CryptES: A Self Learning Application Software for Encryption

Volkan Sozeri
Ege University, Ege Higher Vocational School
35100, Bornova, Izmir, Turkey
E-mail: volkan.sozeri@ege.edu.tr

Coskun Harmansah (Corresponding author)
Ege University, Ege Higher Vocational School
35100, Bornova, Izmir, Turkey
E-mail: coskun.harmansah@ege.edu.tr

Abstract
Students face some difficulties in math theories when they want to learn the methods used in encryption and decryption. Learning these theories is not easy due to nature of encryption process based on mathematics and complexity. It is important to use visual and simulation software tools that helps students understand the steps of encryption techniques. CryptES software developed within the scope of our study can communicate with the user through interactive interfaces related to the encryption and decryption windows in order to provide easily understanding of some stages in text encryption and decryption operations. Thus, the user can trace the steps of the selected encryption technique and clues for the encryption process in these steps. Particularly, implementation of the creation of keys through a visual tool for AES and RSA encryption can help to facilitate the learning of key planning for the students and to comprehend these algorithms in short time.

Keywords: Text Encryption, Text Decryption, MD5, AES, RSA

CryptEs: Şifreleme için Bireysel Öğrenme Uygulama Yazılımı

Özet
Programlama eğitimi alan kişi veya öğrenciler şifreleme ve şifre çözme konusunda kullanılan yöntemleri öğrenmek istediklerinde çok sayıda teorik bilgi ile karşılaşmaktadır. Bu bilgilerin öğrenilmesi; şifreleme işlemlerinin doğası gereği matematik temelli ve karmaşık yapıda olmasından dolayı zordur. Şifreleme tekniklerinin öğrenilmesinde görsel ve simülasyon yazılım araçların kullanılması önemlidir. Bu çalışma kapsamında geliştirilen CryptES yazılımı metin şifreleme ve şifre çözme süreçlerinde yer alan bazı adımların kolay anlaşılmasını sağlamak amacıyla ilgili şifreleme ekranlarında etkileşimli ara yüzler üzerinden kullanıcı ile iletişim kurabilmektedir. Böylece kullanıcı, seçtiği şifreleme tekniğinin uygulama basamaklarını ve bu adımlardaki şifreleme sürecinde neler yapılması gerektiğini görebilmektedir. Özellikle AES ve RSA şifreleme sırasında anahtarların oluşturulma sürecinin görsel bir araç üzerinden gerçekleştirilmesi; öğrencilerin anahtar planlama işlemlerinin öğrenmesini kolaylaştıracak ve daha kısa bir sürede bu algoritmaların kavranılmasını sağlayacaktır.

Anahtar Kelimeler: Metin Şifreleme, Metin Şifre Çözme, MD5, AES, RSA

1. Giriş

Günümüzde veriler kablolu ve kablosuz ağlar aracılığı transfer edilmekte ve bilgiye internet üzerinden erişilmektedir. Metin, görüntü ve ses dosyalarının büyük bir kısmı ağ üzerinden taşınmaktadır. Bu sürecin en önemli sonucu hiç kuşkusuz bilgi güvenliğinin ön plana çıkmasıdır. Verilerin güvenli şekilde ağ

yapıları üzerinden transfer edilmesi amacıyla farklı koruma teknikleri ve algoritmalar kullanılmaktadır. Bu amaçla yapılan çalışmalar, erişim kontrolü, iletişim güvenliği, kimlik doğrulama, veri şifreleme ve veri depolama güvenliği gibi konular üzerine yoğunlaşmaktadır (Teodorescu et. al., 2015; Diesburg & Wang 2010; Karnani & Singh 2015).

Diğer taraftan, şifreleme yöntemleri mesaj, veri dosyası, veri transferi, erişim kontrolü gibi farklı düzeylerde güvenlik sağlamak amacıyla ile de kullanılmaktadır (Teodorescu et. al., 2015; Chaouch et. al., 2016; Singh & Supriya 2013). Kişisel veya kurumsal düzeyde hassas verilerin korunması ve güvenliği için farklı kriptografik yöntemler ve algoritmalar kullanılmaktadır. Bilişim sistemleri ve programlama eğitimi alan öğrenci veya kişilerin veri koruma ve güvenliği alanında bilgi birikimleri geçmiş yıllara kıyasla daha önemli hale gelmiştir. Özellikle programlama eğitimi alan öğrenci veya kişilerin temel veri koruma ve güvenlik ilkelerini öğrenmeleri sayesinde, bu alanda ihtiyaç duyulan uzman sayısının artması sağlanacaktır. Programlama eğitiminde kriptolojinin temeli olan matematik teorilerinin yanı sıra yardımcı simülasyon araçları ile şifreleme tekniklerinin öğretilmesi amacıyla araştırmalar yapılmaktadır. Teodorescu ve arkadaşları (2015) metin şifreleme/şifre çözme metodunun öğretiminde kullanılabilecek grafik tabanlı bir yazılım geliştirdi. Bu çalışmada, simetrik ve asimetrik şifreleme algoritmaları kullanarak verilen bir metnin şifrelenmesi ve ardından şifre çözme işlemlerinin öğretimi amacıyla kullanılacak bir yazılım aracı tasarlanmıştır. Böylece öğrencilerin şifreleme/şifre çözme yöntemlerini, şifreleme gücü, karmaşıklığı, şifreleme/şifre çözme anahtarlarının saklanması ve kullanımları açısından değerlendirmelerine olanak sağlanmıştır. Bilgisayar ve bilişim öğrencilerinin DES ve AES algoritmalarını anlamalarını kolaylaştırmak amacıyla Chok ve Herath (2004) Microsoft Excel altında çalışabilecek bir uygulama geliştirmişlerdir. Gerçekleştirilen öğrenme modülü ile öğrencilerin simetrik şifreleme algoritmalarını anlamalarına katkı sağlaması hedeflenmiştir. Yapılan diğer bir çalışmada, Güzel ve arkadaşları (2010) CrypTool yazılımının eğitim ve geliştirme sürecindeki kullanımını araştırmışlardır. Öğrencilerin görsel bir yazılım aracı kullanmalarının şifreleme/şifre çözme algoritmalarının öğrenilmesinde olumlu yönde katkı sağladıklarını belirtmişlerdir. Tao ve arkadaşları (2011) DES visual görsel aracının simetrik şifrelemenin öğretiminde kullanılmasının bireysel öğrenmeye katkı sağladığı, özellikle bilgisayar bilimleri öğrencileri üzerinde daha pozitif etki yaptığını ortaya koymuşlardır. Şifreleme için görsel araç geliştirilmesine yönelik diğer önemli bir çalışmada Jun Ma ve arkadaşları (2016) AES şifreleme yöntemini öğretmek amacıyla AESVisual adlı görsel bir araç geliştirmişlerdir. AESvisual uygulamasının sınıf içi sunumlarda ve öğrencinin bireysel çalışmalarında olumlu katkıları olduğunu göstermişlerdir.

