E-POSTA GÜVENLİĞİ VE UYGULAMASI

HALİL İBRAHİM ÜLGEN

YÜKSEK LİSANS TEZİ
ELEKTRİK ELEKTRONİK MÜHENDİSLİĞİ

GAZİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

HAZİRAN 2008
ANKARA

Halil İbrahim ÜLGEN tarafından hazırlanan E-POSTA GÜVENLİĞİ VE UYGULAMASI adlı bu tezin Yüksek Lisans tezi olarak uygun olduğunu onaylarım.

Doç. Selma YÜNCÜ ……

Tez Danışmanı, Elektrik-Elektronik Mühendisliği Anabilim Dalı

Bu çalışma, jürimiz tarafından oy birliği ile Elektrik-Elektronik Mühendisliği Anabilim Dalında Yüksek lisans tezi olarak kabul edilmiştir.

Prof. Dr. Şeref SAĞIROĞLU ……

Bilgisayar Mühendisliği Anabilim Dalı, Gazi Üniversitesi

Doç. Selma YÜNCÜ ……

Elektrik-Elektronik Mühendisliği Anabilim Dalı, Gazi Üniversitesi

Yrd. Doç. Dr. Erkan AFACAN ……

Elektrik-Elektronik Mühendisliği Anabilim Dalı, Gazi Üniversitesi

Tarih : 26/06/2008

Bu tez ile G.Ü. Fen Bilimleri Enstitüsü Yönetim Kurulu Yüksek Lisans derecesini onamıştır.

Prof. Dr. Nermin ERTAN ……

Fen Bilimleri Enstitüsü Müdürü

TEZ BİLDİRİMİ

Tez içindeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edilerek sunulduğunu, ayrıca tez yazım kurallarına uygun olarak hazırlanan bu çalışmada orijinal olmayan her türlü kaynağa eksiksiz atıf yapıldığını bildiririm.

Halil İbrahim ÜLGEN

E-POSTA GÜVENLİĞİ VE UYGULAMASI

(Yüksek Lisans Tezi)

HALİL İBRAHİM ÜLGEN

GAZİ ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

Haziran 2008

ÖZET

Bu çalışmada, e-posta iletişimi, e-posta iletişim kuralları, e-posta kullanımında karşılaşılabilecek güvenlik zafiyetleri incelenmiş ve bu zafiyetlerin giderilmesi amacıyla GÜ-Posta adlı güvenli bir e-posta uygulaması geliştirilmiş ve sunulmuştur. Geliştirilen uygulamada kullanılan araçlar ve tekniklerden özet olarak bahsedilmiş, konunun daha anlaşılır olmasını sağlamak amacıyla gerekli temel bilgiler verilmiştir. GÜ-Posta uygulamasında düz mesaj metni bir şifreleme algoritması ile şifrelenir. Daha sonra şifrelemede kullanılan anahtar da RSA algoritması ile (alıcının genel anahtarı kullanılarak) şifrelenir ve oluşturulan e-posta gönderilir. Alıcı e-postayı aldığında, öncelikle şifreli olan anahtarı RSA algoritmasını (kendi özel anahtarı ile birlikte) kullanarak deşifre eder. Elde edilen anahtar mesajın deşifre edilmesinde kullanılır. Bu esnek uygulamada, kullanıcı tercihine göre mesaj DES, 3DES, RC2 veya AES gibi simetrik şifreleme algoritmalarıyla şifrelenebilmektedir. Ayrıca gizlilik, bütünlük, kimlik kanıtlama ve inkar edememezlik gibi güvenliğin temel unsurları sağlanmaktadır.

Bilim Kodu : **905.1.011**
Anahtar Kelimeler : **E-posta, e-posta güvenliği, kriptografi**
Sayfa Adedi : **88**
Tez Yöneticisi : **Doç. Selma YÜNCÜ**

E-MAIL SECURITY AND ITS IMPLEMENTATION

(M.Sc. Thesis)

HALİL İBRAHİM ÜLGEN

GAZİ UNIVERSITY

INSTITUTE OF SCIENCE AND TECHNOLOGY

June 2008

ABSTRACT

In this study, e-mail communication, e-mail protocols and security weaknesses of e-mail usage are examined and a secure e-mail application, GÜ-Posta, is prepared for elimination of these weaknesses. Tools and techniques used in the developed application are summarized and the necessary basic information is given to make the subject more understandable. In GÜ-Posta application, plain message text is encrypted by a cipher. Later on, the key used for encryption is encrypted by RSA algorithm (with using recipient public key) and constituted e-mail is sent. When the receiver gets the e-mail, first of all, he decrypts the encrypted key with using RSA algorithm (with using his private key). Retrieved key is used for decryption of the cipher text. In this flexible application, by the user's preference, message can be encrypted with symmetric encryption algorithms such as DES, 3DES, RC2 or AES. Furthermore, basic elements of security such as confidentiality, integrity, authentication and non-repudiation are ensured.

Science Code : 905.1.011
Key Words : E-mail, e-mail security, cryptography
Page Number: 88
Adviser : Assoc. Prof. Selma YÜNCÜ

TEŞEKKÜR

Çalışmalarım boyunca değerli yardım ve katkılarıyla beni yönlendiren Sayın hocalarım Doç. Selma YÜNCÜ ve Prof. Dr. Şeref SAĞIROĞLU'na, ayrıca beni hep destekleyen ve her zaman yanımda olan değerli aileme teşekkürü bir borç bilirim.

İÇİNDEKİLER

Sayfa

Sayfa

Sayfa

ÇİZELGELERİN LİSTESİ

Çizelge **Sayfa**

ŞEKİLLERİN LİSTESİ

Şekil **Sayfa**

Şekil **Sayfa**

SİMGELER VE KISALTMALAR

Bu çalışmada kullanılmış bazı simgeler ve kısaltmalar, açıklamaları ile birlikte aşağıda sunulmuştur.

Simgeler	**Açıklama**
D	Deşifreleme
E	Şifreleme
K	Anahtar

Kısaltmalar	**Açıklama**
3DES	3'lüVeri Şifreleme Standardı
AAA	Açık Anahtar Altyapısı
AES	Gelişmiş Şifreleme Standardı
DES	Veri Şifreleme Standardı
IBE	Kimlik Temelli Şifreleme
IETF	İnternet Mühendislik çalışma Kolu
IMAP	İnternet İleti Erişim Protokolü
MD5	Mesaj Özetleme Algoritması
MIME	Çok amaçlı İnternet Posta Eklentileri
MOSS	MIME Nesne Güvenlik Servisleri
PEM	Gizliliği Arttırılmış Posta
POP3	Posta Merkezi Protokolü 3
RC2	Rivest Kod 2
S/MIME	Güvenli/Çok amaçlı İnternet Posta Eklentileri
SHA	Güvenli Özet Algoritması
SMTP	Basit Posta Transfer Protokolü
XML	Genişletilebilir İşaretleme Dili

1. GİRİŞ

Sürekli gelişen teknoloji ile birlikte insanoğlunun en temel ihtiyaçlarından biri olan iletişim ihtiyacı da zamanla şekil değiştirmiştir. Her şeyin son derece hız kazandığı günümüzde iletişim yolları da bu hıza uymak zorunda kalarak yazılı iletişim aracı türlerinden biri olan e-posta doğmuştur. E-posta kişilerin sayısal ortamda hazırladıkları ve karşılıklı alışverişte bulunabildikleri iletilere verilen addır. İnternet ile birlikte hızlı bir şekilde gelişerek popülerlik ve önem kazanmıştır.

Birçok kullanıcı için e-posta iletişimi iş ve günlük yaşam içerisinde vazgeçilmezler arasına girmiş ve bilgisayar kullanımında yapılan en önemli etkinlik haline gelmiştir. Yaşanan tüm bu gelişme ve değişim beraberinde de birçok güvenlik sorununu getirmiştir.

Bu yüksek lisans tez çalışmasında, e-posta iletişimi, e-posta iletişim kuralları, e-posta kullanımında karşılaşılabilecek güvenlik zafiyetleri incelenmiş ve bu zafiyetlerin giderilmesi amacıyla geliştirilen GÜ-Posta adlı e-posta uygulaması sunulmuştur.

GÜ-Posta uygulamasında düz mesaj metni bir şifreleme algoritması ile şifrelenir. Daha sonra şifrelemede kullanılan anahtar da RSA algoritması ile (alıcının genel anahtarı kullanılarak) şifrelenir ve oluşturulan e-posta gönderilir. Alıcı e-postayı aldığında ise, öncelikle şifreli olan anahtarı RSA algoritmasını (kendi özel anahtarı ile birlikte) kullanarak deşifre eder. Elde edilen anahtar ise mesajın deşifre edilmesinde kullanılır. Bu esnek uygulamada, kullanıcı tercihine göre mesaj DES, 3DES, RC2 veya AES gibi simetrik şifreleme algoritmalarıyla şifrelenebilmektedir. Ayrıca gizlilik, bütünlük, kimlik kanıtlama ve inkar edememezlik gibi güvenliğin temel unsurları sağlanmaktadır.

İkinci bölümde, e-posta, e-postanın kullanım alanları, neden olabileceği güvenlik zafiyetleri, kullanılan iletişim kurallarından bahsedilmiş daha sonra ise güvenlik gereksinimleri ve kriptografik yaklaşımlardan bahsedilerek konunun daha iyi anlaşılabilmesi amacıyla temel bilgiler verilmeye çalışılmıştır.

Üçüncü bölümde e-posta güvenliği kapsamında kullanılmakta olan yöntemlere değinilmiş; IBE, S/MIME, PEM, MOSS ve PGP konusunda bilgi verilmiştir.

Dördüncü bölümde geliştirilmiş olan e-posta sisteminin (GÜ-Posta) tasarlanması aşamasında kullanılan teknoloji ve yöntemler ile uygulamanın geliştirme süreçleri, mimarisi ve bileşenleri konularına değinilmiştir.

Son bölümde ise GÜ-Posta uygulamasının kullanıcı ara yüzü tanıtılmış, kullanımına yönelik çeşitli bilgiler verilmiştir.

2. TEMEL BİLGİLER

2.1. E-posta

Her şeyin son derece hız kazandığı günümüzde iletişim yolları da bu hıza uymak zorunda kalarak yazılı iletişim aracı türlerinden biri olan e-posta doğmuştur.

2.1.1. E-posta nedir

E-posta kişilerin sayısal ortamda hazırladıkları ve karşılıklı alışverişte bulunabildikleri iletilere verilen addır. Türk Dil Kurumu sözlüğünde e-posta "Bilgisayarlar veya bir ağ içindeki belli gönderim merkezleri arasında elektronik bilgi iletişimi, elmek." şeklinde tanımlanmıştır. Bilgi ve belgelerin kolaylıkla ve vakit geçirmeden hızlıca iletimini sağlar. İnternet ile birlikte hızlı bir şekilde gelişerek popülerlik ve önem kazanmıştır.

"Darwin Magazine:Prime Movers" dergisine göre ilk e-posta mesajı 1971 yılında Ray Tomlinson adlı bir mühendis tarafından gönderilmiştir. İlk önceleri sadece bir makinedeki kullanıcılara mesajlar gönderilebiliyor iken Tomlinson'un büyük buluşu ile internet üzerinden diğer makinelere de mesajlar gönderilebiliyordu. Bunun için alıcı makine "@" işareti ile belirtilmekteydi.

E-posta adresi Internet servis sağlayıcısı (ISS) tarafından verilir. Internet yoluyla size gönderilen e-posta iletisinin sizi bulmasını sağlayan iki önemli bilgi parçası vardır. Bunlardan ilki, ibrahimulg@hotmail.com adresindeki " ibrahimulg " kullanıcı adı gibi bir kullanıcı adı veya @ simgesinin önündeki parçadır. @ simgesinden sonra ise etki alanı veya ana bilgisayar adı gelir. Etki alanı adı, cadde veya kent bilgisine benzer. E-posta iletisinin nereye yönlendirileceğini gösterir. E-posta iletisi etki alanınıza ulaştıktan sonra, kullanıcı adınız ev adresine benzer bir işlev görür. ISS'nizin iletiyi sizin posta kutunuza yönlendirmesine olanak tanır.

2.1.2. Kullanım alanları

Birçok kullanıcı için e-posta iletişimi iş ve günlük yaşam içerisinde vazgeçilmezler arasına girmiş ve bilgisayar kullanımında yapılan en önemli etkinlik haline gelmiştir.

E-posta kullanıcısı sayısı 2007 yılında 1,2 milyar iken 2011 yılında yıllık %7'lik bir büyüme hızı ile 1,6 milyar kişiye ulaşacağı tahmin edilmektedir [1].

İşyerlerinde de gerek iç gerekse dış bilgi paylaşımında e-posta mesajlarının önemli bir yeri vardır. Rapor ve planlar ile artık resmi belgeler de e-posta ile taşınmaktadır.

Bugün ortalama bir çalışanın günlük 156 adet mesaj gönderip aldığı ve bu sayının 2012 yılında günlük 233'e ulaşacağı tahmin ediliyor [2]. Ayrıca iş kullanıcıları arasında yapılan bir araştırma sonucu çalışma zamanının %19'nu ya da 2 saat/gün'e yakın bir zaman harcadıkları görülmüştür [3]. 100 kişiden oluşan göreceli olarak ufak bir şirket için bile bu durum 2008 yılında uğraşılması gereken 5,7 milyon e-posta iletisi anlamına gelmektedir.

2.1.3. Güvenlik zafiyetleri

Kişisel, kurumsal ya da herhangi bir sistem güvenliği üzerinde etkili olan unsurları bir zincire benzetirsek, tüm güvenlik bu zincirin en zayıf halkasının güvenliği kadardır [4]. E-posta iletişimi de gerekli önlemler alınmadığı takdirde zincirin zayıf halkası olmaya adaydır.

E-posta göndermek mektup göndermek gibidir. Tek fark tüm işlemlerin sayısal ortamda gerçekleşiyor olmasıdır. Kullanıcının gönderdiği bir ileti gönderici ve alıcının posta sunucularından başka birçok bilgisayardan geçer. İletinin geçtiği bu güzergâhların güvenliği direkt olarak bizim iletimizin güvenliğini belirler.

E-posta kullanımının sağladığı kolaylıkların yanında, eğer gerekli önlemler alınmaz ise sıkıntılara ve istenmeyen durumlara yol açması kaçınılmazdır.

Bunlardan bazıları aşağıda sıralanmıştır:
Kimlik Hırsızlığı (identity theft): Eğer saldırgan e-posta kullanıcı adınızı ve parolanızı elde ederse e-posta sunucusuna erişerek mesajlarınız üzerinde her türlü işlemi gerçekleştirebilir. Kişisel bilgilerinizi öğrenebilir. Kullanıcı parolasının korunması, tahmin edilmesi güç parola seçimi ve parolaların belirli aralıklarla değiştirilmesi alınabilecek en basit önlemlerdendir.

İzinsiz erişim (eavesdropping): Saldırgan gönderici ile alıcı arasında mesaj trafiğini dinleyebilir. Bu tehdit ağ trafiğinin dinlenmesi veya gönderici/alıcı bilgisayarlarında olabilecek casus yazılımlar ile gerçekleştirilebilir.

Değişiklik yapma (message modification): Saldırgan izinsiz erişim yaptıktan sonra gönderilen mesajın içeriğini de değiştirebilir.

Korumasız durumdaki yedekler (unprotected backups): Kullanılan bilgisayar ya da sunucular e-posta mesajlarını yedekleyebilir ve sistem yöneticisi de düz metin halinde saklanan mesajlara kolay bir şekilde erişebilir. Hatta silindiğini düşündüğünüz mesajlar bile yedeklenmiş olarak yıllarca saklanabilir [5].

SMTP ve POP3 kaynaklı güvenlik problemleri: SMTP ve POP3 kaynaklı güvenlik problemlerinin en önemlisi iletişimde mesaj metninin düz metin olarak tutulması ve güçlü kimlik denetimi kontrollerinin olmamasıdır [5].

Korumasız ağ ortamı: Mesaj trafiğinin geçtiği ortamlardaki her türlü ağ cihazları ve yazılımlarda olabilecek açıklar ile bilgi işlem çalışanlarının göstermesi gerekli ilgi, alaka ve dikkati göstermemeleri mesaj trafiğinde saldırganlar (hackers, newbie, crackers vb kişiler) tarafından gizliliğin ve bütünlüğün ihlal edilmesine sebep olabilir.

Sazan avlama (phishing): Kimlik avcıları, genelde e-posta vb. yollarla kişilere ulaşır ve onların kredi kartı numarası ve benzeri bilgilerini sanki resmi bir kurummuş gibi isterler. Bu "av"a karşılık veren kullanıcıların da hesapları, şifreleri gibi özel bilgileri

çalınır. 2005 yılında Amerika Birleşik Devletleri'nde 2,4 milyon tüketicinin sazan avlama saldırıları (phishing) sonucu etkilendiği, 1,2 milyon tüketicinin toplamda 929 milyon dolara yakın maddi kayıplarının olduğu belirlenmiştir [6]. Sazan avlama saldırılarına karşı bankalar ve benzeri kurumlar hiç bir zaman kullanıcılarından e-posta aracılığı ile özel bilgilerini istemeyeceklerini, böyle bir durumda kaldıklarında kullanıcılarına kendilerini bilgilendirmeleri gerektiğini bildirirler.

E-posta hizmetlerinin verilmesi esnasında; sahtecilik, aldatma, mesaj başlığının ve içeriğinin değiştirilmesi, vb birçok yöntemler kullanılarak farklı kaynaktan, farklı içerikli mesaj gönderilmesi, alıcı ve göndericinin yanıltılması mümkün olabilmektedir [7].

Bu tehditlerin özellikle iş amaçlı kullanılan e-posta trafiğinde yaşanması doğabilecek zararları daha da arttırmaktadır.

Bazı ülkelerde devlet kurumları da çeşitli araçlar kullanarak e-posta trafiğini takip etmektedir. Suç ve suçluların takibinde e-posta özellikle 11 Eylül 2001 tarihinden sonra daha da önem kazanmıştır.

Yurt dışında önemli birçok kurum ve kuruluş işyerinde internet ve e-posta kullanımı ile ilgili politikalar hazırlamakta ve uygulanmasını da sıkı bir şekilde takip etmektedir [8]. Örneğin Nissan Motor şirketinin iki çalışanı yönetimin cinsel içerikli mesajlar gönderip aldıklarını fark etmesi üzerine işlerinden atılmıştır [9].

Yapılan araştırmalar kullanıcıların e-posta gizliliğine ilişkin beklentilerinin en azından dört makul açıklaması olduğu göstermiştir [10].

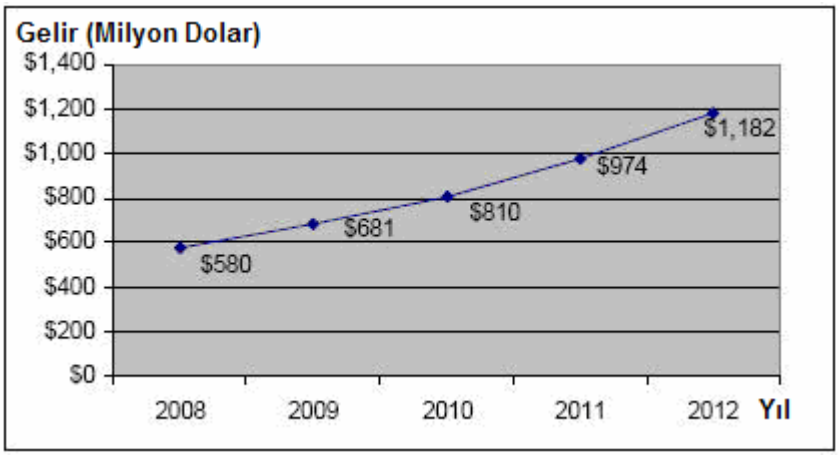
- Teknolojik görüş: Bazı kullanıcılar e-postanın; kullanıcı giriş prosedürlerinin ve parola olmasının verdiği yanılsama ile güvenli olduğunu düşünüyor. Parola kullanılıyor olması sayesinde hiç kimsenin e-posta mesajlarına erişemeyeceğini düşünüyorlar. Ayrıca silinen bir e-postanın varlığına dair hiçbir izin kalmayacağının düşünülmesi de düşülen hatalardan biridir.

- Mektup ile örnekseme: Kullanıcılar, geleneksel mektupla yapılan iletişimde olduğu gibi bir korumanın e-posta için de yapıldığını düşünebiliyorlar. Bir başkasına yollanan bir mektubu açmanın yasak olduğunu düşündükleri için aynı şekilde e-postanın da güvenli olduğunu varsayıyorlar.
- Yanlış yönlendirilmiş güvenlik hissi: E-postanın yarattığı gizlilik yanılsaması ile kimi kişilerin kendilerini sosyal bağlamda daha açık olarak ifade etmeleri ileriki zamanda o kişiler açısından sıkıntılara yol açabilmektedir.
- Organizasyonel görüş: E-postanın doğru, uygun kullanılmasına yönelik düzenlemelerin var olması çalışanların e-postanın özel olduğuna inanmalarını sağlayabilmektedir.

İstenilmeden gönderilen, çoğunlukla çok sayıda kişiye adreslenen, duyuru, ticari ilan vb iletilere spam denilmektedir. Yapılan araştırmalarda Avrupa e-posta trafiğindeki spam ileti sayısının sürekli arttığı görülmektedir. 2006 yılında 16 milyar olan spam ileti sayısının 2010 yılında 38 milyara çıkacağı tahmin edilmektedir. Spam mesajlar tüm avrupadaki e-posta mesajlarının %62'sini oluşturmaktadır [11]. E-posta ağlarına ve internete binen yükün azaltılmasında ve korunmasında spam mesajlardan korunmak çok önemli bir hale gelmiştir.

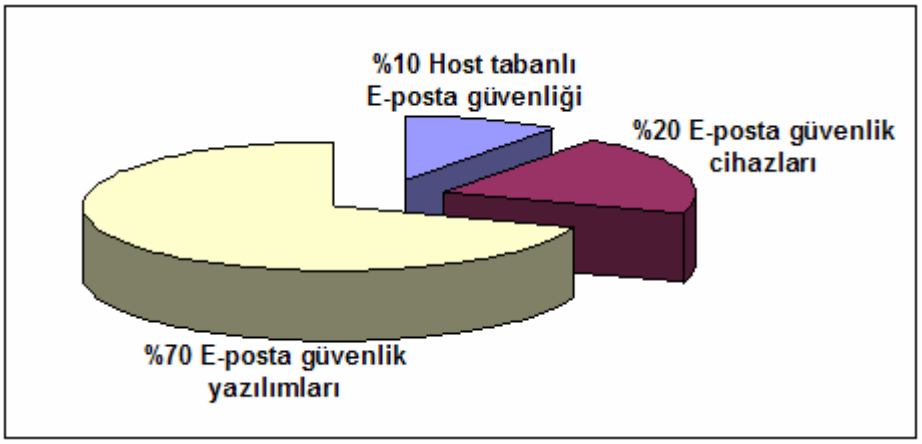
E-posta güvenliğini tehdit eden unsurları iç ve dış tehditler olarak da ikiye ayırabiliriz. İç tehditler; ihmalkâr ve kötü niyetli çalışanlar ile ihlal edilmiş süreçlerdir. Dış tehditler ise; bilgisayar korsanları, rakipler ve casuslardır. Hem iç hem de dış tehdit olarak görebileceklerimiz ise iş ortakları, alt yükleniciler ve eski çalışanlardır.

E-posta güvenliğinin sağlanması amacıyla ortaya çıkan ürünler ve hizmetler büyük bir pazarın oluşmasına neden olmuştur. Yapılan araştırmaya göre Asya Pasifik ülkelerinde e-posta güvenliği pazarının 2008 yılında 580 milyon dolara ulaşacağı ve 2012 yılında yaklaşık 1,2 milyar dolara kadar büyüyeceği tahmin edilmektedir (Şekil 2.1) [12].



Şekil 2.1. E-posta güvenliği gelirleri tahminleri (2008-2012)

E-posta güvenliği kapsamında kullanılan araçların pazar gelirlerinin dağılımı ise Şekil 2.2'de gösterilmiştir.



Şekil 2.2. E-posta güvenlik araçlarına göre pazar gelirlerinin dağılımı [13]

Karşılaşılabilecek olumsuz durumları azaltmak için elektronik ortamlarda saklanan, gönderilen ve alınan bilgilerin güvenilir olması, başkaları tarafından okunamaması ve elde edilememesi, gönderenin ve alanın kimliğinin doğrulanması, taşınan bilgilerin içeriğinin değişmemesi ve elde edilse bile içeriğinin anlaşılamamasının sağlanması gereklidir.

E-posta güvenliği açısından günümüzde en çok kullanılan yöntem ve araçlar S/MIME ve PGP'dir. Bir çok e-posta istemcisinin S/MIME için desteği vardır.

Şifreleme, açık anahtar altyapısı, sertifika ve anahtar gibi kavramlara çok aşina olmayan kullanıcılar için bile uygun bir yöntem olarak değerlendirilmesine rağmen pratik kullanımının kolay olmadığı yapılan çeşitli çalışmalarda görülmüştür [14].

