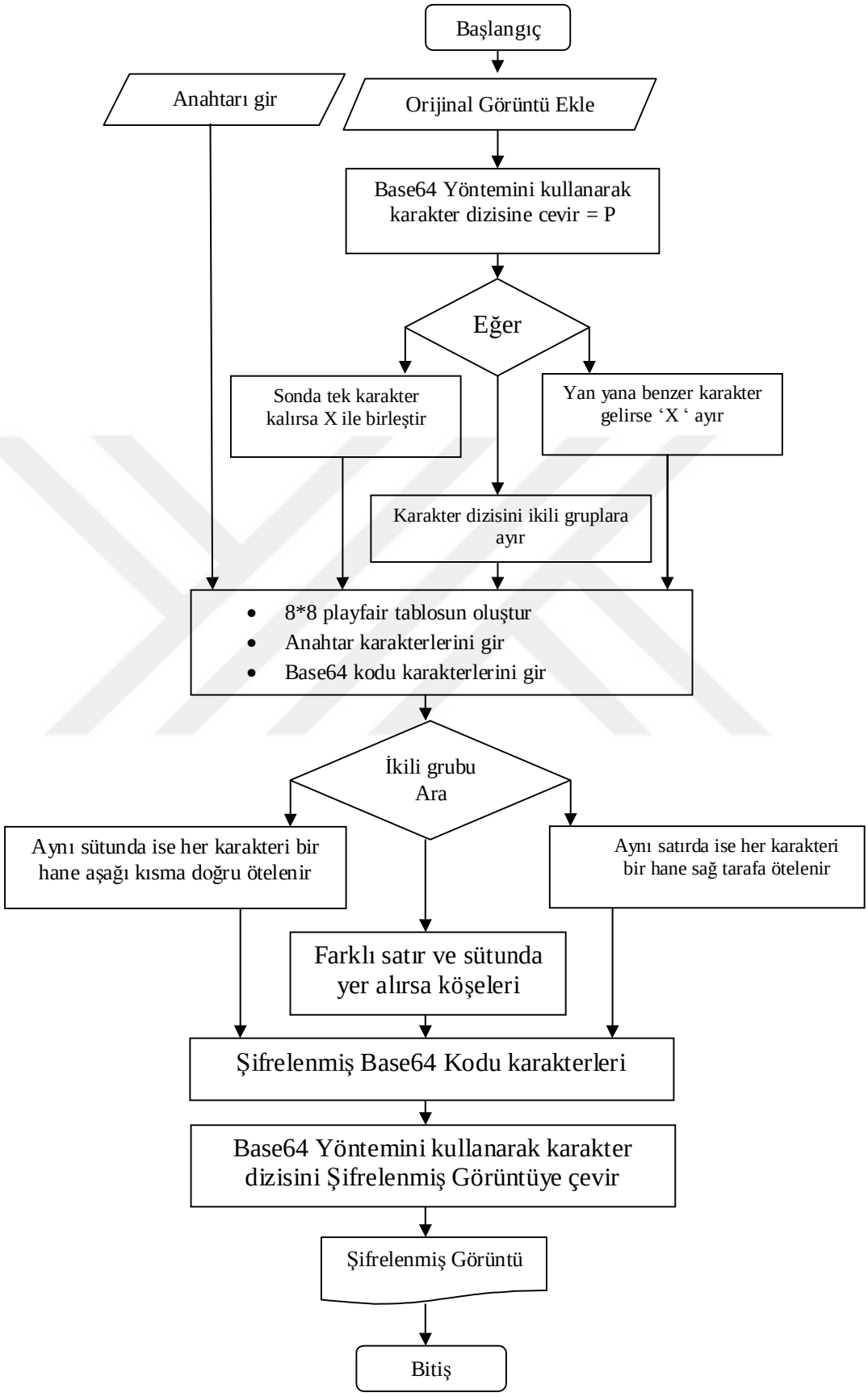
Bu yöntem yerine koyma yönteminin bir türüdür. Orijinal görüntünün Base64 kodunu ikili gruplara ayırarak aynı grupta benzer harfler yan yana geldiğinde X ile ayrılacaktır. Orijinal görüntünün Base64 kod karakter sayısı tekil ise sonuna X eklenerek ikili gruplara ayrılması için kolaylık sağlamış olur.

Daha önceden ikili gruba ayrılan karakterler şifreleme tablosunu kullanarak başka karakterlerle değiştirilerek şifreleme işi gerçekleştirilmiş olur. Aşağıdaki kurallar takip edilerek şifreleme işlemi gerçekleştirilir.

1. İkili grup aynı satırda ise her karakter bir hane sağ tarafa ötelenir.

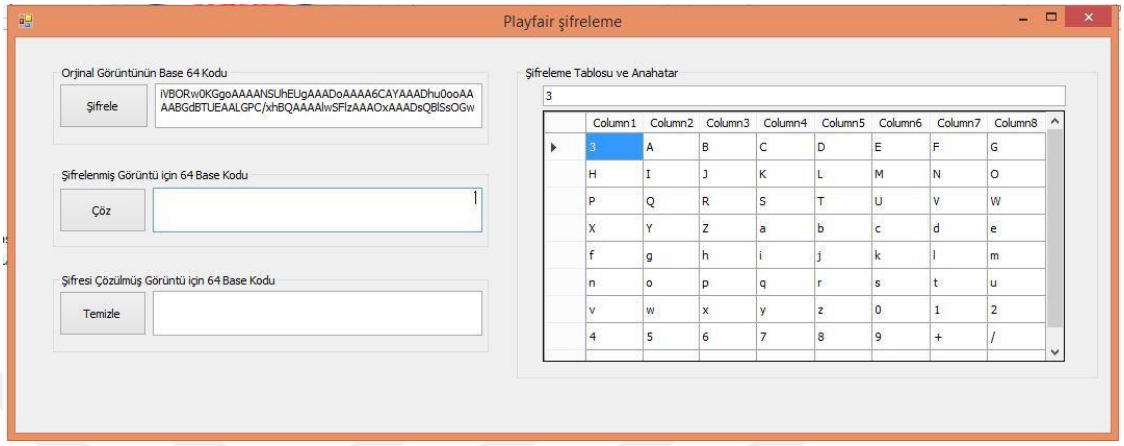
2. İkili grup aynı sütunda ise her karakter bir hane aşağı kısma doğru ötelenir.

3. İkili grup farklı satır ve sütunda yer alırsa köşeleri alınır.



Şekil 2.30.Görüntü şifrelemek için Playfair algoritmasının adımları.

Playfair Algoritması arayüzünde yer alan şifrelemeişlemiyle açılan yeni pencere bu algoritmanın detaylarını içermektedir (Şekil 2.31).

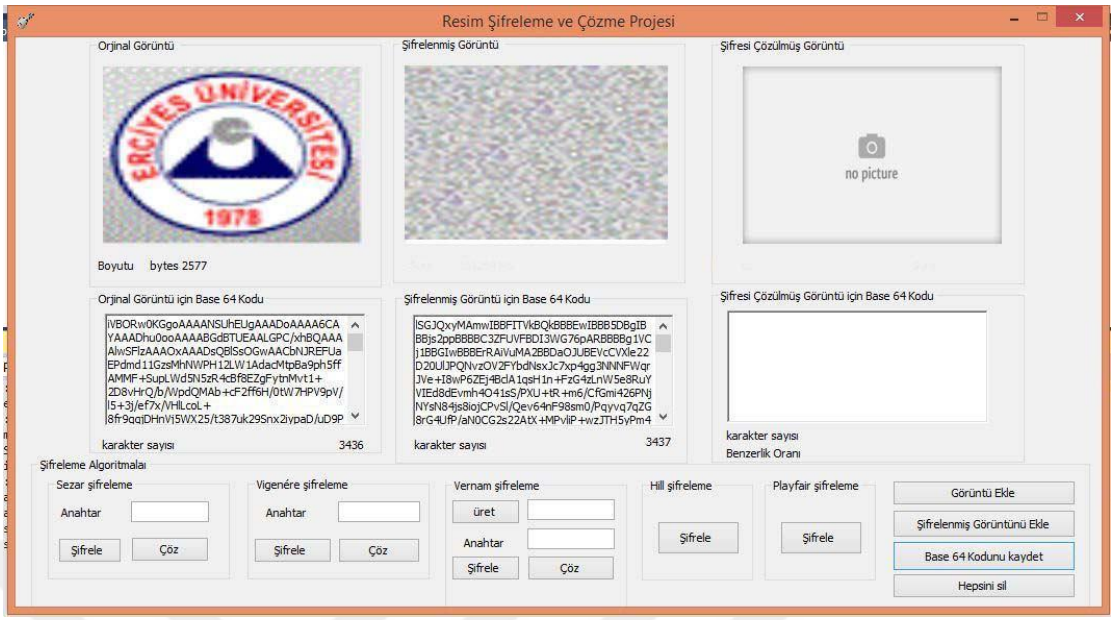


Şekil2.31. Playfair şifreleme penceresi

Programın ara yüzündeki sol tarafta Base64 kodu yer almaktadır. Sağ tarafta ise 8*8'lik bir tablo bulunmaktadır ve 64 karakter içermektedir. Bu 64 karakter A-Z, a-z, 0-9 +/ karakterlerini kapsamaktadır.

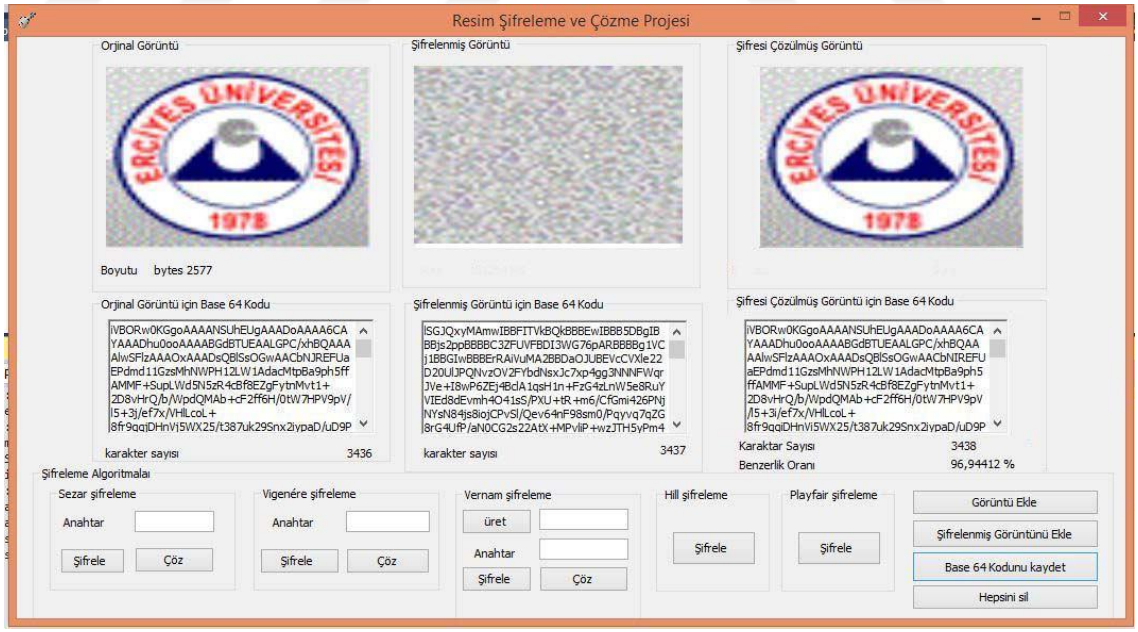
Anahtara ayrılan alana girilince karakterler bu tablonun ilk hanesinden başlayarak yerleştirilir ve tablo Base64 kod karakterleriyle sonuna kadar tekrara gitmeden doldurulur.