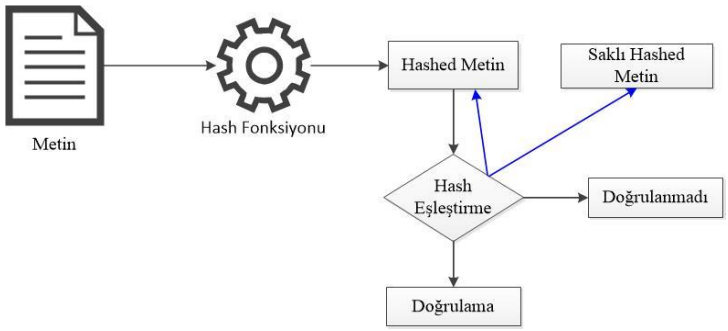
Bu çalışma kapsamında, geliştiren Şifreleme Eğitimi Yazılımı (CryptES), şifreleme konusuna ilgi duyan programlama eğitimi alan öğrenci veya kişilerin şifreleme konusundaki öğrenme süreçlerine katkı sağlamak amacıyla tasarlanmıştır. CryptEs; simetrik, asimetrik ve hashing algoritmalarının şifreleme sürecine ait bilgi ve akış diyagramları ile tanıtıldığı, metin ve dosya şifreleme ve şifre çözme örneklerinin yer aldığı görsel bir öğretim aracıdır. Öğrenciler ilgili ekran üzerinden girdikleri metinleri veya dosya içeriklerini seçtikleri bir şifreleme yöntemi ile şifreleyebilmekte ve ardından şifre çözme işlemini yaparak orjinal metni elde edebilmektedir. Bu çalışma kapsamında geliştirilen CryptES Yazılımı ile Simetrik (Tek Anahtarlı Sistemler), Asimetrik (Çift Anahtarlı Sistemler) ve Hashing (Özetleme) algoritmalarının uygulamalı olarak öğretilmesi hedeflenmiştir.

2. Şifrelemede Kullanılan Yöntemler ve Temel Bileşenleri

Şifreleme işlemi, şifreleme ve şifre çözme olmak üzere iki aşamadan oluşmaktadır. Şifreleme bir verinin orijinal içeriğinin okunamaz başka bir forma dönüştürülmesi, şifre çözme ise şifrelenmiş verinin orijinal haline dönüştürülmesi işlemidir. Şifreleme algoritmaları; Simetrik (Tek Anahtarlı Sistemler) ve Asimetrik (Çift Anahtarlı Sistemler) ve Hash (Özetleme) olmak üzere gruplandırılmaktadır.

2.1. Hashing Algoritmaları

Hashing algoritmaları geniş bir şifreleme uygulama çeşitliliğine sahip olmakla birlikte aynı zamanda genel olarak veri bütünlüğünü doğrulamak için kullanılmaktadır. Bu özellikleri dolayısıyla elektronik imza, belge yönetim sistemlerinde mesajın değişip değişmediğini garanti altına almak için SHA-1, SHA-256 ve MD5 gibi hashing algoritmaları kullanılmaktadır. Programlama eğitimi alanlara yönelik olarak geliştirilen CryptES Yazılımı' nda örnek olması açısından bu şifre algoritmalarından bazıları seçilmiştir. Bu çalışmada örnek olarak seçilen MD5 algoritması (Message-Digest algorithm 5) girdi verisinin boyutundan bağımsız olarak 128 bitlik on altılık karakterde özetler üretir. Bu hash (özet) değeri genellikle mesajın orijinalinden küçüktür. Bu algoritma ile elde edilen şifreler özetlenerek saklandığı için ters fonksiyon kullanılarak tekrar elde edilmeleri teorik olarak çok zordur. Orijinal mesajdaki bir bit değişikliği bile özet değeri değiştirmektedir. Bu özelliği, özet değerin mesaja özel olmasını sağlamaktadır. Şekil 1' de hash algoritmasının çalışması gösterilmektedir.

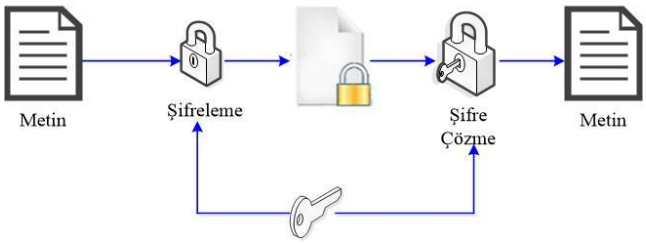


Şekil 1. Hash Algoritmasının İşleyişi

Hash fonksiyonu mesajı her biri belirli uzunlukta bloklara bölmektedir. Bu noktada veri 512 bitlik bloklara ayrılır ve son blok sonuna 64 bit eklenir. Eklenen 64 bit, giriş verisinin uzunluğunu kayıt etmek için kullanılır. İşlem yapılacak veri 512 bitin katları şeklinde değilse ekleme işlemleri yapılmaktadır. Bu algoritmanın en önemli kısmı sıkıştırma fonksiyonudur. Böylece başlangıçta değerleri sabit olan, A, B, C, D olarak adlandırılan 32 bitlik dört değişkenin değerleri her 512 bitlik blok işleme girdiğinde değişir ve algoritma sonunda bu değerler yan yana geldiğinde 128 bitlik şifre elde edilir. MD5 şifreleme de tek yönlü fonksiyonlar kullanıldığından hash değerinden orijinal mesajı elde etmek çok zordur (Wang &Yu 2005; Algolak 2014; Gençoğlu 2017).