E-posta programlarının birçoğunda şifreli mesajlar göndermek için alıcı ile sertifika (.cer dosyaları) alış verişi yapılması gerekir. Sertifikalar, "Sertifika Otoriteleri" (certificate authority-CA)'tarafından üretilirler. Sertifika otoriteleri belirli kurallara ve güven modellerine [15] göre yapılandırılmış kurum ve kuruluşlardır. Yaptıkları işler açısından imza onaylayan notere ve kimlik üreten makamlara benzerler. Sertifika otoritelerinden hizmet almak da belirli bir ücrete tabidir. Bu yüzden güvenli iletişim yapmak isteyen ama bunun için para harcamak istemeyen birçok kullanıcı bu yöntemi tercih etmemektedir. Ayrıca harcanan para ve çabanın korunmak istenen bilginin değerine değmesi gerektiği gerçeği de unutulmamalıdır.

2.2. E-Posta İletişiminde Kullanılan İletişim Kuralları

2.2.1. SMTP

E-posta gönderme protokolüdür. SMTP kısaltması "Simple Mail Transfer Protocol" den gelmektedir. Bu protokol e-posta göndermek için sunucu ile istemci arasındaki iletişim şeklini belirler. Sadece e-posta yollamak için kullanılan bu protokolde, basitçe, istemci bilgisayar SMTP sunucusuna bağlanarak gerekli kimlik bilgilerini gönderir, sunucunun onay vermesi halinde gerekli e-postayı sunucuya iletir ve bağlantıyı sonlandırır.

SMTP'nin doğuşu ve gelişimi

Ağ ortamında ilk e-posta çalışmaları 1971 yılında bir DARPA (Defense Advanced Research Projects Agency) olan ARPANET (Advanced Research Projects Agency Network) ile başlamıştır. Modern SMTP çalışması ise 1980 yılında Dr. Jon Postel tarafından bir RFC ile başlamış sayılır.

RFC “Request for Comments” ifadesinin bir kısaltmasıdır. RFC dokümanları internet teknolojilerine uygulanabilir yeni araştırma, buluş ve metodolojileri kuşatan bildiriler serisidir.

RFC 772, Posta Aktarım İletişim Kuralı (MTP-Mail Transfer Protocol) posta protokolünü FTP’den (File Transfer Protocol) bağımsız yapma yolunda bir teşebbüstü. 1981 yılında ise RFC 788 ile Dr. Jon Postel SMTP’yi tanıttı. SMTP 25 numaralı portu kullanıyordu ve bugün de kullanılan tüm komutlara sahipti. Daha sonraki dönemde ise yayımlanan RFC821 (SMTP) ve RFC 822 (Internet Mail Format) ile e-posta için yeni bir çağ başlamış oldu. SMTP nin en son hali 2001 yılında J. Klensin’in editörlüğünde yayımlanan RFC 2821 de tanımlanmıştır. Yayımlanan tüm dokümanlar Çizelge 2.1’de verilmiştir.

Çizelge 2.1. Yayımlanmış POP RFC dokümanları

Doküman No	Doküman Başlığı	Yayınlanma Yılı
RFC 0821	Basit posta transfer protokolü (STD0010)	1982
RFC 0822	İnternet mesaj biçimi (STD0011)	1982
RFC 1123	İnternet sunucu bilgisayar gereksinimleri (STD0003)	1989
RFC 1652	8 bit MIME için SMTP eki	1994
RFC 1869	SMTP servis ekleri (ESMTP)	1995
RFC 1870	Mesaj boyutu bildirimi için SMTP eki	1995
RFC 1985	Uzak mesaj kuyruğu başlangıcı için SMTP eki	1996
RFC 2034	Gelişmiş hata kodları için SMTP eki	1996
RFC 2045	Çok amaçlı internet posta ekleri (MIME) #1	1996
RFC 2046	Çok amaçlı internet posta ekleri (MIME) #2	1996
RFC 2047	Çok amaçlı internet posta ekleri (MIME) #3	1996
RFC 2048	Çok amaçlı internet posta ekleri (MIME) #4	1996
RFC 2049	Çok amaçlı internet posta ekleri (MIME) #5	1996
RFC 2142	Ortak hizmetler ve roller için posta kutusu isimleri	1997
RFC 2183	İçerik düzenleme için başlık alanı	1997
RFC 2298	Mesaj düzenleme uyarıları	1998
RFC 2476	Mesaj ibrazı	1998
RFC 2505	SMTP MTA’ları için anti spam tavsiyeleri	1999
RFC 2554	Kimlik doğrulama için SMTP servis ekleri	1999
RFC 2821	Basit posta transfer protokolü	2001
RFC 2822	İnternet mesaj biçimi	2001
RFC 2920	Komut işlenmesi SMTP eki(STD0060)	2000
RFC 3030	Geniş ve ikili MIME SMTP ekleri	2000
RFC 3207	TLS üzerinde güvenli SMTP için SMTP ekleri	2002
RFC 3461	Teslim durum uyarıları için SMTP eki	2003
RFC 3463	Gelişmiş posta sistem durum kodları	2003

SMTP'nin genel özellikleri

SMTP protokolü açık, anlaşılması kolay bir protokoldür. Varsayılan port olarak 25 numaralı portu kullanır.
Güncel olarak kullanılan SMTP komutları ve karşılıkları aşağıdaki gibidir:
Komutlar:
EHLO etki alanı
HELO etki alanı (geriye uyumluluk amaçlı olarak kullanılmaktadır)
MAIL FROM: geri dönüş-yolu [posta-parametreleri]
RCPT TO: gidiş yolu [posta-parametreleri]
DATA
RSET
SEND FROM: geri dönüş-yolu
SOML FROM: geri dönüş-yolu
SAML FROM: geri dönüş-yolu
VRFY metin
EXPN metin
HELP [metin]
NOOP [metin]
QUIT

Minimum gerçekleştirim aşağıdaki komutları içermelidir.
EHLO
HELO
MAIL
RCPT
DATA
RSET
NOOP
QUIT
VRFY

2.2.2. POP3

POP3 TCP/IP bağlantısı üzerinden e-posta sunucusu üzerindeki e-posta mesajlarına erişmeyi sağlayan bir iletişim kuralıdır. POP3 kısaltması "Post Office Protocol version 3" ifadesinden gelmektedir.

POP3'ün doğuşu ve gelişimi

POP3 iletişim kuralının geliştirilmesine 1984 yılında başlanmıştır. J.K. Reynolds tarafından yayımlanan RFC 918'i sonraki yıllarda sırası ile RFC 937, RFC 1081, RFC 1939 izlemiştir. Mayıs 1996 tarihinde ise İnternet Standardı (Internet STD 0053) olmuştur. Yayımlanan tüm dokümanlar Çizelge 2.2'de verilmiştir.

Çizelge 2.2. Yayımlanmış POP RFC dokümanları

Doküman No	Doküman Başlığı	Yayınlanma Yılı
RFC 0918	POP	1984
RFC 0937	POP2	1985
RFC 1081	POP3	1988
RFC 1082	POP3 genişletilmiş hizmet önerileri	1988
RFC 1225	POP3	1991
RFC 1460	POP3	1993
RFC 1725	POP3	1994
RFC 1734	POP3 kimlik doğrulama komutu	1994
RFC 1939	POP3 (STD0053)	1996
RFC 1957	POP3 (bilgi niteliğinde)	1996
RFC 2095	IMAP/POP yetkilendirme eki	1997
RFC 2195	IMAP/POP yetkilendirme eki	1997
RFC 2384	POP URL planı	1998
RFC 2449	POP3 ek mekanizması	1998
RFC 2595	POP3 ile TLS'in kullanılması	1999
RFC 3206	POP yanıt kodları (SYS,AUTH)	2002

POP3'ün genel özellikleri

POP3 servis sağlayıcılardan e-posta alımı/erişimi için kullanılan en genel metottur. POP3 genellikle bilgisayarın 110 numaralı portunu kullanır.
Minimum Gerçekleştirimde kullanılan POP3 komutları:
USER isim (yetkilendirme durumunda geçerlidir)
PASS parola

QUIT

STAT (işlem durumunda geçerlidir)
LIST [mesaj]
RETR mesaj
DELE mesaj
NOOP
RSET
QUIT

İsteğe bağlı POP3 komutları:
APOP isim özeti (yetkilendirme durumunda geçerlidir)
AUTH [yetkilendirme türü]
TOP mesaj n (işlem durumunda geçerlidir)
UIDL [mesaj]

POP3 cevapları:
+OK
-ERR

Genişletilmiş komutlar:
CAPA (yetkilendirme ya da işlem durumunda geçerlidir)
STLS (yetkilendirme durumunda geçerlidir)

Genişletilmiş Yanıt Kodları: Bu kodlar –ERR cevabını takip eden köşeli parantez içinde eklenir.
LOGIN-DELAY
IN-USE
SYS/TEMP
SYS/PERM
AUTH

POP3, sunucu ile olan bağlantının devamlı surette sürdürülemediği durumlar için kullanılmaktadır. Bunun için kullanıcı sunucuya bağlanıp e-posta mesajlarını bilgisayarına aktarır ve sonrasında da bağlantıyı koparır. Daha sonra sunucuya bağlanmadan da bu e-postaları görüntüleyebilir veya değiştirebilir.

2.2.3. IMAP

IMAP İnternet İleti Erişim Protokolü (Internet Message Access Protocol), bir e-posta iletişim protokolüdür. 1986 yılında Stanford Üniversitesi'nde geliştirilmiştir. IMAP, POP3 protokolüne alternatif olarak geliştirilmiş bir protokoldür. E-posta mesajlarının sunucuda tutulmasını ve gerektiğinde sunucudan erişilmesini sağlar.

IMAP'ın POP3'e göre birçok avantajı vardır:

- Bir e-posta sunucusuna POP3 ile bağlanıldığında bütün yeni mesajlar istemciye çekilir ve bağlantı kapatılır. IMAP kullanıldığında ise oturum açıldıktan sonra bağlantı sadece istek olduğu durumlarda açık kalır.
- POP3 aynı posta kutusunda aynı anda tek kullanıcıyı destekler. IMAP'ın ise çoklu kullanıcı desteği vardır. Bir kullanıcının yaptığı değişiklik eşzamanlı olarak diğer oturum açmış kullanıcı tarafından görülebilir.
- IMAP, MIME mesajlarına parçasal erişim imkânı sunar.
- IMAP ile mesaj durum bilgisi (mesaj okundu, okunmadı, cevaplandı, silindi, vb.) elde edilebilir. Çoklu kullanıcı kullanımında olan posta kutularında eş zamanlı olarak güncelleme yapılır.
- IMAP kullanıcılara posta kutusunu özelleştirme izni verir. Kullanıcılar posta kutularında klasör oluşturabilir ve mesajlarını bu klasörlere taşıyabilirler.
- IMAP istemcileri sunucu üzerinde istenilen ölçüte göre arama yapabilir.

IMAP, POP3'e göre karmaşık bir protokol olsa da, kullanıcıya e-posta yönetimi için çok daha fazla imkân sağlamaktadır.

IMAP protokolünün ilk sürümünden sonra 1987 yılında IMAP2 tanımlanmıştır. 1988 yılında ise IMAP ile ilgili ilk RFC dokümanı (RFC 1064) yayımlanmıştır. Daha

sonrasında da devam eden çalışmalar sonucunda 1996 yılında yayımlanan RFC 2060 ile IMAP4rev1 ortaya çıkmıştır.

2.2.4. MIME

Basit internet e-posta iletim protokolü (SMTP) sadece 7 bitlik ASCII karakterlerini destekler. Bu durum e-posta mesajında sadece İngiliz alfabesindeki harflerin kullanılabilmesi gibi bir sınırlama getirir. E-posta içerisindeki 7 bitlik ASCII karakterlerinin desteklemediği karakterler gösterilemez.

Multipurpose Internet Mail Extensions (Çok amaçlı İnternet Posta Eklentileri); E-posta uygulamaları aracılığıyla gönderilecek olan iletiye çeşitli türdeki içeriği eklemek için kullanılan bir İnternet standardıdır. MIME SMTP'yi hem metin hem de metin içerikli olmayan birden çok içerik eklenebilecek şekilde genişletir. MIME aracılığıyla e-posta iletilerine resim, ses, görüntü türünde veriler eklenebilmektedir. E-Posta uygulamalarına ek olarak web tarayıcıları da çeşitli MIME türlerini desteklemektedir. MIME türleri, Web tarayıcısına veya posta uygulamasına sunucudan alınan dosyaların nasıl işleneceğini bildirir. Bu sayede tarayıcı dosya HTML biçiminde veya gösterebileceği türde bir dosya olup olmadığını algılayarak ne yapması gerektiğini bilmektedir. Örneğin, bir Web tarayıcısı bir sunucudaki bir öğeyi isterse, aynı zamanda nesnenin MIME türünü de ister. Grafik gibi bazı MIME türleri tarayıcı içinde görüntülenebilir.

MIME'nin doğuşu ve gelişimi

MIME 1992'de Internet Engineering Task Force (IETF) tarafından tanımlanmıştır. Yayımlanan MIME RFC dokümanları Çizelge 2.3'de, MIME ile ilişkili diğer RFC dokümanları ise Çizelge 2.4'de verilmiştir.

Çizelge 2.3. Yayımlanmış MIME RFC dokümanları

Doküman No	Doküman Başlığı
RFC 2045	MIME Bölüm 1: İnternet mesaj gövdelerinin biçimi
RFC 2046	MIME Bölüm 2: Ortam türleri
RFC 2047	MIME Bölüm 3: ASCII olmayan metinler için mesaj başlık ekleri
RFC 2048	MIME Bölüm 4: Kayıt prosedürleri
RFC 2049	MIME Bölüm 5: Uygunluk kriterleri ve örnekleri

Çizelge 2.4. Yayımlanmış MIME ile ilişkili RFC dokümanları

Doküman No	Doküman Başlığı
RFC 1524	Posta başlık dosyalarının biçimsel tanımlaması
RFC 2015	PGP ile MIME güvenliği
RFC 2110	HTML gibi kümelenmiş dokümanların MIME e-posta sarmalanması
RFC 2111	İçerik kimliği ve mesaj kimliği internet kaynak adı
RFC 2112	MIME çok parçalı \ ilişkili içerik türü
RFC 2183	İnternet mesajlarında sunum bilgisinin haberleşmesi: İçerik düzenleme başlık alanı
RFC 2231	MIME parametre değeri ve kodlanmış kelime ekleri: Karakter setleri, dilleri ve devam etmesi.
RFC 2440	OpenPGP Mesaj biçimi
RFC 2633	S/MIME sürüm 3 Mesaj belirtimi
RFC 2821	Basit posta transfer protokolü
RFC 2822	İnternet mesaj biçimi
RFC 3156	OpenPGP ile MIME güvenliği

2.3. Güvenlik Gereksinimleri

Teknolojinin hızla gelişmesiyle birlikte bilgisayar sistemleri ve ağları da gelişerek bilgiye hızlı, kesintisiz ve güvenli bir şekilde erişim ihtiyacı da artmıştır. Artan bu ihtiyaçlar doğrultusunda birçok kurum ve kuruluş güvenliğin tüm unsurları ile sağlanması için çalışmalar yapmaktadır. Güvenliğin en temel unsurları olarak görülen; gizlilik, bütünlük ve mevcudiyet sağlanmadığı takdirde o ortamda güvenlik zafiyetinin olduğundan söz edilebilir.

Bilginin nasıl üretileceği, dağıtılacağı/erişileceği gibi problemlerle birlikte artık bilginin nasıl korunacağı sorusu da dünya gündeminde çok önemli bir yer edinmiştir.

Bilişim teknolojilerinin kullanımının hızla yaygınlaştığı ve arttığı günümüzde, bilgi, bilgisayar ve bilgisayar sistemleri güvenliği en önemli konuların başında yer almaktadır.

Bilgi ve bilgisayar sistemleri güvenliği ile ilgili daha fazla bilgi vermeden önce güvenliğini sağlamaya çalıştığımız “bilgi”, “veri” ve “özbilgi” kavramlarına bakalım.

Türk Dil Kurumunun sözlüğüne göre:
Bilgi:

1. İnsan aklının erebileceği olgu, gerçek ve ilkelerin bütünü, bili, malumat.
2. Kurallardan yararlanarak kişinin veriye yönelttiği anlam.

Veri ise:

1. Bir araştırmanın, bir tartışmanın, bir muhakemenin temeli olan ana öge.
2. Olgu, kavram veya komutların, iletişim, yorum ve işlem için elverişli biçimli gösterimi.

olarak tanımlanmıştır.

İngilizce’de bilgi kelimesinin karşılığı “information”, veri kelimesinin karşılığı ise “data” dır. İngilizce’de sık kullanılan “knowledge” ise Türkçe’ye yine “bilgi” olarak çevrilse de aslında bilginin daha özel bir halini belirttiği için “özbilgi” olarak ifade edilmesi daha doğru olacaktır. Bu üç kelime arasında ilk bakışta bir fark yok gibi görünse de veri, bilgi ve özbilgi olarak adlandırılan şeylerin “değeri” ve “kapsamı” çok farklıdır. Bilgi, verinin işlenmiş halidir.

Bilginin “değerli” veya “değersiz” olduğunun ve onu korumak için ne kadar yatırım yapılarak önlem alınması gerektiğinin belirlenmesi düşünülmesi gerek ilk şeydir.

Bilgi güvenliği; “bilginin bir varlık olarak hasardan korunması, doğru teknolojinin, doğru amaçla ve doğru şekilde kullanılarak, bilginin her türlü ortamda istenmeyen kişiler tarafından elde edilmesini önleme” olarak tanımlanır [16].

Bilgiye ne yapılabilir sorusuna baktığımızda:

- Üretilebilir
- İşlenebilir
- Depolanabilir

- Taşınabilir
- İmha Edilebilir
- Kullanılabilir
- Kaybedilebilir, hasar görebilir

gibi birçok durum karşımıza çıkmaktadır.

Tam güvenli bir ortam oluşturmada:

- Gizlilik
- Bütünlük
- Kimlik kanıtlama ve doğrulama
- İnkâr edememezlik
- Süreklilik

gibi hususlar önemli rol oynamaktadır.

Gizlilik

Bilginin sadece erişmeye hakkı olan kişiler tarafından erişilebilir olduğundan emin olmaktır.

Doğruluk / Bütünlük

Bir dokümanın veya mesajın içeriğinin değiştirilmemesinin sağlanmasıdır. Verinin izinsiz veya yanlışlıkla değiştirilmesini, silinmesini ve veriye ekleme yapılmasını önler.

Kimlik tanımlama, doğrulama ve yetkilendirme

Kimlik tanımlama, kişilerin veya uygulamaların bir sistem ya da hizmeti kullanabilmeleri için o sistem tarafından kayıt edilmeleri ve kimlik verilmesi işlemleridir.

Kimlik doğrulama, belirli hizmet ya da sistemlerden yararlanabilmek için mevcut tanımlı olan kimliklerinin doğruluğunun kanıtlanması işlemleridir.

Yetkilendirme ise aynı sistem içerisinde yer alan kişilerin o sistem üzerindeki tüm kaynaklara erişememeleri veya kullanamamaları, kullanıcı hakları doğrultusunda yetki verilmesidir.

İnkâr edememezlik

Bilgiyi gönderen veya işleyen kişinin yaptığı işi sonradan inkar edememesidir. Bunun sağlanması için mesajı gönderen ve alan kişilerin kayıtları güvenilir bir makam tarafından tutulur. İnkar edememezlik unsurunun sağlanması için genellikle dijital imzalar ve zaman damgaları kullanılır.

Süreklilik

Kişiler ve/veya sistemler arası haberleşme kesintiye uğramadan yapılmalıdır. Her şeyin son derce hız kazandığı günümüzde sürekliliğin kaybına tolerans gösterilemez.

Bulunurluk /Mevcudiyet

Bilgi varlıklarının, izin verilen kullanıcıların ihtiyaç duydukları zamanlarda erişebilir olduklarının güvence altına alınması durumudur.

Kriptoloji kelimesi, köken olarak eski Yunanca'da yer alan “gizli dünya” anlamına gelen “kryptos” kelimesi ile “sebep-sonuç ilişkisi kurma” anlamını gelene “logos” kelimelerinden gelmektedir.

Kriptoloji, verinin bir yerden başka bir yere aktarılması işlemlerinin emniyetli olarak yapmasını sağlayan, temeli çözümü zor matematiksel problemlere dayanan tekniklerin ve uygulamaların bütünüdür.

Günümüzde kriptoloji, matematik, elektronik, optik, bilgisayar bilimleri gibi birçok disiplini kullanan özelleşmiş bir bilim dalıdır. Kriptolojinin iki temel alt dalı vardır. Bunlar kriptografi ve kriptoanaliz'dir.

Kriptografi, belgelerin şifrelenmesi ve şifresinin çözülmesi için kullanılan yöntemlere verilen addır. Kriptografi Yunanca gizli anlamına gelen "kriptos" ve yazı anlamına gelen "graphi" dan türetilmiştir

Kriptoanaliz, kriptografik sistemlerin kurduğu sistemleri inceler ve çözmeye çalışır. Kriptoanalizin amacı kriptografinin tam tersidir.

Günümüzde elektronik bilgi sistemlerinin yaygınlaşması kriptolojinin önemini çok fazla arttırmıştır. Kriptolojinin başlıca kullanım amacı hareket halindeki veya depolanmış bilginin koruma altına alınmasıdır.

Verinin ilk haline düz metin (plaintext) denir. Dönüştürülmüş biçimine ise şifrelenmiş metin veya gizli metin (ciphertext) denir. Dönüştürme işlemine şifreleme dönüştürülmüş metnin ters işlem sonucu açık metin halinin elde edilmesine ise deşifre etme denir. P açık metni, E ve D simgeleri de şifreleme ve deşifreleme işlemlerini temsil ettiğinde, gizli metin C olmak üzere şifreleme işlemi;

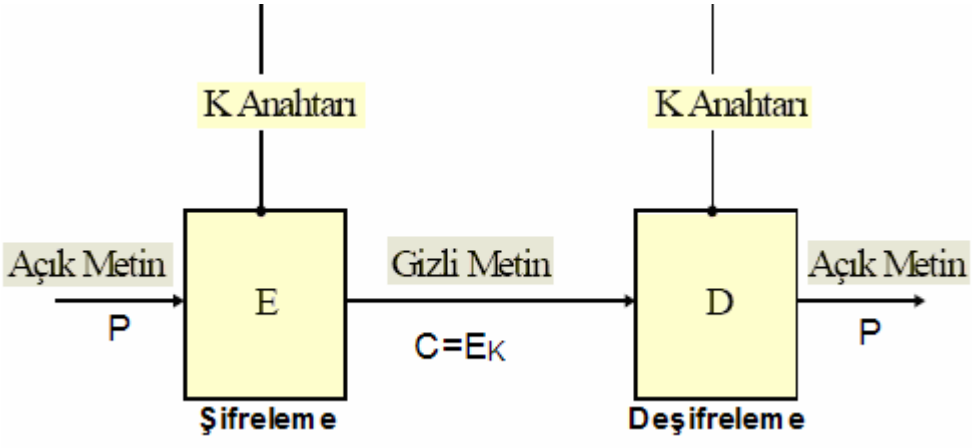
$$C = E_K(P)$$

ile gösterilir. Gizli metinden açık metnin elde edilmesi amacı ile deşifre edilmesi ise;

$$P = D_K(C)$$

olarak gösterilir. Her iki ifadede kullanılan K, kripto anahtarını temsil etmektedir.

Şekil 2.3 bu iki işlemi göstermektedir.



Şekil 2.3. Şifreleme ve şifre çözme işlemleri

Şifre Bilimi Tarihçesi

M.Ö. 100-44 yılları arasında sunulan Sezar yaklaşımı kullanılan ilk şifreleme yaklaşımlarındandır. Yıllar geçtikçe modern kriptografi olarak adlandırılabilecek döneme kadar çeşitli yöntemler geliştirilmiş ve kullanılmıştır. Bu yöntemlerden en bilinenleri:

- Tekli alfabetik yer değiştirme
- Çoklu alfabetik yer değiştirme
- Tek kullanımlı şerit yöntemi
- Dönüşüm şifreleri kullanımı

gibi yöntemlerdir. Bu yöntemlerin ortak noktası; güvenliğin, şifrelemede kullanılan algoritmanın gizli kalmasına bağlı olmasıdır.

Kısaca kriptolojinin tarihçesine bakacak olursak [17]:

- MÖ. 1900 dolaylarında bir Mısırlı katip yazdığı kitabelerde standart dışı hiyeroglif işaretleri kullandı.
- MÖ. 60-50 Julius Caesar (MÖ 100-44) normal alfabedeki harflerin yerini değiştirerek oluşturduğu şifreleme yöntemini devlet haberleşmesinde kullandı. Bu yöntem açık metindeki her harfin alfabede kendisinden 3 harf sonraki harfle değiştirilmesine dayanıyordu.