Şifreleme işlemiyleorijinal görüntünün Base64 kodu Şekil 2.30'ta belirtildiği şekilde uygulanır. Daha farklı bir karakter dizisi elde edilmiş olur ve şifrelenmiş görüntünün Base64 kodu oluşturulur(Şekil 2.32).



Şekil 2.32. Playfair algoritmasıyla görüntü şifreleme aşaması.

Şifrelenmiş görüntünün deşifre edilmesi için şifreleme işleminde kullanılan anahtar kullanılarak orijinal görüntü yeniden elde edilmiş olur.Şekil 2.33'te görüldüğü gibi.



Şekil 2.33. Playfair algoritmasıyla şifre çözme aşaması.

Bu algoritmada orijinal görüntüyle deşifre edilmiş görüntü arasında benzerlik oranı %96-99'dır. Nedeni ise benzer harflerin arasına x ve tekil karakter dizi son yerleştirilmesidir.

3. BÖLÜM

BULGULAR

Bu bölümde algoritmaların yapısal ve performans değerleri bakımından kıyaslaması yapılacaktır. Yapısal Değerler: sisteme uyum, esneklik, güven oranı, görüntüdeki veri kaybı veya artması kriterleridir.

Performans Değerleri: İşlem zamanı, bellek kullanımı ve işlemci kullanımı [15], gibi kriterlere dayanarak performans kıyaslama yapılacaktır. Yapılacak olan kıyaslamaların sonucunda algoritmaların avantaj ve dezavantajları belirlenecektir.Bu testleri gerçekleştirebilmek için farklı boyutta ve renklerde 4 görüntü ve hepsi için aynı anahtar kullanılacaktır, ancak Vernem algoritmasında anahtar otomatik üretilmektedir.

Tez çalışmasında, performans analizleri Windows 8.1işletim sistemine sahip AMD E-350 Processor 1.60 GHz dizüstü bilgisayar kullanılarak yapılmıştır. Klasik şifreleme algoritmalarının performans analizini gerçekleştirmek için MicrosoftVisual Studio 2015 C# ortamında hazırlanan yazılım kullanılmıştır.

Performans analizi iç ve dış olmak üzere farklılık arz edebilmektedir. İçve dış faktörler Tablo3.1'de görülmektedir [32]:

Tablo 3.1. Performans analizini etkileyen faktörler.

İç Faktörler	Dış Faktörler
Çalıştırmak için gereken zaman	Girdi verisinin büyüklüğü
Çalıştırmak için gereken bellek alanı	Bilgisayarın hızı
	Derleyicinin kalitesi

3.1. Şifreleme Algoritmaların Kıyaslaması

Şifreleme algoritmalarının performans analizi genel olarak aşağıda belirtilen kriterler üzerinden yapılır [31, 32, 33]:

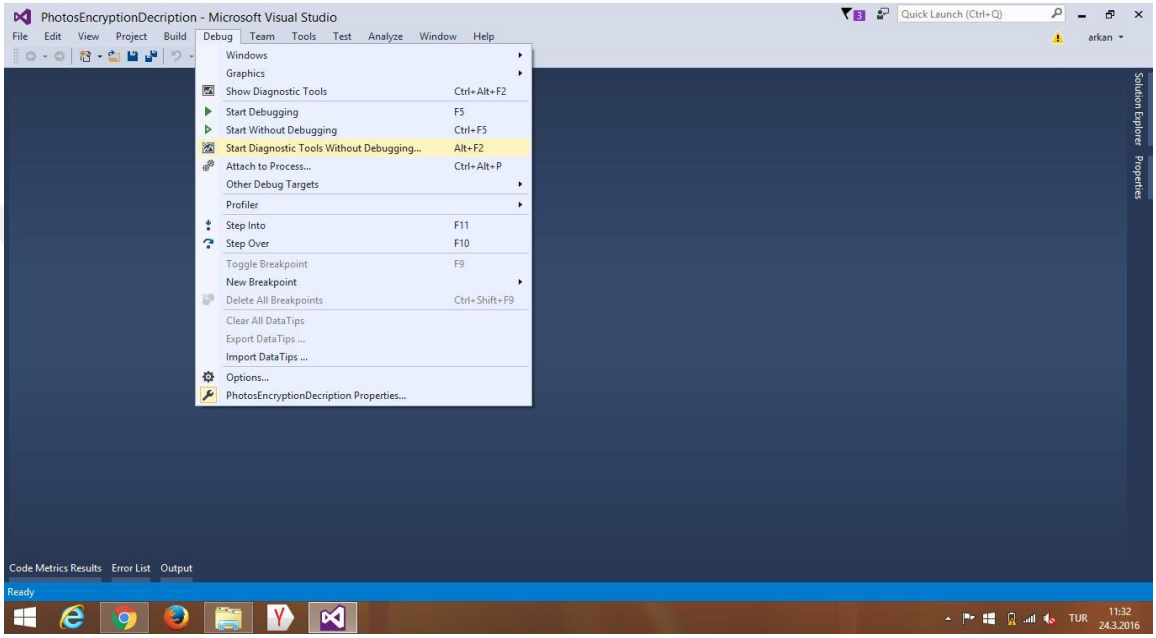
1. Şifreleme sistemin kırılabilme süresinin uzunluğu.
2. Şifreleme ve çözme işlemlerine harcanan zaman (Zaman Karmaşıklığı).
3. Şifreleme ve çözme işleminde ihtiyaç duyulan bellek miktarı (BellekKarmaşıklığı).
4. Bu algoritmaya dayalı şifreleme uygulamalarının esnekliği.
5. Bu uygulamaların dağıtımındaki kolaylık ya da algoritmaların standarthale getirilebilmesi.
6. Algoritmanın kurulacak sisteme uygunluğu.

Performans analizleri, sistem sonuçlarının belirlenen ve kabul edilebilecek olan zaman dilimi içerisinde, üretilebildiğini değerlendirebilmek için yapılananalizlerdir. Performans analizi, yapılan işlem zamanlarının hesaplanması ve sistemin dar boğazlarının belirlenmesini sağlar [34].

Algoritmanın özellikle işlem zamanı boyutu için gerçekleştirilentestindoğru sonuç verebilmesi ilgili algoritmanın en kötü durumu için performans analizi yapılmasını gerektirir. Bu durumda o algoritmanın doğru performansı büyükmiktarda tespit edilmiş olur [35].

Performans ölçütlerinden bir başka algoritma çalışırken gerekli olan bellekgereksinimidir. Bellekte değişkenler, ara sonuçlar ve sonuçlar, bir takımsabitler ve algoritma bilgileri yer almaktadır. İşlem sırasında maksimum bellek kullanımı tespit edildikten sonra ilgili algoritmanın gereken bellek değeri elde edilir. Performansı yüksek olan bir algoritmanın bellek kullanımı az olmalıdır. Optimal bir algoritma hem hızlı olmalı ve de az bellek tüketmelidir [35].

Öncelikle Build ya da Analyze menüsünden Run Code Analysis seçeneği photos Encryption Decrytion.exe projesinde çalıştırılmıştır. Daha verimli sonuçlar elde edebilmek için programın verdiği uyarılar tespit edilmiş ve bir kısmı düzeltilebilmiştir. Programın performans değerleri Debug menüsündeki Start Diagnostic tools without Debugging seçeneği kullanılarak belirlenmiştir.(Şekil 3.1).



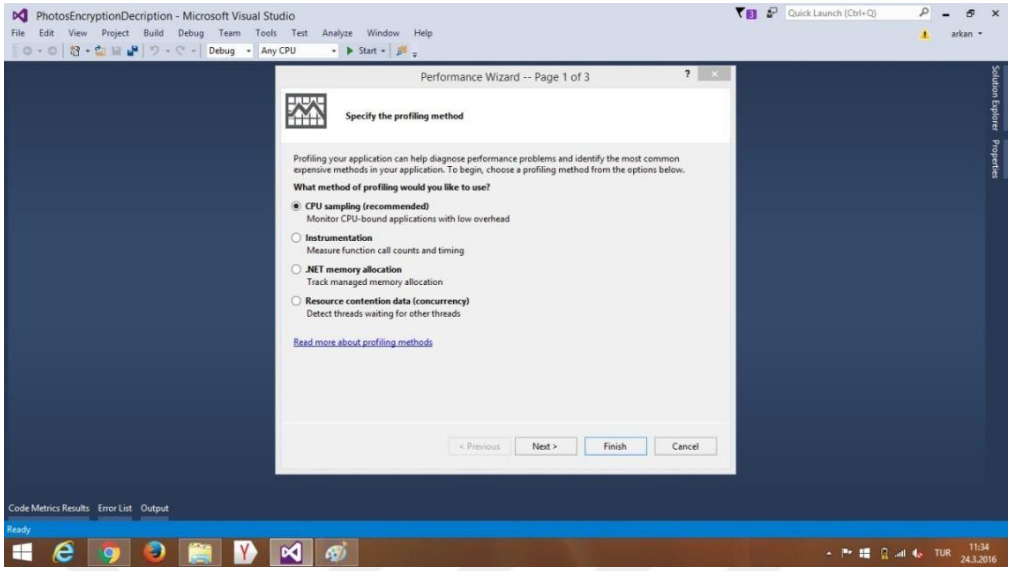
Şekil 3.1. Debug penceresi.

Bu aşamada ekrana değişik dört seçenek gelmektedir.(Şekil 3.2). CPU Sampling seçeneği ile, CPU üzerindeki işlem yükü belirlenmiştir.

Instrumentation seçeneği ile harcanan zaman ve çağrılan fonksiyon sayısı belirlenebilir. Bu bölümde işlem süreleri incelenmiştir.

NET Memory Allocation seçeneği ile bellek kullanımı belirlenmiştir.

