KOCAELİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI

YÜKSEK LİSANS TEZİ

AÇIK ANAHTAR ALTYAPISI VE ELEKTRONİK İMZANIN MOBİL TABANLI UYGULAMASI

MERVE SAĞIR

KOCAELİ 2014

KOCAELİ ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI

YÜKSEK LİSANS TEZİ

AÇIK ANAHTAR ALTYAPISI VE ELEKTRONİK İMZANIN MOBİL TABANLI UYGULAMASI

MERVE SAĞIR

Prof.Dr. Yaşar BECERİKLİ
Danışman, Kocaeli Üniv. .....

Prof.Dr. Hasan OCAK
Jüri Üyesi, Kocaeli Üniv. .....

Prof.Dr. A. Coşkun SÖNMEZ
Jüri Üyesi, İTÜ .....

Tezin Savunulduğu Tarih: 15.07.2014

ÖNSÖZ ve TEŞEKKÜR

Kurumlarda önemli hale gelen gizlilik, güvenlik, kimlik doğrulama ve inkar edilememezlik sorunları elektronik imza uygulamaları ile çözümlenmiştir. Geliştirilen elektronik imzalama uygulaması, elektronik imzalama işlemini PFX dosyası formatıyla Android üzerinde gerçekleştirmektedir. TÜBİTAK BİLGEM tarafından geliştirilmiş Milli ESYA Kütüphanesi kullanılarak gerçekleştirilen Android üzerinde elektronik imzalama yazılımı sektörün bu yöndeki ihtiyaçları için örnek olabilir niteliktedir.

Çalışmalarım boyunca beni teşvik eden, yönlendiren ve bilgilerinden faydalandığım danışmanım Prof. Dr. Yaşar BECERİKLİ'ye ve TÜBİTAK BİLGEM yöneticilerine teşekkür ederim. Manevi desteklerini ömrüm boyunca hissettiğim aileme sonsuz teşekkürlerimi sunarım.

Haziran - 2014 Merve SAĞIR

İÇİNDEKİLER

ŞEKİLLER DİZİNİ

TABLOLAR DİZİNİ

SİMGELER DİZİNİ VE KISALTMALAR

φ : Phi fonksiyonu

Kısaltmalar

AES : Advanced Encryption Standard (İleri Şifreleme Standardı)
API : Application Programming Interface (Uygulama Programlama Arayüzü)
BİLGEM : Bilişim ve Bilgi Güvenliği İleri Araştırmalar Merkezi
CMS : Cryptographic Message Syntax (Kriptografik Mesaj Yazımı)
DES : Data Encyption Standart (Veri Şifreleme Standardı)
EBOB : En Büyük Ortak Bölen
EC : Electronic Commerce (Elektronik Ticaret)
ESHS : Elektronik Sertifika Hizmet Sağlayıcı
ESYA : Elektronik Sertifika Yönetim Altyapısı
ETSI : European Telecommunications Standards Institute (Avrupa Telekomünikasyon Standartları Enstitüsü)
IEEE : Instute of Electrical and Electronics Engineers (Elektrik ve Elektronik Mühendisleri Enstitüsü)
IETF : Internet Engineering Task Force (İnternet Mühendislik Özel Görev Kuvveti)
IPSEC : Internet Protocol Security (İnternet Protokol Güvenliği)
ISO : International Organization for Standardization (Uluslararası Standartlar Teşkilatı)
ITU : International Telecommunication Union (Uluslararası Telekomünikasyon Birliği)
KamuSM : Kamu Sertifikasyon Merkezi
MD5 : Message Digest Algorithm 5 (Mesaj Özetleme Algoritması 5)
NIST : National Institute of Standart and Technology (ABD Ulusal Standart ve Teknoloji Enstitüsü)
OCSP : Online Certificate Status Protocol (Çevrimiçi Seritifika Kontrol Protokolü)
PKI : Public Key Infrastructure (Açık Anahtar Altyapısı)
RC : Rivest' s Cipher (Rivest Şifreleme)
PFX : Personal Information Exchange (Kişisel Bilgi Dosyası)
RSA : Rivest-Shamir-Adleman
SHA : Security Hash Algorithm (Güvenli Özetleme Algoritması)
SİL : Sertifika İptal Listesi
SSL : Secure Socket Layer (Güvenli Giriş Katmanı)
TLS : Transport Layer Security (Ulaşım Düzeyi Güvenliği)
UEKAE : Ulusal Elektronik ve Kriptoloji Araştırma Enstitüsü

AÇIK ANAHTAR ALTYAPISI VE ELEKTRONİK İMZANIN MOBİL TABANLI UYGULAMASI

ÖZET

Günümüzde elektronik imza, uygulamalara login olmaktan, doküman yönetim sistemlerine kadar her alanda kullanılmaktadır. Elektronik imza uygulamalarının yaygınlaşmasıyla ıslak imza gerektiren belgelerin elektronik olarak imzalanarak fazla kağıt tüketiminin önüne geçilmiştir. Kurumlar için de ciddi maliyet kazancı sağlanmıştır. Kurumlarda işleyen yapıda önemli hale gelen gizlilik, güvenlik, kimlik doğrulama ve inkar edilememezlik sorunları elektronik imza uygulamaları ile çözümlenmiştir.

Mobil iletişimin giderek yaygınlaşmasıyla mobil cihaz ve cep telefonlarında kullanılan Android işletim sistemi günlük hayatımızın bir parçası haline gelmiştir. Android işletim sisteminin açık kaynak kodlu olması kullanım esnekliği sağlamaktadır. Geliştiriciler ihtiyaçlara göre Android işletim sistemi üzerinde uygulama geliştirerek kullanıma sunmaktadır.

Yapılan elektronik imzalama uygulamasının katkısı ve yenilikçi yönü; Windows işletim sistemi üzerinde çalışan İmzager, İmzala-Gönder ve PDF İmzalama uygulamalarının yaptığı işlemlerden elektronik imzalama işleminin PFX dosyası formatıyla Android üzerinde gerçekleştirilmesidir. TÜBİTAK BİLGEM tarafından geliştirilmiş Milli ESYA Kütüphanesi kullanılarak gerçekleştirilen Android üzerinde elektronik imzalama yazılımı sektörün bu yöndeki ihtiyaçları için örnek olabilir niteliktedir. Ülkemizde kullanıma girecek elektronik kimlik kartları ile akıllı kart okuyucular kullanılarak elektronik imzalama işleminin sağlanmasıyla elektronik imza kullanımının artacağı düşünülmektedir.

Anahtar Kelimeler: Açık Anahtar Altyapısı, Android, Elektronik İmza, Elektronik İmza Uygulaması.

PUBLIC KEY INFRASTRUCTURE AND ELECTRONIC SIGNATURE APPLICATION BASED ON MOBILE

ABSTRACT

Nowadays, electronic signature is used in all areas which varies from login into the application to document management systems. Due to becoming electronic signature applications popular, paper consumption is prevented by signing documents as electronic. Also, significant cost reduction is ensured for companies. The problems that become critical in companies about privacy, security, identity authentication and non repudiation are solved by electronic signature applications.

Android operating systems that used mobile devices and phones are part of our daily life with becoming mobile communication popular open source operating system structure of Android offers flexible usage. According to the requirements, developers implement applications and put them into use on the Android operating system.

The contribution and innovation of electronic signature application is to realize the electronic signature process which is part of İmzager, İmzala-Gönder and PDF Signature applications on Windows operating systems, is performed on the Android platform, in PFX format. The electronic signature application on the Android, that used with National ESYA library, which is developed by TUBITAK BİLGEM, can be model for the needs of the industry. It is expected to increase the use of electronic signature with electronic ID cards which will come into use in our country and provide electronic signature usage with smart card readers.

Keywords: Public Key Infrastructure, Android, Electronic Signature, Electronic Signature Application.

GİRİŞ

Günümüzde teknoloji dünyasındaki gelişmeler kurumların işleyişlerini de değiştirmektedir. Kurumlar içindeki işleyiş artık elektronik ortam üzerinden yürütülür hale gelmiştir. Elektronik ortam üzerinden işleyen kurumsal yapının güvenliği gün geçtikçe kritikleşmiştir. Yetkisiz kişilerin elektronik ortamlarda bilgi paylaşılan ağlara sızması, bu ağların zarara uğratılması, ağ üzerindeki bilgilerin değiştirilmesi, çalınması ve üçüncü kişilere ulaştırılması, sürekli ağ hizmetlerinin kesintiye uğratılması, ağ üzerinde gönderilen ya da alınan bilginin kim tarafından gönderildiğinin ya da alındığının bilinememesi gibi problemleri ortaya çıkarmıştır.

Güvenlik problemlerinin önüne geçebilmek için kriptoloji yöntem ve araçlarından yararlanılmaya başlanmıştır. Kriptoloji; matematik, elektronik, optik ve bilgisayar bilimleri gibi birçok disiplini kullanan özelleşmiş bir bilim dalı olarak kabul edilmektedir. Kriptoloji algoritmaları ve uygulamaları; verilerin şifrelenmesi ve güvenlikli olarak alıcıya iletilmesi, alıcı tarafından alınan şifreli verilerin çözülmesi işlevlerini yerine getirir.

Kurumlarda işleyen yapıda önemli hale gelen gizlilik, güvenlik, kimlik doğrulama ve inkar edilememezlik sorunları kriptoloji uygulamaları ile çözümlenmiştir. Kurum içindeki veri akışında kimlerin neleri görebileceği gibi yetkilendirme problemlerinin çözümü, hangi verinin kim tarafından gönderildiği, kim tarafından alınabildiği, hizmetin kesintisiz olarak sağlanabilirliğinin güvence altına alınması ve bilgi akışının güvenli yoldan sağlanması gibi hususlar kriptoloji uygulamaları sayesinde sağlanmaktadır.

Ağlarda veri iletimindeki güvenlik ve güvenilirlik problemlerinin çözümü için ilk kez 1976 yılında Whitfield Diffie ve Martin Hellman tarafından Diffie-Hellman anahtar değişimi algoritması (Diffie ve Hellman, 1976) adıyla ilk açık anahtar algoritması yayınlanmıştır. Ron Rivest, Adi Shamir ve Len Adleman tarafından

Diffie-Hellman anahtar değişimi algoritması temel alınarak 1978 yılında RSA algoritması (Rivest ve diğ., 1978) yayınlanmıştır. Diffie-Hellman anahtar değişimi algoritması ve RSA algoritmasının yayınlanması açık anahtar altyapısı ve elektronik imzanın gelişmesine büyük katkı sağlamıştır.

Özel anahtarlı kripto sistem algoritmalarından DES algoritması (Data Encryption Standard) IBM firması tarafından geliştirilen bir şifreleme algoritmasıdır (Mehuron, 2014). DES algoritması, ABD hükümeti NIST (National Institute of Standarts and Technology) tarafından 1977 yılında standart olarak belirlenmiştir. DES algoritması, bugüne kadar en çok kullanılan özel anahtarlı kripto sistemlerden biri olmuştur. Ayrıca 1991 yılında elektronik imza konusunda ilk uluslararası standart olan ISO/IEC9796 standardı oluşturulmuştur. Bu standart en son 2010 yılında gözden geçirilmiştir (URL-1).