- 725-790 Abu Abd al-Rahman al-Khalil ibn Ahmad ibn Amr ibn Tammam al Farahidi al-Zadi al Yahmadi, kriptografi hakkında bir kitap yazdı (Bu kitap kayıp durumdadır). Kitabı yazmasına ilham kaynağı olan, Bizans imparatoru için Yunanca yazılmış bir şifreli metni çözmesidir. Abu Abd al-Rahman, bu metni çözmek için ele geçirdiği şifreli mesajın başındaki açık metni tahmin etme yöntemini kullanmıştır.
- 1000 - 1200 Gaznelilerden günümüze kalan bazı dokümanlarda şifreli metinlere rastlanmıştır. Bir tarihçinin dönemle ilgili yazdıklarına göre yüksek makamlardaki devlet görevlilerine yeni görev yerlerine giderken şahsa özel şifreleme bilgileri (belki şifreleme anahtarları) veriliyordu.
- 1586 Blaise de Vigenère (1523-1596) şifreleme hakkında bir kitap yazdı. İlk kez bu kitapta açık metin ve şifreli metin için otomatik anahtarlama yönteminden bahsedildi. Günümüzde bu yöntem hala DES CBC ve CFB kiplerinde kullanılmaktadır.
- 1623'de Sir Francis Bacon, 5-bit ikili kodlamayla karakter tipi değişikliğine dayanan stenografı buldu.
- 1790'da Thomas Jefferson, Strip Cipher makinesini geliştirdi. Bu makineyi temel alan M-138-A, ABD donanmasınca 2.Dünya savaşında da kullandı.
- 1917'de Joseph Mauborgne ve Gilbert Vernam mükemmel şifreleme sistemi olan "one-time pad"'i buldular.
- 1920 ve 1930'larda FBI içki kaçakçılarının haberleşmesini çözebilmek bir araştırma ofisi kurdu.
- William Frederick Friedman, Riverbank Laboratuarlarını kurdu, ABD için kriptoanaliz yaptı, 2. Dünya savaşında Japonlar'ın Purple Machine şifreleme sistemini çözdü.
- Dünya savaşında Almanlar Arthur Scherbius tarafından icat edilmiş olan Enigma makinasını kullandılar. Bu makine Alan Turing ve ekibi tarafından çözüldü.
- 1970'lerde Horst Feistel (IBM) DES'in temelini oluşturan Lucifer algoritmasını geliştirdi.
- 1976'da DES (Data Encryption Standard), ABD tarafından FIPS 46 (Federal Information Processing Standard) standardı olarak açıklandı.

- 1976’da Whitfield Diffie ve Martin Hellman Açık Anahtar sistemini anlattıkları makaleyi yayınladılar.
- 1978'de Ronald L. Rivest, Adi Shamir ve Leonard M. Adleman RSA algoritmasını buldular.
- 1985'de Neal Koblitz ve Victor S. Miller ayrı ayrı yaptıkları çalışmalarda eliptik eğri kriptografik (ECC) sistemlerini tarif ettiler.
- 1990'da Xuejia Lai ve James Massey IDEA algoritmasını buldular.
- 1991'de Phil Zimmerman PGP sistemini geliştirdi ve yayınladı.
- 1995'de SHA-1 (Secure Hash Algorithm) özet algortiması NIST tarafından standart olarak yayınlandı.
- 1997'de ABD'nin NIST (National Institute of Standards and Technology) kurumu DES'in yerini alacak bir simetrik algoritma için yarışma açtı.
- 2001'de NIST'in yarışmasını kazanan Belçikalı Joan Daemen ve Vincent Rijmen'e ait Rijndael algoritması, AES (Advanced Encryption Standard) adıyla standart haline getirilmiştir..

2.4. Kriptografik Yaklaşımlar

Bu kısımda GÜ-Posta uygulamasında kullanılan kriptografik algoritmalar ve yaklaşımlar için temel bilgiler verilmiştir.

2.4.1. DES ve 3DES

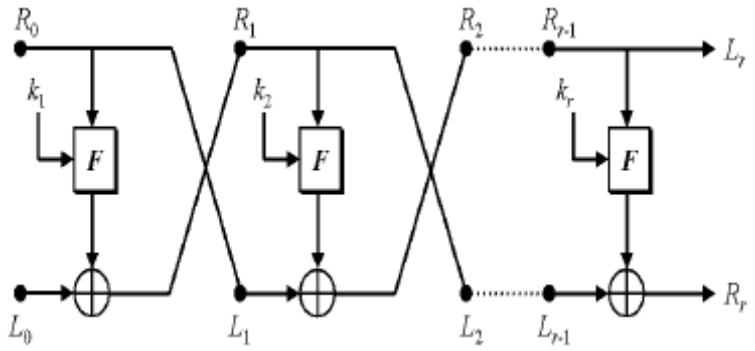
DES (Data Encryption Standard) veri şifreleme standardı, bir blok şifreleme sistemidir.

DES algoritmasının temeli Lucifer şifreleme sistemine dayanmaktadır [18]. 1975 yılında yayınlanan DES algoritması 1976 yılında Veri Şifreleme Standardı olarak seçilmiştir. DES standardının en son tanımı FIPS’in 25 Ekim 1999’da 46-3 numaralı yayınında yapılmıştır.

DES Shannon’un önerdiği yayılma ve karıştırma özelliklerini Feistel adı verilen bir yapı ile sağlamaktadır.

Horst Feistel tarafından ilk defa tasarlanan Feistel yapısı günümüzde bir çok modern sistemde kullanılmaktadır.

k1,k2,...,kn: Döngü anahtarları olmak koşuluyla, Feistel yapıları Şekil 2.4'deki gibi gösterilebilir.

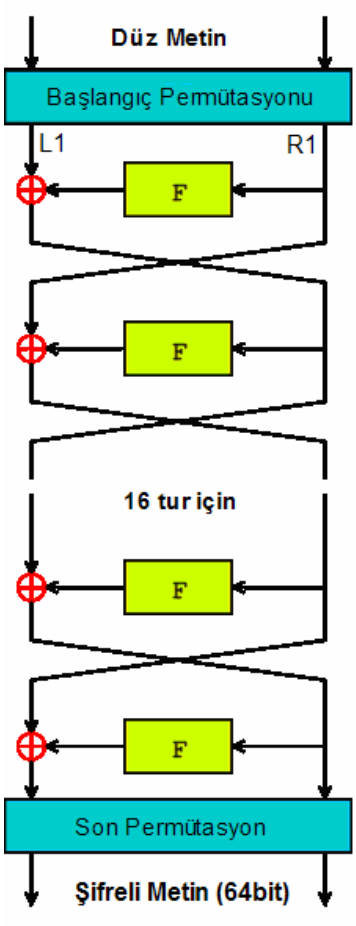


Şekil 2.4. Feistel yapısı

DES, ikilik tabanda olan düz metni 64 bitlik bloklar halinde, 56 bitlik anahtar kullanarak şifreler. Algoritmada kullanılan anahtar rasgele seçilir. DES'in en önemli yanı yayılma ve karıştırma özellikleridir. Bu sayede DES'de her bloğun biti diğer bitlere ve anahtarın bitine bağımlıdır. Yayılma özelliği, anahtarın düz metin ve şifreli metne bağlı olmasını sağlar; bu sayede sistemi karmaşıklaştırır ve kriptoanalizi zorlaştırır. Karıştırma ise her anahtar için düz metin ile şifreli metnin yapıları arasında istatiksel bağlantı olmamasını sağlar; böylece düz metin ve şifreli metini karşılaştırılsa bile aralarında bağıntı bulunamayacağından anahtarın bulunması çok zorlaşır.

64 bitlik bloklar haline ayrılmış mesajın bitlerinin yeri rasgele değiştirilir. Bu işleme başlangıç permütasyonu adı verilir. Bitlerin yer değiştirme işlemi bittikten sonra 64 bit, sol ve sağ olmak üzere iki parçaya ayrılır. Sağ koldan gelen 32 bitlik veri doğrusal olmayan F fonksiyonuna girer. DES'in güvenliği ve gizliliği bu F fonksiyonundaki S-kutularına bağlıdır. S kutuları düz metin ile şifreli metin arasındaki bağlantıyı en aza indirmek için mümkün olan en iyi karıştırmayı yapar. Aynı zamanda fonksiyona giren 32 bitlik veri K anahtarı ile işlem gördükten sonra

sol koldan gelen 32 bitlik veri ile XOR'lanır. Sol koldaki veri herhangi bir işlem görmeden aşağıya iner. Başlangıç permütasyonundan sonra yapılan bu işlemlere bir döngü denir. 1. döngü sona erdikten sonra sağ ve sol yer değiştirilir. Böylelikle 1. döngüde sağ kol olan 32 bitlik veri 2. kol için sol kol olur. Ayrıca her döngüde sadece sağ koldaki 32 bitlik veri aynı kalmakta, sol koldaki veri değişmektedir. 1. döngü için yapılan işlemler 16 döngü için tekrarlanır. Başlangıçta verilen 56 bitlik anahtarlardan her döngü için bir algoritmaya göre farklı döngü anahtarları üretilir. 16 döngü sona erdikten sonra her biri 32 bitten oluşan sağ ve sol kol birleştirilir ve başlangıç permütasyonunun tersi işleme sokulur. Sonunda şifreli metin elde edilir (Şekil 2.5).



Şekil 2.5. DES in yapısı ve çalışması

Ele geçen şifreli metinden düz metne ulaşabilmek için şifreleme işlemini tersten yürütmek gereklidir. Anahtar hem gönderici hem de alıcı tarafından bilinmek zorundadır.

DES'in güvenliği

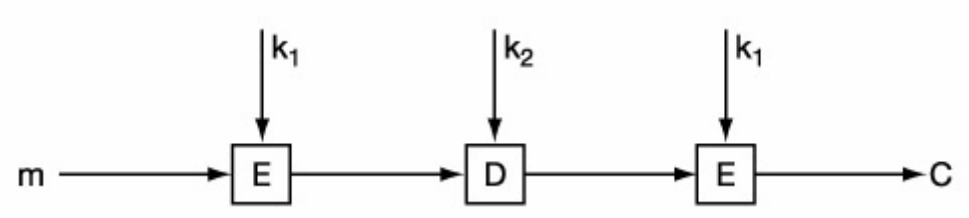
DES simetrik bir sistem olduğu için anahtarın güvenliği çok önemlidir. Anahtar boyutu 56 bit olduğundan DES için toplam 2^{56} tane olası anahtar vardır, çünkü anahtarı bilmeyen birinin şifreli metinden doğru düz metni elde edebilmesi için olası bütün anahtarları denemesi gerekir. Bunu deneme günümüz bilgisayarlarında bile çok uzun bir zaman gerektirir. Sistem başlangıçta güvenilir gibi gözükse de gelişen diferansiyel ve lineer kriptoanaliz yöntemleri yüzünden 1990'lı yıllarda güvenirliğini yitirmiştir.

3DES; DES algoritmasının üç kez kullanılması ile oluşturulmuş bir şifreleme algoritmasıdır. 3DES için DES üzerinde üç kez şifreleme ya da iki kez şifreleme bir kez de deşifreleme gibi varyasyonlar yapılabilir. Örneğin kısaca EDE (encrypt, decrypt, encrypt) olarak bilinen yöntemde ilk önce şifrelenecek mesaj k_1 anahtarı kullanılarak şifrelenir, daha sonra k_2 anahtarı ile deşifre edilir, en son adımda ise k_3 ($k_3 = k_1$) anahtarı ile tekrar şifrelenir (Şekil 2.6). Bu algoritma aşağıdaki gibi ifade edilebilir:

$$DES\left(k_3; DES^{-1}\left(k_2; DES\left(k_1; M\right)\right)\right)$$

Burada k_1, k_2 ve k_3 anahtarları M ise şifrelenecek mesajı ifade etmektedir.

Anahtarların her biri 56'bittir. Eşlik (parity) bitlerinin de eklenmesiyle oluşan anahtarın toplam uzunluğu 192 bit olur.



Şekil 2.6. İki anahtarlı 3DES

2.4.2. AES (Rijndael)

Rijndael, Amerika Birleşik Devletleri'nin Institute of Standards and Technology (NIST) adlı kuruluşu tarafından geleceğin şifreleme standardını oluşturmak amacı ile yapılan yarışmayı kazanan ve Gelişmiş Şifreleme Standardı (Advanced Encryption Standard-AES) adını alan blok şifreleme algoritmasıdır.

2 Ocak 1997'de NIST kriptografi ile uğraşan bilim adamlarına Birleşik Devletlerin hassas uygulamalarda kullanacağı yeni bir blok şifreleyicinin oluşturulması için çağrıda bulundu. Oluşturulacak olan bu Gelişmiş Şifreleme Standardı artık yaşlanmış olan DES ve türevlerinin yerine geçecekti. Bunun öncelikli nedeni DES'in göreceli olarak küçük olan 56 bitlik anahtarının deneme-yanılma ataklarına (Brute force attacks) karşı giderek dayanıksız kalmasıydı. Buna ek olarak DES öncelikle donanım üzerinde uygulamaya göre tasarlandığı için yazılım uygulamaları göreceli olarak yavaş kalmaktaydı. 3DES (TripleDES)'in kullanılması ile küçük anahtar uzunluğu problemi aşılmış olsa bile yazılım uygulamalarının çok yavaş olması ve kısıtlı kaynakları olan platformlarda uygulanmasının uygun olmaması hala problemler teşkil etmekte idi.

AES'den beklenen özellikler bir sır olmadığı için seçilecek olan şifreleyicinin devlet kurumları ve ABD haricinde de ilgi çekeceği ve kullanım alanı bulacağı düşünülüyordu.

Yeni standardın isterleri oldukça sertti. Özetle 1 Ocak 1997 de istenen şartlar aşağıdaki gibidir:

1. AES kamuya açık tanımlı olmalı.

2. AES simetrik blok şifreleyici olmalı.
3. AES anahtar uzunluğu ihtiyaç duyulduğunda arttırılabilecek şekilde tasarlanmalı. (12 Eylül 1997 tarihinde Federal Kayıt Bürosu (Federal Register) blok uzunluğunu 128 bit, anahtar uzunluğunu ise 128, 192 ve 256 bit olarak belirlendiğini duyurdu.)
4. AES hem donanımda hem de yazılımda gerçekleştirilebilir olmalı.
5. AES ya serbestçe kullanılabilir olmalı ya da ANSI patent kurallarına bağıntılı koşullar altında olmalı.
6. Yukarda yazılı isterleri yerine getiren algoritmalar aşağıdaki faktörlere göre değerlendirilecektir:
    a) Güvenlik (Kriptoanalize karşı direnç)
    b) Hesaplama verimliliği
    c) Bellek gereksinimleri
    d) Donanım ve yazılım uygunluğu
    e) Esneklik
    f) Lisanslama gereksinmeleri.

1998 yılında yarışmaya değişik ülkelerden 15 farklı tasarım sunuldu. Sunulan bu tasarımlardan 5 tanesi 1999 yılında finale kaldı. Finalistler alfabetik sırayla aşağıda verilmiştir:

- *MARS*: IBM firmasında çalışan bir ekip (Carolynn Burwick, Don Coppersmith, Edward D'Avignon, Rosario Gennaro, Shai Halevi, Charanjit Jutla, Stephen M. Matyas Jr., Luke O'Connor, Mohammad Peyravian, David Safford, Nevenko Zunic) tarafından geliştirilmiştir [19].
- *RC6*: Ronald L. Rivest (M.I.T. Laboratory for Computer Science, ABD), M.J.B. Robshaw, R. Sidney ve Y.L. Yin (RSA Laboratories, ABD) tarafından geliştirilmiştir [20].
- *Rijndael***:** Joan Daemen (Proton World Int., Belçika), Vincent Rijmen (Katholieke Universiteit Leuven, Belçika) tarafından geliştirilmiştir [21].

- *Serpent*: Ross Anderson (Cambridge Üniversitesi, İngiltere), Eli Biham (Technion, Hayfa, İsrail) ve Lars Knudsen (Bergen Üniversitesi, Norveç) tarafından geliştirilmiştir [22].
- *Twofish:* Bruce Schneier, John Kelsey, Chris Hall, Niels Ferguson (Counterpane Systems, ABD), Doug Whiting (Hi/fn, Inc., ABD), David Wagner (University of California Berkeley, ABD) tarafından geliştirilmiştir [23].

NIST tarafından 2 Ekim 2000 tarihinde önerilen AES olarak Rijndael algoritmasının seçildiği ve resmi standart olma yolunda gerekli sürece geçildiği duyuruldu. Daha sonra 26 Kasım 2001 tarihinde AES'in FIPS PUB 197 (Federal Information Processing Standards Publication 197) olarak onaylandığı duyuruldu.

Yapılan testler ve analizler sonucunda performans, uygulama kolaylığı, esneklik, güvenlik gibi unsurlar açısından en iyi kombinasyonu ortaya koyması ile Rijndael AES olarak seçilmiştir [24].

İlerleyen zamanda finalist algoritmalarla ilgili yapılan başka çalışmalar sonucu yine Rijndael algoritmasının diğerlerine nazaran başarılı olduğu görülmüştür. [25]

Bu bölümden itibaren AES ile Rijndael algoritması kastedilmektedir.

AES-Genel Özellikleri

Genel Güvenlik Özellikleri

AES'in bilinen bir güvenlik açığı yoktur. AES doğrusal olmayan bileşenler olarak S-kutularını kullanır. AES yeterli güvenliğe sahip görünüyor ama matematiksel yapısının saldırılara yol açabileceği yönünde bazı eleştiriler de almıştır. Diğer taraftan geliştirme sürecinde basit yapı AES'in güvenlik analizini kolaylaştırmıştır.

Yazılım gerçekleştirimleri

AES şifreleme ve deşifreleme işlemlerini 8-bit, 64-bit ve DSP'ler gibi değişik platformlarda çok iyi gerçekleştirebilmektedir. Ancak artan anahtar uzunluğu nedeniyle tur sayılarının da artması ile performansta bir azalma olmaktadır. AES'in içsel paralelliği verimli işlemci kaynakları kullanımına olanak sağlar. Bu da iyi bir yazılım performansına sebep olur. AES'in anahtar oluşturma zamanı hızlıdır.

Kısıtlı Alan-Gömülü sistem gerçekleştirimleri

Genel olarak AES ister şifreleme isterse de deşifreleme uygulaması olsun kısıtlı alana, kaynaklara sahip olan ortamlara çok uygundur. Çok düşük RAM ve ROM gereksinimi vardır. Bir dezavantajı eşzamanlı olarak şifreleme ve deşifreleme işlemi uygulanmak istendiğinde ROM gereksiniminin artmasıdır. Buna rağmen bu tür ortamlara hala uygun görünmektedir.

Donanım Gerçekleştirimleri

AES finalistler içinde geri besleme modunda (feedback modes) en yüksek üretim kapasitesine (throughput), geri besleme olmayan modda (non-feedback modes) ise en yüksek ikinci üretim kapasitesine (throughput) sahip algoritmadır. 192 ve 256 bit anahtar uzunlukları için ek tur sayılarından dolayı kapasitesi düşer. Tamamen ardışık düzenlenmiş (pipelined) gerçekleştirimler için alan isterleri artarken kapasitesi etkilenmez.

Gerçekleştirimlere yapılan saldırılar

Güç ve zamanlama saldırıları karşında savunmak için AES tarafından kullanılan işlemler en basit işlemler arasındadır. Maskeleme tekniklerinin AES'e bu saldırılara karşı biraz savunma sağlaması için kullanılışı diğer finalistlere göre önemli performans düşüşüne neden olmaz ve RAM ihtiyacı makul kalır. AES'in bu gibi

korunmalar düşünüldüğünde rakipleri üzerinde büyük bir hız avantajı kazandığı görünür.

Şifreleme ve Deşifreleme

AES'in şifreleme ve deşifreleme fonksiyonu farklıdır. Bir FPGA çalışmasının bildirdiğine göre şifreleme ve deşifrelemenin birlikte gerçekleştirimi şifrelemenin tek başına gerçekleştiriminden %60 daha fazla yer gerektirmektedir. AES'in hızı şifreleme ve deşifreleme arasında önemli bir değişiklik göstermez. Buna karşın anahtar hazırlanış performansı deşifreleme için şifrelemeye göre daha yavaştır.

Anahtar çevikliği

AES şifreleme için anında (on the fly) alt anahtar hesaplamasını destekler. AES belirli bir anahtar ile ilk olarak deşifrelemeden önce bütün alt anahtarları oluşturmak için zaman çizelgesinin bir kez koşturulmasını ister.

Diğer Özellikler

AES 128 bit blok uzunluğu ile değişken anahtar uzunlukları (128, 192 ve 256 bit) arasındaki tüm kombinasyonlara tam destek verir. Prensipte AES'in yapısı belirtilen tur sayılarının değiştirilmesi ile 32'nin katı olan herhangi bir blok ve anahtar uzunluğunu destekler. Döngü (Tur) sayısı anahtarın büyüklüğüne yani içerdiği bit sayısına göre değişmektedir. Anahtar uzunluğuna göre değişen AES parametreleri Çizelge 2.5'de verilmiştir.

Çizelge 2.5. AES parametreleri [26]

Anahtar büyüklüğü (word/byte/bit)	4/16/128	6/24/192	8/32/256
Düz metin blok büyüklüğü (word/byte/bit)	4/16/128	4/16/128	4/16/128
Tur sayısı	10	12	14
Tur anahtar büyüklüğü (word/byte/bit)	4/16/128	4/16/128	4/16/128
Genişletilmiş anahtar büyüklüğü (word/byte)	44/176	52/208	60/240

Feistel ağ yapı temelli olan DES'den farklı olarak AES yerine koyma-permütasyon (substitution-permutation) ağ yapılıdır. Bu ağ yapısı yerine koymalar (S-kutuları ile) ve permütasyonlar (P-kutuları ile) kullanılarak bir dizi matematiksel işlemin yapıldığı bir yapıdır. Her bir çıktı biti giriş bitine bağlıdır.

2.4.3. RC2

Ron Rivest tarafından 1987 yılında RSA Güvenlik (RSA Security) şirketi için tasarlanmış bir blok şifreleyicidir. RC kısaltması bazı kaynaklara göre "Rivest Cipher" bazı kaynaklara göre de "Ron's Code" ifadesinden gelmektedir. RC1 Ron Rivest'in sadece not defterinde kalmış RC3 ise RSADSI'da (RSA Data Security, Inc) geliştirme aşamasında kırılmıştır.

RC2 DES'den daha hızlıdır ve DES'in yerini alması için düşünülmüştür. Anahtar boyutlarının ayarlanması ile geniş kapsamlı anahtar aramaları karşısında DES'ten daha güvenli ya da daha az güvenli yapılabilir. 64 bitlik blok uzunluğu sahiptir ve yazılım gerçekleştirimlerinde DES'den 2 ya da 3 kat daha hızlıdır. Anahtara ek olarak 40 ile 88 bit arası uzunluklarda değişen "salt" adında ayrı bir string daha kullanılabilmektedir. "Salt" şifreleme anahtarına eklenir ve bu uzatılan anahtar mesajı şifrelemede kullanılır. Daha sonra "salt" şifrelenmemiş olarak mesajla birlikte gönderilir. Yaygın olarak ürünlerini dışarıya satmak isteyen geliştiriciler tarafından RC2 ve RC4 kullanılmıştır. Bunun sebebi DES kullanan uygulamaların ihracında daha sıkı koşulların uygulanmasıydı. Daha sonra Yazılım Yayımcıları Birliği (Software Publishers Association - SPA) ile Birleşik Devletler hükümeti arasında

yapılan bir anlaşmayla RC2 ve RC4'e (40 bit anahtar kullanan) özel bir statü verilmiştir. Bu özel statü ile alışılmış kriptografik ihracat sürecinden daha hızlı ve basit bir onay süreci tanınmıştır.

Başlangıçta algoritmanın detayları RSA Güvenlik şirketi tarafından gizli tutulmuştu. Ama daha sonra 29 Ocak 1996 tarihinde kimliği gizli biri tarafından RC2'nin kaynak kodları internete verildi. Bu kişinin algoritmanın özelliklerine mi eriştiği yoksa tersine mühendislik yaparak mı bu bilgilere sahip olduğu bilinmemektedir.

2.4.4. RSA

RSA açık anahtarlı (asimetrik) bir şifreleme sistemidir. Algoritması 1977'de Ron Rivest, Adi Shamir ve Len Adleman tarafından oluşturan RSA adını da bu üçlünün soy isimlerinin baş harflerinden almıştır. RSA en çok bilinen ve kullanılan bir açık anahtar şifreleme sistemidir. RSA algoritmasında biri özel (private) biri de genel (public) anahtar olmak üzere iki adet anahtar kullanılır. Şifreleme işleminde bu anahtarlardan herhangi biri kullanıldığında deşifreleme işlemi ancak diğer anahtarla yapılabilir.

RSA algoritmasının işlem basamakları aşağıda verilmiştir.

1. p ve q gibi iki tane rasgele ve birbirinden bağımsız büyük asal sayı seçilir. $(p \neq q)$. Daha fazla güvenlik için p ve q sayıları çok büyük ve eşit uzunlukta seçilmelidir.
2. $n = p.q$ hesaplanır.
3. Totient fonksiyonundan yararlanılarak $\phi(n)(p-1).(q-1)$ hesaplanır.
4. $1 \langle e \langle \phi(n)$ koşuluna uyan ve $\phi(n)$ ile aralarında asal $(OBEB(e, \phi(n)) = 1)$ olan bir e tamsayısı seçilir.
5. $d.e \equiv 1 \pmod{\phi(n)}$ olacak şekilde bir d sayısı üretilir.

Burada; n ve e genel anahtarı, d ve n verileri de özel anahtarı ifade eder.
Bu algoritmada şifreleme işlemi, m mesaj, c ise şifreli mesaj olmak üzere:

$$c_i = m_i^{\,e} \bmod n$$

ile gerçekleştirilir.
Mesajın deşifre edilmesi ise:

$m_i = c_i^{\,d} \bmod n$

ile gerçekleştirilir.

RSA algoritmasının temelleri aşağıdaki teoremlere dayanmaktadır:

Teorem 1(Küçük Fermat teoremi): Eğer p bir asal sayı ve a bir tam sayı ise, $(a, p) = 1$

$a^{p-1} = 1 \pmod{p}$ dir.