Concurrency seçeneği thread uygulamaları için kullanılabilir. Projede thread uygulaması yer almadığından dolayı bu bölümle ilgili herhangi inceleme yapılmamıştır.



Şekil 3.2. Performans wizard penceresi.

Beş farklı algoritma için şifreleme ve çözme işlemlerinin performans analizi ve değerleri bulmak için aynı adımlar kullanılacaktır. Ancak Sezar algoritması kullanılarak şifreleme işlemi için performans analizi adım adım görseller vasıtasıyla açıklanacaktırtest işleminde 2.5 KB'lık (Erciye Logo) görüntüsü kullanılacaktır. Aynı adımlar diğer algoritmalar için tekrarlanması ve sonuçlar Tablo 3.3,3.4,3.5 ve 3.6'daverilmiştir.

Performans analizi uygulamalarında 4 farklı görüntüye yer verilmiştir. Bu görüntüler örnek olması amacıyla seçilmiştir. Seçilmesinde herhangi bir özel durum söz konusu değildir. Yapılan incelemeler sadece şifreleme ve çözme işlemi için sunulmuştur. Çalışmada kullanılan görüntüler Şekil 3.3'te verilmiştir.

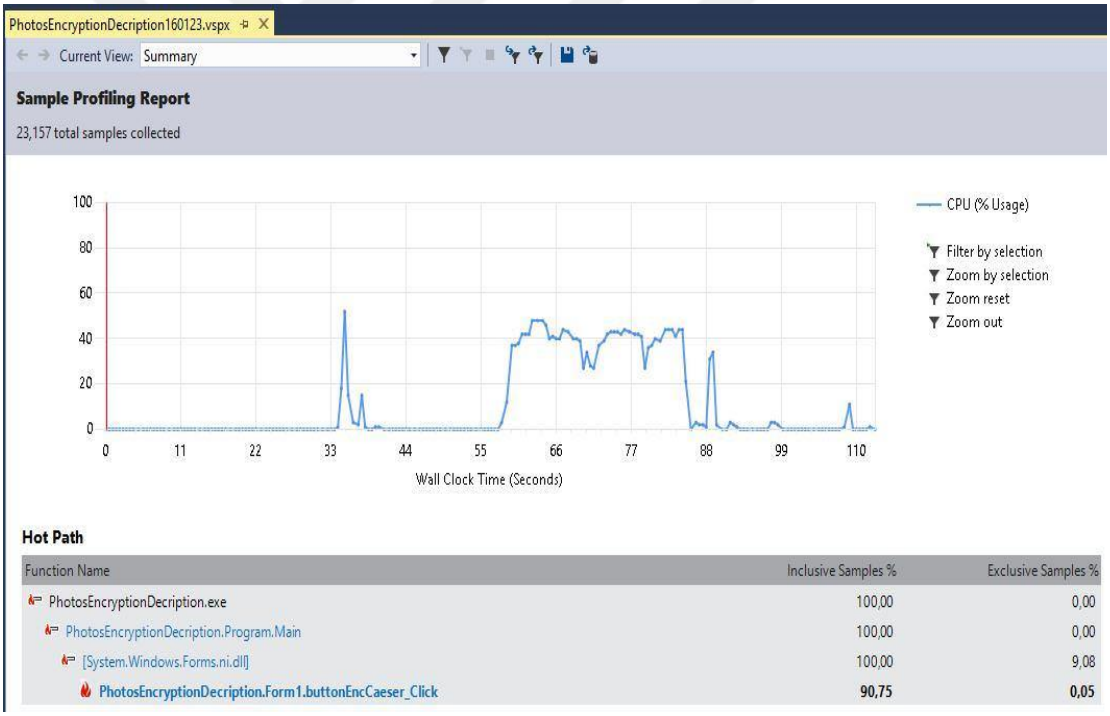
Şekil 3.1. Çalışmada kullanılan görüntüler.

3.2. Sezar Şifreleme Algoritması Kullanılarak Görüntü Şifreleme İşlemi için Performans Değerleri

CPU kullanımı, bellek kullanımı ve harcananişlem zamanı açısından performans analizleri yapılmıştır. Performans analizi, değişik boyuttaki görüntüler için yapılmış olup görüntünün boyut ve rengine göre Base64 kodundaki karakter sayısı değişiklik göstermektedir. Karakter sayısının performans değerlerini etkilediği görülmüştür.

3.2.1. CPU Örnekleme (Sampling)Verileri

Şekil 3.4'te CPU Örneklemeseçeneğine ait sonuçlar görülmektedir:



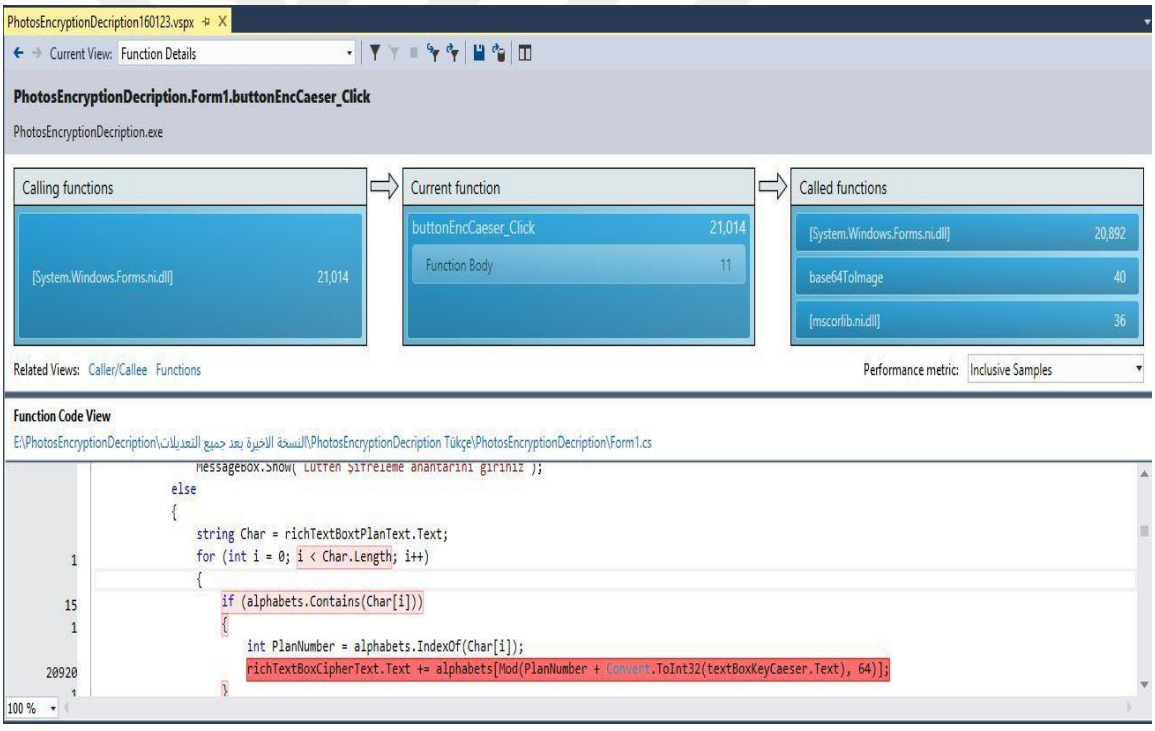
Şekil 3.4. "Erciyes Logo" 2.5 KB'lıkgörüntünün şifrelenmesine ait CPU grafiği. Toplam

23.157 (sınıf,fonksiyon) için CPU kullanımı Şekil 3.4'teki grafikte görülmektedir. CPU kullanımı program ilk çalıştırıldığında 0 dan başlayıp , daha sonra şifreleme işlemlerinde kullanılan fonksiyon ve sınıflara göre değişkenlik göstermektedir.

Anlık görünüm (Current View)kısmında farklı seçenekler bulunmaktadır. Summary, CPU ile ilgili kullanım bilgilerinin özet şeklinde ve en genel haliyle görülmesini sağlar. Hot Path bölümündekiInclusive Samples yüzdesi yüksek olan sınıflar sistemi en çok zorlayan sınıflardır. Exclusive Samples bölümünde ise performans darboğazı yaratan fonksiyonlar en yüksek yüzdeye sahiptir.

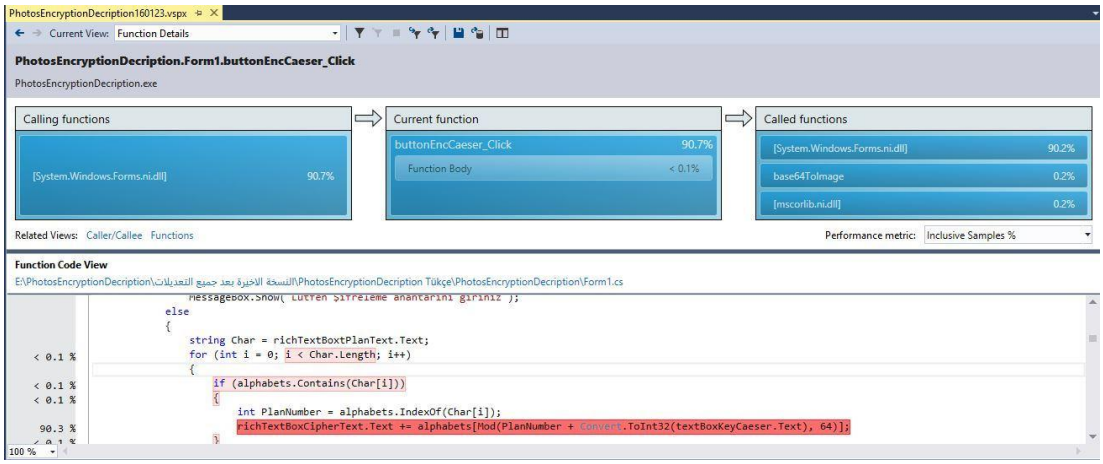
Daha detaylı görebilmek için yanlarında ateş simgesi bulunan yazıların üzerine tıklamak yeterlidir ya da Current View kısmında Function Details seçeneği ile detaylara ulaşmak mümkündür.

Şekil 3.5'teSezar Şifreleme sınıfına ait fonksiyonlar görülmektedir:



Şekil 3.5.Sezarşifreleme sınıfında darboğaz oluşturan fonksiyonlar.