Elektronik imza, haberleşmede alınan mesajı oluşturan kişinin kanuni olarak tespitini sağlayan mekanizmadır. Alıcı tarafından alınan mesajın değiştirilmemiş olduğundan emin olunması sağlanır. Islak imza ile gelen bir dokümanı oluşturan kişinin belli olması gibi elektronik imzalı dokümanı oluşturan kişi de bellidir (Sağır ve Becerikli, 2013). Elektronik imzalama işlemi ile dokümanın kimden geldiği belli olduğu için haberleşmenin temel ilkelerinden olan inkar edilememezlik, kimlik doğrulama ve bütünlük işlevleri yerine getirilir.

Dünyada 1996 yılında, ülkemizde 2004 yılında hazırlanan mevzuatlarla hukuki altyapısı belirlenmeye başlanan elektronik imza, halihazırda birçok ülkede yasal olarak uygulanmaya başlamıştır. E-imza, Birleşmiş Milletler Uluslararası Ticaret Hukuku Komisyonu tarafından, 1996 yılında Elektronik Ticaret Model Yasası'nın ve 2001 yılında Elektronik İmza Model Yasası'nın çıkarılmasıyla, dünya ülkelerince gerekli hukuki düzenlemeler yapılarak uygulamaya geçirilmeye başlanmıştır. Avrupa Birliği, elektronik imzanın kullanılmasını kolaylaştırmak ve hukuken tanınmasına katkıda bulunmak amacıyla 13 Aralık 1999 tarihli ve 99/93/EC sayılı Elektronik İmza Direktifi'ni yayınlamıştır. Direktif; elektronik imza sertifikaları, sertifika hizmet sağlayıcıları ve bunların denetimi ile ilgili esasları belirlemektedir. Bilişim toplumu hizmetlerinin üye ülkeler arasında serbest dolaşımını sağlamak amacıyla hazırlanan 8 Haziran 2000 tarihli 2000/31/EC sayılı Elektronik Ticaret Direktifi ile

de elektronik sözleşmeler ve bunların hukuki neticelerine ilişkin önemli hususlar belirlenmiştir (Karakoçak ve diğ., 2005).

Dünyada elektronik imzanın yaygınlaşması için çalışmalar devam etmektedir. Ek-A'da verilen çeşitli ülkelerde yapılan çalışmalar ve yılları tablo olarak görülmektedir. Tabloda ayrıca dünyada elektronik imza yasalarının uygulama yılları verilmiştir (Sağıroğlu ve Alkan, 2005).

Türkiye'de Başbakanlık tarafından görevlendirilmiş TÜBİTAK Kamu Sertifikasyon Merkezi tarafından kamu ve özel sektöre açık anahtar altyapısı hizmeti verilmektedir. 5070 sayılı elektronik imza kanununa göre nitelikli elektronik sertifika dağıtımı yalnızca kamu kurumlarına yapılabilmektedir. 2013 yılı içinde Adalet Bakanlığı, Sosyal Güvenlik Kurumu, İçişleri Bakanlığı gibi birçok kamu kurumuna 171.037 adet nitelikli elektronik sertifika dağıtılmıştır (Tübitak Bilgem Kamu SM, 2013).

Estonya, elektronik kimlik kartının kullanıldığı ilk ülkedir ve kamu hizmetlerinin neredeyse tamamı bu kart ile verilmektedir. Estonya'da 2005 yerel seçimleri ile internet üzerinden seçim yapılan ilk ülke olmuştur (Boyacı ve diğ., 2012).

Elektronik seçimin Estonya'da 2005, 2007 ve 2009 yılında uygulanması Norveç hükümeti için ilham kaynağı olmuş bu başarıyı kendi ülkelerinde de kullanmak istemişlerdir. ABD yurtdışındaki vatandaşlarının kullanabilmesi için bir çalışma yürütmektedir. İngiltere, ABD, Belçika, Estonya ve Norveç başta olmak üzere gelişmiş ülkeler elektronik seçim için geniş konsorsiyumlar oluşturmuştur (Boyacı ve diğ., 2012).

Finlandiya, e-imza kanunu 2003 yılında yürürlüğe girmiştir. 1998 yılında, Fin Hükümeti elektronik kimlik tanımlama, veri transferinde şifreleme ve elektronik işlemler için elektronik imza kullanımına imkan tanıyan bir sistem yaratmaya karar vermiş ve Nüfus Kayıt Merkezince Açık Anahtar Altyapısı tabanlı sertifikasyon hizmetleri sunulmaya başlanmıştır (Yeşil ve diğ., 2006).

Hollanda, e-imza kanunu 2003 yılında yürürlüğe girmiştir. Hollanda'da e-imza kanununa göre, ESHS (Elektronik Sertifika Hizmet Sağlayıcı) tarafından sertifika

verilecek olan kişi, kimlik tespiti esnasında bizzat hazır bulunmak zorundadır (Yeşil ve diğ., 2006).

Estonya, e-imza kanunu 2000 yılında yürürlüğe girmiştir. Estonya devleti elektronik kimlik kartını tüm vatandaşlarına zorunlu hale getirmiştir (Yeşil ve diğ., 2006).

Siemens ve Siemens Business Services Kurumsal PKI (Public Key Infrastructure) Projesi ile dünyanın en büyük kurulu sitelerinden birini kurmuş ve işletmektedir. Sertifika Otoritesi, tek merkezden tüm dünyadaki Siemens ve Siemens Business Services çalışanlarına sayısal imzalar ve şifreleme yoluyla güvenlik çözümü sağlanmıştır (Yıldırım ve diğ., 2009).

Avrupa Birliği'nde kullanılan ve elektronik imza çeşitlerini tanımlayan ETSI (Avrupa Telekomünikasyon Standartları Enstitüsü) 101733 standardı kullanılarak Politecnico di Torino Üniversitesinde geliştirilmiş Elektronik Doküman Yönetim Sistemi bulunmaktadır. Elektronik Doküman Yönetim Sistemi projesi, güvenilir elektronik imzaların uygulanabilirliğini göstermek amacıyla teknik bir altyapı oluşturmak için başlatılmıştır. Tıp dünyası için Elektronik Doküman Yönetim Sistemi entegrasyonu sağlanmıştır. Elektronik Doküman Yönetim Sistemi ile verimlilik ve servis kalitesi arttırılabilir. Elektronik belge ve elektronik dokümanlar kullanılarak doktorların işleri hızlandırılabilir ve herhangi bir veri kaybına uğramadan güvenilir veriye ulaşılabilir. Elektronik Doküman Yönetim Sistemiyle XML formatında dosyaların oluşturulması, saklanması ve aranması sağlanabilir. Doktorlar mobil cihazları kullanarak e-MR'ları elektronik olarak imzalayabilir. Politecnico di Torino'da kullanılan elektronik doküman yönetim sistemi, gerçek elektronik belgelerde elektronik imza kullanımında öncü olmuştur ve statik belgeler için kullanılmıştır (Berbecaru ve diğ., 2002).

Hazırlanan tezde ilk bölümünde; açık anahtarlı ve özel anahtarlı kripto sistemler anlatılmıştır. Açık anahtarlı ve özel anahtarlı kripto sistemlerde kullanılan algoritmaların çalışmalarına yer verilmiştir.

Tezin ikinci bölümünde; elektronik imza altyapısına değinilmiştir. Elektronik imzanın ne olduğu, Windows işletim sistemi üzerinde uygulama geliştirmek için

gerekli olan altyapı anlatılmıştır. Açık anahtar altyapısından ve altyapıyı oluşturan bileşenlerin üzerinde durulmuştur.

Tezin üçüncü bölümünde; Android işletim sistemi üzerinde Açık Anahtar Altyapısı kullanılarak elektronik imzalama uygulamasının geliştirilmesi anlatılmıştır. Uygulama geliştirilirken kullanılan altyapı ile uygulamanın çalışması anlatılmıştır. Uygulamanın ekran görüntülerine yer verilmiştir. Uygulama geliştirilirken kullanılan ortam hakkında bilgi verilmiştir.

Tezin sonuç bölümünde; geliştirilen uygulamanın genel olarak değerlendirmesi ve ileriki çalışmalar için neler yapılabileceği anlatılmıştır.

1. KRİPTO SİSTEMLER

Kriptoloji uygulamalarının gerçeklendiği kripto sistemler; bir anahtarı kullanarak düz metni şifreli hale, şifreli metni düz metne çeviren algoritmik sistemlerdir. Düz metin, gizlilik nedeniyle saklanmak istenen metindir. Şifreli metin, anlamsız söz dizimi olarak görülür. Şifreli metin ya da düz metinden tüm mümkün anahtarları bir bir sayarak bulmak ya da anahtar olmadan şifreli metinden düz metni çıkarmak mümkün değildir. Olası tüm anahtarları sıralamak mümkün değildir. Şifreli metin doğru rastgele değerlerden çıkarılamaz (URL-2, URL-4).

Kripto sistemler;

- Açık Anahtarlı (Asimetrik - Public Key) Kripto sistemler,
- Özel Anahtarlı (Simetrik - Private Key) Kripto sistemler,

olmak üzere ikiye ayrılır.

T.C. 5070 sayılı Elektronik İmza Kanunu'nda özel anahtar "imza oluşturma verisi", açık anahtar "imza doğrulama verisi" olarak isimlendirilmiştir (Kaya ve Topcan, 2010).

Açık anahtarlı kripto sistemler ile haberleşmenin temelleri olan gizlilik, bütünlük, kimlik doğrulama, inkar edilememezlik ve süreklilik hizmetleri sağlanır. Özel anahtarlı kripto sistemlerle ise gizlilik ve güvenlik hizmetleri sağlanır. Özel anahtarlı kripto sistemler performans olarak açık anahtarlı kripto sistemlere göre daha hızlıdırlar. Güvenlik açısından açık anahtarlı kripto sistemler ile özel anahtarlı kripto sistemleri karşılaştırdığımızda anahtar uzunluğuna bağlı olarak güvenlik düzeyinin değiştiği görülür. Tablo 1.1'de Açık Anahtarlı Kripto sistemler ile Özel Anahtarlı Kripto sistemlerin haberleşmenin temel unsurları açısından değerlendirilmesi görülmektedir (Gülaçtı, 2006a).