2.2. Simetrik Algoritmalar

Bu algoritmada şifreleme ve şifre çözmek için tek anahtar kullanılmaktadır. Bu anahtar şifreleme ve şifreyi çözecek kişiler arasında ortak olarak kullanılmaktadır. Gönderilecek şifrelenmiş metin ile birlikte ortak anahtar alıcıya gönderilmektedir. Simetrik algoritma öteleme, yer değiştirme, xor işlemleri gibi işlemler gerçekleştirilerek şifreleme yapılmaktadır (Şekil 2).



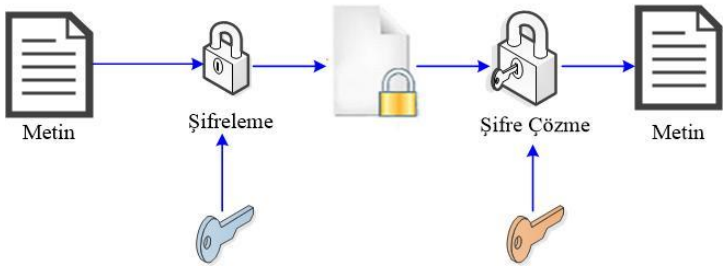
Şekil 2. Simetrik Şifreleme Yönteminin İşleyişi

Bu çalışmada örnek olarak seçilen AES (Advanced Encryption Standard) yaygın olarak kullanılan simetrik bir şifreleme algoritmasıdır. Bu algoritma John Daemen ve Vincent Rijmen tarafından Rijndael adıyla geliştirilmiş ve 2002 yılında standart haline gelmiştir. AES şifrelemede uzunluğu 128 bitte sabit olan blok ile uzunluğu 128, 192 ya da 256 bit olan anahtar kullanır. Bu şifreleme tekniğinde baytların yer değiştirmesi ve 4x4' lük matrisler üzerine yayılmış metin parçalarının, satırlarına uygulanan kaydırma işlemleri gibi teknikler kullanılmaktadır. Döngü sayısı anahtar genişliğine göre değişmektedir. 128 bit anahtar için 10 döngüde şifreleme yapılırken 192 ve 256 bit anahtarlar için sırasıyla 12 ve 14 döngüde şifreleme yapılmaktadır. Şifreleme işleminde ilk olarak 128 bit veri 4x4 byte matrisine dönüştürülür. Her döngüde sırasıyla byte'ların yer değiştirmesi, satırların ötelenmesi, sütunların karıştırılması ve anahtar planlamadan gelen ve o döngü için belirlenen anahtar ile XOR işlemleri gerçekleştirilir. Byte'ların yer değiştirilmesi işlemlerinden sonra satırların ötelenmesi işlemleri yapılır ve sütunlar karıştırılır. Döngünün son katmanında ise o döngüye ait anahtar ile XOR işlemleri yapılmaktadır. AES algoritması yer değiştirme ve karıştırma işlemlerini temel almasından dolayı büyük boyuttaki verileri çok kısa sürede şifreleyebilmektedir. Diğer taraftan, şifre çözme işlemini de aynı hızda gerçekleştirebilmektedir (Mahajan & Sachdeva 2013; Thambrija et. al., 2012; Akleylek et. al., 2011; Chaouch et. al., 2016).

AES donanımda ve yazılımda hızlı olması, kolay uygulanabilir olması, çok daha küçük belleğe gerek duyması, pratik ve doğrudan (brute force) saldırılara karşı dayanıklılığı dolayısıyla en yaygın kullanılan simetrik şifreleme algoritmasıdır. Diğer taraftan, DES, 3DES ve RC4 gibi simetrik şifreleme algoritmaları da yaygın olarak farklı alanlarda kullanılmaktadır. Geliştirilen eğitim yazılımında bu şifreleme algoritmalarından bazıları seçilerek örnek şifreleme uygulamaları yapılmıştır.

2.3. Asimetrik Algoritmalar

Asimetrik algoritmalarda şifreleme ve şifre çözmek için farklı anahtarlar kullanılmaktadır. Şifrelemek için kullanılan anahtar açık (public) olmakla birlikte şifre çözme anahtarı gizli (private) olup sadece alıcı tarafından bilinmelidir (Şekil-3). Başka bir deyişle şifreleme için kullanılan anahtar ile şifre çözme için kullanılan anahtar birbirinden farklıdır. Burada anahtar çiftlerini üreten algoritmaların matematiksel özelliklerinden dolayı açık-gizli anahtar çiftleri her kişi için farklı olacaktır. Bir kullanıcı tarafından açık anahtarla şifrelenen bir mesajı sadece ona ait gizli anahtar çözebilir.



Şekil 3. Asimetrik Şifreleme Yönteminin İşleyişi

Herhangi bir kişinin gizli anahtarıyla attığı sayısal imzanın doğrulanabilmesi, sadece onun açık anahtarını kullanarak gerçekleştirilebilir. Açık anahtar kamuya açıktır, elektronik kimlik belgelerinin içinde diğer kişisel bilgilerle birlikte tutulur ve herkes birbirinin açık anahtarını e-kimliklerine ulaşarak elde edebilir. Bu çalışma kapsamında örnek olarak seçilen RSA algoritması Ron Rivest, Adi Shamir, Leonard Adleman tarafından 1977 yılında duyurulan bir asimetrik şifreleme algoritmasıdır. Çift anahtarla çalışan algoritma anahtar planlaması ve şifreleme işlemlerini de içermektedir. RSA algoritmasına gücünü aldığı noktalardan biri de asal sayılardır. Asal sayıların 1 ve kendisi dışında tam bölenleri bulunmaz. Bu algoritmada ilk olarak çok büyük iki asal sayı seçilmektedir. Bu seçim sonrası çarpma, tersini alma gibi çeşitli matematik işlemlerin ardından açık ve gizli anahtar oluşturulur. Elde edilen açık anahtar ile gönderilmek istenen veri şifrelenir. Açık anahtar ile şifrelenen metin sadece gizli anahtar ile çözülebilmektedir (Kakkar et. al., 2012; Singh & Supriya 2013; Blomer et. al., 2003; Boscher et. al., 2007).