Teorem 2 (Fermat teoremi): Eğer $(a, m) = 1$ ise $a^{\phi(m)} = 1 \pmod{m}$ şeklinde ifade edilebilir. ($\phi(m)$, m 'den küçük, m ile aralarında asal olan sayıların adedidir.)

Teorem 3 (Çinlilerin kalan teoremi): p ve q gibi $(p, q) = 1$ olan iki sayı ele alalım (asal sayı olmalarına gerek yok). Daha sonra eğer $a = b \pmod{p}$ ve $a = b \pmod{q}$ ise $a = b \pmod{p.q}$ olur.

Bir örnekle algoritmanın işleyişini daha fazla açıklayalım. Bu örnek durum için Ayşe ve Burak adlı iki hayali kişi olduğunu varsayalım. Ayşe ve Burak aralarında RSA algoritmasını kullanarak mesajlaşmak istiyor olsunlar. Yapılacak işlemleri adım adım anlatacak olursak:

1. Ayşe genel ve özel anahtarlarını oluşturmak için p ve q gibi iki tane asal sayı seçer.
2. $p = 23$ ve $q = 41$ (Gerçek uygulamalarda bu sayıların çok büyük asal sayılar olması gerekmektedir. Bu örneğin anlaşılır olması için küçük asal sayılar seçtik.)
3. p ve q çarpımı hesaplanır. $n = p.q = (23).(41) = 943$.
4. $\phi(n) = (p-1).(q-1)$ hesaplanır. $\phi(n) = (23-1).(41-1) = 880$
5. $1 < e < \phi(n)$ koşuluna uyan ve $\phi(n)$ ile aralarında asal $(OBEB(e, \phi(n)) = 1)$ olan bir e tamsayısı seçilir. $e = 7$.

6. Artık Burak Ayşe'ye mesaj göndermek için yeterli bilgiye sahiptir. (Ayşe'nin genel anahtarına $(n \text{ ve } e)$). Göndermek istediği mesaj $M = 35$ olduğunu varsayalım.
7. Burak şifreli mesaj değeri C'yi hesaplar. $C = M^e \pmod{n} = 35^7 \pmod{943}$.
8. $35^7 = 64339296875$ dir. $64339296875 \pmod{943} = 545$ olur. 545 Burak'ın Ayşe'ye göndereceği şifreli mesajdır.
9. Ayşe Burak'tan gelen şifreli mesajı görebilmek için kendi özel anahtarını kullanması gerektiğini biliyor bu yüzden $d.e \equiv 1 \pmod{\phi(n)}$ eşitliğindeki d değerini bulması gerekmekte.
10. $d.e \equiv 1 \pmod{\phi(n)}$ eşitliğinde bilinen değerler yerine konulursa $d.7 \equiv 1 \pmod{880}$ buradan $d = 503$ bulunur.
11. $M = C^d \pmod{n} = 545^{503} \pmod{943}$. Buradan M değerinin hesaplanabilmesi için kolaylık olması açısından 545^{503} değeri şu şekilde açılabilir.

$545^{503} = 545^{256+128+64+32+16+4+2+1} = 545^{256}.545^{128}...545^{1}$

$545^{1} \pmod{943} = 545$

$545^{2} \pmod{943} = 923$

$545^{4} \pmod{943} = 400$

$545^{16} \pmod{943} = 857$

$545^{32} \pmod{943} = 795$

$545^{64} \pmod{943} = 215$

$545^{128} \pmod{943} = 18$

$545^{256} \pmod{943} = 324$

Buradan $545^{503} \pmod{943} = 545.923.400.857.795.215.18.324 \pmod{943} = 35$ bulunur. Böylece Ayşe Burak'ın kendisine göndermiş olduğu mesajı deşifre etmiştir.

RSA sisteminin güvenliği çok büyük sayıların asal çarpanlarına ayırma işleminin zorluğuna dayanmaktadır. Asal sayıları elde etmede bilinen bir formül, yöntem yoktur. RSA ile günümüzde 1024 bitlik (yaklaşık 300 basamaklı bir sayı) ya da 2048

bitlik anahtarlar kullanılabilir. RSA algoritması, Amerika' da 1983 yılında MIT'ten patent almıştır. Bu patent 21 Eylül 2000 de son bulmuştur.

RSA kripto sistemi pratikte en çok:

1. DES, 3DES gibi simetrik anahtarlı kripto sistemlerle birlikte kullanılır. Mesaj önce simetrik algoritmalardan biri ile şifrelendikten sonra şifrelemede kullanılan anahtar RSA ile şifrelenir.
2. Hash fonksiyonları ile birlikte mesaj özetlerinin şifrelenmesi (imzalanması) amacıyla kullanılır.

Bunlardan dolayı RSA en çok elektronik imzada uygulama alanı bulmuştur. Uygulama alanlarının yukarıda bahsettiğimiz alanlarda yoğunlaşmasının sebebi RSA kripto sisteminin simetrik kripto sistemlere göre anahtar güvenliği ve yönetimi karşısında getirdiği avantajlara rağmen hesaplamada yavaş olmasıdır.

RSA kripto sisteminin hızı

RSA nın temelinde modüler çarpım serileri ile gerçekleştirilen modüler üs alma işlemi yer almaktadır. Genellikle genel anahtar için genel üs (public exponent) küçük seçilir. Bütün kullanıcı grupları her birinin farklı modu olan aynı genel üssü kullanabilirler. Bu da şifrelemenin deşifrelemeden, doğrulamanın ise imzalama işleminden daha hızlı olmasını sağlar.

Tipik modüler üs alma algoritmaları ile:

- Genel anahtar işlemleri $O(k^2)$ adım.
- Özel anahtar işlemleri $O(k^3)$ adım.
- Anahtar üretimi işlemleri $O(k^4)$ adım sürer.

Burada k mod işlemindeki bit sayısıdır.

FFT (Fast Fourier Transform) temelli metotlar gibi hızlı çarpım (Fast multiplication) metotlarının kullanılması asimptotik olarak işlemlerin daha az adımda yapılmasını

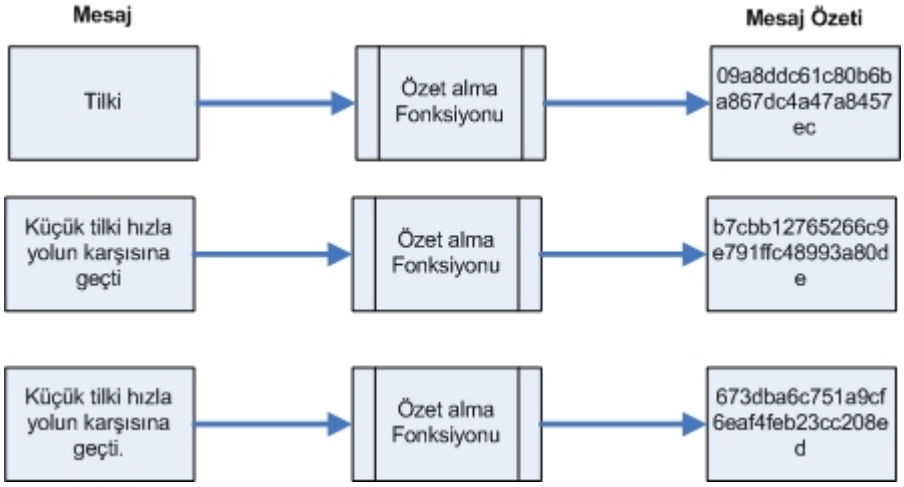
sağlar. Ancak bu yöntemler yazılım ve donanım uygulamalarının çok karmaşık olması sebebi ile pratikte çok fazla kullanılmamaktadır.

Donanımda RSA, DES'den 1000 kez daha yavaş kalmaktadır. Yazılım uygulamalarında ise RSA, DES'den 100 kez daha yavaştır.

Yeni geliştirilen yazılım ve donanım uygulamaları ile RSA algoritması uygulamalarının hızı ve verimi arttırılmaktadır.

2.4.5. Özet (Hash) fonksiyonları

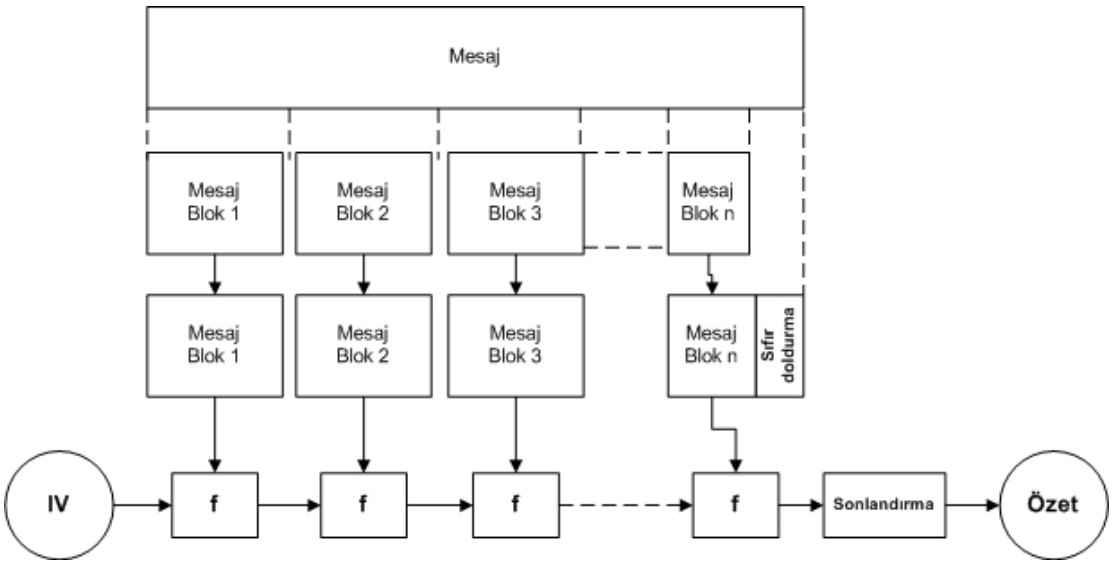
Özet fonksiyonları büyük boyuttaki verileri küçük mesaj özetlerine dönüştürmeye yarayan özetleme algoritmalarıdır. SHA (Secure Hash Algorithm) ve MD5 (Message Digest) bu özetleme algoritmalarından ikisidir. Özet fonksiyonları genellikle veri bütünlüğünü sağlamak ve doğrulamak amacı ile kullanılır. Özet fonksiyonları girdi olan verinin büyüklüğüne bağlı olmaksızın sabit uzunlukta çıktı verirler. Örneğin SHA1 algoritması 160 bitlik çıktı üretir. Veri üzerindeki en ufak bir değişiklik bile verinin özetini değiştirir. Şekil 2.7'de üç farklı mesaj için özet değerleri verilmiştir. Son iki örnek arasındaki tek fark bir "nokta" olduğu halde özet değerinin oldukça farklı olduğu görülmektedir.



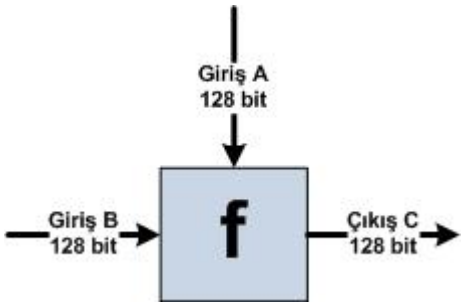
Şekil 2.7. Hash fonksiyonları

Merkle-Damgard yapısı

Bir özet fonksiyonu rasgele seçilmiş uzunluktaki bir mesajı alarak sabit uzunlukta bir çıktı üretebilmelidir. Bu işlem giriş verisini eşit uzunlukta blok serileri halinde parçalara ayırarak ve üzerlerinde tek yönlü sıkıştırma fonksiyonlarını işleterek gerçekleştirilir (Şekil 2.8). Tek yönlü sıkıştırma fonksiyonları sabit uzunlukta iki faklı girdiyi girişlerden biri ile aynı uzunlukta olan bir çıktı haline dönüştüren fonksiyonlardır (Şekil 2.9). Dönüşümün tek yönlü olmasının anlamı verilen bir çıktı için girdinin hesaplanmasının çok güç olmasıdır.



Şekil 2.8. Merkle-Damgard özet (hash) yapısı



Şekil 2.9. Tek yönlü sıkıştırma fonksiyonu

Sıkıştırma fonksiyonu özel olarak özetleme için tasarlanmış ya da bir blok şifreleyiciden inşa edilmiş olabilir. Merkle-Damgard yapısı ile inşa edilmiş bir özet fonksiyonu çakışmalara, sıkıştırma fonksiyonunun olduğu kadar dirençlidir.

İşleme tabi tutulan son bloğun uzunluğu kısa ise aradaki fark doldurulmalıdır. Bu işleme "sıfır doldurma" (padding) denilir ve yapının güvenliği açısından çok önemlidir. SHA1 ve MD5 algoritmalarında Merkle-Damgard yapısı kullanılmaktadır.

Hash fonksiyonlarının genel özellikleri

Özet fonksiyonlarından beklenen bazı genel özellikler vardır. Bunlar M özeti alınacak bilgi, $H(\ )$ özet fonksiyonu ve MD özet bilgi olmak üzere:

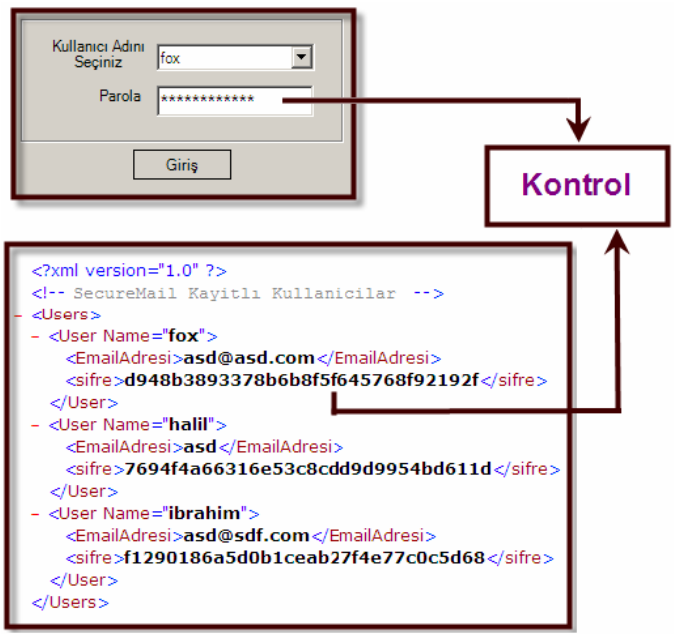
- Verilen bir M için $H(M)$yi hesaplamak kolay olmalı.
- Tek yönlülük (One-Way)
  - M 'den $H(M) = MD$ 'yi bulmak kolayken
  - $H(M) = MD$ formülünden M 'nin bulunmasının zorluğu
- Çakışmazlık (Collision free)
  - M ve M^* mesajlar olmak üzere
  - $H(M) = H(M^*)$ olasılığının çok zor olması

gibi özelliklerdir.

Genel olarak özet fonksiyonlarının kullanım alanları aşağıdaki gibidir:

- Bilgisayar programlarının güvenirliğinin sağlanması: Mesaj özetleri yazılım dünyasında geniş bir şekilde kullanılmaktadır. Özellikle internet sitelerinden çeşitli programlar indirmek istenildiğinde sıklıkla karşılaşılır. Kullanıcı internetten indirdiği programın özel yazılımlar kullanarak özetini alır ve aynı internet sitesinde yayınlanan özet değeri ile karşılaştırır. Böylece indirdiği program veya dosyaların içeriğinin değişip değişmediğinden emin olabilir.
- Veri bütünlüğünün sağlanması

- Parola doğrulama: Parolalar belirli nedenlerden dolayı açık metin olarak saklanmazlar. Bunun yerine parolanın özeti saklanır ve her kullanımda girilen parolanın tekrar özeti alınarak mevcut kayıtlı özet ile karşılaştırılır. Sonuçta iki özet de aynı ise yetkilendirme işlemi tamamlanmış olur (Şekil 2.10).
- Sayısal imza
- Şifreleme algoritmaları



Şekil 2.10. Özet fonksiyonlarının parola doğrulama amaçlı olarak kullanılması

Saldırı Metotları

Özet fonksiyonları üzerine yapılabilecek başarılı bir saldırının anlamı özet fonksiyonunun iddia edilen güvenlik özelliği üzerinde tahrifat yapabilmenin bir yolunu bulmak anlamına gelir. Örnek olarak; mesaj özetinden mesajı tekrar elde edebilecek bir yöntemin bulunması veya geliştirilmesi özet fonksiyonunun tek yönlü olma gibi en önemli özelliklerinden birini başarısızlığa uğrattığı için saldırı olarak adlandırılabilir.

Özet fonksiyonunun temel kriptografik özellikleri tek yönlü (one-way) ve çakışmasız (collision-free) olmasıdır. Yapılabilecek en basit saldırı çok sayıda giriş deneyerek aradığımız çıkışa (özete) uygun girişi bulmak şeklinde olabilir.

Özet fonksiyonunun çıkışının n bit uzunluğunda olduğunu varsayarsak bu durumda 2^n farklı giriş deneyerek çıkış değeri bulunabilir.

Diğer bir tehdit de veritabanına kaydedilmiş özet değerlerinin kullanılarak mesajın bulunmasıdır.

SHA ve MD5

SHA, NSA (National Security Agency) tarafından tasarlanmış, NIST (National Institute of Standards and Technology) tarafından yayınlanmıştır. Algoritmanın tüm türevlerinin özellikleri 2002 yılında yayınlanan FIPS (Federal Information Processing Standards) 180-2 de belirtilmiştir.

MD5 ise 1991 yılında Ron Rivest tarafından geliştirilmiş, diğer bir özetleme algoritmasıdır.

GÜ-Posta uygulamasında kullanıcı hesaplarına ilişkin parola bilgileri kullanıcının isteğine bağlı olarak MD5, SHA1, SHA256, SHA384 veya SHA512 algoritmalarından biri ile özeti alınarak saklanmıştır. Çizelge 2.6'da GÜ-Posta uygulamasında kullanılabilecek algoritmalara için blok ve özet çıktısı uzunlukları verilmiştir.

Çizelge 2.6. Özet alma algoritmalarının giriş blok ve çıkış özet uzunlukları

Algoritma ismi	Giriş Blok Uzunluğu (bit)	Özet uzunluğu (bit)
MD5	512	128
SHA-1	512	160
SHA-256	512	256
SHA-384	1024	384
SHA-512	1024	512

3. E-POSTA GÜVENLİĞİ KAPSAMINDA KULLANILMAKTA OLAN YÖNTEMLER

3.1. IBE (Identity Based Encryption) – Kimlik Temelli Şifreleme

1984 yılında, RSA nın mucitlerinden biri olan Adi Shamir sertifika kullanımı yerine direk kullanıcıların kimliklerinin genel anahtar olarak kullanımını önerdi. Bu açık anahtar şifrelemenin basitleştirilmesi açısından belki de en iyi yaklaşımdır. Uzun bir süre bu yaklaşımın problemlerini çözebilecek bir metot geliştirilemedi. 2000 yılına kadar IBE kriptografinin çözülemeyen problemlerinden biri olarak kaldı. 2001 yılında Stanford ve California Üniversitesi Bilgisayar Bilimleri profesörleri Dr. Boneh ve Dr. Matt Franklin tarafından pratik bir şema bulundu. Yayınladıkları bu çığır açıcı makalede Eliptik eğri ve "Weil Pairing" temelli şemayı tanımladılar ve güvenliğini gösterdiler [27].

"Weil pairing" yerine "Tate pairing" de kullanılabilir. "Tate pairing", "Weil pairing"'e benzer ama hesaplanması daha kolaydır. "Tate pairing" "bilinear map" olarak bilinen özel bir metodu temel alır.

"Bilinear mapping" spesifik matematiksel özelliklerin çiftleştirilmesi (eşlenmesi) olarak tanımlanmıştır.

$$Pair(r \bullet I, s \bullet P) = Pair(r \bullet P, s \bullet I)$$

Bu özellik kripto sistemin çalışması için zorunlu bir unsurdur. Kripto sistemin arkasındaki temel teorik hesaplamalar "bilinear map"'ler kullanılarak gerçekleştirilebilir. Kullanılabilir bir IBE sisteminin inşasındaki zorluk güvenli, hesaplanabilir ve etkin bir "bilinear mapping" algoritmasının bulunmasındadır. Günümüzde güvenli "bilinear map"'ler için bilinen yöntemler eliptik eğriler üzerinde "Weil" ve "Tate" algoritmalarıdır.

"Tate pairing" eliptik eğri üzerinde özel noktalar grubunda tanımlanmış "bilinear map"dir. Bu noktalar grubunun, kripto sistemin çalışmasını sağlayacak belirli

özelliklerinin temin edilmesi için yoğun hesaplamaların yapılması gereklidir. Bu özelliklerden biri de grup derecesini belirten P dir. P büyük asal bir sayıdır. Bu noktalar için "$\bullet$" operatörü özel bir çarpma operatörüdür. Tam sayıları eliptik eğri üzerindeki noktalarla çarpar.

$$3 \bullet P = P + P + P$$

Her ne kadar matematiksel temelde RSA algoritmasından çok daha karmaşık olsa bile IBE kripto sistemi benzer bir prensibe dayanır. Verilen bir P ve $s \bullet P$ için s değerinin hesaplanmasının neredeyse imkânsız olması. Kripto sistemin güvenliği Bilinear Diffie-Hellman probleminin özelliklerinden kanıtlanabilir.

IBE algoritmasının arkasındaki prensibin basitleştirilmiş bir hali aşağıda açıklanmıştır. Gerçek uygulama çok daha karışıktır, noktalar grubu ile ilgili ek bilgi, çeşitli kriptografik özet fonksiyonları ve IBE genel anahtar parametrelerindeki protokole özgü bilgileri gerektirir.

IBE özel anahtarları Özel Anahtar Üretecinde (Private Key Generator) (ÖAÜ) oluşturulur. Bunun için ÖAÜ "s" (master secret) ve eliptik eğri üzerinde sabit bir "p" noktasını kullanır. Bu bilgi, kriptografik rassal sayı üreteci kullanılarak, ilk kullanımda üretilir. Kimlik doğrulaması yapılmış bir kullanıcının IBE özel anahtarının üretilmesi için PKG ilk önce kullanıcının kimliğini özetler ve özeti eliptik eğri üzerindeki $ID_{KULLANICI}$ noktasına eşler. Daha sonra PKG $ID_{KULLANICI}$ ile s'i çarpar.

Örneğin Burak adlı kullanıcı için özel anahtar $s \bullet ID_{Burak}$ olur.

Bir mesajı şifrelemek için; göndericinin sadece kullanıcının kimliğini ve ÖAÜ'ya özgü IBE genel parametrelerini bilmesi gerekir. P, $s \bullet P$ ve eğri içindeki kriptografik özet fonksiyonunu içeren bu parametreler herkese (kamuya) açık hale getirilebilir. IBE genel anahtarları $ID_{KULLANICI}$ 'nin hesaplanması için kriptografik özet

fonksiyonu kullanılarak üretilir. IBE genel parametreleri ile birleştirilmiş bu bilgi mesajın şifrelenmesi için yeterlidir.

IBE ile güvenli mesaj gönderilmesi ve alınması işlemi gönderici ve alıcı açısından aşağıda anlatılmıştır:

Gönderici (Sender) Ayşe

- Burak'a gönderilecek bir mesajı şifrelemek için Ayşe rasgele bir r sayısı alır ve k anahtarını hesaplar. $k = Pair(r \bullet ID_{Burak}, s \bullet P)$
- Daha sonra Burak'a k anahtarı ile şifrelenmiş mesajı gönderir. $E_k[Mesaj]$
- Mesajla birlikte ayrıca $r \bullet P$ değerini de gönderir.

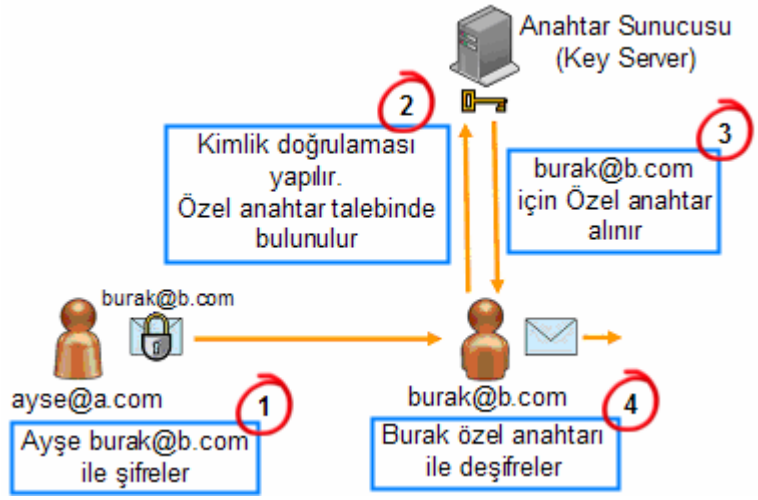
Alıcı (Receiver) Burak

- Mesaj alındıktan sonra Burak anahtarı $k = Pair(s \bullet ID_{Burak}, r \bullet P)$ hesaplayarak yeniden oluşturur.
- Mesajı k anahtarını kullanarak deşifre eder. Burak özel anahtarı $s \bullet ID_{Burak}$ bilen tek kişi olduğu için k 'yı başka hiç kimse hesaplayamaz.