Tablo 1.1. Özel Anahtarlı ile Açık Anahtarlı Kripto sistemlerin Karşılaştırılması

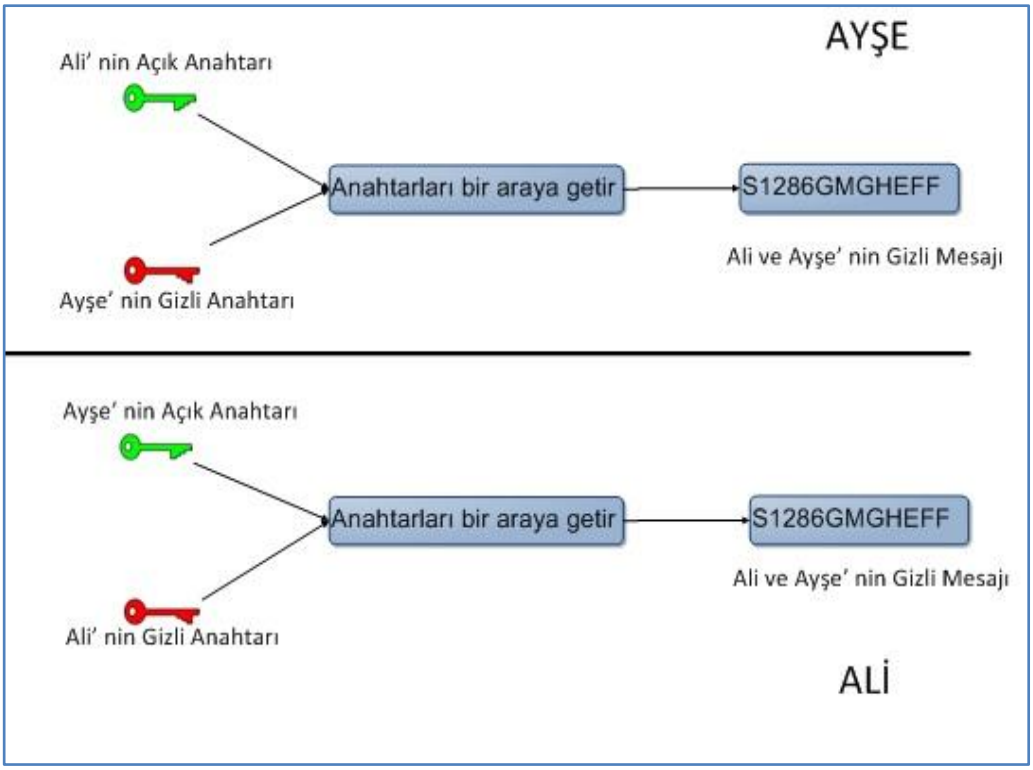
	Özel Anahtarlı Kripto sistemler	**Açık Anahtarlı Kripto sistemler**
Gizlilik	Sağlar	Sağlar
Bütünlük	---	Sağlar
Kimlik Doğrulama	---	Sağlar
İnkar Edememezlik	---	Sağlar
Performans	Hızlı	Yavaş
Güvenlik	Anahtar uzunluğuna bağlı	Anahtar uzunluğuna bağlı

1.1. Açık Anahtarlı Kripto Sistemler

Açık anahtarlı kripto sistemler asimetrik kripto sistemler olarak da adlandırılır. Açık anahtarlı kripto sistemlerde açık ve özel anahtar çifti kullanılır. Açık ve özel anahtar çifti ile şifreleme ve şifre çözme işlemleri gerçekleştirilir. Şifreleme ve şifre çözme işlemlerinde kullanılan anahtarlar birbirinden farklıdır. Asimetrik algoritmalar kullanılarak sertifika sahibine ait, açık ve özel anahtar çifti oluşturulur.

1.1.1. Diffie-Hellman anahtar değişim protokolü

Diffie-Hellman anahtar değişim protokolünde, iki kullanıcı için anahtar değişimi güvenli bir şekilde sağlanır. Böylece mesajın gizliliği sağlanmış olur. Gönderici ve alıcı durumundaki sertifikalar için açık ve özel anahtar çifti üretilir. Açık anahtar paylaşılır. Alıcı ve gönderici durumundaki iki taraf da birbirinin açık anahtarını elde ettiğinde, iki taraf da birbiriyle gizli mesaj alışverişinde bulunabilir (Çağlar, 2004).



Şekil 1.1. Diffie - Hellman Anahtar Değişim Protokolü

Diffie-Hellman gizli iletişimlerde kullanılabilecek ortak özel anahtar üretir. Bu anahtar da ortak ağlarda (güvenli olmayan kanaldan) güvenli veri alışverişini sağlar. Şekil 1.1'de Diffie-Hellman anahtar değişim protokolünün nasıl gerçekleştirildiği görülmektedir.

Diffie-Hellman anahtar değişim protokolünde Ali ve Ayşe'nin birbiriyle iletişim kurmak istediğini varsayalım. Üçüncü bir kişinin mesajdan haberdar olmaması gerekmektedir.

p'nin primitif sayı olduğunu ve g'nin mod p'ye göre primitif kök olduğunu varsayalım.

- Ali rastgele a değerini seçer ve $u \equiv g^a \pmod{p}$ eşitliğini hesaplar. U değerini Ayşe'ye gönderir.
- Ayşe rastgele b değerini seçer ve $v \equiv g^b \pmod{p}$ eşitliğini hesaplar. V değerini Ali'ye gönderir.
- Ayşe anahtar k'yı hesaplar.

$k \equiv u^b \equiv (g^a)^b \pmod{p}$

- Ali anahtar k'yı hesaplar.

$k \equiv v^a \equiv (g^b)^a \pmod{p}$

- İşlemler sonucunda Ali ve Ayşe k isimli aynı anahtara sahip olmuşlardır.

$k \equiv u^b \equiv g^{ab} \pmod{p}$

Eğer üçüncü bir kişi k değerini bulmak isterse a ya da b değerine ihtiyacı olacaktır. Diğer türlü, ayrık logaritma problemini çözmek zorunda kalacaktır. Ayrık logaritma problemi modüler aritmetikte üs alma operasyonudur. Kabul edilebilir bir sürede bu problemi çözen bilinen bir algoritma yoktur (Garner, 2013).

Bu algoritmanın çalışma mantığı şöyledir:

- Ayşe, Ali'nin mesaj gönderme işlemlerinde kullanacağı açık anahtarını öğrenir.
- Ayşe, kendi özel anahtarı ile Ali'nin açık anahtarını birleştirerek bir hesaplama yapar. Bu hesaplamanın sonucu bir şifreli mesaj olarak saklanır.
- Şifrelenmiş mesaj Ali'ye gönderilir.
- Ali şifreli mesajı aldığında kendi özel anahtarı ve Ayşe'nin açık anahtarı ile bir hesaplama yapar ve böylece Ayşe'den gelen mesajı okuyabilir.

1.1.2. RSA algoritması

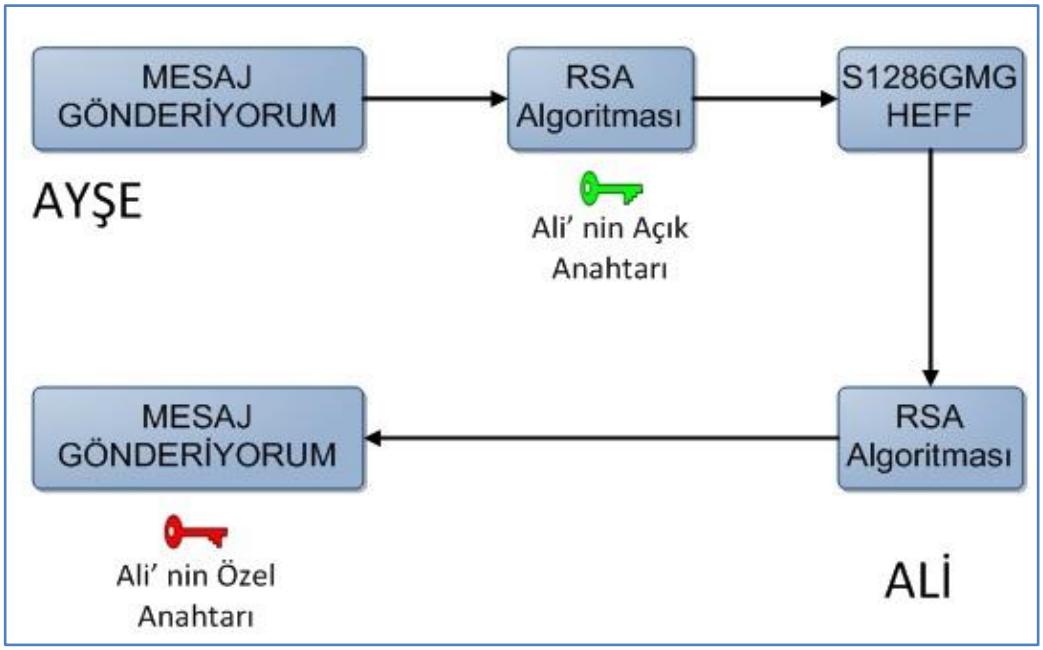
RSA algoritması en yaygın kullanılan açık anahtarlı kripto algoritmasıdır. Adını, algoritmayı bulan R. Rivest, A. Shamir, L. Adleman isimlerinin ilk harflerinden alır.

RSA algoritması, elektronik imzalama yöntemi olarak kullanılır. Çarpanlara ayırma problemini temel alır (Rivest ve diğ., 1978).

Şekil 1.2'de algoritmanın çalışması şekil olarak gösterilmiştir. Algoritma şu şekilde çalışır:

- n ve e değerleri açık anahtarı, d değeri özel anahtarı gösterir.
- n bileşik bir tamsayıdır ("modulus")
- e bir tamsayıdır ("açık üs ifadesi")
- d bir tamsayıdır ("gizli üs ifadesi")
- $ed \equiv 1 \bmod (p-1)(q-1)$ denkliğini sağlayan e ve d değerleri seçilir.
- e=65537 imzalama işlemlerini kolaylaştırdığı için tavsiye edilir.
- n=p.q eşitliğini sağlayacak şekilde en az 512 bitlik asal p ve q sayıları üretilir.

- Mesaj m'i imzalama için $c = m^e \bmod n$ işlemi uygulanır.
- Mesaj m'i imza doğrulama için $m = c^d \bmod n$ işlemi uygulanır.



Şekil 1.2. RSA Algoritması

- Ayşe, Ali'ye mesajı şifreli olarak göndermek için:

Mesaj m'nin e'inci üssünü alır, yani m'yi Ali'nin açık anahtarı ile şifreler. Bu işlemler şu şekilde gerçekleşir:

- Şifreli mesaj c'yi Ali'ye gönderir. $c = m^e \bmod n$ işlemi gerçekleştirilir.
- Ali c sayısının d'inci üssünü alır. Bunun anlamı c'nin şifresini kendi özel anahtarını kullanarak çözmesidir. $m = c^d \bmod n$ işlemi gerçekleştirilir.

RSA algoritması işletilirken Ayşe iki tane modüler çarpma işlemi yapar. Ali 1,5 log n tane modüler çarpma yapar. Eğer işlemler sırasında kullanılan p ve q asal sayıları bilinirse işlem daha hızlı yapılabilir. Üçüncü kişi gizli mesajı elde etmek için kök bulma ya da çarpanlarına ayırma problemini çözmelidir. Bunun için etkili bir çözüm bulunamamıştır. İşlemlerde kullanılan modülüs n sayısı arttıkça algoritmanın güvenliği de artar.

RSA algoritmasının çalışmasına bir örnek verilmiştir (URL-3).

İlk adım olarak N tamsayısı oluşturulacak p ve q sayıları seçilir. Gerçekte çok büyük iki farklı asal sayı olması gereklidir fakat burada örnek olduğu için küçük sayılar kullanılır. p=7 ve q=19 seçilebilir.