Asimetrik şifreleme hız konusunda simetrik algoritmalara göre daha başarılıdır. Diğer taraftan, simetrik şifreleme içerdiği basit işlemlerden dolayı elektronik cihazlarda uygulanması daha kolaydır. RSA algoritmasının en büyük dezavantajı, asimetrik bir şifreleme algoritması olarak çok büyük sayılarla işlem yapması nedeniyle yavaş olmasıdır.

Simetrik şifrelemedeki anahtar dağıtım problemi asimetrik şifrelemede ortadan kaldırılmaktadır. Bu algoritma şifreleme amaçlı kullanımı ile birlikte doğrulama amaçlı elektronik imzalarda kullanılabilmektedir. Şifreleme amacıyla DSA, ECDSA ve RSA gibi asimetrik algoritmalar kullanılmaktadır.

3. Geliştirilen Şifreleme Eğitim Yazılımı (CryptES) ve Arayüzü

CryptES; simetrik, asimetrik ve hashing algoritmalarının çalışmasına ait bilgi ve akış diyagramlarının, metin ve dosya şifreleme ile şifre çözme örnek uygulamalarının bulunduğu görsel bir öğretim aracıdır. CryptEs, .NET platformunda C# programlama dili kullanılarak geliştirilmiştir.

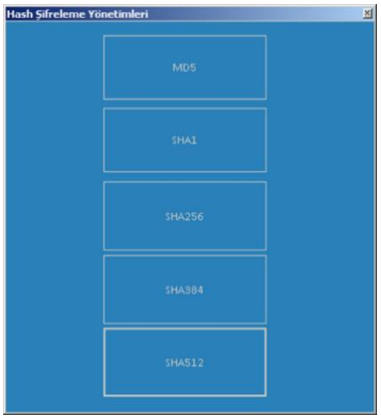
CryptES yazılımının "Şifreleme Listesi" giriş ekranında Hash, Asimetrik ve Simetrik şifreleme yöntemlerine ilişkin butonlar bulunmaktadır (Şekil-4).

Şekil 4. CryptEs Giriş Ekranı

Kullanıcı şifreleme yapmak istediği yöntemi seçerek ilgili şifreleme algoritma ekranına geçmektedir.

3.1. Hash Algoritma Örneği Olarak MD5 Uygulaması

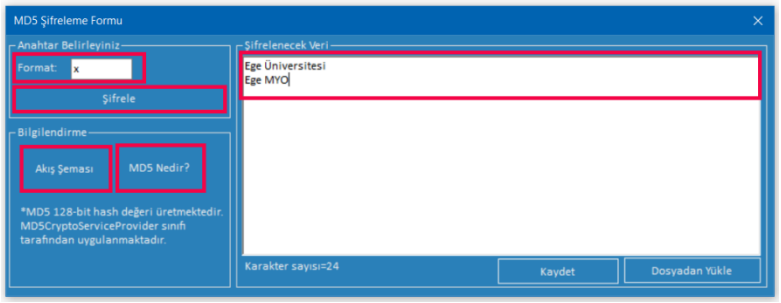
Geliştirilen CryptES yazılımı hashing algoritmaları kapsamında MD5, SHA-1, SHA-256, SHA-384 ve SHA-512 şifreleme yöntemlerine ait uygulamalar içermektedir. Böylece, seçilen bir hash şifreleme yöntemi ile orijinal metnin şifrelenmiş biçimini görebilmektedir. Kullanıcı "Hash Şifreleme Yöntemleri" ekranından öğrenmek istediği şifreleme tekniğini seçebilmektedir (Şekil-5).



Şekil 5. Hash Şifreleme Yöntemleri Giriş Ekranı

Bu çalışmada örnek olarak MD5 şifreleme uygulaması seçilmiştir. MD5 şifreleme belge yönetim sistemlerinde mesajın değişmediğinin garanti altına alınması ve kimlik denetiminin sağlanmasına yönelik uygulamalarda yaygın olarak tercih edilmektedir.