3.1.1. Uygulamada IBE

Ayşe, Burak ile güvenli bir şekilde mesajlaşmak istediğinde Ayşe'nin tek yapması gereken Burak'ın e-mail adresini genel anahtar olarak kullanarak mesajı şifrelemek ve şifreli mesajı Burak'a göndermektir. Burak mesajı aldığında ise IBE anahtar sunucusundan özel anahtarını talep eder. Burak IBE sunucusunun kimlik doğrulamasından geçtikten sonra (akıllı kart ya da parola kullanımı ile) sunucu deşifreleme için gerekli anahtarı üretir ve Burak'a gönderir (Şekil 3.1). Pratikte IBE sistemi kısa ömürlü anahtarlar kullanır. Bu durum anahtar tazeliği (key freshness) kavramını tanımlar. Anahtar ömürleri güvenlik politikasına bağlı olarak saatlik, günlük ya da haftalık olabilir. Burak o zaman periyodunda geçerli olan anahtarı elde etmek için anahtar sunucusuna tekrar bağlanarak talepte bulunur. Bu durum

göndericinin (Ayşe), alıcının sistemdeki durumunun kontrolüne ihtiyaç duymamasını sağlar. Eğer alıcı işten ayrılmışsa IBE anahtar sunucusu otomatik olarak yeni özel anahtar yayımlamayı (kullanıcı tekrar kimlik doğrulamasından geçemeyeceği için) durdurur. Göndericinin geleneksel AAA'dan farklı olarak özel bir şey yapmasına gerek yoktur.



Şekil 3.1. IBE algoritmasının genel işleyişi

Özetle, IBE genel anahtarları kullanıcıların kimlikleridir. IBE özel anahtar üretimi kuruluş tarafından yapılır ve genel parametreler kuruluşun erişim alanından (domain) kolayca alınır. IBE gizlilik, kimlik denetimi, bütünlük, inkar edememezlik gibi güvenliğin temel özelliklerini sağlar.

3.1.2. IBE potansiyel avantajları

Sertifikalar olmadan basitlik

Geleneksel AAA dan farklı olarak, IBE karışık ön-kayıt (pre-enrollment) ya da iptal (revocation) listesi kontrolü gerektirmez. Esasen sertifikalara ihtiyaç yoktur. Bunun yerine alıcının genel anahtarı kendi kimliğinden üretilir.

IBE sisteminde aynı zamanda AAA daki genel anahtarların üretimi, onaylanması ve saklanması gibi karmaşık süreçler yoktur. IBE, genel anahtar e-mail adresini (ya da başka bir kimlik bilgisini) temel aldığı için çok basittir.

IBE sunucusunun sürekli olarak saklayacağı tek bilgi "master secret" dır. "master secret" esasen güvenli bölgede (secure domain) büyük rassal bir sayıdır. Sunucu sırrı (secret) iki amaçla kullanır:

- IBE yazılımını kuran kullanıcıların genel anahtar parametrelerinin üretilmesi.
- Alıcıdan talep geldiğinde her bir alıcı için o haftanın özel anahtarının üretilmesi.

Her bir alıcı ve hafta için üretilen yeni genel anahtar üç bileşenden oluşur.

- Genel anahtar çekirdeği (The public key "seed")
- Haftanın numarası
- Alıcının kimliği (e-mail adresi)

Ön kayıt işleminin gerekmemesi (No pre-enrollment)

Alıcının kendisine gönderilen bir mesajı deşifre etmesi için önceden kayıt olmuş olmasına gerek yoktur.

Anahtar iptali yerine anahtar süre bitimi (Key expiration Instead Of Revocation)

AAA da karşılaşılan en zor problemlerden biri gizliliğin ihlali sonucunda ya da çalışanın işten ayrılması gibi durumlarda genel anahtarların iptal edilmesidir. IBE bu soruna çözüm için genel anahtarın üretiminde anahtar bilgisine hafta numarası ilave eder. Böylece IBE sistemi her bir alıcı için her hafta yeni bir özel anahtar yayımlamalıdır.

IBE ile zamanda yolculuk (Time Travel Using IBE)

IBE sisteminde ancak ileriki bir tarihte deşifre edilebilecek şekilde şifrelemeler de yapılabilir.

3.1.3. IBE sisteminde kullanılan veri türleri

Genel Parametreler (Public Parameters): Genel sistem üzerinde tanımlı parametreler kümesidir. Özel anahtar sunucusu (Private Key Server) tarafından üretilir ve yayımlanır.

Ana Sır (Master Secret): Anahtar sunucusu tarafından üretilen ve gizli bir şekilde tutulan ana anahtar (master key)'dır. Kullanıcıların özel anahtarlarının üretilmesinde kullanılır.

Kimlik (Identity): Genellikle sistem kullanıcısını ifade eden, insan tarafından anlaşılabilir, belirsiz olmayan keyfi bir dizgi (string) dir. Zaman damgası ve diğer nitelikler eklenebilir.

Genel Anahtar (Public Key): Algoritmik olarak kimlikten (identity) türetilen bir dizgi (string) dir. Türetim herhangi biri tarafından özerk (otonom) olarak yapılabilir.

Özel Anahtar (Private Key): Özel anahtar, yayımlanmış genel parametreler kümesi altında verilen kimliğe uyuşacak şekilde anahtar sunucusu tarafından yayımlanır.

Düz Metin (Plaintext): Güvenli bir şekilde transfer edilmek istenen verinin şifrelenmemiş halidir.

Şifreli Metin (Ciphertext): Güvenli bir şekilde transfer edilmek istenen verinin (düz metinin) şifrelenmiş halidir.

3.2. PEM

PEM Gizliliği Arttırılmış Posta (Privacy Enhanced Mail), açık anahtarlı şifreleme yöntemi kullanarak e-posta iletişiminin güvenliğinin sağlanması için IETF tarafından önerilen bir standarttır. IETF tarafından önerilen bir standart olmasına rağmen yaygın olarak kullanılmamıştır.

PEM gizlilik, kimlik kanıtlama ve veri bütünlüğünün sağlanması amacıyla bünyesinde bir dizi kriptografik tekniği barındırır.

PEM metin tabanlı e-posta mesajlarının güvenliğinin sağlanmasında kullanılan ilk standartlardan biridir. Sadece 7-bitlik metin tabanlı mesajları şifreleyebilmektedir. Ayrıca sayısal imzaların dağıtımının ve doğrulanmasının hiyerarşik yapısını da tanımlamıştır. PEM internet gibi geniş ağlarda anahtar yönetimi için bir açık anahtar alt yapısı da belirlemiştir. Ancak daha sonrasında yapılan çalışmalar yetersiz olmuş yeni başka standartlar geliştirilmiştir.

MIME sayesinde ikili (binary) eklentilerin de e-posta mesajına eklenebilmesi ile PEM önemsiz bir hale gelmiştir. PEM ile ilgili yayımlanmış olan RFC dokümanları Çizelge 3.1'de verilmiştir.

Çizelge 3.1. PEM RFC dokümanları

Doküman No	Doküman Başlığı
RFC 1421	İnternet elektronik postası için gizlilik iyileştirmesi: Bölüm 1: Mesaj şifreleme ve kimlik doğrulama yöntemleri.
RFC 1422	İnternet elektronik postası için gizlilik iyileştirmesi: Bölüm 2: Sertifika temelli anahtar yönetimi
RFC 1423	İnternet elektronik postası için gizlilik iyileştirmesi: Bölüm 3: Algoritmalar, biçimler ve belirteçler.
RFC 1424	İnternet elektronik postası için gizlilik iyileştirmesi: Bölüm 1: Anahtar sertifikasyonu ve ilgili servisler

3.3. MOSS

MOSS, MIME Nesne Güvenlik Servisleri (MIME Object Security Services), MIME nesnelerine sayısal imza ve şifreleme hizmetleri vermek üzere geliştirilmiş bir protokoldür.

MOSS, mesaj göndericisi ve alıcısı arasında (end to end) uygulama katmanında hizmet verir.

MOSS, PGP'nin popülaritesi ve yaygınlaşması ile geniş bir kullanım alanı bulamamıştır.

3.4. S/MIME

S/MIME Güvenli / Çok amaçlı İnternet Posta Eklentisi (Secure / Multipurpose Internet Mail Extension) temel olarak RSA Data Security firması tarafından geliştirilen standartlar ve MIME üzerine inşa edilmiştir.

IETF (Internet Engineering Task Force) tarafından geliştirilen MIME standardına ek olarak PKCS#7 (Cryptographic Message Syntax) içerisinde tanımlanan güvenlik eklentilerini içerir.

S/MIME, MIME formatına sayısal imza ve şifreleme özelliklerini ekler. S/MIME sayesinde farklı e-mail yazılımı kullanan kullanıcılar birbirleriyle iletişim sağlayabilir. S/MIME protokolünü destekleyen yazılımlar ve sertifikalar kullanılarak güvenli e-mail alıp yollanabilir. Birçok Web tabanlı e-posta programı ve bazı e-posta programları dijital imzaları desteklemez.

S/MIME, açık anahtarların sahiplerinin doğruluğunu garantileme mekanizması olarak e-imza uygulamasında olduğu gibi güvenli sertifika otoritelerini (Certification Authority – CA) kullanmaktadır.

S/MIME'ın ikinci sürümü anahtar uzunlukları ile ilişkin güvenlik riskleri taşıdığı düşünüldüğünden IETF standardı olarak kabul edilmemiştir. S/MIME üçüncü sürümü IETF standartıdır.

S/MIME ile GÜ-Posta uygulamasının karşılaştırması Çizelge 3.2'de verilmiştir.

Çizelge 3.2. S/MIME ile GÜ-Posta uygulamasının karşılaştırılması

	S/MIME	GÜ-Posta
Gizlilik	Sağlar	Sağlar
Bütünlük	Sağlar	Sağlar
Kimlik kanıtlama	Sağlar	Sağlar
İnkâr edememezlik	Sağlar	Sağlar
Ücret	• Ücretli ve ücretsiz e-posta istemci programları ile tümleşik.	• Ücretsiz
Gerekli altyapı	• S/MIME destekleyen bir e-posta istemci programı • X-509 sertifikası	• Windows işletim sistemi • .NET framework 3.5
Desteklenen şifreleme algoritmaları	• NIST tarafından önerilen algoritmalar: DES, 3DES • Geriye uyumluluk için: RC2-40 bit ve RC2-56 bit • Güvenlik ihtiyacının üst seviye olduğu organizasyonlarda AES	• DES, 3DES, RC2, AES
Desteklenen imzalama algoritmaları	• RSA	• RSA
Uygulama-kullanım kolaylığı	• E-posta istemci programlarında ilk ayarların yapılması deneyimsiz kullanıcılar açısından zordur. • Kullanımı kolay.	• Kolay
Anahtar Yönetimi	Kullanıcı sertifika otoritesinden dijital sertifikasını alır.	Uygulamanın “Anahtar yönetimi” ekranı kullanılarak genel anahtar dağıtımı ve alımı kolaylıkla yapılabilir.
Sertifika Kullanımı	Var	Yok
E-posta istemci programı ihtiyacı	Var	Yok
Ek bilgiler	• Geniş grup ve organizasyonlar için uygundur. • En yaygın kullanılan e-posta şifreleme standardıdır. • Yaygın kullanılan e-posta istemci programları ile kullanılabilir.	• Az kullanıcılı, küçük gruplar için uygundur • Gelecek çalışmalarda nitelikli e-imza ile entegrasyonu düşünülmektedir.

3.5. PGP

Pretty Good Privacy (PGP) kriptografik gizlilik ve kimlik doğrulama sağlayan, 1991 yılında Philip Zimmermann tarafından oluşturulmuş bir yazılımdır.

PGP şifreleme ve sayısal imzalama programıdır. E-posta gönderme özelliği yoktur. Sadece E-posta olarak gönderilebilir halde dosyalar yaratır. Değişik platformlarda (MS-DOS, UNIX, Macintosh) çalışabilmektedir.

PGP şu anda kullanılan en yaygın E-posta şifreleme programıdır. Ancak, bu gelişim süreci içinde yaratıcısı Philip Zimmermann'ın başını oldukça ağrıtmıştır. PGP güçlü şifreleme algoritmaları içerdiğinden, ABD gümrük kurallarına göre ABD dışına çıkartılması yasak bir üründür. 1991 yılında internet üzerinde kaynak kodunun yayımlanması ile Amerikan hükümeti soruşturma başlatmış, açılan dava 1996 yılında düşmüştür.

PGP'nin ticari olmayan kişiler ve kuruluşlar tarafından edinilmesi ve kullanımı tüm dünya da ücretsizdir.

PGP emir bazında işlem yapan bir arayüze sahiptir. Orjinal PGP'nin grafik arayüzü yoktur. O yüzden kullanımı gayet zordur.

PGP mesaj şifreleme ve imzalama özelliklerinin yanı sıra açık ve gizli anahtarlarla ilgili işlemler de yapabilmektedir.

PGP'de şifreleme ve imzalama

PGP, temel olarak RSA algoritmasına dayanır. Ancak PGP, RSA'nın performansının diğer özel anahtar tabanlı şifreleme algoritmalarına göre düşük olması nedeniyle, mesaj şifrelerken IDEA özel anahtar algoritmasını kullanır. Bu şifrelemede kullanılan 128 bitlik anahtar ise RSA kullanılarak şifrelenir ve mesaj paketinin başına eklenir. IDEA'nin kullandığı 128 bitlik bu oturum anahtarı rasgele yaratılır ve sadece bir kere kullanılır. Mesaj imzalanırken ise mesajın tümü değil 128 bitlik bir özü (hash) imzalanır ve bu imza mesaj paketine konur. Öz yaratmak için ise MD5 hash algoritması kullanılır.

PGP'nin başka bir özelliği ise mesajı şifrelemeden önce sıkıştırmasıdır. Bu şekilde, hem saklama alanından, hem de bant genişliğinden tasarruf sağlanmış olur. Sıkıştırmanın diğer bir yararı ise harflerin sıklığını kullanan şifre çözümleyici ataklara karşı şifreyi koruyabilmesidir.

PGP, şifrelenmiş mesajı ve sayısal imzayı aynı pakete koyabildiği gibi ayrı ayrı şifrelenmiş mesaj ve imza da üretebilmektedir.

PGP imzanın ve şifrelenmiş mesajın başına zaman damgaları (time stamp) koyabilmektedir. Böylelikle, alıcı mesajın ve imzanın ne zaman gönderildiğinden emin olur.

PGP'de anahtar işlemleri

PGP kullanarak RSA anahtarları yaratmak mümkündür. Bu şekilde gizli anahtar ve ona karşılık gelen açık anahtar yaratılır. RSA anahtarlarının boyu kullanıcı isteğine bağlı olarak 512 ile 2048 bit arasında değişebilir.

GnuPG ve OpenPGP

PGP ortaya çıkmasından sonra PGP'nin değişik türevleri de geliştirilmiştir. Bunlardan en önemlileri GnuPG ve OpenPGP'dir. GnuPG (GNU Privacy Guard) PGP'nin açık kaynak uyarlamasıdır ve "http://www.gnupg.org/" adresinden ücretsiz olarak indirilebilir. GnuPG sunucuda çalışan bir uygulamadır (back-end application) şifreleme ve imzalama yetenekleri vardır. OpenPGP e-posta şifrelemesi için kullanılan tescilli olmayan bir standarttır. PGP'yi temel almıştır. OpenPGP protokolü şifrelenmiş mesajlar, imzalar, sertifikalar ve genel anahtarların değiştirilmesi için standart biçimleri tanımlar. İstemci tarafında (front-end) e-posta istemcileri ile çalışan şifreleme ve imzalama işlemlerini gerçekleştiren bir çözüm sunar. IETF tarafından RFC 4880 standardı olarak yayımlanmıştır. Herkes tarafından herhangi bir ücret ya da lisans harcı ödenmeden kullanılabilir.

PGP ile GÜ-Posta uygulamasının karşılaştırması Çizelge 3.3'de verilmiştir.

Çizelge 3.3. PGP ile GÜ-Posta uygulamasının karşılaştırması

	PGP	**GÜ-Posta**
Gizlilik	Sağlar	Sağlar
Bütünlük	Sağlar	Sağlar
Kimlik kanıtlama	Sağlar	Sağlar
İnkâr edememezlik	Sağlar	Sağlar
Ücret	• Ücretli ve ücretsiz sürümleri var.	• Ücretsiz
Gerekli altyapı	• PGP destekleyen bir e-posta istemci programı	• Windows işletim sistemi • .NET framework 3.5
Desteklenen şifreleme algoritmaları	• DES, 3DES, AES	• DES, 3DES, RC2, AES
Desteklenen imzalama algoritmaları	• DSA,RSA	• RSA
Uygulama-kullanım kolaylığı	• Ücretsiz olan sürümlerinin kullanımı zordur.	• Kolay
Anahtar Yönetimi	Kullanıcılar iletişim kurmak istedikleri diğer kullanıcıların yayımladıkları genel anahtarı temin etmeleri gerekir.	Uygulamanın "Anahtar yönetimi" ekranı kullanılarak genel anahtar dağıtımı ve alımı kolaylıkla yapılabilir.
Sertifika Kullanımı	Yok	Yok
E-posta istemci programı ihtiyacı	Var	Yok
Ek bilgiler	• PGP'de merkezi bir anahtar otoritesi yoktur. Güvenli e-posta iletişimi kurmak isteyen kullanıcılar birbirlerinin genel anahtarlarını alırlar. • Az kullanıcılı, küçük gruplar için uygundur. • Gerektiğinde harici AAA'sı kullanabilir. • Birçok e-posta istemci programı ile birlikte kullanılabilir.	• Az kullanıcılı, küçük gruplar için uygundur • Gelecek çalışmalarda nitelikli e-imza ile entegrasyonu düşünülmektedir.

4. GÜVENLİ BİR E-POSTA SİSTEMİNİN (GÜ-POSTA) TASARLANMASI

Tez çalışması çerçevesinde geliştirilmiş olan GÜ-Posta uygulamasının genel mimarisi ile kullanılan teknoloji ve yöntemler bu bölümde anlatılmıştır.

4.1. Kullanılan Teknoloji ve Yöntemler

GÜ-Posta uygulamasının geliştirilme aşamasında Microsoft Visual Studio 2008 Professional Edition adlı geliştirme ortamı kullanılmıştır. Genel anlamda. NET olarak bilinen platformda bu geliştirme ortamı (IDE) sayesinde desteklenen dillerden dilediğinizi kullanarak çok kapsamlı ve güçlü programlar hazırlanabilmektedir. Bu çalışma kapsamında C# programlama dili tercih edilmiştir.

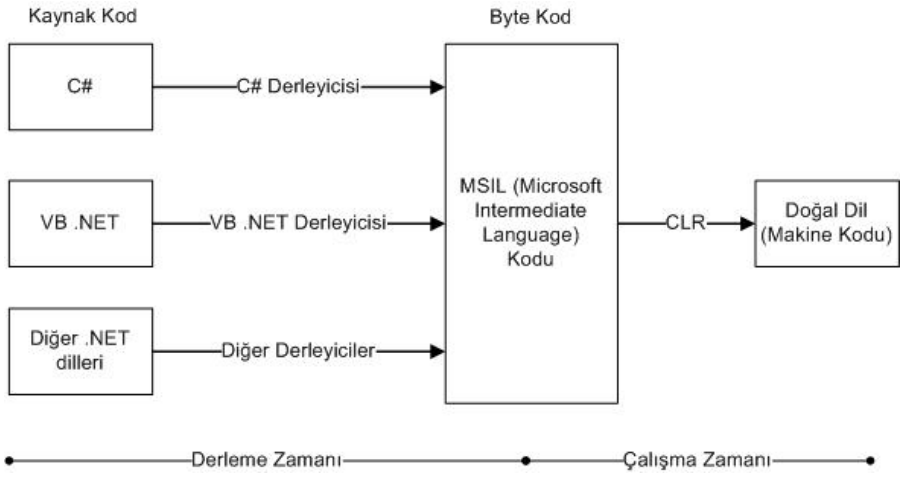
4.1.1. .NET teknolojileri ve C# programlama dili

.NET uygulama geliştiricilerin yazılım geliştirme sürecinde altyapı işlemleri için harcadığı çabayı en aza indirgemek ve daha güvenli, güvenilir ve sağlıklı uygulamalar geliştirebilmelerini sağlamak için Microsoft tarafından geliştirilmiş bir altyapıdır. 1990 yılında başlayan çalışmalar ile Microsoft uygulama geliştiricilerin ve son kullanıcıların işlerini kolaylaştıracak olan platform geliştirmeye başladı. Çok geniş analizler ve geliştirme ekibinin yoğun çalışmaları sonucu ortaya çıkan ürünler 2000 yılında sunuldu. Büyük bir beğeni ile karşılanan bu altyapı sürekli olarak geliştirilmeye devam etmekte ve yaygın bir şekilde kullanılmaktadır.

CLR (Common Language Runtime)

CLR .NET altyapısında programların çalışmasını kontrol eden ve işletim sistemi ile programımız arasında yer alan arabirimdir. Normalde bir uygulamanın farklı işletim sistemlerinde (Windows, Linux, MacOS vb.) çalışması istendiğinde her bir işletim sistemi için programın ayrı ayrı yazılıp kullanılacak olan işletim sistemine uygun olan derleyici kullanılarak derlenmesi gerekmektedir. Bu durumda platformdan bağımsız uygulamalar geliştirebilmek için bir ara dil kullanmak ve her bir platform için bu ara dile çevrilmiş programın kodunu çalıştıracak altyapıları hazırlamak

çözüm olarak düşünülmüş ve .NET teknolojisi doğmuştur. .NET platformu MSIL(Microsoft intermediate Language) olarak adlandırılan ara dili kullanmaktadır. .NET platformunda uygulama geliştirmek için hangi dili kullanılırsa kullanılsın yazılan programın kodu direkt olarak makine dilinden önce MSIL'e çevrilir. Daha sonra ise program çalıştırılan sistemde kurulu olan CLR ile çalışma anında MSIL kodlarını çevirerek program çalıştırılır, çalışma anında derleme işlemini ise JIT derleyicileri (Just in Time compilers) üstlenir (Şekil 4.1).



Şekil 4.1. .NET mimarisi

C#; çok karmaşık bir yapıda olmayan, hata yapma olasılığı düşük, son derece güçlü, yüksek performanslı, "Nesneye Yönelik Programlama" gibi güçlü paradigmalara destek veren bir dildir. C#'ın bu kadar güçlü ve işlevsel olmasını sağlayan unsurların başında ise NetFramework olarak adlandırılan framework gelmektedir.

Bu framework; en çok hata yapılan noktalar olan kaynak yönetimini otomatik olarak gerçekleştirir, kendisini destekleyen diller arasında kusursuz bir iletişim sağlar, yapılabilecek hemen her tür işlem için (dosya ve veritabanı erişimi, grafik programlama, network programlama vb.) hazır fonksiyon kütüphaneleri ve işletim sistemleri arasında taşınabilirlik sunar.

.NET Framework

.NET platformu dilden bağımsızdır ve programların yürütülebilmesi için ihtiyaç duyulan tüm ortak servisler .NET Framework tarafından sağlanır. .NET Framework, Microsoft (Microsoft Visual Basic.NET, Microsoft Visual C#, vb) ve başka üçüncü parti birçok dil (COBOL, Pascal, Perl, Python vb) için de destek sağlar.

.NET Framework .NET Platformunun bütünleşik parçası olan teknolojiler kümesidir. .NET Framework bileşenleri:

- Common Language Runtime (CLR),
- .NET Framework Sınıf Kütüphanesi,
- ADO.NET (veri ve XML),
- ASP.NET (web formları-ya da mobil web formlar- ve servisleri)
- Kullanıcı ara yüzü

şeklinde sınıflandırılabilir.

Namespace (İsim Uzayı)

Programlama dillerinde, programcıların işlerini kolaylaştırmak için bir takım hazır kütüphaneler mevcuttur, bu kütüphanelerden bazıları standart olmakla birlikte bazıları programcılar tarafından sonradan geliştirilmiş ve kullanıcıların hizmetine sunulmuştur. Örneğin MFC ve ATL gibi kütüphanelerin kendilerine has amaçları vardır, MFC kütüphanesi ile bir takım hazır C++ sınıflarına ulaşarak temelde zor olan bir takım Windows platformuna özgü işlemler yapılabilir. Bu da programcılara uygulama geliştirmek için daha az zaman harcatır. Bu tür kütüphaneler Visul Basic'te ve Java dilinde de vardır. Fakat bu dillerin aksine C# dili ile gelen hazır bir takım sınıf kütüphaneleri bulunmamaktadır, kısacası standart bir C# kütüphanesi mevcut değildir. NET Framework'ün programcılara sunduğu temel sınıflar mevcuttur. Bu sınıfları iyi organize edebilmek için .NET, namespace kavramı kullanılmaktadır. .NET teki sınıf kütüphaneleri bir dilden bağımsız bir yapıdadır. Visual Basic.NET kullanıcısı ile C# kullanıcısı aynı kütüphaneden faydalanırlar.

Namespace'ler .NET Framework sınıf kütüphanesindeki veri türlerini ve sınıfları kullanabilmemiz için C# dilinde "using" anahtar sözcüğü ile birlikte kullanılır ve derleyiciye bildirilir. Diğer dillerde ise bu isim alanları farklı şekilde derleyiciye bildirilir, ama temelde yapılan iş .NET Framework sınıf kütüphanelerini kullanma hakkı almaktır. Farklı programları incelendiğinde namespace'lerin sadece eklenme biçimi ve namespace'lerde ki sınıfların söz dizim olarak kullanımının farklı olduğun görülebilir.