İkinci adım olarak bu sayıların çarpımını bularak N tam sayısının değeri bulunur.
N = pxq => N = 7x19 => N = 133

Bu N sayısının Totient değeri hesaplanır. Totient, sayılar teorisinde bir tam sayının o sayıdan daha küçük ve o sayı ile aralarında asal olan sayı adedini belirten fonksiyondur. Genellikle Euler Totient ya da Euler'in Totienti olarak adlandırılan Totient, İsveçli matematikçi Leonhard Euler tarafından tanımlanmıştır. Totient fonksiyonu, Yunan harflerinden phi(φ) ile simgelendiği için Phi fonksiyonu olarak da anılabilir (URL-18). φ(N) = (p-1)x(q-1) => φ(N) = 6x18 = 108

Son olarak bu Totient değeri ile aralarında asal olan e sayısı seçerek e'nin tersine eşit olan d sayısı bulunur. $1 < e < 108$ ve EBOB(e, 108) = 1 denklemlerini sağlayan e sayısı 5 olarak seçilir. Öyleyse 5'in mod 108'e göre tersi bulunursa, d sayısı da bulunmuş olur. $dx5 \equiv 1 (\text{mod } 108)$ denkliğini sağlayan d sayısı Genişletilmiş Öklid Algoritması kullanılarak bulunabilir. Önce Öklid Algoritması uygulanarak bu iki sayının en büyük ortak bölenini bulunur. Bu sayılar aralarında asal olacak şekilde seçildiği için sonucun 1 olması gereklidir. Denklem (1.1) ile (1.2a), (1.2b) ve (1.2c) de EBOB değeri bulunmaktadır.

$$108 = 21x5 + 3 \quad (1.1)$$

$$5 = 1x3 + 2 \quad (1.2a)$$

$$3 = 1x2 + 1 \quad (1.2b)$$

$$2 = 2x1 + 0 \quad (1.2c)$$

Sıfırdan bir önceki değer bu iki sayının EBOB'unu vermektedir. Öyleyse; EBOB(108,5) = 1

Yani beklendiği gibi bu iki sayı aralarında asaldır. Genişletilmiş Öklid Algoritmasını uygulanarak ve d sayısı bulunmaya çalışılır. Bunun için de EBOB değerinden yola

çıkarak geriye doğru giden matematiksel/mantıksal işlemler yapılması gereklidir. Denklem (1.3)'te yapılan işlemler görülmektedir.

$$
\begin{aligned}
1 &= 3 - 1x2 \\
&= 3 - 1x(5 - 1x3) \\
&= 3 - 5 + 3 \\
&= 2x3 - 5 \\
&= 2x(108 - 21x5) - 5 \\
&= 2x108 - 42x5 - 5 \\
&= 2x108 - 43x5
\end{aligned}
\tag{1.3}
$$

Hangi sayının tersi aranıyorsa onun yanındaki çarpan kendisinin tersi olmaktadır. 5 sayısının mod 108'de tersi arandığı için 5 sayısının yanındaki çarpan aranan değerdir. Bu sayı görüldüğü üzere (-43)'tür veya başka bir deyişle 108+(-43) = 65'tir. Sonuç olarak özel anahtar için aranan d sayısı 65 olarak bulunmuş oldu. Tüm değerleri bulunduğuna göre artık açık ve özel anahtarları oluşturarak şifreleme ve şifre çözme işlemlerine geçilebilir.

Şifrelemede kullanılacak açık anahtarı : (N,e) = (133,5)
Şifre çözmede kullanılacak özel anahtar : (N,d) = (133,65)

Anahtarlar ile şifreleme ve şifre çözme işlemini gerçekleyerek şifrelenen metin ile şifre çözme sonucunda elde edilen metnin aynı olup olmadığını test edelim. Bu örnek için m=15 sayısını açık anahtarı ile şifreleyip özel anahtar ile açılır ve çıkan sonuçları karşılaştırılır. Denklem (1.4) ve (1.5)'te c ve m sayıları hesaplanır.

$$c = mxe = 155 = 78 \pmod{133} \tag{1.4}$$

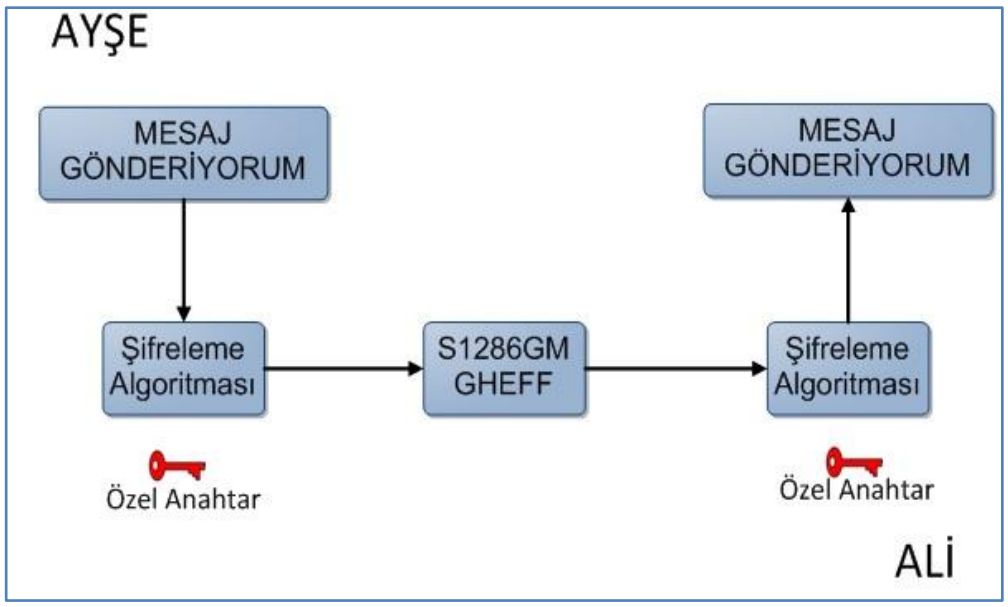
$$m = cxd \Rightarrow m = 7865 \pmod{133} \Rightarrow m = 15 \tag{1.5}$$

1.2. Özel Anahtarlı Kripto Sistemler

Özel anahtarlı kripto sistemler, simetrik kripto sistemler olarak da adlandırılır. Özel anahtarlı kripto sistemlerde, şifreleme ve şifre çözme için tek anahtar kullanılır. Bu anahtar özel anahtar ya da simetrik anahtar olarak adlandırılır ve yalnızca şifreleme-

şifre çözme işlemlerini yapan taraflar bu anahtarı bilir. Özel anahtarlı kripto sistemlerde; şifrelenmiş mesaj, mesaja hiç bir açık anahtar eklenmeden yollanır.

Özel anahtarlı kripto sistemler, açık anahtarlı kripto sistemlere göre daha hızlıdır. Özel anahtarlı kripto sistemlerde, şifrelenmiş veri üçüncü taraf tarafından ele geçirilme riski olsa bile belli bir hat üzerinden gönderilir. Özel anahtarlı kripto sistemlerde, güvenilir bir şekilde anahtar dağıtımı yapmak zordur. İki tarafın haberleşmesinde rol alan bütünlük ve kimlik doğrulama güvenlik gereksinimlerinin gerçeklenmesi zordur. Şekil 1.3'te özel anahtarlı kripto sistemde bir mesajın iletimi görülmektedir.



Şekil 1.3. Özel Anahtarlı Kripto Sistem

1.2.1. DES algoritması

DES algoritması (Data Encryption Standard) IBM tarafından geliştirilen bir şifreleme tekniğidir. ABD hükümeti tarafından 1977 yılında standart olarak belirlenmiştir. DES, özel anahtarlı kripto sistemdir ve bugüne kadar en çok kullanılan kripto sistemlerden biridir.

DES algoritması, 64 bitlik bir blok şifrelemeli algoritmadır. Verileri 64 bit'lik bloklar halinde şifreler. Kullanılan anahtar ise 56 bit uzunluğundadır. 8 bit parity bit olarak kullanılır. Bu teknikte, her şifreleme adımına döngü denilir ve her döngüde kullanılan anahtar farklıdır. Gönderilecek metin, belirli uzunluktaki bloklara bölünür

ve ayrı ayrı şifrelenen bloklar ile şifreli metin elde edilir. Blok uzunluğu, kullanılan işlemci hızına göre değişebilir (Mehuron, 2014).

DES algoritmasının dezavantajı yavaş olmasıdır. Ayrıca DES algoritması, Brute Force ataklarına karşı da güvensizdir. DES'in güvenilirliğini arttırmak için 3DES tekniği geliştirilmiştir (URL-5).

1.2.2. Üçlü DES (3DES) algoritması

3DES algoritması, bir simetrik şifreleme algoritmasıdır. DES algoritmasının 56 bit'lik anahtar uzunluğundan kaynaklanan güvenlik eksikliğini azaltmak için 3DES algoritması geliştirilmiştir. 3DES algoritması ile 168 (56x3) bit uzunluğunda kripto anahtarları kullanılır ve çalışma prensibi DES algoritmasına benzerdir (Kırımlı, 2007).

Bu yöntemde, şifrelenen veri tekrar geri çözülür ve DES şifrelemesi 3 sefer ard arda yapılır. Şifreleme için kullanılan ve uzunluğu 24 byte olan anahtar, 3 bloğa ayrılır. İlk 8 byte ile şifreleme yapılır, buraya kadar olan kısım DES işlemidir. Daha sonra şifrelenen metin ortadaki 8 byte ile çözülür ve son 8 byte ile tekrar şifrelenerek 8 byte'lık blok elde edilir. DES'e göre güvenilirliği fazladır fakat hız 3 kat daha azalmıştır. Her byte bir parity biti bulundurur. Dolayısı ile kullanılan anahtar 168 bittir.

DES'i kırmak için yüksek maliyetle son teknoloji makineler geliştirilmiş olmasına rağmen 3DES, bankalar ve devlet daireleri olmak üzere birçok ortamda kullanılmaya devam etmektedir (URL-5).

1.2.3. AES algoritması

AES (Advanced Encryption Standard - Gelişmiş Şifreleme Standardı), ABD Ulusal Standart ve Teknoloji Enstitüsü tarafından yayınlanmıştır. 26 Kasım 2001 tarihinde US FIPS PUB 197 kodlu dokümanla duyurulmuştur. AES, uluslararası alanda şifreleme standardı olarak kullanılmaktadır. AES ile tanımlanan şifreleme algoritması, hem şifreleme hem de şifreli metni çözmede kullanılan anahtarların birbiriyle ilişkili olduğu, simetrik-anahtarlı bir algoritmadır. AES için şifreleme ve şifre çözme anahtarları aynıdır. Halihazırda birçok şifreleme paketinde yer alan

algoritma Amerikan Ulusal Güvenlik Teşkilatı (NSA - National Security Agency) tarafından çok gizli bilginin şifrelenmesinde kullanımı onaylanan kamuya açık ilk şifreleme algoritmasıdır.

AES, DES'e oranla daha büyük anahtar boyutu kullanmaktadır. AES'in hem yazılım hem de donanım performansı yüksektir. 128-bit girdi bloğu; 128, 192 ve 256 bit anahtar uzunluğuna sahiptir.

AES, durum denilen 4x4 sütun-öncelikli bayt matrisi üzerinde çalışır. Matristeki işlemler de özel bir sonlu cisim üzerinde yapılmaktadır.