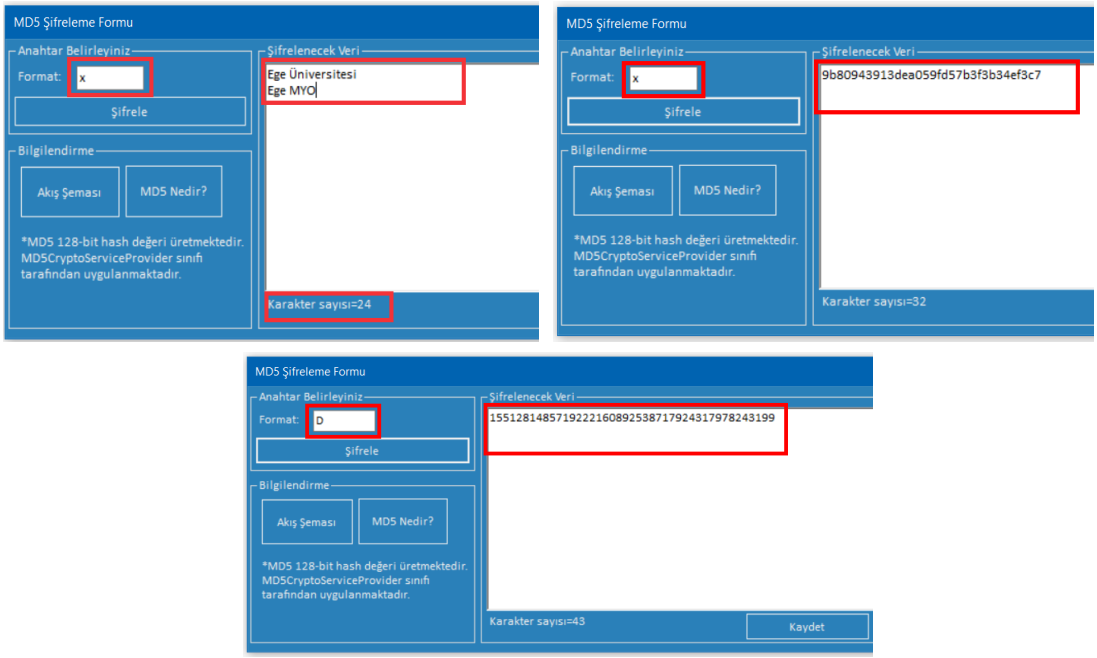
"Hash Şifreleme Yöntemleri" ekranından MD5 butonu tıklandığında MD5 "Şifreleme Formu" penceresi açılmaktadır (Şekil-6). Bu ekranda örnek bir metin şifreleme uygulamasının gerçekleştirilmesinin yanı sıra MD5 şifreleme tekniğinin basit bir anlatımı (MD5 Nedir?) ve bu tekniğin işleyişinin gösterildiği bir akış diyagramı da yer almaktadır. Kullanıcı şifrelemek istediği metni "Şifrelenecek Veri" alanına doğrudan klavye ile girebileceği gibi dosya aracılığı ile de veri girişi yapabilmektedir. Şifrelenecek veri girişi yapıldıktan sonra "Şifrele" butonu tıklanarak girilen veri şifrelenecek ve şifreli metin ekranda görüntülenecektir.



Şekil 6. MD5 Şifreleme Formu Ekranı

Şifrelenecek metnin görüntülenme biçimi "Format" alanına girilen değerle belirlenebilmektedir. CryptES

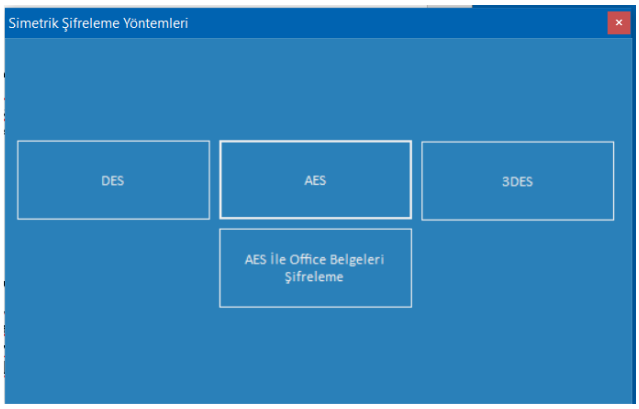
yazılımında, MD5 128-bit hash değeri üretimi .NET platformunda şifreleme işlemleri için kullanılan Windows Cryptographic Service Providers içerisindeki sınıflardan *MD5CryptoServiceProvider* sınıfı kullanılarak gerçekleştirilmiştir. Burada şifrelenecek metin Onaltılı Sayı (Hex) biçiminde görüntülenmek istenirse Format kutusuna "X" veya "x" yazılmaktadır. Bu alana Onlu Sayı için "D" veya "d", yazılarak şifrelenmiş metin onlu sayı biçiminde görüntülenebilmektedir. Böylece bu sınıfın bazı özelikleri şifreleme öğrenmek isteyenlere yol göstermek amacıyla kullanılabilmektedir (Şekil 7).



Şekil 7. MD5 Şifreleme Formu Örnek Uygulama Ekranları

3.2. Simetrik Şifreleme Örneği Olarak AES Uygulaması

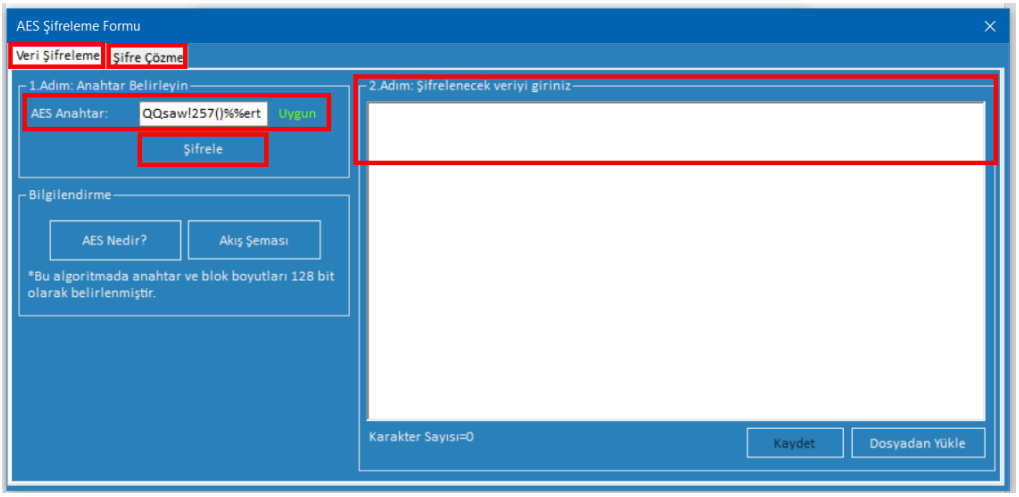
CryptES yazılımında simetrik şifrelemeye örnek olarak AES(Advanced Encryption Standard), DES ve 3DES şifreleme teknikleri sunulmaktadır. Kullanıcı "Simetrik Şifreleme Yöntemleri" penceresinde, AES, DES veya 3DES butonlarından birini tıklayarak ilgili şifreleme yöntemi ekranına geçecektir (Şekil-8). Bu çalışmada simetrik şifreleme örneği olarak AES algoritması ile şifreleme ve şifre çözme uygulaması verilmiştir.