4.1.2. XML

XML (Genişletilebilir İşaretleme Dili-Extensible Markup Language) HTML ile benzerlik gösteren bir işaretleme (markup) dilidir. Bağımsız bir kuruluş olan W3C (World Wide Web Consortium) organizasyonu tarafından tasarlanmıştır. XML verinin tanımlanması ve tarif edilmesi için kullanılır. HTML'deki yapının aksine XML'de kullanılacak olan etiketler (tag'ler) önceden tanımlı değildir. XML dokümanının yapısı tamamıyla kullanıcı tarafından oluşturulur. Verinin tarif edilmesi için DTD (Document Type Definition) adı verilen yapılar kullanılmaktadır. XML ve DTD'nin birlikte kullanılması ile dokümanlar kendini tarif eden bir yapı halini alırlar. XML ve HTML arasındaki en belirgin fark XML'in verinin kendisiyle ilgilenmesi HTML'in ise verinin sunumuyla ilgilenmesidir. XML bir dil olmakla birlikte aynı zamanda bir teknolojiyi de ifade eder.

XML'in genel özellikleri:

- Belge ve verilerin yapılandırılmasını sağlayan evrensel bir formattır.
- Metin (text) tabanlı işaretleme dilidir.
- Veri alış verişinde kullanılan bir standarttır.
- Bilginin yapısını tanımlamak için kullanılan bir teknolojidir.
- İşaretleme (markup) dillerini tanımlayan bir meta dildir.

Örnek bir XML dokümanı Şekil 4.2'de verilmiştir.

Şekil 4.2. Örnek bir XML dokümanının içeriği

Bu yapıda HTML'de kullanılan <h1> ve <p> gibi standart etiket (tag) yapıları yerine kendi hazırladığımız etiketler kullanılmıştır. Bu da XML dokümanlarının genişletilebilir bir yapıya sahip olduğunun en güzel örneğidir.

Çok farklı tipteki verileri orijinal formatlarında tek bir çatı altında tutabilen XML, bilgiye hızlı, kolay ve ortamdan bağımsız olarak erişebilme imkânı sunar. Günümüzde birçok bilgisayar uygulaması ve internet servisleri XML dokümanlarını ve teknolojilerini kullanmaktadır. Özellikle; ortamdan bağımsız olarak farklı tipteki verilerin hiyerarşik bir yapıda kullanılabilmeleri, taşınabilmeleri ve hızlı bir şekilde sorgulanabilmeleri XML teknolojisini vazgeçilmez kılmaktadır.

4.2. GÜ-Posta Geliştirme Süreçleri, Mimarisi ve Bileşenleri

4.2.1. Genel bilgiler

GÜ-Posta uygulaması ile mesajların istenmeyen kişiler tarafından okunması, mesaj içeriğinin değiştirilmesi gibi tehditlere karşı güvenli bir e-posta uygulaması geliştirilmiştir. Uygulama çerçevesinde her biri kendi alanında başarılı olan ve yaygın bir şekilde kullanılan DES, 3DES, RC2, RSA, MD5 ve SHA algoritmaları kullanılmıştır. GÜ-Posta geliştirilmiş bazı uygulamaların aksine [28], MS Outlook programından bağımsız çalışan, kendi başına e-posta hizmeti sunan, bir yazılımdır. Ayrıca kullanıcı farklı şifreleme algoritmaları ve anahtar uzunluğunu (RSA anahtarları için) da seçebilir.

4.2.2. Geliştirme süreçleri

GÜ-Posta yazılımının geliştirilmesi iki ana süreçte yapılan faaliyetler sonucu olmuştur. Bu süreçler aşağıda verilmiştir:

- Planlama süreci.
- Gerçekleştirme süreci.

Planlama süreci

Planlama sürecinde yapılanları sırası ile inceleyecek olursak:

- Geliştirilecek olan uygulamadan beklenen gereksinimler belirlenmiştir.
- Yazılım geliştirme ortamı seçimi: Microsoft Visual Studio 2008 Professional tercih edilmiştir.
- Yazılım geliştirme metodolojisinin belirlenmesi: Nesne yönelimli yazılım geliştirme metodolojisi seçilmiştir.
- Yazılım mimarisinin belirlenmesi: Programın hangi modüllerden oluşması gerektiği üzerinde çalışma yapılmıştır.
- Programın, kullanıcı ara yüzleri için taslakların hazırlanması.
- Gerekli olacak araç ve bileşenlerin kullanımının öğrenilmesi için bir çalışma programı hazırlanması.

Geliştirme süreci

Geliştirme sürecinde yapılanları sırası ile inceleyecek olursak:

- Planlama sürecinde taslağı hazırlanan kullanıcı ara yüzlerinin Visual Studio Designer ile hazırlanması.
- Gerekli modüllerin ve sınıfların hazırlanması.
- Her modül için hazırlanan kodların kullanıcı ara yüzünde ilgili bileşenlere bağlanması.
- Test aşaması
- Hazırlanan programın test edilmesi ve hatalı kodların düzeltilmesi.

- Gerekli optimizasyonların yapılması.
- Programın kullanıcı ara yüzünde bulunan, kullanıcının veri girişi yapması gereken kısımlarda veri girişi kontrolünün yapılması ve olabilecek hataların minimuma indirgenmesi.

4.2.3. Yazılım mimarisi

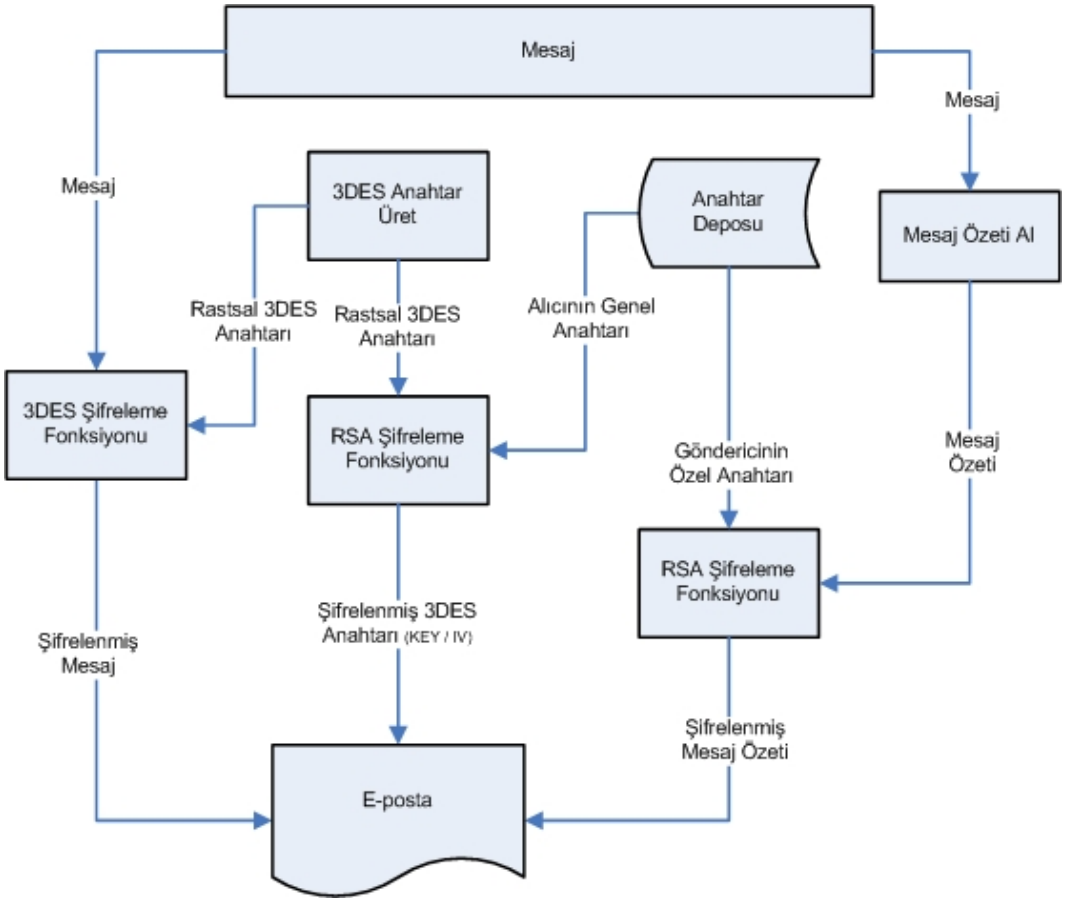
Geliştirilen GÜ-Posta yazılımı üç adet kullanıcı ara yüzünden (Windows formundan) oluşmaktadır.
Bu ara yüzler ile kullanıcı mesaj gönderme, mesaj alma/okuma, yeni kullanıcı kaydı ve anahtar yönetimi olmak üzere dört ana fonksiyonu yerine getirebilir:

Mesaj gönderme

GÜ-Posta uygulamasında mesaj gönderme işleminin gerçekleştirilebildiği modül genel olarak mesaj metninin şifrelenmesi ve e-posta olarak gönderilmesi işlemlerinden oluşur.

Kullanıcının, şifreleme işlemi için dört alternatifi vardır; DES, 3DES, RC2 veya AES algoritmalarından biri seçebilir.

Şekil 4.3'de mesaj gönderme modülünde yapılan işlemler akış şemasında gösterilmiştir. Verilen örnekte kullanıcının şifreleme algoritması olarak 3DES seçtiği farz edilmiştir (diğer algoritmalar için de benzer bir akış ve işleyiş vardır).



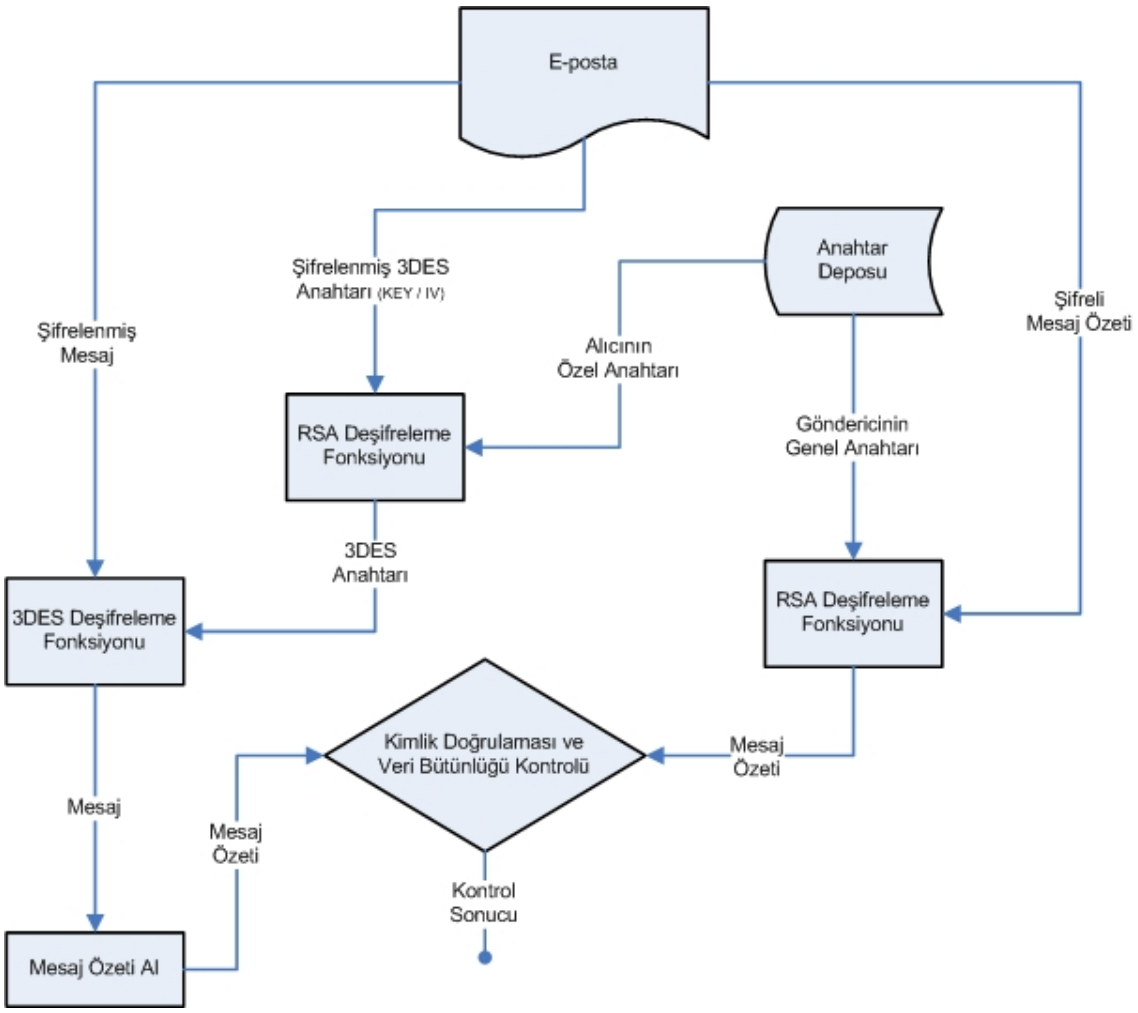
Şekil 4.3. Mesaj gönderme akış şeması

Sonuç olarak oluşturulan e-posta SMTP iletişim kuralı ile alıcının posta kutusuna iletilir.

Mesaj alma/okuma

GÜ-Posta uygulamasında mesaj alma / okuma işleminin gerçekleştirildiği modül genel olarak mesaj metninin şifrelenmesi e-posta iletisinin alınması ve alınan şifreli mesaj metninin deşifre edilmesi işlemlerinden oluşur.

Şekil 4.4'de mesaj alma / okuma modülünde yapılan işlemler akış şemasında gösterilmiştir. Verilen örnekte deşifre edilmek istenen mesajın 3DES şifreleme algoritması ile şifrelenmiş olduğu farz edilmiştir (diğer algoritmalar için de benzer bir akış ve işleyiş vardır).



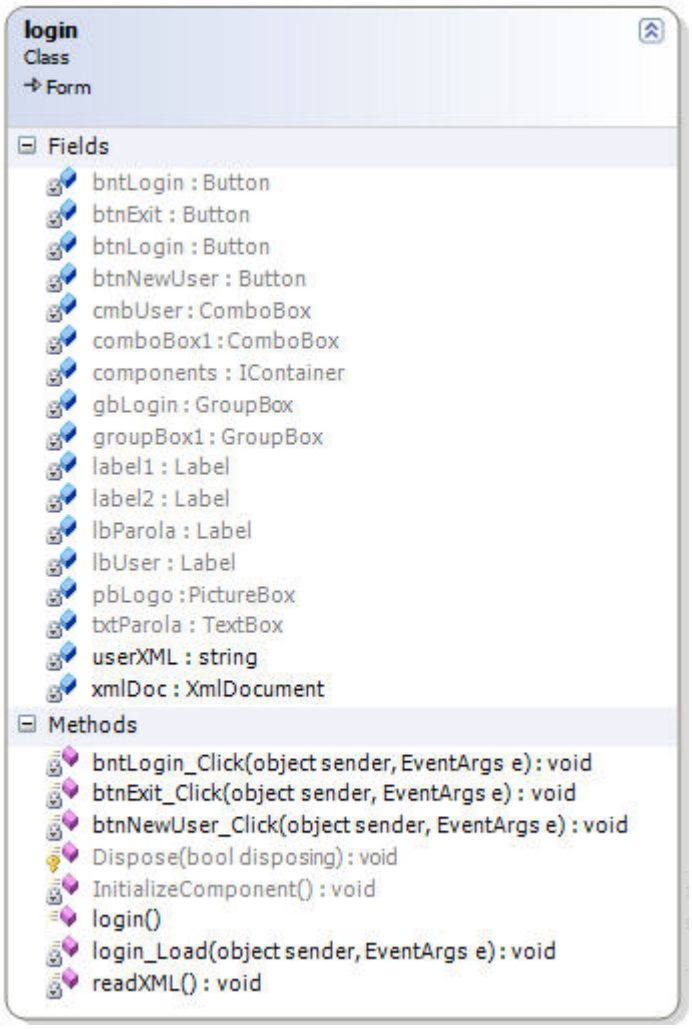
Şekil 4.4. Mesaj alma/okuma akış şeması

E-posta mesajında yer alan ekli dosya dan mesaj metninin şifreleme işleminde hangi yöntemin kullanıldığı anlaşılarak deşifre işlemi de o yönteme uygun olarak gerçekleştirilir. Şekil 4.4'de verilen akışta şifreleme işleminde 3DES kullanılmış olduğu anlaşılarak deşifreleme işlemi bu şifreleme algoritmasına göre yapılmıştır.

4.2.4. Yazılım bileşenleri

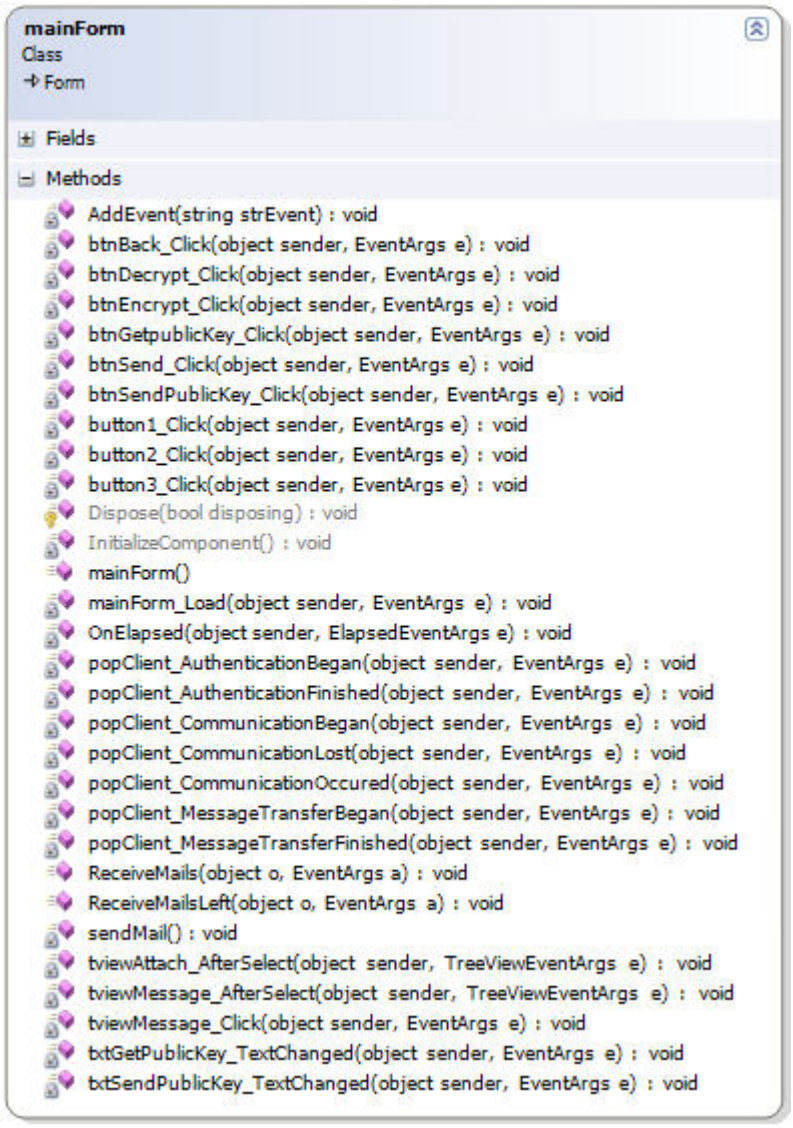
GÜ-Posta yazılımında kullanılan bileşenler ve sınıflar aşağıda verilmiştir. Her bir bileşen ve sınıfın alanları, özellikleri ve metotları ilgili şekillerde gösterilmiştir.

- Kullanıcı giriş ekranını oluşturan “login.cs” dosyası ile ilgili bilgiler (Şekil 4.5).



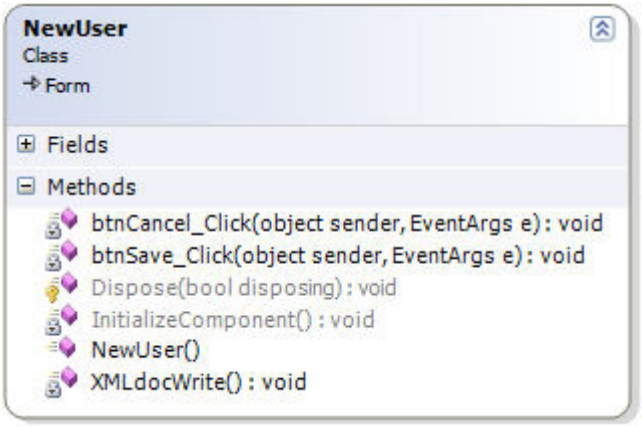
Şekil 4.5. Login.cs sınıf diyagramı

- Mesaj gönderme, alma, okuma ve anahtar yönetimi ara yüzünü oluşturan "mainForm.cs" dosyası ile ilgili bilgiler (Şekil 4.6).



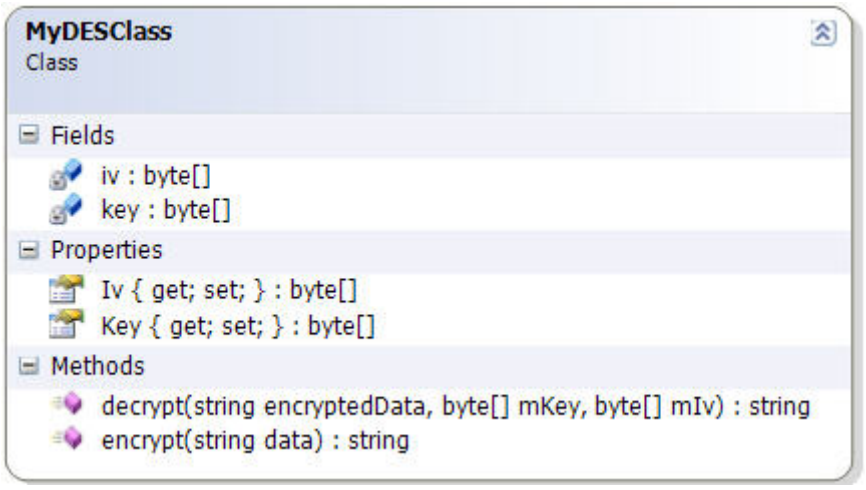
Şekil 4.6. mainForm.cs sınıf diyagramı

- Yeni kullanıcı kaydı ara yüzünü oluşturan “NewUser.cs” dosyası ile ilgili bilgiler Şekil (4.7).



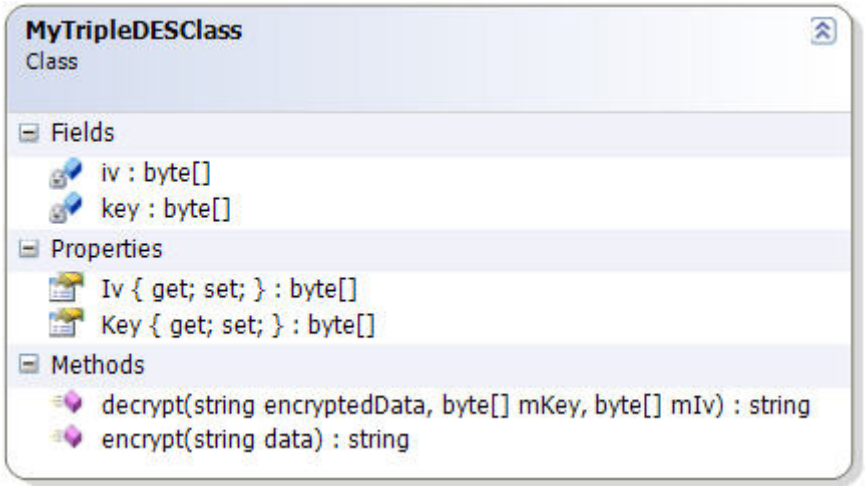
Şekil 4.7. NewUser.cs sınıf diyagramı

- DES şifreleme algoritması ile yapılan şifreleme ve deşifreleme işlemleri için hazırlanmış olan sınıfla ilgili bilgiler (Şekil 4.8).



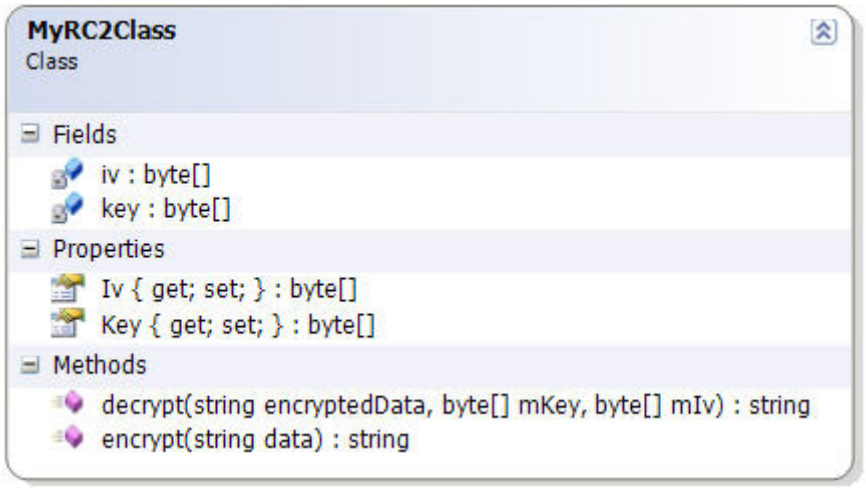
Şekil 4.8. MyDESClass sınıf diyagramı

- 3DES şifreleme algoritması ile yapılan şifreleme ve deşifreleme işlemleri için hazırlanmış olan sınıfla ilgili bilgiler Şekil (4.9).