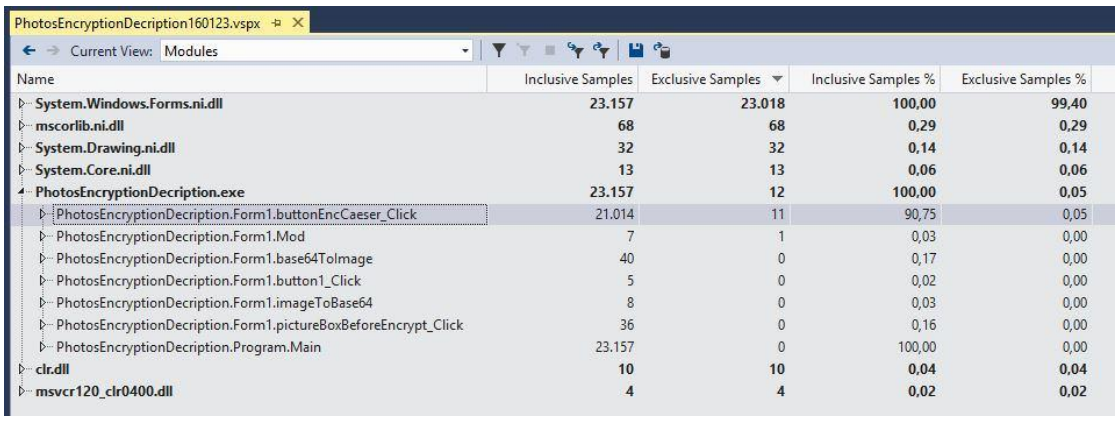
Şekil 3.5'te görüldüğü üzere Sezar Şifreleme sınıfında toplam 21 fonksiyon çağırılmaktadır. Çağırılan fonksiyonların yüzdelik dilimleri Şekil 3.6'da görülmektedir. Bu kısma ulaşmak için Performance metric kutusundan Inclusive Samples % seçimini yapmak yeterlidir.



Şekil 3.6.Sezar şifreleme sınıfında darboğaz oluşturan fonksiyonların yüzdeleri.

Şekil 3.6'da görülen değerlere göre sistemde en büyük darboğazı oluşturan fonksiyon Sezar Şifreleme sınıfında bulunmaktadır. Şekil 3.6'dabuttonEncCaser_Click olayı şifreleme algoritmasının çalıştırılmasını sağlayan olaydır. Şifreleme algoritmasının CPU kullanımı, geliştirilen demo programın toplam CPU kullanımının %90.7'ini oluşturmaktadır.

Current View kısmında Modules seçeneği seçilirse sistemi zorlayan sınıfların hangileri olduğu ve fonksiyonların hangi satırlarda yer aldığı bilgisine ulaşılabilir. Şekil 3.7'de Modules seçeneği ile daha detaylı olarak elde edilen bilgiler görülmektedir:



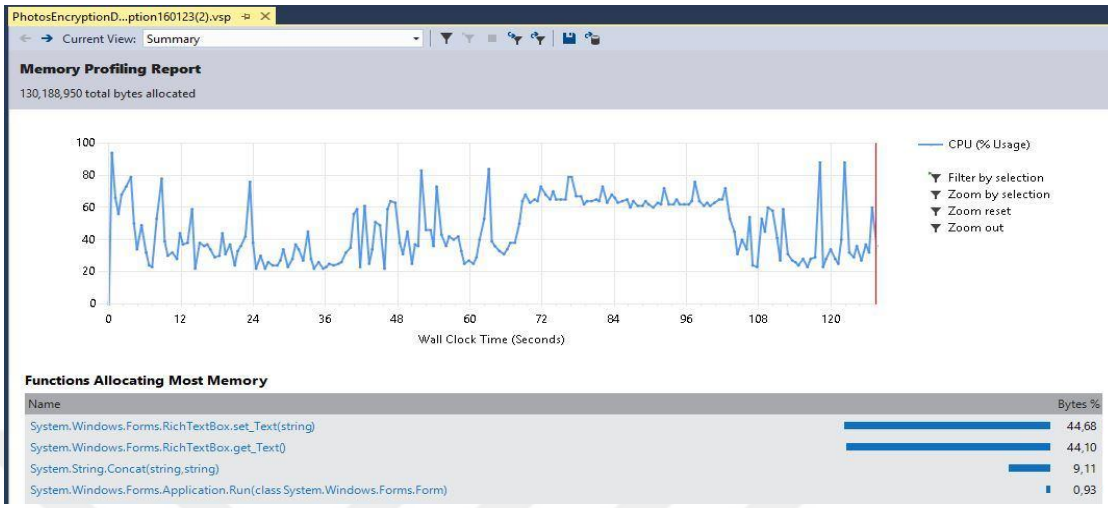
Name	Inclusive Samples	Exclusive Samples	Inclusive Samples %	Exclusive Samples %
System.Windows.Forms.ni.dll	**23.157**	**23.018**	**100,00**	**99,40**
mscorlib.ni.dll	**68**	**68**	**0,29**	**0,29**
System.Drawing.ni.dll	**32**	**32**	**0,14**	**0,14**
System.Core.ni.dll	**13**	**13**	**0,06**	**0,06**
PhotosEncryptionDecription.exe	**23.157**	**12**	**100,00**	**0,05**
PhotosEncryptionDecription.Form1.buttonEncCaeser_Click	21.014	11	90,75	0,05
PhotosEncryptionDecription.Form1.Mod	7	1	0,03	0,00
PhotosEncryptionDecription.Form1.base64ToImage	40	0	0,17	0,00
PhotosEncryptionDecription.Form1.button1_Click	5	0	0,02	0,00
PhotosEncryptionDecription.Form1.imageToBase64	8	0	0,03	0,00
PhotosEncryptionDecription.Form1.pictureBoxBeforeEncrypt_Click	36	0	0,16	0,00
PhotosEncryptionDecription.Program.Main	23.157	0	100,00	0,00
clr.dll	**10**	**10**	**0,04**	**0,04**
msvcr120_clr0400.dll	**4**	**4**	**0,02**	**0,02**

Şekil 3.7.Sınıf ve fonksiyonların bulundukları satırları gösteren CPUkullanımı.

Şekil 3.7'de çerçeve içinde görülen alan şifreleme algoritmasının çalıştırıldığı button EncCaser_Click olayına ait CPU kullanımını göstermektedir. Geliştirilen algoritmanın, demo programın CPU kullanımının sadece %90.75≈%91'ini kullandığı tespit edilmiştir Diğer fonksiyon ve sınıfların değerleri genel işlemci değerine göredir.

3.2.2. Bellek Kullanımı (Net Memory allocations)Verileri

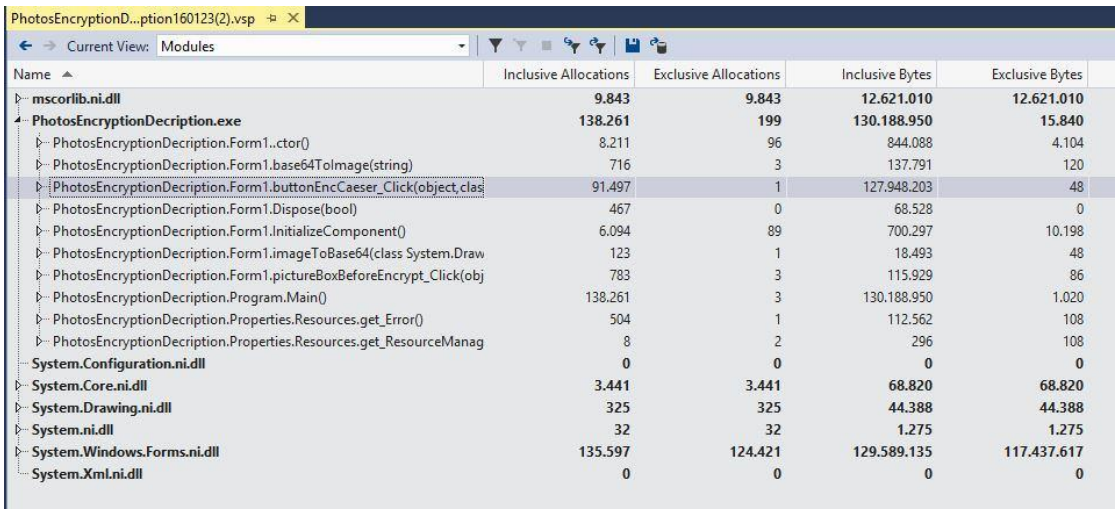
Şekil 3.8'de Sezar Algoritması için bellek kullanımına ait grafik görülmektedir:



Şekil 3.8."Erciyes Logo" 2.5 KB'lıkgörüntünün şifrelenmesine ait bellek kullanımı.

Gerçekleştiren PhotosEncryptionsDecription programı toplamda 130.188.95 byte miktarında bellek kullanmıştır.

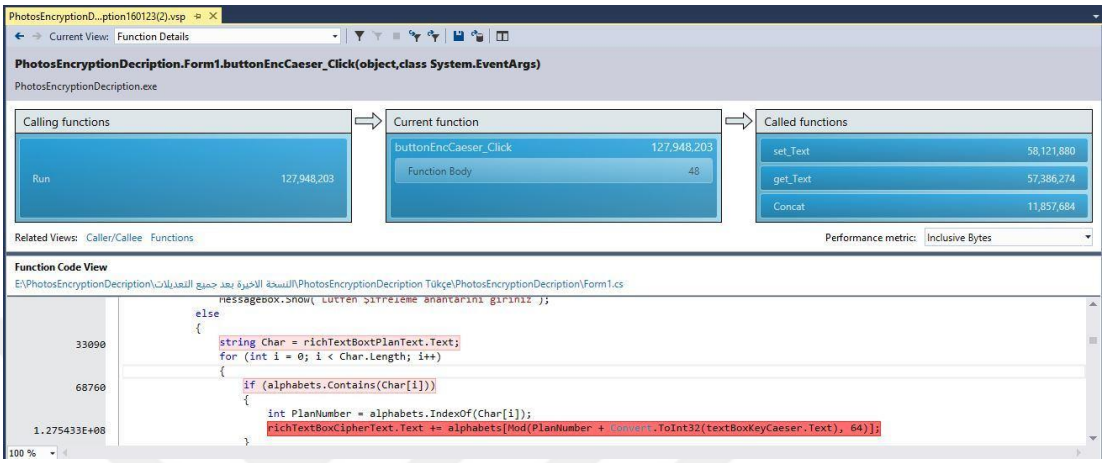
Modules seçeneği ile fonksiyon ve sınıfların bellek kullanımı ile ilgili detaylı bilgiye ulaşılabilir. Şekil3.9'da Modules seçeneği ile bellek kullanımına ait daha detaylı olarak elde edilen bilgiler görülmektedir:



Şekil 3.9. Fonksiyon ve sınıfların bellek kullanımına ait detaylar.