Şekil 8. Simetrik Şifreleme Yöntemleri Ekranı

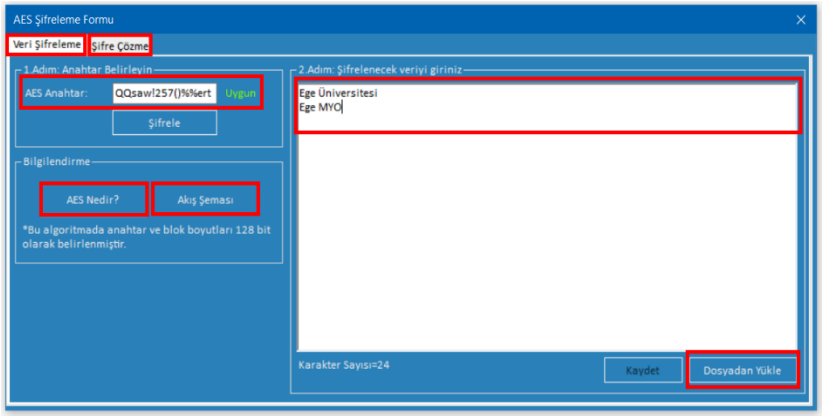
Kullanıcı "Simetrik Şifreleme Yöntemleri" ekranında AES butonunu tıkladığında "AES Şifreleme Formu" penceresi açılmaktadır (Şekil-9). Bu ekranda örnek bir AES metin şifreleme uygulamasının

gerçekleştirilmesinin yan ısıra kullanıcı bu şifreleme tekniğinin basit bir anlatımı (AES Nedir?) ve bu tekniğin işleyişinin gösterildiği bir akış diyagramı da bulunmaktadır.



Şekil 9. AES Şifreleme Formu Ekranı

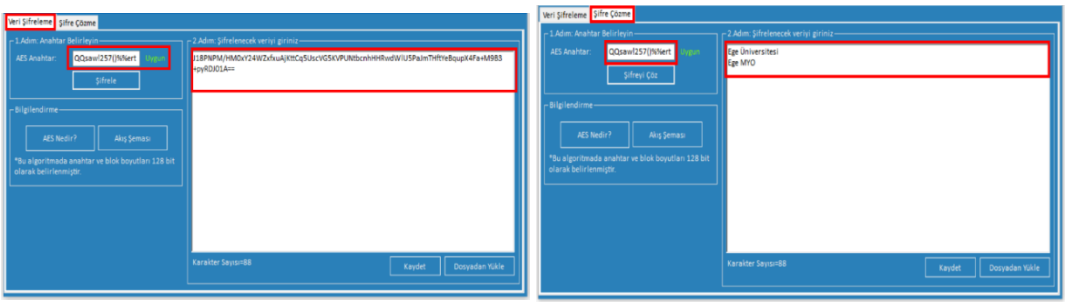
AES Şifreleme Formu "Veri Şifreleme" ve "Şifre Çözme" olmak üzere iki sekmeden oluşmaktadır. Kullanıcı şifrelemek istediği metni "Şifrelenecek Veri" alanına doğrudan klavye ile girebileceği gibi dosya aracılığı ile veri girişi yapabilmektedir. Şifrelenecek veri girişi yapıldıktan sonra kullanıcı tarafından bir AES Anahtarı belirlenmelidir. Kullanıcı "AES Anahtar" kutusuna bir anahtar değeri girmek zorundadır. Girilen anahtar uygun olduğunda program otomatik olarak anahtarın uygun olduğunu gösterir bir mesaj görüntülemektedir (Şekil 10).



Şekil 10. AES Şifreleme Formu Örnek Ekranı

AES Şifreleme Formu ekranında şifreleme süreci iki adımda gerçekleştirilmektedir. İlk adımda, kullanıcı şifreleme için bir anahtar girecektir. AES 128-bit şifreleme için .NET platformunda kullanılan Windows Cryptographic Service Providers içerisindeki sınıflardan *AesCryptoServiceProvider* sınıfı kullanılarak gerçekleştirilmiştir. Bu uygulamada örnek olarak anahtar ve blok boyutu 128-bit seçildiği için buna uygun bir anahtar değeri girilerek ikinci adıma yani şifrelenecek metin girişine geçilmektedir. 2. adımda şifrelenmek istenen metin veri girişi penceresine doğrudan girebileceği gibi istenirse dosya aracılığı ile de veri girişi yapılabilmektedir. Veri girişi yapıldıktan sonra "Şifrele" butonu tıklanarak girilen veri şifrelenerek ekranda görüntülenecektir (Şekil 11a). AES Şifreleme Formu' nun diğer önemli ekranı ise "Şifre Çözme" ekranıdır. Bu ekranda şifrelenmiş metnin şifre anahtarı ile yeniden açılarak başka bir deyişle şifresi çözülerek orijinal metnin tekrar elde edilmektedir. Şekil 11b' de görüldüğü gibi Şifre Çözme ekranına diğer ekrandan şifrelenen metin kopyalandıktan sonra şifrelemede kullanılan anahtar ile

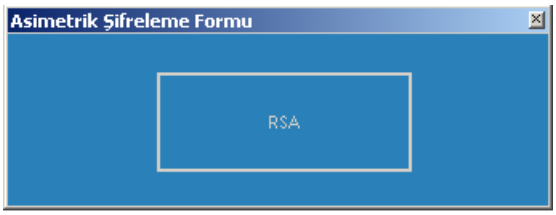
Şifreyi Çöz butonuna tıklandığında şifrelenmiş metin çözülerek orijinal metin görüntülenecektir.