Algoritma belirli sayıda tekrar eden girdi açık metni, çıktı şifreli metne dönüştüren özdeş dönüşüm çevirimlerinden oluşmaktadır. Her çevirim, son çevirim hariç, dört adımdan oluşmaktadır. Şifreli metni çözmek için bu çevirimler ters sıra ile uygulanır. Çevirimlerin tekrar sayıları 128-bit, 192-bit ve 256-bit anahtar uzunlukları için sırası ile 10, 12 ve 14'tür (Federal, 2001).

2. ELEKTRONİK İMZA ALTYAPISI

Elektronik imza literatürüne bakılınca tanımının çok geniş tutulduğu ve farklı teknolojiler kullanılarak oluşturabildiği görülür. Örneğin, kağıt üzerinde oluşturulmuş ıslak imzanın tarayıcıdan geçirilerek resminin elektronik ortama alınması ve bu resmin elektronik dokümanın sonuna eklenmesi, bilgisayar ekranına atılan imzalar, kişinin parmak izinin bilgisayar ortamına aktarılması ile elde edilen veriler de elektronik imza kapsamında değerlendirilmektedir. Ancak ülkemizde bu tarif edilen yöntemlerle oluşturulan elektronik imzalar yasal olarak geçerli değildir (Kaya ve Topcan, 2010).

2.1. Elektronik İmza Nedir?

Elektronik imza; T.C. 5070 sayılı Elektronik İmza Kanununda, başka bir elektronik veriye eklenen veya elektronik veriyle mantıksal bağlantısı bulunan ve kimlik doğrulama amacıyla kullanılan elektronik veri olarak ifade edilmiştir (URL-6).

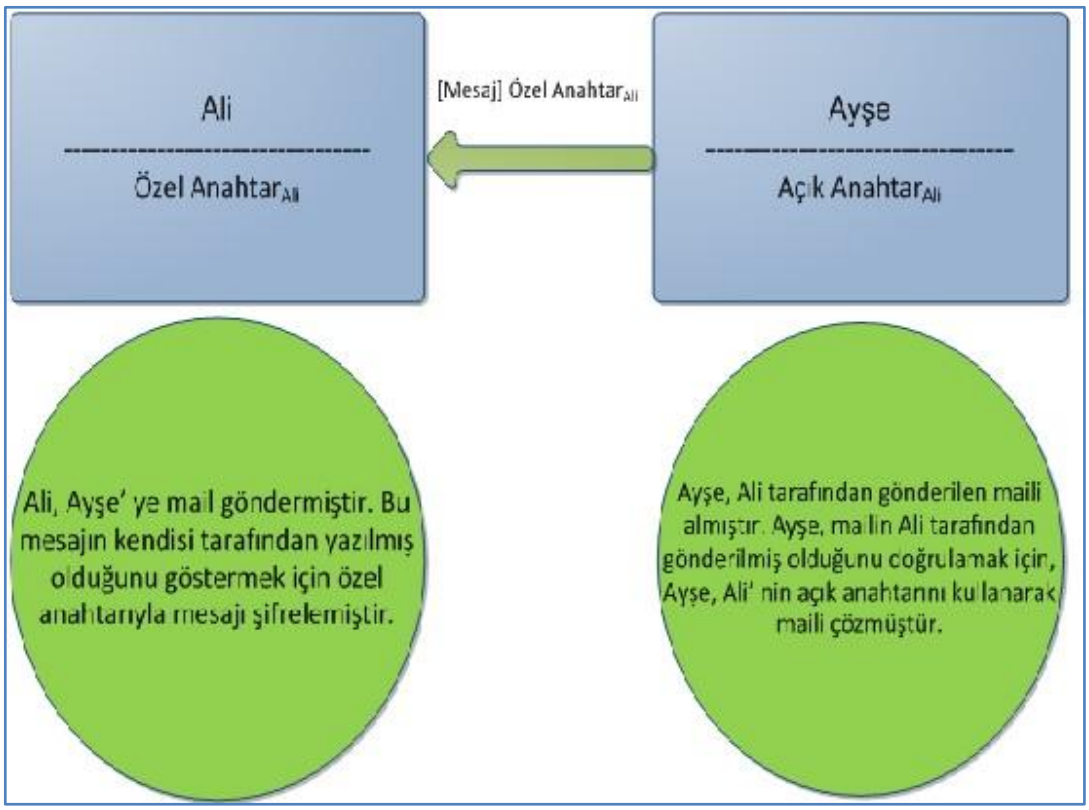
Nitelikli elektronik sertifika 5070 sayılı Elektronik İmza Kanunu'nun 9uncu maddesine göre aşağıdaki maddeleri sağlaması zorunludur.

- Sertifikanın "nitelikli elektronik sertifika" olduğuna dair bir ibarenin,
- Sertifika hizmet sağlayıcısının kimlik bilgileri ve kurulduğu ülke adının,
- İmza sahibinin teşhis edilebileceği kimlik bilgilerinin,
- Elektronik imza oluşturma verisine karşılık gelen imza doğrulama verisinin,
- Sertifikanın geçerlilik süresinin başlangıç ve bitiş tarihlerinin,
- Sertifikanın seri numarasının,
- Sertifika sahibi diğer bir kişi adına hareket ediyorsa bu yetkisine ilişkin bilginin,
- Sertifika sahibi talep ederse meslekî veya diğer kişisel bilgilerinin,
- Varsa sertifikanın kullanım şartları ve kullanılacağı işlemlerdeki maddî sınırlamalara ilişkin bilgilerin,
- Sertifika hizmet sağlayıcısının sertifikada yer alan bilgileri doğrulayan güvenli elektronik imzasının bulunması zorunludur.

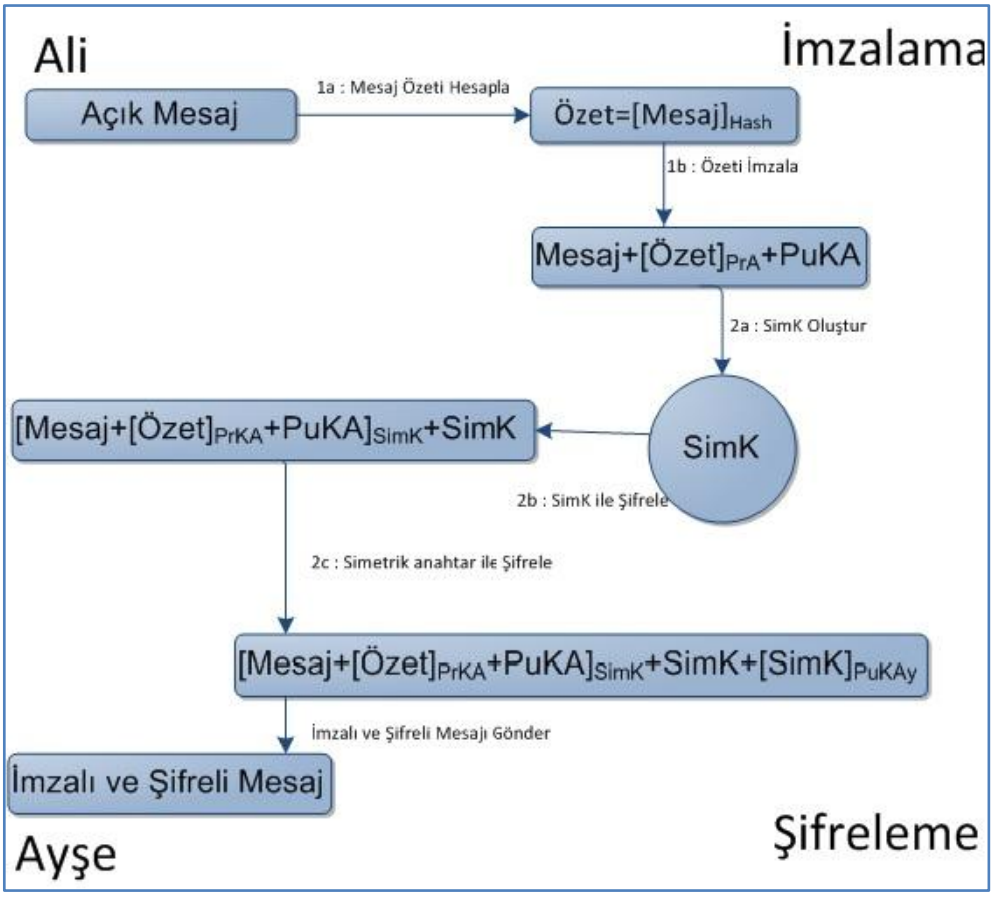
Elektronik imza aşağıdaki özelliklere sahiptir (Gülaçtı, 2006a) :

- Mesajın sonuna eklenir.
- Mesaj alıcısının, mesajın göndericisinin kimliğini doğrulamasını ve mesajın bütünlüğü kontrolünü sağlar.
- İnkar edememezlik hizmetini sağlar.
- Açık anahtarlı kriptografi kullanır.

Açık anahtar altyapısının örnek kullanım senaryosu Şekil 2.1'de görülmektedir.



Şekil 2.1. Açık Anahtar Altyapısı Örnek Kullanım Senaryosu



Şekil 2.2. Mesajın Şifrelenmesi

Mesajın şifrelenmesi akışı Şekil 2.2'de görülmektedir. Şekil üzerinde yer alan;

- PuKA: Ali'nin Açık Anahtarı
- PrKA: Ali'nin Özel Anahtarı
- PuKAy: Ayşe'nin Açık Anahtarı
- SimK: Simetrik Anahtar
- Hash: Özet Alma Algoritması

gösterir.

Mesaj imzalama : Elektronik imza aşağıdakileri içerir (CGI, 2004):

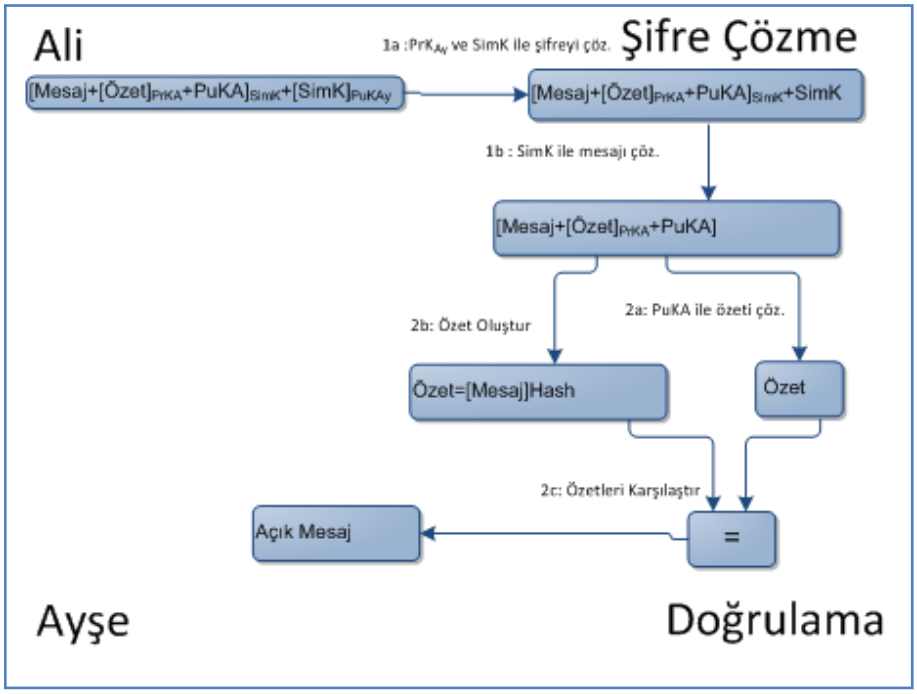
Mesaj özeti oluşturma: Özet oluşturmanın asıl amacı, mesajın değiştirilmemiş olduğundan emin olmaktır. Buna mesaj bütünlüğü denir.