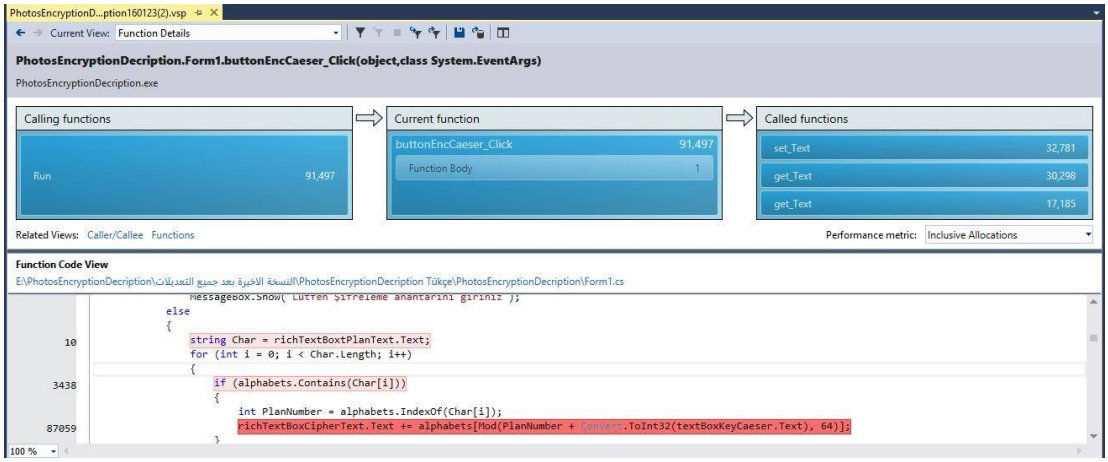
Current View kısmında Functions ya da Allocation seçilirse tüm fonksiyonların tek tek görüntülendiği daha detaylı bilgi ekranına ulaşılabilir. Şekil 3.10'da arka planı koyu görülen değerler şifreleme algoritmasına aittir. Şifreleme işlemleri button

EncCaser_Click olayında gerçekleşmektedir. PhotosEncryptionsDecription programı 130MByte'lık bellek alanı kullanırken bu kullanımın sadece 127 MByte'lık kısmı şifreleme işlemlerinde kullanılmaktadır.



Şekil 3.10. Sezar algoritması için bayt cinsinden bellek kullanımı.

Şekil 3.10'de görüldüğü üzere byte cinsinden incelendiğinde Photos Encryptions Decription programı ait fonksiyonlar toplam bellek kullanımının 127.948.203'ına sahiptir.Şifreleme algoritmasına ait bellek verileri (button EncCaser_Click). Şekil 3.11'de yüzde görülmektedir:



Şekil 3.11.Sezar algoritması için bellek kullanımın yüzdelik değerleri.

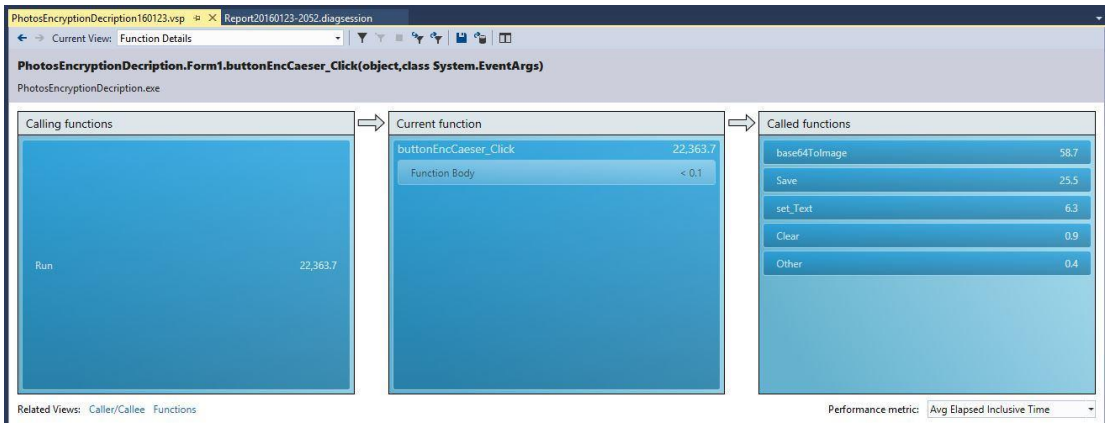
Bu değerlere göre, byte cinsinden incelendiğinde şifrelemealgoritmasının kullandığı bellek, programın toplam bellek kullanımının%91.497 'ini oluşturmaktadır.

3.2.3. İşlem Zamanı (Instrumentation)Verileri

Şekil 3.12'de işlem zamanına ait grafiği göstermektedir. Erciyes Logo Görüntüsününşifrelenmesi için geçen işlem zamanı süresince algoritmanın içermiş olduğu işlemlere göre CPU kullanımında dalgalanmalar oluşmaktadır.

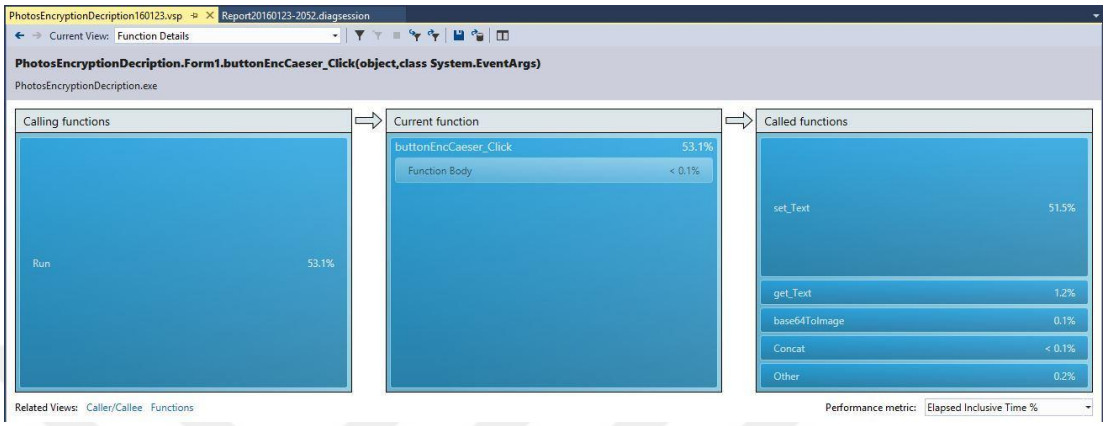
Şekil 3.12. Sezar şifreleme algoritmanın işlem zamanı grafiği.

Şekil 3.12'degörüldüğü gibi Sezar Şifreleme algoritmasınınşifreleme işlemini tamamlama süresi 42.1 saniye olarak tespit edilmiştir. Bu süre, ara yüzün ekrana yüklenmesinden kapatılmasına kadar geçen toplam süredir.Button EncCaser_Click ait detaylı bilgiler Şekil 3.13 ve Şekil 3.14'te görülmektedir:



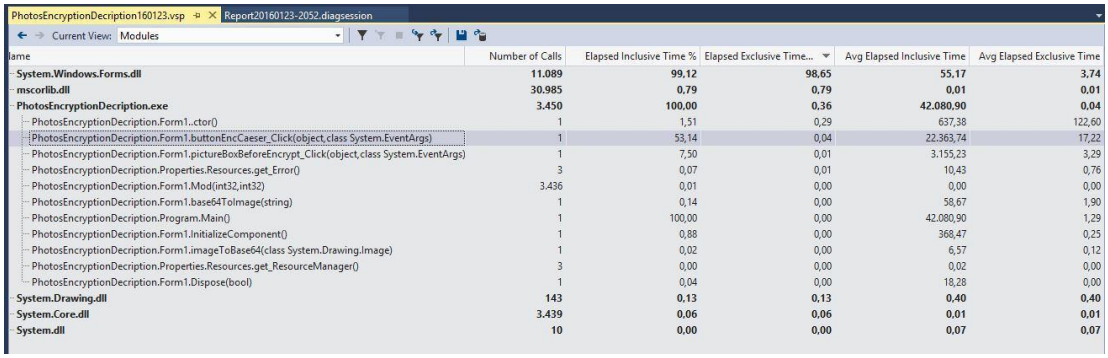
Şekil 3.13.Sezar algoritması için gerekli işlem zamanı.

buttonEncCaser_Click toplam işlem zamanının %53.1'üne sahiptir. Dolayısıyla şifreleme algoritmasının işlem zamanı %53.1 olarak belirlenmiştir. Şekil 3.13'tebutton EncCaser_Click olayına ait fonksiyon ve olayları bayt cinsinden göstermektedir:



Şekil 3.14.Sezar algoritması için işlem zamanın yüzdelik değerleri.

Bu işlemlerin yüzdelik değerleri Şekil 3.14'te görüldüğü gibi tespit edilmiştir.Resim 3.15'te şifreleme işlemlerine ait (button EncCaser_Click) işlem zamanı değerleri milisaniye ve yüzde olarak görülmektedir.



lame	Number of Calls	Elapsed Inclusive Time %	Elapsed Exclusive Time...	Avg Elapsed Inclusive Time	Avg Elapsed Exclusive Time
System.Windows.Forms.dll	**11.089**	**99,12**	**98,65**	**55,17**	**3,74**
mscorlib.dll	**30.985**	**0,79**	**0,79**	**0,01**	**0,01**
PhotosEncryptionDecription.exe	**3.450**	**100,00**	**0,36**	**42.080,90**	**0,04**
PhotosEncryptionDecription.Form1..ctor()	1	1,51	0,29	637,38	122,60
PhotosEncryptionDecription.Form1.buttonEncCaeser_Click(object,class System.EventArgs)	1	53,14	0,04	22.363,74	17,22
PhotosEncryptionDecription.Form1.pictureBoxBeforeEncrypt_Click(object,class System.EventArgs)	1	7,50	0,01	3.155,23	3,29
PhotosEncryptionDecription.Properties.Resources.get_Error()	3	0,07	0,01	10,43	0,76
PhotosEncryptionDecription.Form1.Mod(int32,int32)	3.436	0,01	0,00	0,00	0,00
PhotosEncryptionDecription.Form1.base64ToImage(string)	1	0,14	0,00	58,67	1,90
PhotosEncryptionDecription.Program.Main()	1	100,00	0,00	42.080,90	1,29
PhotosEncryptionDecription.Form1.InitializeComponent()	1	0,88	0,00	368,47	0,25
PhotosEncryptionDecription.Form1.imageToBase64(class System.Drawing.Image)	1	0,02	0,00	6,57	0,12
PhotosEncryptionDecription.Properties.Resources.get_ResourceManager()	3	0,00	0,00	0,02	0,00
PhotosEncryptionDecription.Form1.Dispose(bool)	1	0,04	0,00	18,28	0,00
System.Drawing.dll	**143**	**0,13**	**0,13**	**0,40**	**0,40**
System.Core.dll	**3.439**	**0,06**	**0,06**	**0,01**	**0,01**
System.dll	**10**	**0,00**	**0,00**	**0,07**	**0,07**

Şekil 3.15. Sezar algoritmasının işlem zamanı için milisaniye ve yüzde değerleri. Şekil

3.15'te görülen ilk üç sütun işlem zamanını milisaniye olarak göstermektedir. Son iki sütun ise yüzdelik değerleri vermektedir. Şekil3.15'ten görüleceği üzere algoritmanın işlem zamanı 53.14 milisaniye olarak tespit edilmiştir. Yüzdelik değer olarak, görüldüğü gibi toplam işlem zamanının %22.36'sı şifreleme algoritmasına aittir. 53.14 milisaniyelik süre programın toplam işlem zamanının %22.36'sını oluşturmaktadır.