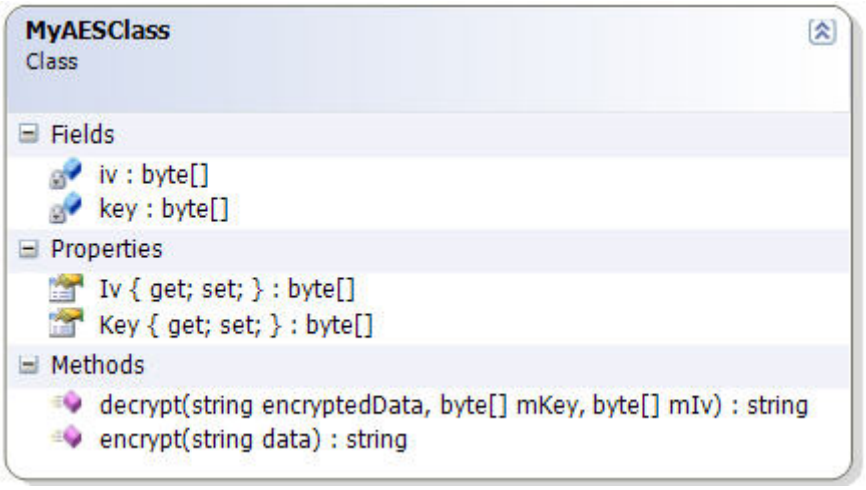
Şekil 4.9. MyTripleDESClass sınıf diyagramı

- RC2 şifreleme algoritması ile yapılan şifreleme ve deşifreleme işlemleri için hazırlanmış olan sınıfla ilgili bilgiler (Şekil 4.10).



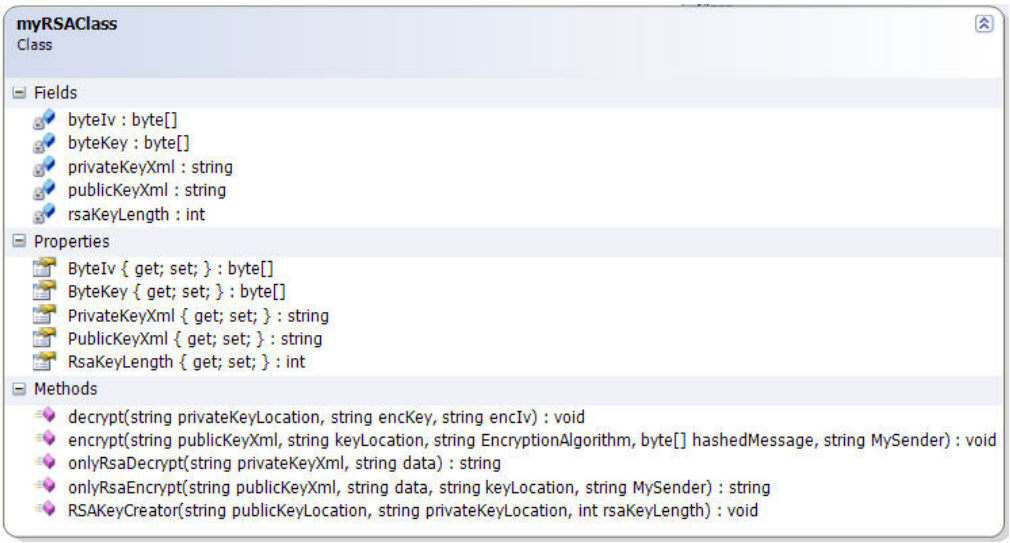
Şekil 4.10. MyRC2Class sınıf diyagramı

- AES şifreleme algoritması ile yapılan şifreleme ve deşifreleme işlemleri için hazırlanmış olan sınıfla ilgili bilgiler (Şekil 4.11).



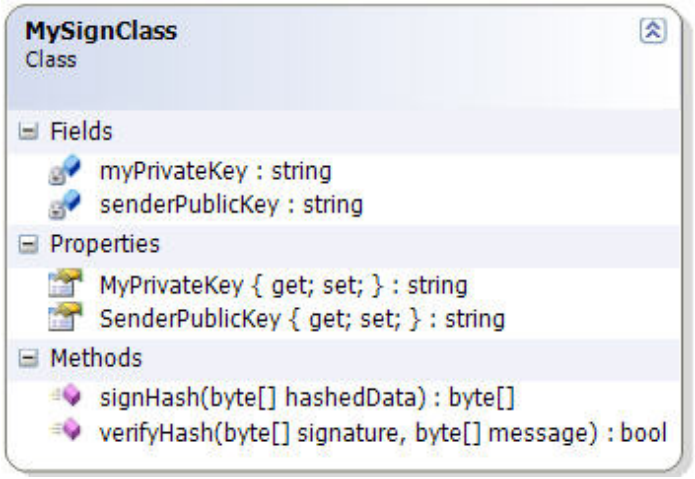
Şekil 4.11. MyAESClass sınıf diyagramı

- RSA algoritması ile yapılan şifreleme ve deşifreleme işlemleri için hazırlanmış olan sınıfla ilgili bilgiler (Şekil 4.12).



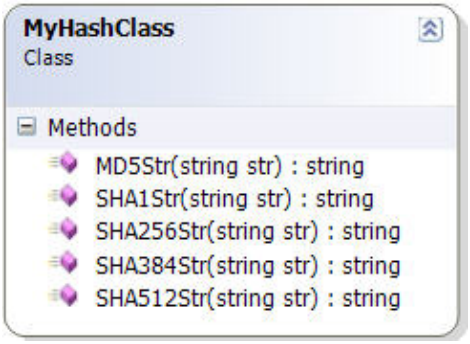
Şekil 4.12. MyRSAClass sınıf diyagramı

- Mesaj özetinin imzalanması işlemi için hazırlanmış olan sınıfla ilgili bilgiler (Şekil 4.13).



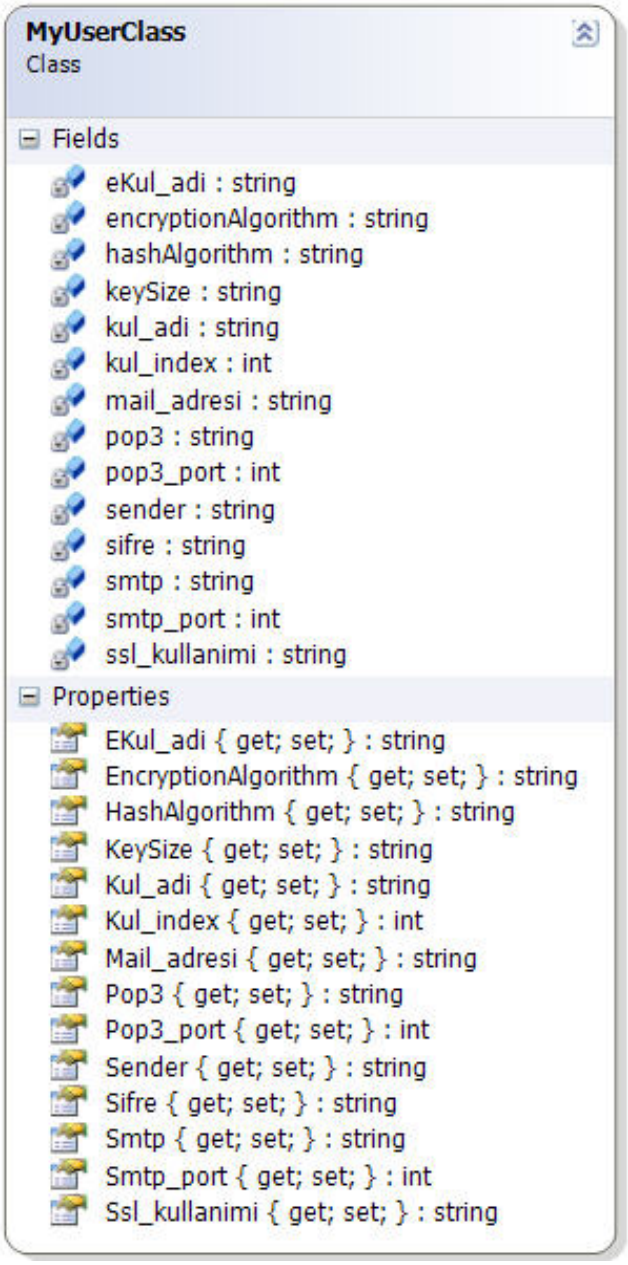
Şekil 4.13. MySignClass sınıf diyagramı

- Mesaj özetinin alınması işlemi için hazırlanmış olan sınıfla ilgili bilgiler (Şekil 4.14).



Şekil 4.14. MyHashClass sınıf diyagramı

- Kullanıcı bilgilerinin yönetimi için hazırlanmış olan sınıfla ilgili bilgiler (Şekil 4.15).



Şekil 4.15. MyUserClass sınıf diyagramı

4.3. GÜ-Posta'nın Tasarlanmasında Temel Alınan Güvenlik Kriterleri

Geliştirilen uygulamada gizlilik, kimlik kanıtlama, bütünlük ve inkar edememe gibi güvenlik kriterlerinin sağlanması amaçlanmıştır. Bu amaçla:

- Gizlilik için; mesajın asimetrik şifreleme algoritmaları (DES, 3DES, RC2, AES) kullanılarak şifrelenmesi ve şifrelemede kullanılan anahtarın da RSA ile şifrelenmesi.
- Kimlik kanıtlama, bütünlük ve inkar edememe unsurlarının sağlanması için ise:
    - Kullanıcı erişim kontrolünün yapılması (GÜ-Posta uygulaması için),
    - Mesaj özetinin alınması ve göndericinin özel anahtarı ile şifrelenmesi,
    - Alıcı tarafında, deşifre edilen mesajın tekrar özetinin alınarak göndericiden gelen mesaj özeti ile karşılaştırılıp bütünlük kontrolünün yapılması

işlemlerinin yerine getirilmesi tasarlanmıştır.

4.4. GÜ-Posta Uygulaması İle Hedeflenen Kullanıcılar

GÜ-Posta uygulaması tasarlanırken uygulamayı kullanması beklenen kullanıcılar için bir hedef kullanıcı profili de belirlenmiştir. Bu profilin belirlenmesinde değişik yaş, cinsiyet ve eğitim durumundan kişilerle görüşülmüş ve sözlü olarak alınan geribildirimler birer tasarım kriteri olarak ele değerlendirilmiştir.

Hedef kullanıcılar açısından belirlenen tasarım kriterleri aşağıda verilmiştir:

- Uygulamanın kullanıcı ara yüzlerinin yaygın kullanılmakta olan e-posta uygulamaları ile benzerlik göstermesi.
- Kullanıcı ara yüzlerinin kullanışlı olması ve sıradan kullanıcıların güvenlik ile ilgili karmaşık gelen kavramlardan (özel anahtar, genel anahtar vb.) kendisini soyutlayabilmesine imkan tanınması.

- Uygulamanın ücretsiz olması ve ücret ödemeyi gerektirecek nitelikli sertifikalar gibi ücretli ürünler/sistemler kullanmaya ihtiyaç duymayacak şekilde kullanılabilir olması.

Bu kriterlere göre geliştirilen GÜ-Posta uygulaması aşağıda belirtilen şartlara uyan kullanıcılar için uygundur:

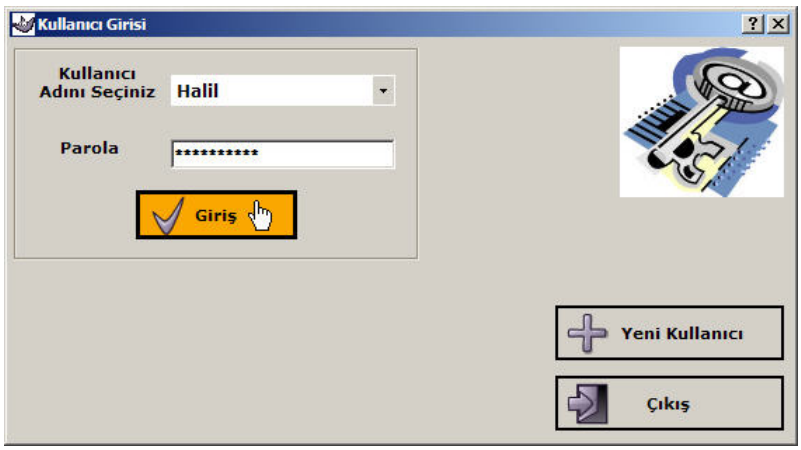
- Öğrenci, akademisyen, ev kullanıcıları gibi güvenlik gereksinimleri çok üst seviye olamayan tüm kullanıcılar rahatlıkla uygulamayı kullanabilirler.
- Az sayıda kullanıcının (2 ila 10 kişi) karşılıklı kullanımına uygundur.
- Kullanıcıların karşılıklı olarak genel anahtarlarını birbirlerine dağıtmaları gerekmektedir.

5. GÜ-POSTA UYGULAMASININ GERÇEKLEŞTİRİMİ

Bu bölümde sunulan GÜ-Posta uygulamasının kullanıcı arayüzleri ve programın genel işleyişi anlatılmaktadır.

5.1. Kullanıcı Girişi

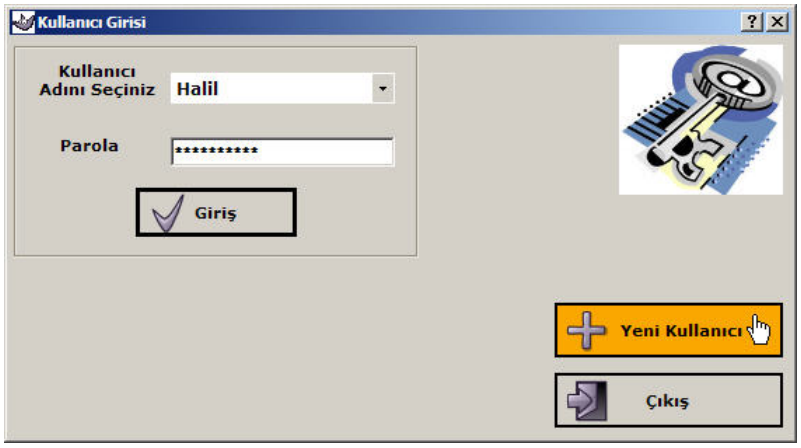
GÜ-Posta uygulamasını çalıştırdıktan sonra ilk olarak ekrana kullanıcı giriş penceresi gelir (Şekil 5.1). Kullanıcı daha önceden kayıt yapmış ise açılır listeden kullanıcı adını seçer ve parolasını girdikten sonra "Giriş" düğmesine tıklar.



Şekil 5.1. GÜ-Posta kullanıcı giriş ekranı

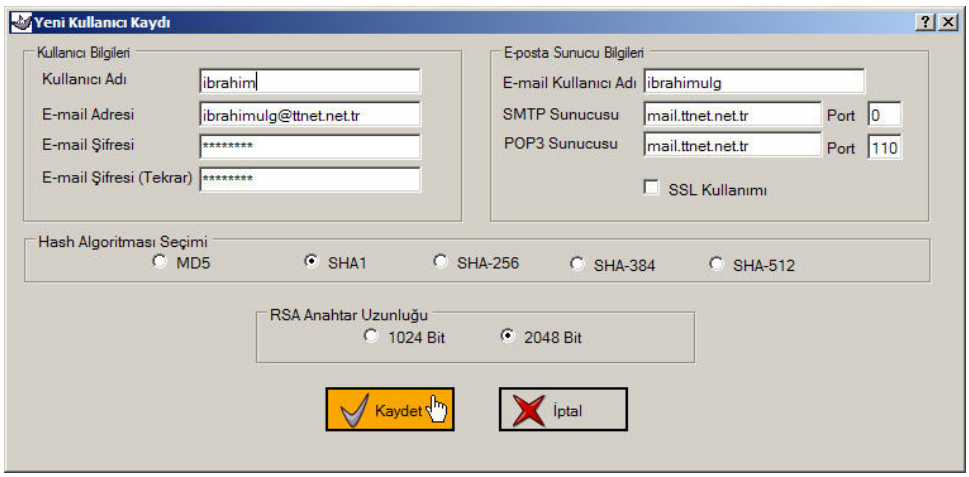
5.2. Yeni Kullanıcı Kaydı

Kullanıcı programı ilk defa kullanıyorsa (sisteme kayıtlı değil ise) "Yeni Kullanıcı" düğmesine tıklar (Şekil 5.2). "Yeni Kullanıcı Kaydı" penceresi açılır (Şekil 5.3).



Şekil 5.2. GÜ-Posta yeni kullanıcı girişi

Açılan pencerede (Şekil 5.3) ilgili alanları doldurduktan sonra "Kaydet" düğmesine tıklayarak kayıt işlemini bitirir. Burada sunucu adı ve port numaraları girilmeden önce üyesi / müşterisi olduğunuz e-posta sağlayıcısından gerekli bilgileri öğrenmeniz gerekmektedir. Eksik ya da hatalı bilgilerin kullanılması mesaj gönderme ve alma işlemlerinde başarısızlığa yol açar.



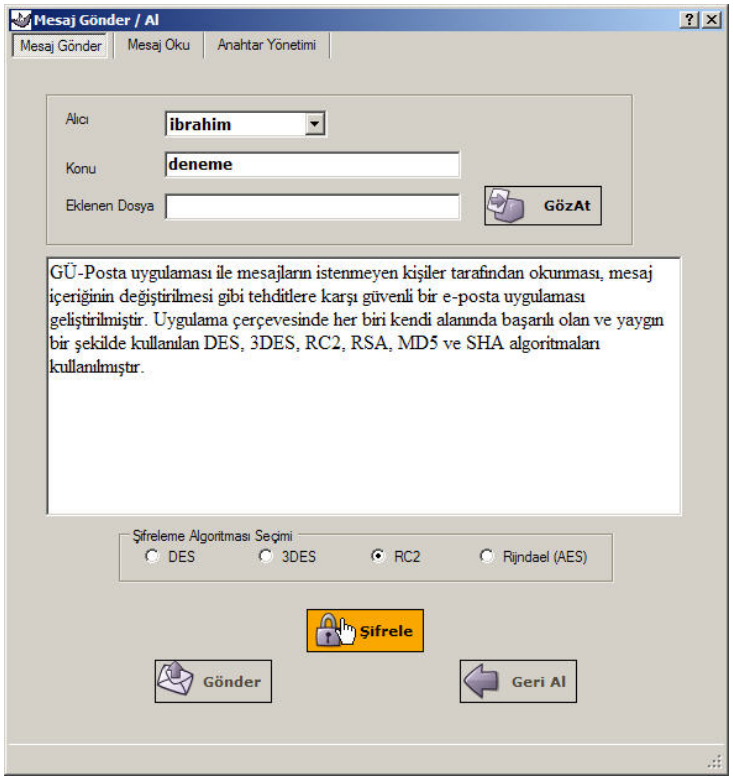
Şekil 5.3. GÜ-Posta yeni kullanıcı kaydı

Kaydet düğmesine tıkladığınızda bilgileriniz "Users.xml" adlı dosyaya kaydedilir ve anahtar çiftleri oluşturulur.

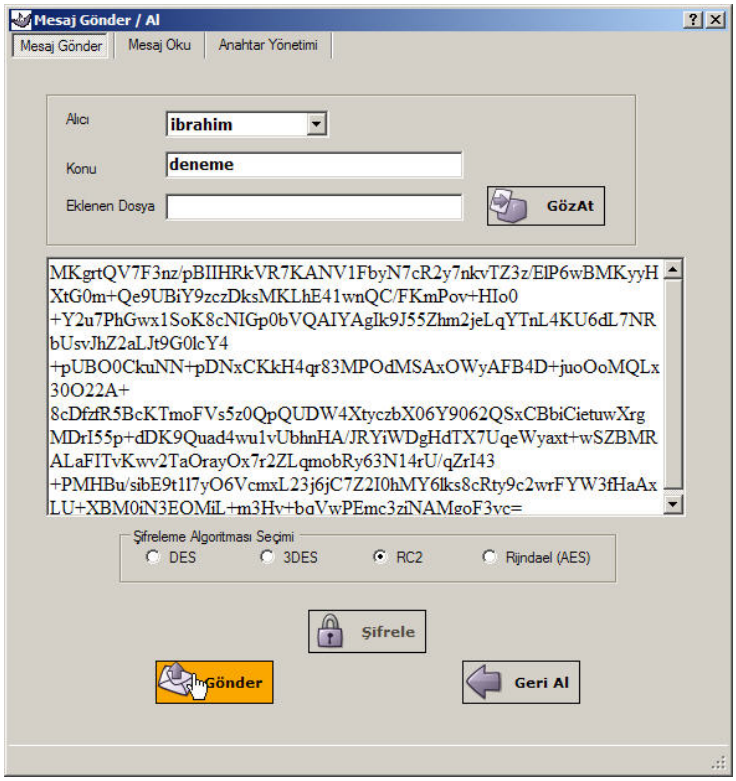
5.3. Mesaj Gönderme İşlemleri

Kullanıcı Girişi ekranında kullanıcı adı seçildikten ve parola girildikten sonra (eğer yetkilendirme işlemi başarılı olursa) Şekil 5.4'deki pencere ekrana gelir.
Mesaj Gönder ekranında yapılacak işlemler sırası ile:

- Alıcı seçilmesi: Mesajın alıcısı seçilir. Alıcının programda kayıtlı kullanıcı olması gereklidir.
- Konu: Gönderilecek olan mesajın konusu yazılır.
- Eklenen Dosya ve Göz at: Mesajla birlikte dosya da göndermek isterseniz Göz at düğmesine tıklayın. Açılacak olan iletişim kutusunu kullanarak dosya seçimini yapabilirsiniz.
- Mesaj Alanı: Göndermek istenilen mesaj yazılır.
- Şifreleme Algoritması Seçimi: Şifreleme işleminde kullanılacak olan algoritma seçimi buradan yapılır.
- Şifrele: İlgili tüm bilgileri girdikten sonra Şifrele düğmesine tıklayarak mesajı şifreleyebilirsiniz (Şekil 5.5). Burada mesajın, alıcının genel anahtarının kullanılarak şifrelendiğini hatırlatalım.
- Geri Al: Şifreleme işleminden sonra mesaj gönderilmeden tekrar düz metne dönülmek istenildiğinde bu düğmeye tıklanılır.
- Gönder: Mesaj şifrelendikten sonra aktif olan Gönder düğmesine tıklanmasıyla mesaj alıcıya gönderilir (Şekil 5.5).



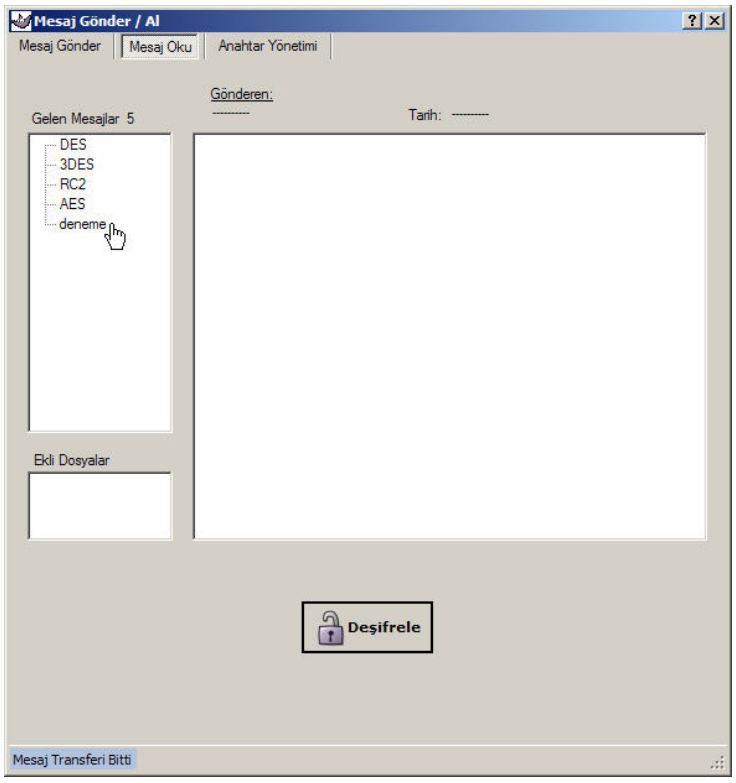
Şekil 5.4. GÜ-Posta mesaj gönderme ekranı - şifreleme işlemi



Şekil 5.5. GÜ-Posta mesaj gönderme ekranı - mesaj gönderme işlemi

5.4. Mesaj Alma / Okuma İşlemleri

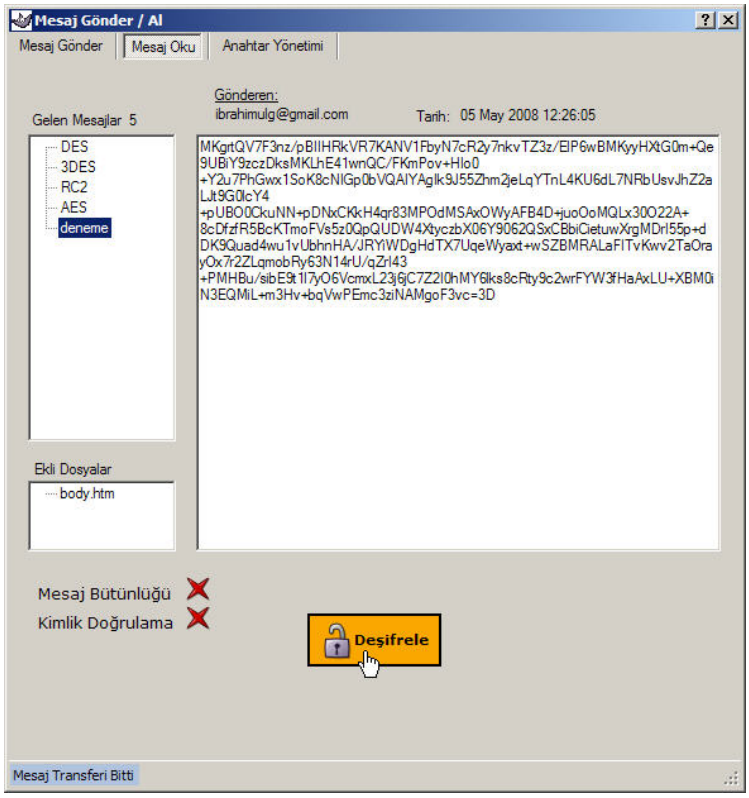
Kullanıcı e-posta hesabındaki şifreli mesajlara erişmek ve bu mesajları deşifre etmek istediğinde GÜ-Posta uygulamasını çalıştırır. Kullanıcı Girişi ekranında kullanıcı adı seçildikten ve parola girildikten sonra açılan pencerenin (Şekil 5.6) üst kısmında bulunan düğmelerden Mesaj Oku düğmesine tıklayarak mesaj okuma ekranına erişilmiş olunur.