Şekil 11. a) AES Şifreleme Formu Şifreleme Örnek Ekranı b) AES Şifreleme Formu Şifre Çözme Örnek Ekranı

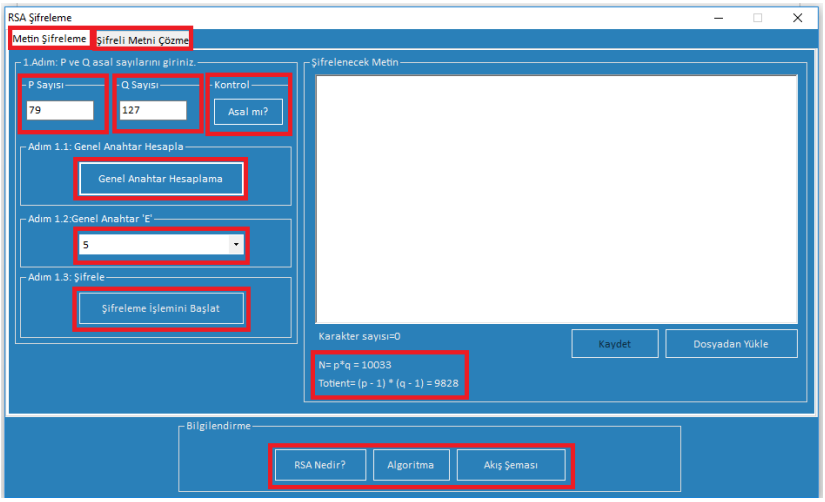
3.3. Asimetrik Algoritma Örneği Olarak RSA Uygulaması

Geliştirilen uygulamada asimetrik şifrelemeye örnek olarak RSA (Ron Rivest, Adi Shamir, Leonard Adleman) şifreleme tekniği verilmiştir. "Asimetrik Şifreleme Formu" penceresinde, RSA butonuna tıklandığında giriş ekranına geçilmektedir (Şekil 12).



Şekil 12. Asimetrik Şifreleme Formu Ekranı

Asimetrik Şifreleme Formu ekranında RSA butonuna tıklandığında RSA Şifreleme giriş ekranı görüntülenecektir (Şekil-13). Bu ekran Metin Şifreleme ve Şifreli Metin Çözme olmak üzere iki panelden oluşmaktadır. Diğer şifreleme tekniklerine ait uygulama ekranlarında olduğu gibi RSA Şifreleme ekranında bu tekniğin basit bir anlatımı (RSA Nedir?), algoritması ve bu tekniğin işleyişinin gösterildiği bir akış diyagramı bulunmaktadır. Şifrelemek istenen metin "Şifrelenecek Veri" alanına doğrudan girebileceği gibi dosya aracılığı ile de veri girişi yapılabilmektedir.



Şekil 13. RSA Şifreleme Formu Örnek Ekranı

RSA şifreleme tekniği, anahtarların planlanması ve şifreleme işlemlerinden oluşmaktadır. Bu

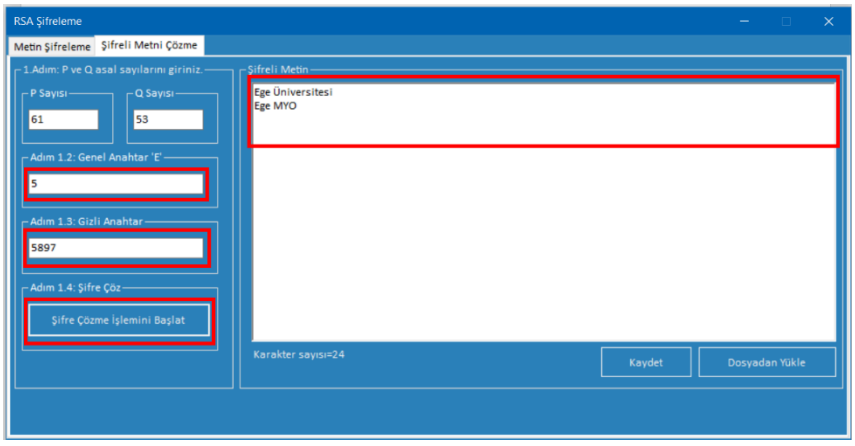
algoritmanın nasıl gerçekleştirildiğinin öğretimi amacıyla uygulamada RSA şifreleme süreci adımlar şeklinde tasarlanmıştır. Birinci adımda şifrelemede kullanılacak anahtarların planlanması için p ve q olmak üzere iki asal sayı girilmelidir. Kullanıcı tarafından girilen p ve q değerlerinin asal sayı olup olmadığı kontrol edilmekte ve uygun p ve q değerleri girilmiş ise Genel Anahtar Hesaplama butonuna tıklanarak anahtar oluşturma işlemleri gerçekleştirilmektedir (Şekil 14a). Genel Anahtar 'E' isimli liste kutusunda p ve q değerlerinin ardından yapılan matematik işlemleri ile elde edilen asal sayı listesi bulunmaktadır. Bu liste içerisinden seçilen bir sayı açık anahtar olarak kullanılacaktır (Şekil 14b). Şifrelenecek Metin alanına girilen veri için bu aşamada Şifreleme İşlemini Başlat butonuna tıklanarak şifreleme işlemi gerçekleştirilecektir (Şekil 14c).

a) b)

c)

Şekil 14. a) Açık Anahtar Hesaplama b) RSA Şifreleme Formu Örnek Ekranı c) RSA Şifreleme Formu Şifreleme Örnek Ekranı

Şifreli metini çözmek için kullanılan Şifreli Metni Çözme penceresi Şekil 15' de görüntülenmektedir. Kullanıcı, şifreleme ekranındaki şifrelenmiş metni bu penceredeki veri alanına kopyaladıktan sonra şifreleme sırasında kullandığı açık ve gizli anahtarları ilgili alanlara girmelidir. Bu aşamadan sonra Şifre Çözme İşlemini Başlat butonuna tıklanarak şifreli metin çözülecek ve orijinal metin görüntülenecektir.



Şekil 15. RSA Şifreleme Formu Şifreli Metni Çözme Örnek Ekranı

RSA algoritması ile şifreleme ve şifre çözme süreçlerinin işleyişi, şifreleme öğretimini desteklemek amacıyla adım adım yürütülebilecek şekilde tasarlanmıştır. Böylece RSA algoritmasının önemli adımlarından birisi olan anahtarların planlanması sürecinin anlaşılması ve ardından şifreleme işlemlerinin uygulanması sağlanmıştır.