Tablo3.2'de Sezar şifreleme algoritması için görüntü şifreleme işlemine ait performans analiz değerlerinden elde edilen değerler (işlem zamanı, bellek kullanımı, CPU kullanımı) yer almaktadır.

Tablo 3.2. Sezar şifreleme algoritması için görüntü şifreleme işlemine ait performans analiz değerlerinden.

Algoritma	İşlem Zamanı (MS)	RAM Kullanımı (Byte)	CPU Kullanımı (%)
Sezar	22.36	127.94	90.75

Tablo3.2'de yer alan veriler yalnız Sezar algoritmasına ait değerlerdir. Aynı adımları kullanarak diğer algoritmalar için farklı boyuttaki görüntüler için elde edilmiş performans analiz değerleri Tablo 3.3,3.4,3.5 ve 3.6'dayer almaktadır.

3.3. Performans Değerlerinin Karşılaştırılması

Tablo 3.3,3.4,3.5 ve 3.6'dakullanılan algoritmanın görüntülerüzerinde yapılmış şifreleme ve çözme işlemlerine ait elde edilen değerler görülmektedir. Tablolar da yer alan işlem zamanındana ait değerler milisaniye, işlemci kullanımına (CPU) ait değerler yüzde (%), hafıza kullanımına (RAM) ait değerler ise (Byte) türünden ifade edilmiştir.

Tablo3.3. 2.51KB'lık görüntü (Erciyes Logo) için şifreleme ve çözme işlemeleri için algoritmaların performans analiz değerleri.

Algoritma	İşlem Zamanı (MS)		RAM Kullanımı (Byte)		CPU Kullanımı (%)	
	Şifreleme	Çözme	Şifreleme	Çözme	Şifreleme	Çözme
Sezar	22.36	25.59	127.94	127.76	90.75	92.99
Vigenere	24.68	26.52	241.46	241.20	88.91	91.15
Vernam	22.23	23.44	128.96	126.63	87.81	91.15
Hill	205.81	186.84	227.62	227.38	0.36	0.12
Playfair	160.84	139.43	194.45	194.43	0.53	0.37

Tablo3.4. 1.83 KB'lıkgörüntü (Cameraman) için şifreleme ve çözme işlemeleri için algoritmaların performans analiz değerleri.

Algoritma	İşlem Zamanı (MS)		RAM Kullanımı (Byte)		CPU Kullanımı (%)	
	Şifreleme	Çözme	Şifreleme	Çözme	Şifreleme	Çözme
Sezar	124.3	84.38	515.0	2.032	15.93	16.60
Vigenere	89.79	99.95	1.020	2.537	17.17	16.92
Vernam	87.48	95.09	517.0	2.034	15.93	16.81
Hill	1.371	1.266	808.8	808.7	0.03	0.04
Playfair	771.8	1.041	814.0	814.0	0.10	0.11

Tablo3.5. 1.10 KB'lıkgörüntü (Lenna) için şifreleme ve çözme işlemeleri içinalgoritmaların performans analiz değerleri.

Algoritma	İşlem Zamanı (MS)		RAM Kullanımı (Byte)		CPU Kullanımı (%)	
	Şifreleme	Çözme	Şifreleme	Çözme	Şifreleme	Çözme
Sezar	62.51	59.89	345.0	1.432	46.80	51.06
Vigenere	56.09	61.29	706.9	1.793	47.10	51.70
Vernam	54.78	67.96	346.6	1.433	46.51	51.78
Hill	770.0	766.0	529.8	829.6	0.03	0.36
Playfair	525.9	556.3	534.0	1.621	0.07	0.57

Tablo3.6. 1.19 KB'lıkgörüntü (Peppers) için şifreleme ve çözme işlemeleri için algoritmaların performans analiz değerleri.

Algoritma	İşlem Zamanı (MS)		RAM Kullanımı (Byte)		CPU Kullanımı (%)	
	Şifreleme	Çözme	Şifreleme	Çözme	Şifreleme	Çözme
Sezar	95.53	118.5	505.4	1.998	17.45	18.67
Vigenere	91.60	105.3	1.002	2.495	18.94	19.36
Vernam	83.63	101.4	507.3	2.000	16.01	8.79
Hill	1.301	1.280	792.5	792.4	0.05	0.04
Playfair	1.058	733.7	797.7	797.6	0.19	0.11

Performans değerlerinin karşılaştırılmasında doğru sonuçları elde edebilmek için, her beş algoritmada dört farklı görüntü ile çalışılmıştır.

Farklı görüntüler için farklı sonuçlar elde edilmektedir.Tablo 3.1'de performans değerlerini etkileyen faktörler verilmiştir. Tablo 3.1'e göre, girdi verisinin büyüklüğü performans analizinde etkili bir değerdir.

Her algoritma için 2.5 KB'lık, 1.83 KB'lık, 1.19 KB'lık ve 1.10 KB'lık farklı görüntüler kullanılmıştır, çünkü görüntüdeki renklere göre Base64 Kodunda karakter sayıları değiştirmektedir.

Performans analizi çalışmalarında elde edilen sonuçlar değişkenlik gösterdiği için dört algoritmada ve dört görüntü için aynı anahtar ile çalışma tercih edilip değeri 3 olarak belirlenmiştir. Anahtarın 3 olarak belirlenmesinde herhangi özel bir durum olmamakla birlikte Vernem Algoritması için otomatik üretilen anahtarlar kullanılmıştır.

Algoritmaların beşi de aynı bilgisayarda ve aynı yazılım ortamında test edilmiştir. Tablo 3.1'de performansı etkileyen dış faktörler arasında bilgisayarın hızı ve derleyicinin kalitesi görülmektedir. Bu faktörlerin şifreleme ve çözme işlemlerine olan etkisini sabit tutabilmek için algoritmaların beşi de aynı ortamda test edilmiştir.

Tablo3.3, Tablo 3.4, Tablo 3.5 ve Tablo 3.6 incelendiği zaman değerlerler farklık göstermektedirler. Bunu nedeni farklı boyuttaki görüntüler üzerinde çalışılmasıdır. Algoritmalar arasında olan değer değişiklikleri aynı oranda olmasından dolayı, Tablo3.3 üzerinden analiz yapılacaktır.

Tablo 3.3'teişlem zamanı sütunundan görüldüğü üzere şifreleme işlemlerini tamamlamak için geçen işlem zamanı Sezar için 22.36, Vernam için 22.23 milisaniye ve çözme işlemi için Sezar Algoritması için 25.59 milisaniye, Vernam Algoritması için 23.44 milisaniye olarak tespit edilmiştir, Vigenere Algoritması ile şifreleme işlemi 24.68 milisaniye, çözme işlemi 26.52 milisaniye olarak tespit edilmiştir yani işlem zamanı bakımından Vigenere, Sezar ve Vernem Algoritmalarından daha fazla zaman harcayan ancak Hill ve Playfair Algoritmalarına göre daha iyi ve avantajlı bir algoritmadır. Diğer algoritmalar daha büyük değerler üretmiştirler, farklı boyutta görüntüler için de aynı durum söz konusu olmuştur.

En fazla zaman harcayan algoritmalar: Playfair Algoritması şifreleme işleminde 160.84 milisaniye, çözme işleminde 139.43 milisaniye, daha sonra Hill algoritması, şifreleme işlemi için 205.81 milisaniye ve çözme işlemi için 186.84 milisaniye olarak görülmektedir.

Bu durumda Vernam Algoritmasının şifreleme ve çözme işlemlerinde diğer algoritmalara göre daha hızlı olduğu görülmektedir. Sebebi Vernam algoritmasının bitler (1 ve 0) üzerinde çalışmasından kaynaklanmaktadır. Bu durum Vernam Algoritması için işlem zamanı anlamında kazanç sağlamakta ve geçen sürenin azaltılmasında katkı sağlamaktadır. Bahçetepe'ye göre [35].performans açısından hızlı çalışan bir algoritma, kullanım için daha idealdir. Bu durumda karşılaştırma sonuçlarına göre, işlem zamanı açısından bakıldığında Vernam Algoritması performans sıralamasında birinci sırada, Sezar Algoritması ikinci sırada, Vigenere üçüncü, Playfair ise dördüncü ve en son da Hill Algoritması beşinci sırada yer almaktadır.

Tablo 3.3'te RAM kullanım sütunundan görüldüğü üzere Sezar Algoritmasında şifreleme işlemlerinin tamamlaması için kullanılan bellek miktarı Bytecinsinden 127.76 ve çözme işlemi için 127.76 Byteolarak tespit edilmiştir.

Vernam Algoritmasında şifreleme işlemleri için 128.96Byteve çözme işlemi için 126.63 Byte, Vigenere Algoritmasında şifreleme işlemleri için 241.46 Byteve çözme işlemi için 241.46 Byte, Playfair Algoritmasında şifreleme işlemleri için 194.45 Byteve çözme işlemi için 194.43 Byte, Hill Algoritmasınd şifreleme işlemleri için 227.62 Byteve çözme işlemi için 227.38 Byteolarak tespit edilmiştir.

Bahçetepe[35].modüler çarpma algoritmaları ve kriptolojide uygulamaları üzerine yazmış olduğu tez çalışmasında bellek tüketimi az olan algoritmanın performansının daha yüksek olduğunu belirtmiştir. Buna dayanarak bellek kullanım performansı açısından yapılan karşılaştırmada bellek kullanımı en düşük çıkan algoritma Sezar Algoritmasıdır. Bu durumda performansı en yüksek olan algoritma Sezar'dır. Bellek kullanımı açısından performans sıralamasına bakıldığında, Sezar birinci, Vernam ikinci, Playfair üçüncü, Hill dördüncü ve en son da Vigenere beşinci sıradadır. Vernem Algoritmasının ilk sırada olma sebebiyse verilerin tek bir dizide tutulması ve yer değişim işlemleri yapılmamasıdır. Diğer algoritmalarda ise birden fazla dizide tutularak birden fazla işlemin yapılması ve öteleme ile yer değişim işlemlerinin bulunmasıdır.