Özet imzası: İmza, göndericinin özel anahtarıyla şifrelenmiştir. İmza ile birlikte gönderici tarafından kullanılan özet alma algoritmasıdır. Göndericinin açık anahtarı da imzaya eklenir. Böylece herkes, göndericinin açık anahtarı ve özet alma

algoritmasını kullanarak imzayı çözebilir ve doğrulayabilir. Açık anahtar şifrelemesi ve özet alma algoritmaları göndericinin özel anahtarının özetinin şifrelendiğini ve mesajın herhangi bir değişikliğe karşı korunduğunu kanıtlar.

Mesaj şifreleme : Şifreleme aşağıdaki 3 adımı içerir:

- Bir kez kullanımlık şifreleme/şifre çözme anahtarının oluşturulması, uzun mesajlarda açık anahtarların şifreleme/şifre çözme için fazla zaman alır. Bu sebeple özel anahtar algoritmalarının kullanılması daha uygundur.
- Mesaj şifre çözme; tüm mesaj(mesaj ve imzası) SimK kullanılarak şifrelenir, yukarıdaki özel anahtar oluşturulur.
- Simetrik anahtar şifreleme; SimK mesajı çözen kullanıcı tarafından da kullanılır. SimK sadece gönderici Ali için erişebilir olmalıdır. SimK'yı gönderici hariç herkesten gizlemek için göndericinin açık anahtarı kullanılır. SimK, mesaja oranla küçük bir bilgi olsa da açık anahtar algoritmalarının verimsizliği nedeniyle oluşan performans kaybı kabul edilebilirdir.



Şekil 2.3. Mesajın Doğrulanması

Mesajın doğrulanması işleminin akışı Şekil 2.3'te görülmektedir.

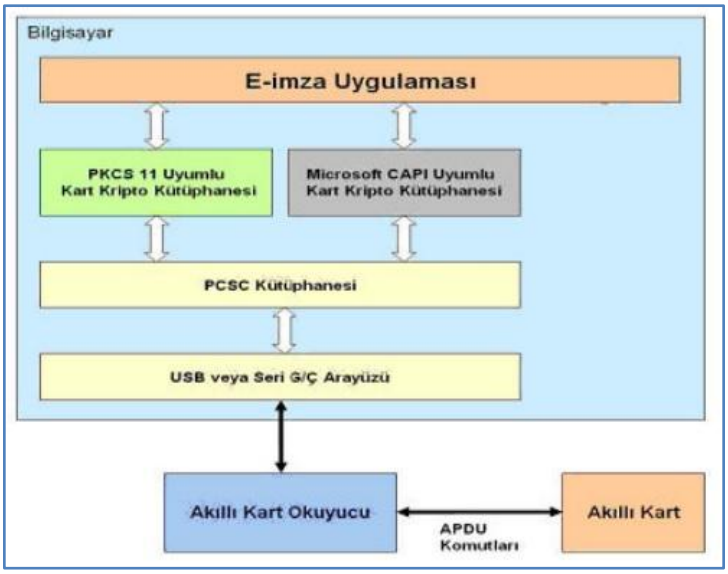
Mesajın çözülmesi adımları şöyledir: Bir kez kullanılan özel anahtar mesajı çözer. SimK anahtarı Ayşe'nin açık anahtarı kullanılarak çözülmüştür. Sadece Ayşe SimK'yı çözebilir ve mesajı çözmek etmek için kullanır. Mesaj çözülmesi işlemi; mesaj ve imzası SimK kullanılarak gerçekleştirilmiştir.

İmza doğrulaması için mesaj özeti çözülmesi işlemi şöyle gerçekleştirilir: Özet Ali'nin özel anahtarı kullanılarak şifrelenmiştir. Özet, mesajda kullanılan göndericinin açık anahtarıyla çözülmüştür. Özet alma tek taraflı bir işlem olduğundan mesaj tek başına özetten çıkarılamaz. Alıcı göndericinin kullandığı aynı özet alma algoritmasını kullanarak özeti çıkarır.

Özetlerin karşılaştırılması işleminde çözülen özet ile oluşturulan özet karşılaştırılır. Eğer ikisi de aynıysa, imza doğrulanmıştır. Alıcı, göndericiden gelen mesajın değiştirilmemiş mesaj olduğunu kabul eder. Eğer özetler aynı değilse; mesaj gönderici tarafından imzalanmamıştır ya da mesaj değiştirilmiştir. Her iki durumda da mesaj reddedilir.

2.2. Windows Üzerinde Elektronik İmza Uygulama Geliştirilmesi

Bilgisayarda akıllı kartların çalışabilmesi için aşağıdaki Şekil 2.4'tekine benzer bir mimariye ihtiyaç vardır.



Şekil 2.4. Akıllı Karta Erişim (Özbey, 2006)

Bir akıllı kartın işletim sisteminde kullanılabilmesi için akıllı kart okuyucu sürücüsü, işletim sistemi akıllı kart bileşenleri ve akıllı kart kripto kütüphanesi yazılımlarına ihtiyaç vardır.

Bilgisayara bağlanan akıllı kart okuyucular için sürücü yüklenmesi gereklidir. Yaygın olarak PCSC standardı kullanılır. Akıllı kart kripto kütüphanesi; yazılımcının akıllı karta erişmesini sağlayıp imzalama, şifreleme gibi işlemleri yaptırabileceği fonksiyonların bulunduğu kütüphanelerdir. Bunun için iki kütüphane vardır. Bunlardan ilki olan PKCS#11 uyumlu kütüphane genellikle açık kaynak kodlu ürünlerin akıllı karta erişim için tercih ettikleri kütüphanedir. Bu kütüphane, platform bağımsız akıllı kart uygulaması geliştirmek için kullanılan kütüphane olduğu için Windows ve Linux gibi farklı ortamlarda kullanılabilir. Diğer kütüphane ise Microsoft CAPI - Crypto API uyumlu kütüphanedir. Bu tip kütüphane Microsoft Windows işletim sistemi üzerinde kullanılmak üzere tanımlanmış bir standarda uygun yazılmıştır (Özbey, 2006).

Elektronik imza oluşturulurken kart ile iletişim APDU - Application Protocol Data Unit komutları üzerinden sağlanmaktadır. APDU komutları ISO/IEC 7816-4 standardında belirtilmiş, akıllı kart ile akıllı kart okuyucu arasında iletişimi sağlayan yapıdır. Tablo 2.1'de karta iletilen APDU komutunun içeriği ayrıntılı olarak verilmiştir (URL-12)

Tablo 2.1. APDU Komutu Detayları

Tip	**İsim**	**Uzunluk**	**Detaylar**
CLA	Class	1 byte	Komutun sınıfını gösterir
INS	Instruction	1 byte	Komut talimatı
P1	Parametre 1	1 byte	Talimat için ilk parametre
P2	Parametre 2	1 byte	Talimat için ikinci parametre
Lc	Length command	0-3 byte	Komut verisinin uzunluğu
Data	Data	Lc byte	APDU isteği
Le	Length expected	0-3 byte	APDU cevabının uzunluğu

APDU komutu; karta gönderilen APDU komutu ve karta gönderilen komuttan dönen APDU cevabı olmak üzere iki çeşittir. Karta gönderilen APDU komutu zorunlu olarak 4 byte başlık (CLA, INS, P1, P2) ve 0-255 byte arası data alanlarını içerir.

APDU komutunun data alanında bulunan byteların sayısı Lc ile gösterilir. Data alanında beklenen en fazla byte uzunluğu Le ile gösterilir. CLA alanı komutun kapsamının ne olduğunu ve ISO/IEC 7816 standardına göre uyumlu olup olmadığı gösterir. Güvenli haberleşme ve lojik kanal numarası için uygunluğunu gösterir. INS alanı ISO/IEC 7816 standardında tanımlanan iletişim protokollerinin kullanımına izin verilip verilmeyeceğini gösterir. P1-P2 parametrik byteları herhangi bir değere sahip olabilirler. Parametre değeri herhangi bir bilgi içermiyorsa '00' değeri verilebilir. Data alanı, P1 parametresi ile karta işlem yaptırılacak komutun içeriğine göre değişir. Aşağıdaki listede en çok kullanılan komutlar ve komutların hangi standartta belirtildiği gösterilmiştir.

En çok kullanılan APDU komutları hangi standartta tanımlanmış olduğunu belirtir liste olarak Tablo 2.2'de verilmiştir (URL-10).

Tablo 2.2. En Çok Kullanılan APDU Komutları

Komut	Instruction (INS)	Standard
ACTIVATE FILE	‘44’	ISO/IEC 7816-9
APPEND RECORD	‘E2’	ISO/IEC 7816-4
CREATE FILE	‘E0’	ISO/IEC 7816-9
CREATE RECORD	‘E2’	EN 726-3
DEACTIVATE FILE	‘04’	ISO/IEC 7816-9
DELETE	‘E4’	OP
DELETE FILE	‘E4’	ISO/IEC 7816-9
ERASE BINARY	‘0E’	ISO/IEC 7816-4
EXTERNAL AUTHENTICATE	‘82’	ISO/IEC 7816-4
GET CHALLENGE	‘84’	ISO/IEC 7816-4
GET DATA	‘CA’	ISO/IEC 7816-4
GET RESPONSE	‘C0’	TS 51.011
MANAGE SECURITY ENVIRONMENT	‘22’	ISO/IEC 7816-8

Tablo 2.2. (Devam) En Çok Kullanılan APDU Komutları

Komut	Instruction (INS)	Standard
PUT DATA	'DA'	ISO/IEC 7816-4
PUT KEY	'D8'	OP
REACTIVATE FILE	'44'	ISO/IEC 7816-9
READ BINARY	'B0'	TS 51.011
READ RECORD	'B2'	TS 51.011
READ RECORD(S)	'B2'	ISO/IEC 7816-4
RESET RETRY COUNTER	'2C'	ISO/IEC 7816-8
SEARCH BINARY	'A0'	ISO/IEC 7816-9
SEARCH RECORD	'A2'	ISO/IEC 7816-9
SELECT (FILE)	'A4'	ISO/IEC 7816-4
SET STATUS	'F0'	OP
STATUS	'F2'	TS 51.011
TERMINATE CARD USAGE	'FE'	ISO/IEC 7816-9
TERMINATE DF	'E6'	ISO/IEC 7816-9
TERMINATE EF	'E8'	ISO/IEC 7816-9
UPDATE BINARY	'D6'	TS 51.011, ISO/IEC7816-4
UPDATE RECORD	'DC'	TS 51.011, ISO/IEC 7816-4
VERIFY	'20'	ISO/IEC 7816-4, EMV
WRITE BINARY	'D0'	ISO/IEC 7816-4
WRITE RECORD	'D2'	ISO/IEC 7816-4

SW1-SW2 alanları kartta işlenen komutun cevabını gösteren bytelardır. Tablo 2.3'te APDU komutu alanları verilmiştir.