Şekil 5.6. GÜ-Posta mesaj okuma ekranı

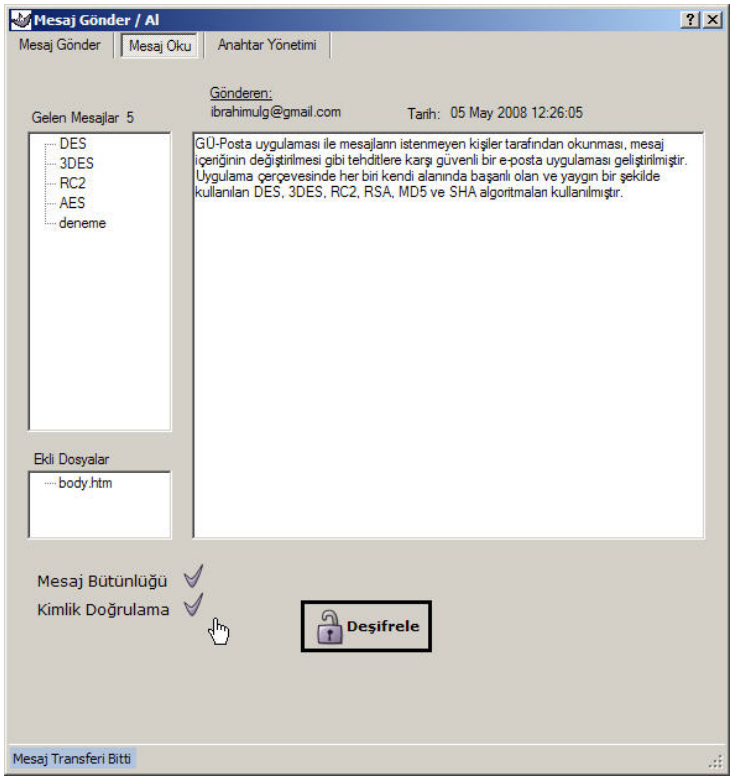
Mesaj transferi bittikten sonra ekranın sol alt köşesinde bilgi mesajı görüntülenir.
Mesaj oku ekranında yapılan işlemler aşağıdaki gibidir:

- Gelen Mesajlar: Posta kutusunda bulunan mesajları bu alanda görebilirsiniz.
- Ekli Dosyalar: Mesajla birlikte ekli dosyalar da varsa bu alanda listelenir.
- Mesaj: Okumak istenilen mesaj Gelen Mesajlar alanındaki listeden seçildikten sonra bu alanda görüntülenir.
- Gönderen: Seçilen mesajı gönderenin e-posta adresi bu alanda gösterilir.
- Tarih: Seçilen mesajın gönderilme tarihi bu alanda gösterilir.



Şekil 5.7. GÜ-Posta şifreli mesajın deşifre edilmesi

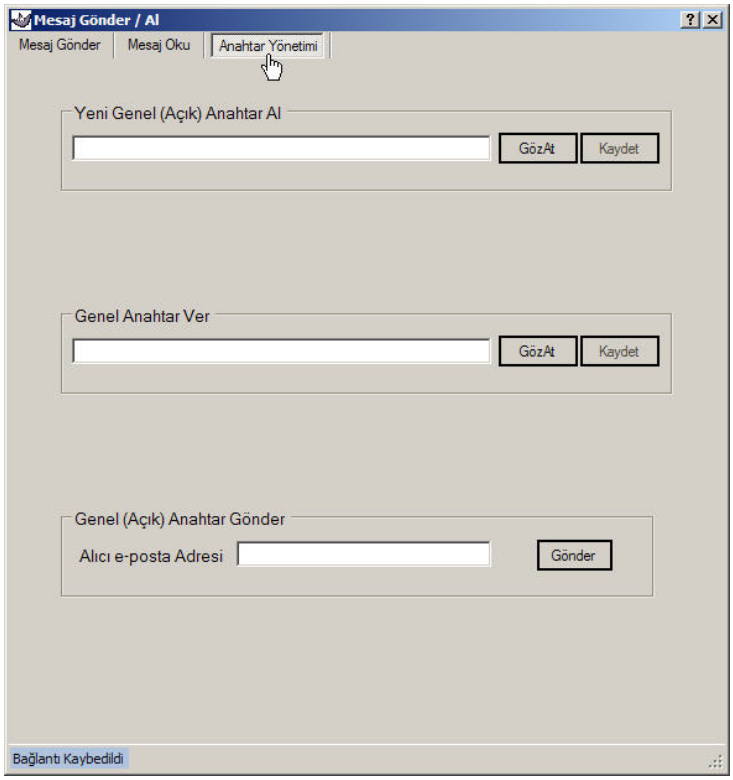
Burada örneğimizde kullandığımız mesaj şifreli bir metin olduğu için deşifre etmemiz gerekiyor. Bu işlemi "Deşifrele" düğmesine tıklayarak (Şekil 5.7) kolay bir şekilde gerçekleştirebiliriz. Deşifreleme işleminde giriş yaptığımız kullanıcının özel anahtarının kullanıldığını belirtelim. Daha sonra deşifreleme işlemi sonrası elde edilen düz metin ile mesaj bütünlüğü ve kimlik doğrulamaya yönelik onay bilgileri kullanıcıya sunulur (Şekil 5.8).



Şekil 5.8. GÜ-Posta deşifre edilen mesajın görüntülenmesi

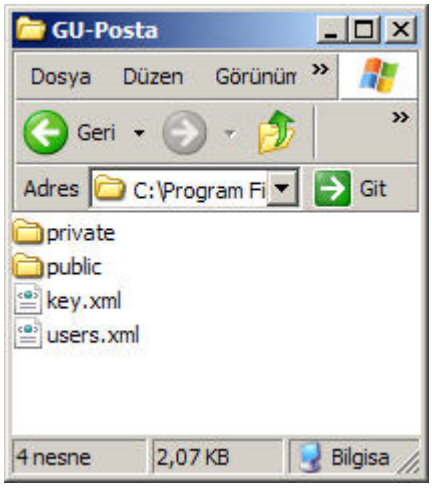
5.5. Anahtar Yönetimi ve Kullanıcı Bilgilerinin Saklanması

"Anahtar Yönetimi" sekmesinde ise anahtar yönetimi ile ilgili işlemler yapılabilmektedir. Şekil 5.9'da verildiği gibi kullanıcı başka bir kullanıcının genel anahtarını herhangi bir konumdan (disket, optik medya, flash disk vb.) alarak sisteme ekleyebilir. Yine aynı şekilde kendi genel anahtarını sistemden başka bir ortama gönderebilir. Son olarak kullanıcı isterse kendi genel anahtarını başka bir kullanıcıya e-posta ile gönderebilir.



Şekil 5.9. GÜ-Posta anahtar yönetimi ekranı

GÜ-Posta uygulaması için gerekli olan tüm dosyalar (konfigürasyon ve kullanıcı bilgileri ile genel ve özel anahtarların yer aldığı dosyalar) uygulamanın kurulu olduğu dizinde saklanmaktadır (Şekil 5.10).



Şekil 5.10. GÜ-Posta konfigürasyon ve kullanıcı bilgileri dizini

Getirdiği kolaylıklar açısından dosyalar XML tabanlı olarak oluşturulmuştur. Örneğin kullanıcı bilgilerinin yer aldığı XML belgesinin içeriği Şekil 5.11'deki gibidir.

```
<?xml version="1.0" ?>
- <Users>
  - <User Name="Halil">
      <EmailAdresi>ibrahimxxg@gmail.com</EmailAdresi>
      <sifre>ffe97822f5b262f1017e99900f71bf1f2e3263d9</sifre>
      <eKul_adi>ibrahimxxg@gmail.com</eKul_adi>
      <smtp>smtp.gmail.com</smtp>
      <smtp_port>587</smtp_port>
      <pop3>pop.gmail.com</pop3>
      <pop3_port>110</pop3_port>
      <SSL_kullanimi>1</SSL_kullanimi>
      <HashAlgorithm>SHA1</HashAlgorithm>
      <KeySize>2048</KeySize>
    </User>
  - <User Name="ibrahim">
      <EmailAdresi>ibrahimxxg@ttmail.com</EmailAdresi>
      <sifre>ce346e47011939fb9671fdf438d1c66f9791ea68</sifre>
      <eKul_adi>ibrahimxxg@ttmail.com</eKul_adi>
      <smtp>smtp.ttmail.com</smtp>
      <smtp_port>0</smtp_port>
      <pop3>pop3.ttmail.com</pop3>
      <pop3_port>110</pop3_port>
      <SSL_kullanimi>0</SSL_kullanimi>
      <HashAlgorithm>SHA1</HashAlgorithm>
      <KeySize>2048</KeySize>
    </User>
  </Users>
```

Şekil 5.11. GÜ-Posta Users.xml belgesinin içeriği

Bu XML dosyası incelendiğinde GÜ-Posta uygulamasında kullanılan kullanıcı bilgilerinin burada tutulduğu görülebilir. Böylece kullanıcının bir kez kayıt olduktan sonra çeşitli bilgileri tekrar tekrar girmesine gerek kalmamaktadır. Ayrıca kullanıcıların parolaları açık metin olarak saklanmamakta bunun yerine parolaların SHA1 algoritmasına göre alınmış özetleri saklanmaktadır. Bunun başlıca sebebi gizliliktir. Böylece kullanıcı parolaları XML belgesinin ikinci bir kişinin eline geçme ihtimaline karşı güvenlik altına alınmış olur.

Programda kullanılan anahtarlar (özel ve açık anahtar çiftleri) ise yine programın kurulu olduğu dizinde public ve private adlı klasörlerde saklanmaktadır (Şekil 5.10). Bu durum güvenlik zafiyetine sebep olabilir gibi görünse de yapılan çalışma sonucu geliştirilen uygulamanın deneysel bir ürün olduğu unutulmamalıdır. Uygulama, anahtarlar başka bir ortamda (flashdisk, smartcard vb.) saklanabilecek şekilde kolaylıkla yeniden düzenlenebilir.

5.6. Uygulama Kullanıcı Deneyimi

Geliştirme aşamasının bitişiyle GÜ-Posta uygulaması kullanıcıların deneyimine sunulmuştur. Bölüm 4.4'de belirtilen hedef kullanıcı profiline uyan kullanıcılar programı denemişlerdir. Denek kullanıcılar ile ilgili bilgiler ve uygulamayı kullandıktan sonra verdikleri yorumlar Çizelge 5.1'de verilmiştir:

Çizelge 5.1. Denek kullanıcı bilgileri ve kullanıcı deneyimleri

	Cinsiyeti	Yaşı	Mesleği	Kullanıcı Deneyimi
Kullanıcı 1	Kadın	21	Öğrenci	Şifreleme konusuna uzun bir süredir ilgi duyuyordum. Bu program sayesinde şifreleme, özel ve genel anahtar, bütünlük, inkar edememezlik gibi kavramlar konusunda daha fazla bilgi sahibi oldum.
Kullanıcı 2	Erkek	27	Mühendis	Uygulama fonksiyonlarını yerine getirebiliyor. Yalnız kullanıcı ara yüzünde ve genel sistem olarak iyileştirmeler yapılabilir.
Kullanıcı 3	Erkek	52	Öğretmen	Program çok kullanışlı. Bilgisayar kullanmayı yeni öğrenmeme rağmen sıkıntı yaşamadım.
Kullanıcı 4	Kadın	15	Öğrenci	Şifreli mesajlar göndermek çok eğlenceli, en yakın arkadaşımla karşılıklı e-posta alışverişi yapabiliyoruz.

Verilen yorumlardan geliştirilen uygulamanın gündelik hayatta kullanılabilir olduğu görülmüştür. Yorumların verdiği bilgiler ışığında uygulamanın ilerde eklenecek yeni özelliklerle daha da geliştirilmesi düşünülmektedir.

5.7. Test Sonuçları

Uygulama geliştirme aşamasında ve sonrasında sürekli olarak test edilmiş ve fonksiyonel olarak doğru çalıştığı garanti altına alınmıştır. Kullanıcı testlerinde ise bazı problemlerle karşılaşılmıştır. Bu problemler aşağıda verilmiştir:

- SMTP sunucusuna bağlanmak için kullanılan 25 numaralı portun kapalı olması durumunda (bazı kurumlarda güvenlik nedeniyle firewall tarafından bu port kapatılmakta) mesaj gönderme işlemi yapılamamaktadır.
- Kullanıcı ara yüzünde veri girişi yapılan alanlarda verilerin istenilen nitelikte olduğunun kontrolünün yapılması gerektiği testler sonucu ortaya çıkmıştır. Örneğin e-posta adresinin uygun biçimde girilip girilmediğinin kontrolü gibi.

6. SONUÇ VE ÖNERİLER

Bu tez çalışmasında güvenlik gereksinimleri ve mevcut uygulamalar incelendikten sonra güvenli bir e-posta uygulaması geliştirilmiştir.

Geliştirilmiş olan GÜ-Posta uygulamasında; düz mesaj metni bir şifreleme algoritması ile şifrelenir, daha sonra şifrelemede kullanılan anahtar da RSA algoritması ile (alıcının genel anahtarı kullanılarak) şifrelenir ve oluşturulan e-posta gönderilir. Alıcı e-postayı aldığında ise, öncelikle şifreli olan anahtarı RSA algoritmasını (kendi özel anahtarı ile birlikte) kullanarak deşifre eder. Elde edilen anahtar ise mesajın deşifre edilmesinde kullanılır.

GÜ-Posta' nın genel özellikleri aşağıda verilmiştir:

- Gizlilik (şifreleme) için DES (56 bit), 3DES, RC2 (128 bit), AES (256 bit) ve RSA kullanımı.
- İmzalama işlemleri için RSA kullanımı, 1024 ve 2048 bitlik anahtar uzunluğunun seçilebilmesi
- Mesaj özetleri almak ve bütünlük kontrolü için SHA1 kullanımı
- Kullanıcı erişim kontrolü işlemleri için MD5, SHA1, SHA-256, SHA-384 ve SHA-512 kullanımı
- Kullanıcıların özel ve genel anahtarlarının XML formatında saklanması: Özel anahtar, programın kurulu olduğu dizinde (C:\Program Files\FoxSoft\GU-Posta) "private" adlı bir klasörde "Kullanıci_adi.pr" adlı dosyalarda saklanır. Özel anahtar mesajların şifrelenmesinde, deşifrelenmesinde veya dijital olarak imzalanmasında kullanılır. Genel anahtar, programın kurulu olduğu dizinde (C:\Program Files\FoxSoft\GU-Posta) "public" adlı bir klasörde "Kullanıci_adi.pb" dosyalarda saklanır. Gönderilen mesajın şifrelenmesi, alınan mesajın bütünlüğünün kontrol edilmesi ve kimlik kanıtlama amaçlı olarak kullanılır.
- Mesajları şifrelerken ve deşifre ederken özel ve genel anahtarları e-posta mesajının "Gönderici" ve "Alıcı" alanlarını kullanarak otomatik olarak

seçebilmesi: Kullanıcının özel ve genel anahtar gibi kavramlardan kendisini soyutlayabilmesini sağlamaktadır.

- Uygulamanın MS Outlook'tan bağımsız olarak kullanılabilmesi
- Genel anahtar dağıtımının kolaylıkla yapılması
- Güvenlik için üçüncü bir otoriteye ihtiyaç duyulmaması

şeklinde özetlenebilir.

Geliştirilen uygulama bir önceki sürümüne göre farklılıklar içermektedir [29]. Şifreleme işlemlerinde kullanılan algoritmaların sayısı arttırılmış, kimlik kanıtlama ve inkar edememezlik özellikleri eklenmiştir.

E-posta güvenliği açısından günümüzde en çok kullanılan yöntem ve araçlar S/MIME ve PGP'dir. S/MIME ile GÜ-Posta uygulamasının karşılaştırılması Çizelge 3.1'de, PGP ile GÜ-Posta uygulamasının karşılaştırması ise Çizelge 3.3'de verilmiştir. Yapılan karşılaştırmalarda da görüleceği üzere geliştirilen uygulama e-posta iletişimi ile ilgili güvenlik ihtiyaçlarını karşılamaktadır. Uygulama ücretsiz oluşu ve temel ihtiyaçları karşılanması ile kişisel mahremiyetin sağlanmasında önemli rol oynayacaktır.

Gelecekte yapılması düşünülen çalışmalar ile GÜ-Posta uygulaması daha da geliştirilecektir. Yapılması düşünülen çalışmalar özetle aşağıda anlatılmıştır:

- Kimlik doğrulama ve gizlilik için akıllı kart (Smart card) kullanımı düşünülmektedir. Akıllı kartların kullanımının gizliliğe katkısı özel anahtarların kart üzerinde saklanması ve dış dünyadan daha izole hale gelmesidir. Akıllı kart kullanımı ile kullanıcı uygulama yazılımlarının yüklü olduğu herhangi bir bilgisayarda güvenli e-posta iletişimini yapabilir hale gelecektir.
- Bir web servisi ile sertifika otoritesi benzeri hizmetler verilmesi ve mevcut uygulamanın bu servis ile entegrasyonu düşünülmektedir.

- Kullanıcı bilgilerinin saklanmasında “izole saklama/depolama” (isolated storage) olarak adlandırılan yapının oluşturulması düşünülmektedir. Bir uygulama verileri bir dosya içerisinde sakladığında dosya ismi ve saklanacak dosyanın konumu başka uygulamaların veya kişilerin erişimini engelleyebilmek için dikkatli bir şekilde seçilmelidir. Depolama alanının izole edilmesi ile veri kişilerden ve diğer uygulamalardan izole edilmiş ve dolayısıyla korunmuş olur. Uygulamaya izole depolama özelliğinin kazandırılması için .NET framework’ün sunduğu “System.IO.IsolatedStorage” adlı isim uzayının kullanılması düşünülmektedir. Bu yapı özel ve genel anahtarların saklanmasında da kullanılabilir.

KAYNAKLAR

1. The Radicati Group Inc, "Market Numbers Summary Update Q3 2007", ***Radicati***, United States, 23-24 (2007).
2. The Radicati Group Inc, "Email Archiving Market 2008-2012", ***Radicati***, United States, 19 (2008).
3. The Radicati Group Inc, "Messaging & Collaboration - Business User Survey 2007", ***Radicati***, United States, 20 (2007).
4. Schneier, B., "Cryptographic Design Vulnerabilities", ***IEEE Computer***, 31(9): 29-33 (1998).
5. Bahadur, G., Chan, W., Weber, C., "Privacy Defended: Protecting Yourself Online", ***Que***, 275 (2002).
6. BT Group, "Security Report Online Identity Theft", ***BT***, United States, 4-27 (2006).
7. Levi, A., "Nasıl bir E-posta güvenliği?", ***Bilişim Güvenlik***, 38-40 (2003).
8. Flynn, N., Kahn, R., "E-Mail Rules A Business Guide to Managing Policies, Security, and Legal Issues, for E-Mail and Digital Communications", ***Amacom***, 27-28 (2003).
9. Weisband, S.P., Reinig, B.A., "Managing User Perceptions of Email Privacy", ***Communications of the ACM***, 38(12): 41-47 (1995).
10. Agarwal, R., Rodhain, F., "Mine or Ours: Email Privacy Expectations, Employee Attitudes, and Perceived Work Environment Characteristics", ***Proceedings of the 35th Annual Hawaii International Conference on***, United States, 7(10): 2471 - 2480 (2002).
11. The Radicati Group Inc, "European E-mail Security Market, 2006-2010", ***Radicati***, United States, 18 (2006).
12. The Radicati Group Inc, "A Technology Market Research Firm, Email Security Market in Asia/Pacific, 2008-2012", ***Radicati***, United States, 14 (2006).
13. The Radicati Group Inc, "Email Security Appliances Market Quadrant - 2007", ***Radicati***, United States, 4 (2007).
14. Kapadia, A., "A Case (Study) For Usability in Secure Email Communication", ***IEEE Computer Society, IEEE Security & Privacy,*** 80-84 (2007).
15. Perlman, R., "An Overview of PKI Trust Models", ***IEEE Network Magazine***, 13(6): 38-43 (1999).

16. Sağıroğlu, Ş., Alkan, M., "Her Yönüyle Elektronik İmza", ***Grafiker Yayınları***, Ankara, 2 (2005).

17. TÜBİTAK UEKAE, "Açık Anahtar Altyapısı Eğitim Kitabı", http://www.kamusm.gov.tr/tr/Bilgideposu/Belgeler/teknik/aaa/ (2006).

18. Schneier, B., "Applied Cryptography, Second Edition: Protocols, Algorithms, and Source Code in C", ***John Wiley & Sons, Inc***, United States, 265-270 (1996).

19. Burwick, C., Coppersmith, D., D'Avignon, E., Gennaro, R., Halevi, S., Jutla, C., Matyas, Jr S. M., O'Connor, L., Peyravian, M., Safford, D., Zunic, N., "MARS - A candidate cipher for AES", ***NIST AES Candidates***, 1-63 (1999).

20. Rivest, R.L., Robshaw, M.J.B., Sidney, R., Yin, Y.L., "The RC6 Block Cipher", ***NIST AES Candidates***, 1-21 (1998).

21. Daemen, J., Rijmen, V., "AES Proposal – Rijndael", ***NIST AES Candidates,*** 1-45 (1999).

22. Anderson, R., Biham, E., Knudsen, L., "Serpent- A Proposal for the Advanced Encryption Standard", ***NIST AES Candidates***, 1-23 (1999).

23. Schneier, B., Kelsey, J., Whiting, D., Wagner, D., "Twofish: A 128-Bit Block Cipher", Hall C., Ferguson, N., ***NIST AES Candidates***, 1-68 (1998).

24. Jamil, T., "The Rijndael Algorithm", ***Potentials, IEEE***, 23 (2): 36-38 (2004).

25. Yerlikaya, T., Buluş, E., Arda, D., "AES Aday Şifreleme Algoritmalarının Yazılım Ve Donanım Performans Karşılaştırılması Ve Uygulamaları", ***Elektrik Elektronik Bilgisayar Mühendisliği Sempozyumu (ELECO 2004)***, Bursa, 1-5 (2004).

26. Stallings, W., "Cryptography and Network Security Principles and Practices", Fourth Edition, ***Prentice Hall***, United States of America, 140 (2005).

27. Boneh, D., Franklin, M., "Identity based encryption from the Weil pairing", ***Crypto 2001***, USA, 213-229 (2001).

28. Sarısakal, M.N. , Savaşan, V., Sertbaş, A., "RSA ve IDEA Algoritmalarını birlikte kullanan güvenli bir e-posta uygulaması: VMAIL", ***Istanbul University Engineering Faculty, Journal of Electrical & Electronics***, 297-305 (2001).

29. Ülgen, H. İ., Sağıroğlu, Ş., Yüncü, S., "Güvenli bir e-posta uygulaması: GÜ-Posta", ***Ulusal Elektronik İmza Sempozyumu***, Ankara, 142-148 (2006).

ÖZGEÇMİŞ

Kişisel Bilgiler

Soyadı, adı	: ÜLGEN, Halil İbrahim
Uyruğu	: T.C.
Doğum tarihi ve yeri	: 12.05.1981, Karabük
Medeni hali	: Bekar
Telefon	: 0 (312) 430 62 30
Faks	: -
e-mail	: ibrahimulg@hotmail.com

Eğitim

Derece	Eğitim Birimi	Mezuniyet tarihi
Lisans	Gazi Ünv. / Elektrik-Elektronik Müh. Bölümü	2005
Lise	S. Demirel Fen Lisesi	1999

İş Deneyimi

Yıl	Yer	Görev
2007-	TAI-TUSAŞ Türk Havacılık ve Uzay Sanayi A.Ş.	Yazılım Müh.
2005-2007	Gazi Üniversitesi	Araş. Gör.

Yabancı Dil

İngilizce

Yayınlar

1. Ülgen H. İ., Sağıroğlu Ş., Yüncü S., "Güvenli Bir E-Posta Uygulaması:GÜ-Posta", Ulusal Elektronik İmza Sempozyumu, 2006.

Hobiler

Bilgisayar teknolojileri, Futbol.