Tablo 3.3'teCPU sütunundan görüldüğü üzere Sezar algoritmasının şifreleme işlemlerinin tamamlanması için CPU üzerinde oluşturduğu iş yükü miktarı %90.75 ve çözme işlemi için %92.99, Vigenere Algoritmasının şifreleme işlemlerinin

tamamlanması için CPU üzerinde oluşturduğu iş yükü miktarı %88.91 ve çözme işlemi için %91.15, Vernam Algoritmasının şifreleme işlemlerinin tamamlanması için CPU üzerinde oluşturduğu iş yükü miktarı %87.81 ve çözme işlemi için %91.15, Playfair Algoritmasının şifreleme işlemlerinin tamamlanması için CPU üzerinde oluşturduğu iş yükü miktarı %0.53 ve çözme işlemi için %0.37, Hill Algoritmasının şifreleme işlemlerinin tamamlanması için CPU üzerinde oluşturduğu iş yükü miktarı %0.36 ve çözme işlemi için %0.12, olarak tespit edilmiştir.

CPU üzerindeki iş yükü durumları açısından karşılaştırıldığında daha az değere sahip olan algoritma daha iyi performansa sahip olmak anlamına gelmektedir. Sezar Algoritması CPU kullanımı açısından en yüksek değere sahiptir. Daha sonra sırasıyla Vigenere, Vernam, Playfair, veHill Algoritmaları CPU üzerindeki iş yükü oranlarına göre performansı en iyi olandan en kötü olana doğru sıralama yapıldığında Hill Şifreleme Algoritması birinci, Playfair Şifreleme Algoritması ikinci, Vernam Şifrelem Algoritması üçüncü, Vigenere Şifreleme Algoritması dördüncü ve en son Sezar Şifrelem Algoritması beşince sıradadır.

Bu sıralamaların nedeniyse daha çok matematiksel işlemi kullanan ve gruplar şeklinde işleyen algoritmalar daha az CPU kullanımı gözlemlenmiş ve tüm dizini birden şifreleyen öteleme, yer değiştirme, harf değiştirme işlemlerine dayanan algoritmalarsa yüksek değerlere sahip olduğu görülmüştür.

Bellek kullanım oranları ve CPU kullanım oranları arasında ters ilişki ortaya çıkmıştır. CPU kullanımı en yüksek olan Sezar Algoritmasının bellek kullanımı en düşüktür. Bu durumun sebebi ise algoritmaların kaynak kodlarındaki adım ve işlem sayısı olabilir. Daha az kod satırı ile daha çok iş yapılması bu durumun temel nedeni olabilir.

Algoritmaların yapısıyla ilgili bir takım kriterler(kurulacak sisteme uygunluğu, esnekliği, güvenlik oranı, şifreleme ve çözme işlemlerinde veri artışı veya kaybı kriterlerine göre kıyaslama) yapılmıştır.

Tablo 3.7, Tablo 3.8, Tablo 3.9 ve Tablo 3.10'da verilen sonuçlar, anlatılan tüm algoritmalar için Cameraman. JPEG, Peppers. JPEG, Lenna. JPEG, Erciyes Logo. PNG Görüntüleri örnek olarak alınmıştır.

Genelde kullanılan algoritmalar gücü ve hızı düşük algoritmalardır, yalnız bu algoritmaları yukarıda belirtilenkriterlere göre karşılaştırdığımız zaman değişiklik göstermektedirler.

Örneğin Sezar, Vigenere ve Playfair Algoritmalarının görüntü şifrelemek için esnek ve kolay algoritmalar olduğunu görebiliriz. Ancak Vernam ve Hill Algoritmalarında bu esneklik bulunmamaktadır. Bunun sebebi Vernam Algoritması için çök uzun bir anahtar kullanılmasının gerekli oluşudur. Özellikle görüntü şifrelemede, görüntüye denk gelen Base 64 Kodu karakter sayısı 3436 olan (Erciyes Logo) görüntüsü için, 3436 karaktere sahip olan bir anahtar kullanmak zorundayız. Hill Algoritmasında bazen kullanılan anahtar için tersdeğer bulunamamaktadır.

Tablo 3.7, Tablo 3.8, Tablo 3.9 ve Tablo 3.10'da orijinal görüntünün şifrelenmesi ve çözerken de orijinal görüntünün tekrardan elde edilmesi, oranı olangüven oranı (%), verilmektedir. Sezar,Vigenere ve Hill Algoritmaları için %100 olduğu tespit edilmiş, ancak Vernam %99 ve Playfair için %97 tespit edilmiştir. Ayrıca algoritmalarda görüntü şifreleme ve çözme işlemlerinde veri artışı bulunmaktadır.

Tablo 3.7. 2.5 KB'lık Erciyes Logo görüntüsü için kıyaslama değerleri.

Orjinal Görüntü : Erciyes Logo
Size : 2.52 KB
Base 64 Karakter Sayısı : 3436
Tip : PNG
Boyut : 58 * 58

Algoritma	Şifrelenme	Çözme	Veri artışı		Veri kaybı		güven oranı (%)
			Şifreleme	Çözme	Şifreleme	Çözme	
Sezar	 Boyut : 58 * 58 Size : 2.97 KB	 Boyut : 58 * 58 Size : 2.52 KB	0.45 KB	-	-	-	100%
Vigenere	 Boyut : 58 * 58 Size : 2.97 KB	 Boyut : 58 * 58 Size : 2.52 KB	0.45 KB	-	-	-	100%
Vernam	 Boyut : 58 * 58 Size : 2.97 KB	 Boyut : 58 * 58 Size : 2.52 KB	0.45 KB	-	-	-	100%
Hill	 Boyut : 58 * 58 Size : 2.97 KB	 Boyut : 58 * 58 Size : 2.52 KB	0.45 KB	-	-	-	100%
Playfair	 Boyut : 58 * 58 Size : 2.97 KB	 Boyut : 58 * 58 Size : 2.52 KB	0.45 KB	-	-	-	96.9 %

Tablo 3.8.1.10 KB'lık Lennagörüntüsü için kıyaslama değerleri.

Orjinal Görüntü : Lenna
Size : 1.10 KB
Base 64 Karakter Sayısı : 5728
Tip : JPEG
Boyut : 50 * 29

Algoritma	Şifrelenme	Çözme	Veri artışı		Veri kaybı		güven oranı (%)
			Şifreleme	Çözme	Şifreleme	Çözme	
Sezar	 Boyut : 50 * 29 Size : 1.85 KB	 Boyut : 58 * 58 Size : 1.66 KB	0.75 KB	0.56 KB	-	-	100%
Vigenere	 Boyut : 58 * 58 Size : 1.83 KB	 Boyut : 58 * 58 Size : 1.66 KB	0.73 KB	0.56 KB	-	-	100%
Vernam	 Boyut : 58 * 58 Size : 1.83 KB	 Boyut : 58 * 58 Size : 1.66 KB	0.73 KB	0.56 KB	-	-	99 %
Hill	 Boyut : 58 * 58 Size : 1.85 KB	 Boyut : 58 * 58 Size : 1.66 KB	0.75 KB	0.56 KB	-	-	100%
Playfair	 Boyut : 58 * 58 Size : 1.82 KB	 Boyut : 58 * 58 Size : 1.66 KB	0.72 KB	0.56 KB	-	-	97 %

Tablo 3.9.1.19 KB'lık Peppersgörüntüsü için kıyaslama değerleri.

Orjinal Görüntü : Peppers
Size : 1.19 KB
Base 64 Karakter Sayısı : 11932
Tip : JPEG
Boyut : 49 * 37

Algoritma	Şifrelenme	Çözme	Veri artışı		Veri kaybı		güven oranı (%)
			Şifreleme	Çözme	Şifreleme	Çözme	
Sezar	 Boyut : 49 * 37 Size : 2.07 KB	 Boyut : 49 * 37 Size : 1.83 KB	0.86 KB	0.64 KB	-	-	100%
Vigenere	 Boyut : 49 * 37 Size : 2.06 KB	 Boyut : 49 * 37 Size : 1.83 KB	0.85 KB	0.64 KB	-	-	100%
Vernam	 Boyut : 49 * 37 Size : 2.07 KB	 Boyut : 49 * 37 Size : 1.83 KB	0.86 KB	0.64 KB	-	-	99 %
Hill	 Boyut : 49 * 37 Size : 2.05 KB	 Boyut : 49 * 37 Size : 1.83 KB	0.84 KB	0.64 KB	-	-	100%
Playfair	 Boyut : 49 * 37 Size : 2.07 KB	 Boyut : 49 * 37 Size : 1.83 KB	0.86 KB	0.64 KB	-	-	97 %

Tablo 3.10.1.19 KB'lık Cameramangörüntüsü için kıyaslama değerleri.

Orjinal Görüntü : Cameraman
Size : 1.83 KB
Base 64 Karakter Sayısı : 10572
Tip : JPEG
Boyu : 72 * 72

Algoritma	Şifrelenme	Çözme	Veri artışı		Veri kaybı		güven oranı (%)
			Şifreleme	çözme	Şifreleme	çözme	
Sezar	 Boyut : 58 * 58 Size : 4.09 KB	 Boyut :58 * 58 Size : 3.04 KB	2.26 KB	1.21 KB	-	-	100%
Vigenere	 Boyut : 58 * 58 Size : 4.08 KB	 Boyut :58 * 58 Size : 3.04 KB	2.25 KB	1.21 KB	-	-	100%
Vernam	 Boyut : 58 * 58 Size : 4.10 KB	 Boyut :58 * 58 Size : 3.04 KB	2.27 KB	1.21 KB	-	-	99 %
Hill	 Boyut : 58 * 58 Size : 4.08 KB	 Boyut :58 * 58 Size : 3.04 KB	2.25 KB	1.21 KB	-	-	100%
Playfair	 Boyut : 58 * 58 Size : 4.08 KB	 Boyut :58 * 58 Size : 3.04 KB	2.25 KB	1.21 KB	-	-	97 %

4. BÖLÜM

TARTISMALAR-SONUÇ ve ÖNERİ

4.1. Tartışma-Sonuç

Güvenlik,iletişimde ve görüntü depolamasında önemli bir unsurdur. Kriptoloji bu güvenliği sağlayan bir yöntemdir. İnternet, tıbbi görüntüler, multimedya ve askeri iletişimlerde şifreleme kullanılmaktadır.

Klasik şifreleme algoritmalarından en yaygın olan Sezar, Vigenere, Vernam, Hill, Playfair Algoritmaları yapısı hakkında bilgi vererek metin üzerinde ve farklı boyuttaki görüntüler üzerinde uygulama ve performans analizi yapılıp sonuçları karşılaştırılmıştır. Bu çalısmada bilgi güvenliğini sağlama amaçlı olarak kullanılan klasik algoritmalarından sık kullanılanları zaman karmaşıklığı, islemci karmaşıklığı ve hafıza karmaşıklığı bakımından incelenmiş ve uygulamaları gerçekleştirilmiştir.