4. Sonuçlar ve Öneriler

Bu çalışmada geliştirilen yazılım içinde bulunan ve şifrelemede çok sık kullanılan MD5, AES ve RSA şifreleme tekniklerine ilişkin örnek uygulamalara yer verilmiştir. CryptES yazılımı şifreleme ve şifre çözmede kullanılan hashing, simetrik ve asimetrik yöntemlerin kullanıldığı uygulamaları kapsamaktadır. Bu çalışmamızda geliştirilen yazılım içerisinde yer alan ve şifrelemede çok sık kullanılan MD5, AES ve RSA şifreleme tekniklerine ilişkin örnek uygulamalara yer verilmiştir. Programlama eğitimi alan kişi veya öğrenciler şifreleme ve şifre çözme konusunda kullanılan yöntemleri öğrenmek istediklerinde çok sayıda teorik bilgi ile karşılaşmaktadırlar. Bu bilgiler öğrenilmesi şifreleme işlemlerinin doğası gereği matematik temelli ve karmaşık yapıda olmasından dolayı zor olmaktadır. Bundan dolayı, şifreleme tekniklerinin öğrenilmesinde görsel ve simülasyon yazılım araçların kullanılması oldukça önemlidir. Çalışmamız kapsamında tasarlanan CryptES yazılımı metin şifreleme ve şifre çözme süreçlerinde yer alan bazı adımların kolay anlaşılmasını sağlamak amacıyla ilgili şifreleme ekranlarında etkileşimli arayüzler üzerinden kullanıcı ile iletişim kurabilmektedir. Böylece kullanıcı seçtiği şifreleme tekniğinin uygulama basamaklarını ve bu adımlarda şifreleme sürecinde neler yapılması gerektiğini görebilmektedir. Özellikle AES ve RSA şifreleme sırasında anahtarların oluşturulma sürecinin görsel bir araç üzerinden gerçekleştirilmesi öğrencilerin anahtar planlama işlemlerinin öğrenmesini kolaylaştıracak ve daha kısa süre içinde bu algoritmaların anlaşılmasını sağlayacaktır.

References

Akleylek, S. et. al., (2011). Kriptoloji ve Uygulama Alanları: Açık Anahtar Altyapısı ve Kayıtlı Elektronik Posta. Akademik Bilişim'11 Akademik Bilişim Konferansı Bildirileri, İnönü Üniversitesi, Malatya.

Algolak, H., (2014). Increasing the Security of Chatting Programs by Using a Hybrid of MD5, Shift and Diffie-Hellman Algorithms, M. Sc. Thesis, Gazi University Institute of Informatics.

Blomer, J. et. al. (2003). A new CRT-RSA algorithm secure against bellcore attacks, CCS'03 Proceedings of the 10th ACM conference on Computer and communications security, New York, USA. 311-320.

Boscher, A. et. al. (2007). CRT RSA Algorithm Protected Against Fault Attacks, Information Security Theory and Practices. Smart Cards, Mobile and Ubiquitous Computing Systems, Lecture Notes in Computer Science, 446, 229-243

Chaouch, A., Bouallegue, B., & Bouraoui, O. (2016). Software application for simulation-based AES, RSA and elliptic-curve algorithms. 2nd International Conference on Advanced Technologies for Signal and Image Processing (ATSIP'2016).

Chok, O. S., & Herath, S. (2004). Computer Security Learning Laboratory: Implementation of DES and AES Algorithms using Spreadsheets. In Proceedings of the 37th Midwest Instruction and Computing Symposium.

Diesburg, S. M., Wang, A. A. (2010). A survey of confidential data storage and deletion methods. ACM Computing Surveys, 43(1), 1–37.

Gençoğlu, H., (2017). Hibrid Şifreleme Algoritması, Doktora Tezi, Fen Bilimleri Enstitüsü, Trakya Üniversitesi.

Güzel, E. et. al., (2010). Şifreleme Eğitiminde Açık Kaynak Kodlu Araç Kullanımı: CrypTool. XII Akademik Bilişim Konferansı Bildirileri, Muğla Üniversitesi, Muğla.

Kakkar, A. et al. (2012). Comparison of Various Encryption Algorithms and Techniques for Secured Data Communication in Multinode Network, International Journal of Engineering and Technology, 2(1), 87-92.

Karnani, S., & Singh, C. P. (2015). Data Security Through Encryption Technique, International Journal Online of Science,III, 1-5.

Ma, J. et. al., (2016). AESvisual: A Visualization Tool for the AES Cipher. Proceedings of ACM Conference on Innovation and Technology in Computer Science Education.

Mahajan, P., & Sachdeva, A. (2013). A Study of Encryption Algorithms AES, DES and RSA for Security. Global Journal of Computer Science and Technology Network, 13(15).

Singh, G., & Supriya, S. (2013). A Study of Encryption Algorithms (RSA, DES, 3DES and AES) for Information Security, International Journal of Computer Applications, 67 (19), 33-38

Tao, J. et. al. (2011). DESvisual: A visualization tool for the DES cipher, Journal of Computing Sciences in Colleges, 27(1), 81-89.

Teodorescu, R. M., Lita, I., Cioc, I. B., & Visan, D. A. (2015). Virtual Instrumentation Application for Symmetrical and Asymmetrical Text Encryption/Decryption Studying, International Conference, 7th Edition Electronics, Computers and Artificial Intelligence.

Thambiraja, E. et. al., (2012). A Survey on Various Most Common Encryption Techniques. International Journal of Advanced Research in Computer Science and Software Engineering, 2(7).

Wang, X., & Yu, H. (2005). How to Break MD5 and Other Hash Functions. EUROCRYPT'05 Proceedings of the 24th Annual International Conference on Theory and Applications of Cryptographic Techniques. LNCS, vol. 3494, Springer, Heidelberg 19-35.