Tablo 2.3. APDU Komutu Alanları

APDU Komutu						
CLA	INS	P1	P2	Lc	Data	Le

APDU cevabı alanları Tablo 2.4'te verilmiştir.

Tablo 2.4. APDU Cevabı Alanları

APDU Cevabı		
Cevap	SW1	SW2

APDU cevabı detayları Tablo 2.5'te görülmektedir.

Tablo 2.5. APDU Cevabı Detayları

APDU Cevap Detayları			
Tip	İsim	Uzunluk	Detaylar
Veri	Gövde	0-3 byte	Dönen cevabın uzunluğudur (Le)
SW1	Status Word 1	1 byte	
SW2	Status Word 2	1 byte	

Örnek APDU komutu Tablo 2.6'da görülmektedir.

Tablo 2.6. Örnek APDU Komutu

C0	20	00	01	08	303030300000000	
CLA	INS	P1	P2	Lc	Veri	Le

Örnek APDU cevabı Tablo 2.7'de görülmektedir.

Tablo 2.7. APDU Cevabı

90	00

2.3. Açık Anahtar Altyapısı

Açık Anahtar Altyapısı (Public Key Infrastructure - PKI) (AAA) dijital sertifikaları yaratmak, yönetmek, dağıtmak, kullanmak, saklamak ve iptal etmek için ihtiyaç duyulan donanım, yazılım, insan, politika ve prosedürlerdir (Toorani,M., 2008).

Açık Anahtar Altyapısı kullanılarak elektronik haberleşmenin temel güvenlik öğeleri olan

- Gizlilik,
- Bütünlük,
- Kimlik Doğrulama,
- İnkar Edilememezlik,
- Süreklilik

hizmetleri sağlanır.

Açık Anahtar Altyapısı temel güvenlik öğelerini aşağıda listelenen çözümlerle yerine getirir (Erol, 2004).

- Gizlilik – Veri Şifreleme,
- Bütünlük – Sayısal İmzalama, Sertifikalar, Kimlikler,
- Kimlik Doğrulaması – Özetleme Algoritmaları, Mesaj Özetleri, Sayısal İmzalar,
- İnkar Edilememezlik – Sayısal İmzalama, İşlem Kayıtları,
- Süreklilik – Yedek Sistemler, Bakım, Yedekleme.

Açık Anahtar Altyapısı (Public Key Infrastructure - PKI) (AAA) açık anahtar kriptografisini temel alır. Çeşitli kriptografi algoritmaları ve protokolleri ile çalışır. Açık Anahtar Altyapısı servisleri DH, RSA, DSA, ECDSA gibi asimetrik kriptografi algoritmaları, DES, RC2, AES gibi simetrik kriptografi algoritmaları, SHA1, SHA2, RIPEMD160 gibi özetleme algoritmaları, SSL, CMP, OCSP gibi protokolleri kullanır. Bu algoritma ve protokollerin nasıl kullanılacağını anlatan RSA PKCS 1, IETF RFC 3280, ITU.T X.509, S/MIME gibi birçok standart da AAA sistemleri tarafından gerçeklenir (Gülaçtı, 2006b).

Açık Anahtar Altyapısı (Public Key Infrastructure - PKI) (AAA) tarafından aşağıdaki hizmetler sağlanır (Gülaçtı, 2006b):

- Sertifika Üretimi ve Yaşam Döngüsü
- Elektronik İmza Sertifikaları
- Şifreleme Sertifikaları
- SSL Sertifikaları
- Zaman Damgası Sunucu Hizmetleri
- OCSP Sunucu Hizmeti
- E-posta İçin Elektronik İmza ve Şifreleme İstemci Yazılımı
- Dosya ve Klasörler için Elektronik İmza ve Şifreleme İstemci Yazılımı
- Zaman Damgası İstemci Yazılımı

Açık Anahtar Altyapısının (Public Key Infrastructure - PKI) (AAA) sunduğu uygulama alanları aşağıda listelenmiştir (Erol, 2006):

- Güvenli posta haberleşmesi (S/MIME),
- Verilerin imzalı ve şifreli saklanması (Masaüstü güvenliği/Desktop security),
- Akıllı kartla güvenli oturum açma (Windows logon, Kerberos),

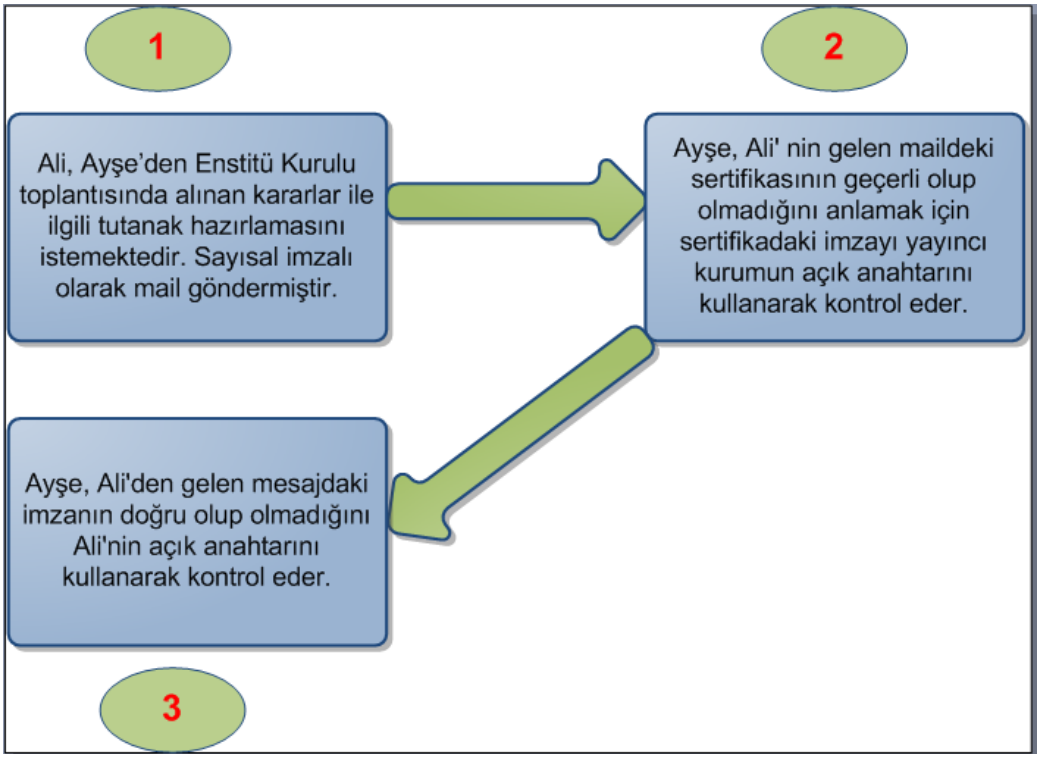
- İmzalı formlar (Signed forms),
- Sayısal noter,
- Kod imzalama,
- XML imzalama,
- İnternet protokolü güvenliği (Internet protocol security, IPSEC),
- Taşıma katmanı güvenliği (Transport layer security, TLS),
- Güvenli yuvalar katmanı (Secure socket layer, SSL),
- Güvenilir zaman damgası (Time stamp protocol, TSP),
- Sanal özel ağ (Virtual private network, VPN),
- E-birey veya e-vatandaş,
- E-ticaret, e-iş, e-bankacılık, e-kimlik, e-devlet, e-kurum, e-cüzdan, e-sınav, e-eğitim, e-imza, e-bilim, e-üniversite, e-yaşam, e-sağlık, vb.

2.4. Açık Anahtar Sertifikaları

İdeal sertifika tanımına uygun bir elektronik sertifika oluşturabilmek için uluslararası ITU-T (ITU Telecommunication Standardization Sector) kurumu X.509 standardını tanımlamıştır. Bu standarda uygun olarak hazırlanan bir sertifika aşağıdaki özellikleri taşır (Gülaçtı, 2006a):

- Sayısaldır, bilgisayarda veya elektronik bir cihazda hazırlanır.
- Sahibinin adını ve açık anahtarını içerir.
- Sahibinin çalıştığı şirketin/kurumun adını içerir.
- Genelde sahibinin e-posta adresini içerir.
- Kullanıma giriş tarihini ve son kullanım tarihini içerir.
- Yayınlayan güvenilir kurumun adını içerir.
- Yayınlayan kuruluş tarafından verilmiş tekil bir seri numarasına sahiptir.
- İçeriğin bütünlüğü yayınlayan kurulusun sayısal imzasıyla koruma altına alınmıştır.

Şekil 2.5'te açık anahtar sertifikasının bir kullanım senaryosu görülmektedir.



Şekil 2.5. Açık Anahtar Sertifikası Örnek Kullanım Senaryosu

Bir sertifikanın üretilmesinden iptaline kadar geçirdiği aşamalara "Sertifika Yaşam Çevrimi" denir. Aşağıda bir sertifika yaşam çevrimi, genel hatlarıyla tarif edilmiştir. Gerçek uygulamalarda bazı farklılıklar olabilir (Öğretmen, 2006).

Sertifikanın yayınlanması için şu gerçekleştirilir: Kullanıcı, elektronik sertifika almak için Elektronik Sertifika Hizmet Sağlayıcısı'na başvurur. Bu başvuru adımında, kimlik doğrulamanın güvenliği için yüz yüze görüşme gereklidir. Elektronik Sertifika Hizmet Sağlayıcısı, kullanıcının kimliğini doğrular ve sertifikayı yayınlar. Elektronik Sertifika Hizmet Sağlayıcısı, sertifikayı, herkese açık bir ortama (örneğin bir dizin hizmeti üzerine) aktarır.

Sertifikanın kullanılması için şu adımlar gerçekleştirilir: Kullanıcı gerçekleştirdiği işlemi ya da göndereceği mesajı kendi özel anahtarı ile imzalar ve alıcıya iletir. Alıcı mesajı alır, mesajdaki sayısal imzayı göndericinin açık anahtarıyla doğrulaması gerekir. Bunun için kullanıcının nitelikli elektronik sertifikasını Elektronik Sertifika Hizmet Sağlayıcısı'nın herkese açık dizin hizmeti üzerinden sorgular ve sertifikayı elde eder. Sertifikadaki geçerlilik süresini, nitelikli sertifika olup olmadığını ve imzalayan Elektronik Sertifika Hizmet Sağlayıcısı'nın imzasının doğruluğunu kontrol eder.

Yukarıdaki aşamalardan sorunsuz geçildikten sonra, mesajdaki elektronik imzanın alıcıya ait olup olmadığını kontrol eder. Bunu imza onaylama işlemi ile gerçekleştirir. İmza onaylanırsa alıcı, mesajı gönderenin kimliğinden, mesajın alıcıdan çıktığı şekliyle kendisine ulaştığından ve göndericinin bu mesajı gönderdiğini daha sonra inkâr edemeyeceğinden emin olur.