Performans analizi incelemelerinde farklı boyuttaki görüntüler için CPU iş yükü, bellek kullanımı ve işlem süresi açısından farklı değerlere sahip oldukları görülmüştür. Bu nedenle tez çalışmasında örnek olması amacıyla performans analizi işlemlerinde sabit bir anahtar kullanılmıştır.

CPU üzerindeki iş yükü açısından Hill Algoritması birinci, Playfair Algoritması ikinci, Vernam üçüncü, Vigenere dördüncü ve Sezar Algoritması beşinci olarak tespit edilmiştir.

Bellek tüketimi açısından birinci SezarAlgoritması, ikinci Vernam, üçüncü Playfair, dördüncü Hill ve beşince Vigenere Algoritması olarak tespit edilmiştir.

İşlem zamanı açısından karşılaştırma yapılırsa en ideal algoritma Vernam'dir.İkinci sırada Sezar, üçüncü Vigenere, dördüncü Playfair ve beşinci Hill Algoritması olarak tespit edilmiştir.

Klasik şifreleme sistemleri görüntüler için kullanılabilse de, iki nedenden dolayı sıkıntı çıkabilmektedir. Birincisi;görüntü yazı verilerinden çok daha büyüktür, bu nedenle klasik algoritmalar görüntüleri şifrelemek için yavaş kalmaktadırlar. Öteki problem ise; yazı verisinin şifresi çözüldüğünde aynen geri gelmesi gerekli iken, görüntü dosyalarında böyle bir zorunluluk yoktur. İnsan doğası nedeniyle şifresi çözülmüş bir görüntü verisinin tarz olarak gerçek görüntü ile aynı olması gerekmemektedir ve fark edilemeyecek kadar farklılıklar olabilmektedir.

4.2. Öneri

Yapılan çalışma ve incelemeler sadece görüntü şifrelemek için yapılmıştır. Çalışma: dosya, ses ve video için de geliştirilebilir ve amaçlara yönelik metotlarla işlem zamanının indirgenmesi veya daha az bellek tüketmesi ile algoritmaların dezavantajları da yok edilebilir.

KAYNAKLAR

1. Yerlikaya, T., Buluş, E., Buluş, N., "Kripto Algoritmalarının Gelişimi veÖnemi", *Akademik Bilişim Konferansı*, Denizli, (2006).
2. Yılmaz, R.,"Kriptolojik Uygulamalarda Bazı İstatistik Testler" , YüksekLisans Tezi, Selçuk Üniversitesi Fen Bilimleri Enstitüsü, Konya, 1-31(2010).
3. Canbek, G., Sağıroğlu, Ş., "Bilgi ve Bilgisayar Güvenliği Casus Yazılımlarve Korunma Yöntemleri", Grafiker Yayınları, Ankara, 16-22,33,198-204,(2006).
4. Çeşmeci, M.Ü., "Kriptoloji Tarihi" , **UEKAE Dergisi, 1**(1): 20-31, (2009).
5. Çimen, C., Akleylek, S., Akyıldız, E., "Şifrelerin Matematiği Kriptografi6.Basım", ODTÜ Yayıncılık, Ankara, 1,5-66, (2008).
6. Sağıroğlu, Ş., Alkan, M., "Bilgi Güvenliği Bilimi(Kriptoloji)", Her YönüyleElektronik İmza, Grafiker Yayınları, Ankara, 1-19, 25-39 (2005).
7. Dutta, A., McCrohan, K., "Management's Role In Information Security in aCyber Economy", **California Management Review, 45**(1): 67-87, (2002).
8. Whitman, M.E., "Enemy At The Gate: Threats To Information Security",**Communıcations Of The Acm, 46**(8): 91-95, (2003).
9. İnternet: Frank Lin, "Cryptography's Past, Presents, And Future Role InSociety", **http://engineering.wustl.edu/contentfiles/ecc/Li**(Dateaccessed: December2015).
10. Civelek, Y. D., "Kişisel Verilern Korunması ve Bir Kurumsal YapılanmaÖrneği", Uzmanlık Tezi, TC Başbakanlık Devlet Planlama TeşkilatıMüsteşarlığı Bilgi Toplumu Dairesi Başkanlığı, Ankara, 30-57 (2011).
11. Vural, Y., Sağıroğlu, Ş., "Kurumsal Bilgi Güvenliği ve Standartları ÜzerineBir İnceleme", **Gazi Üniversitesi Mühendislik Mimarlık FakültesiDergisi, 23**(2): 507-522 (2008).
12. Tekerek, M., "Bilgi Güvenliği Yönetimi", **KSU Fen ve Mühendislik Dergisi,11**(1): 132-137 (2008).
13. Kara, O., "2. Dünya Savaşı'ndan Günümüze Kriptoloji: Enigma'dan AES'eŞifreleme", **Bilim ve Teknik Dergisi,** (500): 28-33 (2009).

14. Aslandağ, K., "Bilgi Güvenliği Kavramı ve Bilgi Güvenliği YönetimSistemleri ile Şirket Performansı İlişkisine Dair Bir Uygulama", YüksekLisans Tezi, Gebze Yüksek Teknoloji Estitüsü, Sosyal BilimlerEnstitüsü, Gebze, 13-21 (2010).

15. Günden, Ü., "Şifreleme Algoritmalarının Performans Analizi", YüksekLisans Tezi, Sakarya Üniversitesi Fen Bilimleri Enstitüsü, Sakarya, 1-18, 28-34, 77-78 (2010).

16. Başar, M.S., "Yer Değiştirme Esaslı ve Rasgele Anahtarlı Yeni BirŞifreleme Algoritması", Doktora Tezi, Atatürk Üniversitesi SosyalBilimler Enstitüsü, Erzurum, 1-17 (2004).

17. Bauer, F.L., Decrypted Secrets Methods and Maxims of Cryptology ThirdEdition, Springer-Verlag Berlin Heidelberg, Germany, 1-7 (2002).

18. Denning, R. D.E., "Encyption Algorithms", Cryptography and DataSecurity, Addison-Wesley Publishing, United States of America, 1-3, 59-79, 86-89 (1982).

19. İnternet: Toyran, M., Pedersen, T.P., Hasekioğlu, A., Can, M.A., Berber,S., "Bilgi GüvenliğindeKuantumTeknikler", **http://www.emo.org.tr/ekler/ebc1c 93c2e9ad67_ek.pdf**(Date accessed: January2016).

20. Şen,Ş.,"İndirgenmiş SPN(Substitution Permutatin Network) Algoritması için Lineer Kriptanaliz Uygulaması", Yüksek Lisans Tezi, Trakya Ünv.Fen Bil. Enst., Edirne, (2006).

21. Öztürk İ, Soğukpınar İ, 2004. Analysis and Comparison of Image EncryptionAlgorithms, **IJIT,**1(2) ISSN:1305 – 239X.

22. Dalkılıç, G., Akın, O., "Anahtar Tabanlı Gelişmiş Rotor Makinesi", *Akademik Bilişim Konferansı*, Gaziantep, (2005).

23. Bayar, E., "Modern Kriptosistemlerle Şifrelemenin Modellenmesi İle VeriGüvenliğinin Sağlanması", Yüksek Lisans Tezi, Marmara ÜniversitesiFen Bilimleri Enstitüsü, İstanbul, 1-32 (2012).

24. Buluş, H.N., "Temel Şifreleme Algoritmaları ve Kriptoanalizlerininİncelenmesi", Yüksek Lisans Tezi, Trakya Üniversitesi Fen BilimleriEnstitüsü, Edirne, 1, 20, 23-43, 61-73 (2006).

25. Singh, Y.K., “Generalization Of Vigenere Cipher”, ARPN(Asian ResearchPublishing Network) **Journal Of Engineering and Applied Sciences,7**(1): 39-44 (2012).

26. Ritter, T., “Substitution Cipher With Pseudo-RandomShuffling: TheDynamicSubstitution Combiner”, **Cryptologia, 14**(4): 289-303 (1990).

27. Stallings ,W, Cryptography And Network Security, Third Edition, New Jersey 2323-42,(2003).

28. Henk C.A. van Tilborg, Fundamentals Of Cryptology, A Professional Reference And Interactive Tutorial, Eindhoven University of Technology The Netherlands, Kluwer Academic Publishers Boston,Dordrecht, 9-13,20-21, London.

29. Kenneth H. Rosen, An Introduction To Cryptography, Second Edition, Discrete Mathematics And Its Applications,79-91,131-157,172(2007).

30. What is base64 encoding and how can we benefit from it**http://inchoo.net/magento/what-is-base64-encoding-and-how**(Date accessed: November2015).

31. Kodaz, H., Botsalı, F.M., “Simetrik ve Asimetrik Şifreleme AlgoritmalarınınKarşılaştırılması”, **Selçuk Üniversitesi Teknik Bilimler MeslekYüksekokulu Teknik-Online Dergi, 9**(1): 10-23 (2010).

32. Yerlikaya, T., “Yeni Şifreleme Algoritmalarının Analizi”, Doktora Tezi,Trakya Üniversitesi Fen Bilimleri Enstitüsü, Edirne, 1-3, 6-9, 51-58(2006).

33. Yerlikaya, T., Buluş, E., Buluş, N., “Asimetrik Şifreleme AlgoritmalarındaAnahtar Değişim Sistemleri” , **Akademik Bilişim Konferansı,** Denizli,(2006).

34. Öztürk, M.M., “Uzaktan Eğitimde Ölçme Değerlendirme Sistemi Tasarımıve Yazılım Test Teknikleri İle Performans Analizi”, Yüksek Lisans Tezi,Sakarya Üniversitesi Fen Bilimleri Enstitüsü, Sakarya, 5-29, 71-73(2012).

35. Bahçetepe, H., “Modüler Çarpma Algoritmalarının İncelenmesi veKriptolojide Uygulanması”, Yüksek Lisans Tezi, İstanbul Üniversitesi FenBilimleri Enstitüsü, İstanbul, 71-72 (2006).

ÖZGEÇMİŞ

KİŞİSEL BİLGİLER

Adı, Soyadı: Zainab Hashim OBAID

Uyruğu: Iraq (IQ)

Doğum Tarihi ve Yeri: 16Şubat 1983, Kerkük

Medeni Durumu: Evli

Tel: +905372596954

E–mail: zainabark@yahoo.com

Yazışma Adresi: Iraq- Kerkük

EĞİTİM

Derece	Kurum	Mezuniyet Tarihi
Yüksek Lisans	EÜ Fen Bilimler Enstitüsü	2016
Lisans	Kerkük Üniversitesi Bilgisayar Bilimleri Bölümü	2006
Lise	Kerkük Kız Lisesi, Iraq	2002

İŞ DENEYİMLERİ

Yıl	Kurum	Görev
2009- Halen	İsken Kız Lisesi,Iraq-Kerkük	Öğretmen

YABANCI DİL

Türkçe, İngilizce