Bir nitelikli elektronik sertifika, özel anahtarın kaybolması veya sertifikanın geçerlilik süresinin sonuna gelmesi durumlarında iptal edilebilir. Geçici süreyle kullanımdan kaldırılacak olan sertifikalar ise askıya alınır. İptal edilen sertifika tekrar kullanılamaz. Askıya alınan sertifika, askıda olduğu süre boyunca kullanılamaz. Ancak askıda olan bir sertifika askıdan indirildiğinde normal kullanımına devam edilebilir. Elektronik Sertifika Hizmet Sağlayıcısı, iptal edilen sertifikaları Sertifika İptal Listesi (SİL) adında bir liste ile periyodik olarak yayınlar. Elektronik Sertifika Hizmet Sağlayıcısı, aynı yayınladığı sertifikalarda olduğu gibi yayınladığı SİL'leri de elektronik olarak imzalar. Bir sertifika iptal edildiğinde, bir sonraki SİL içerisinde iptal edilen sertifikaya ilişkin bilgileri yayınlanır.

SİL'de iptal edilen sertifikalar, geçerlilik süreleri dolan sertifikalar ve geçici olarak kullanımdan kaldırılan (askıya alınan) sertifikalar yer alır.

2.5. Açık Anahtar Sertifikası Özellikleri

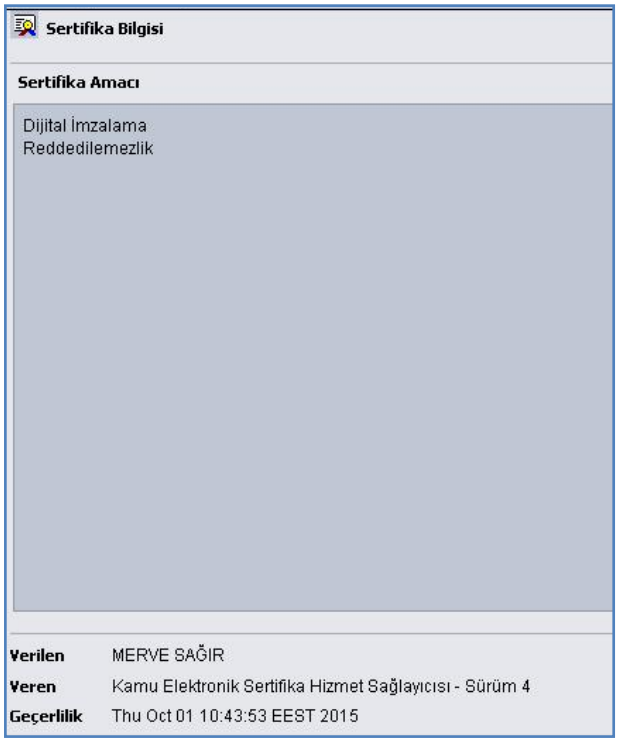
Açık anahtar sertifikaları aşağıdaki özellikleri taşır (URL-7):

- Seri numarası : Sertifikayı belirtmek için tekil olarak kullanılır.
- Başlık : Kişi ya da nesneyi belirtir.
- İmza Algoritması : İmza yaratmak için kullanılır.
- İmza : İmzanın yayıncıdan geldiğini doğrulamak için gerçek imzadır.
- Veren : Sertifikanın ve bilgisinin doğrulanmasını yapan kurum.
- Geçerlilik Başlangıç : Sertifika geçerliliğinin başladığı tarih.
- Geçerlilik Bitiş : Sertifika geçerliliğinin bitiş tarihi.
- Sertifika Amacı : Açık anahtarın amacını belirtir.
- Açık Anahtar : Açık anahtarı gösterir.
- Parmakizi Algoritması : Açık anahtarı özetleyen algoritmadır.
- Parmakizi : Açık anahtarın kısaltması olarak kullanılır.

Örnek bir sertifika içeriği Şekil 2.6'da görülmektedir.

Şekil 2.6. Örnek Sertifika

Örnek bir sertifikanın bilgisi Şekil 2.7'de görülmektedir.



Şekil 2.7. Sertifika Bilgisi

Sertifika yaşam çevrimi yönetiminde yer alan sertifikaların kaydı, yayınlanması ve iptal edilmesi ile SİL'lerin yaratılması ve yayınlanması, sertifika ve SİL'lerin

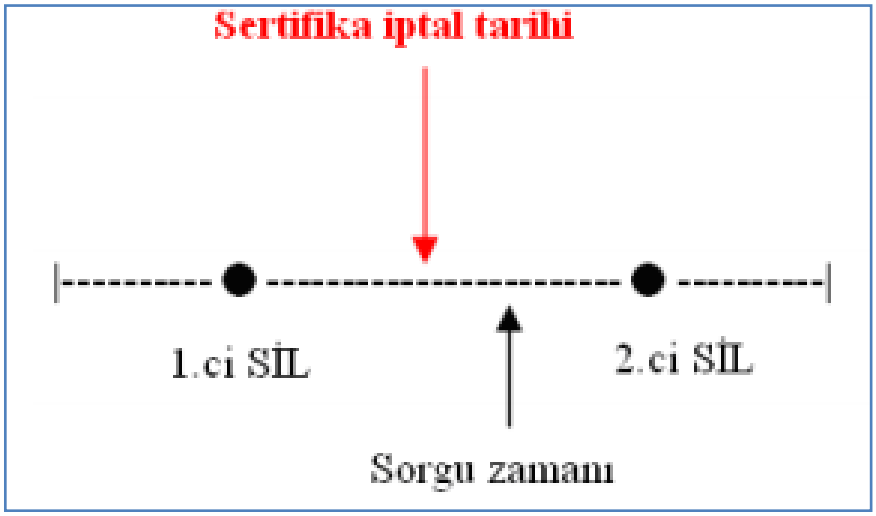
saklanması ve getirilmesi işlemlerini içerir. Bazı gelişmiş fonksiyonlar zaman damgası ve politika tabanlı sertifika validasyonunu içerir (Hunt, 2001).

Bu fonksiyonlar Açık Anahtar Altyapısında üç basit işlemi için tanımlanır:

- Sertifikalandırma açık anahtar çiftini bir nesne ile ilişkilendirme prosesidir.
- Validasyon sertifikanın geçerli olduğunun doğrulanması ve gerektiğinde iptal edilmesi prosesidir.
- Anahtar yönetimi güncelleme, yedekleme ve arşivleme.

Sertifika İptal Listeleri (SİL), X.509 tipindeki sertifikalarla birlikte ilk olarak 1988'de ITU-T (International Telecommunication Union) tarafından ortaya atılmıştır. 1993 yılında ise ikinci sürümüne erişilmiştir. SİL'ler, iptal edilmiş sertifikaların listesini taşır ve çevrimdışı olarak belirli periyotlarla üretilir. Bir SİL iptal edilmiş sertifikalara ait seri numaralarını, iptal tarihlerini, ve kendi oluşturulma ve bir sonraki güncelleme tarihlerini içerir. İsteğe bağlı olarak, sertifikaların iptal nedenlerini de içerebilir. SİL yayınlayıcısı tarafından sayısal olarak imzalanır, böylece SİL listesinin geçerliliği de kontrol edilebilir. Herhangi bir sertifikanın geçerliliği kontrol edilirken, söz konusu sertifikayı yayınlayan Sertifika Makamı'nın yayınladığı SİL'in imzası kontrol edilir, eğer imza doğruysa sorgulanan sertifikanın SİL'de yer alıp almadığı kontrol edilir. Eğer sertifikaya ait seri numarası SİL içinde bulunamazsa söz konusu sertifika geçerli kabul edilir, aksi durumda sertifika geçersizdir (Öğretmen, 2006).

SİL'den iptal kontrolü yaparken; SİL'ler Elektronik Sertifika Hizmet Sağlayıcılar tarafından belirlenen saatlerde yayınlandığı için, bir SİL'in yayınlanmasından diğer SİL'in yayınlanmasına kadar olan süre içerisinde iptal edilen sertifikaların geçerliliği bir sonraki SİL yayın tarihinden önce kontrol edildiğinde sertifikaların geçerlilik bilgisi doğru olarak görüntülenemeyecektir. Kısaca iptal edilmiş olan bir sertifika geçerli görünecektir. Bu sorun geçmiş zamana yönelik olmayan o anki sorgular için geçerlidir, geçmişe yönelik sorgu yapılıyorsa böyle bir sorun yaşanmayacaktır. Çözüm için; eğer o andaki geçerlilik kontrolü yapılacaksa geçerlilik kontrolünde OCSP kullanılması önerilmektedir.



Şekil 2.8. Bir Sertifikanın İptal Durumu (Çelebi Başçı, 2006)

Birinci SİL'in yayınlanmasından 2.ci SİL'in yayınlanmasına kadar olan süre içinde bir sertifika iptal edildiği Şekil 2.8'de görülmektedir. Hemen sonrasında bir sorgu yapılmış ve bu arada henüz yeni SİL yayınlanmadığı için sertifika geçerli sayılmıştır. Bu durumda OCSP kullanılmış olsaydı sertifikanın iptal edildiği anlaşılacak ve hatalar önlenmiş olacaktı (Çelebi Başçı, 2006).

OCSP, bir sertifikaya ait güncel iptal bilgisinin çevrimiçi elde edilmesini sağlayan bir protokoldür. OCSP, SİL yönteminin yerine veya bu yöntemle birlikte kullanılabilir, böylece sorgulanan sertifikaya ait en güncel durum bilgisi elde edilebilir.

OCSP istek ve cevap mesajları RFC 2560 standardında (Myers ve diğ., 1999) tanımlandığı gibi olmalıdır.

OCSP istemcisi bir sayısal sertifikanın kontrolü sırasında, OCSP sunucusuna bir geçerlilik durum isteği gönderir ve söz konusu sertifikayla herhangi bir işlem yapmadan önce OCSP sunucusundan gelecek olan geçerlilik durum bilgisini bekler.

OCSP isteği içinde bulunan alanlar aşağıda listelenmiştir (Öğretmen, 2006):

- Protokol sürüm bilgisi
- Hizmet isteği (Hizmeti almak isteyen)
- Sorgulan sertifikaya ait ayırt edici bir özellik (Yayıncı-Seri No, Açık Anahtarın Özeti vb...)
- OCSP Sunucusu tarafından işlenebilecek eklentiler

İstek mesajının sunucuya ulaşmasından sonra OCSP sunucusu,

- Mesaj biçiminin düzgün olup olmadığını kontrol eder.
- İstenilen hizmeti sağlar. Sorgulanan sertifika(lar) hakkında geçerlilik durum bilgisi oluşturur.
- OCSP istemcisine sertifika durum bilgisini içeren bir cevap gönderir. Aksi durumda hata mesajı üretir.

OCSP yanıtları farklı tiplerde olabilir. Bir OCSP yanıtı, yanıt tipi ve yanıt sekizlilerinden oluşur. Tüm OCSP istemci ve sunucularının desteklemesi beklenen bir OCSP yanıtı bulunmaktadır. Tüm "anlamlı" OCSP yanıtları elektronik olarak imzalanmalıdır.

Anlamlı bir yanıt mesajı aşağıdaki bileşenleri içerir:

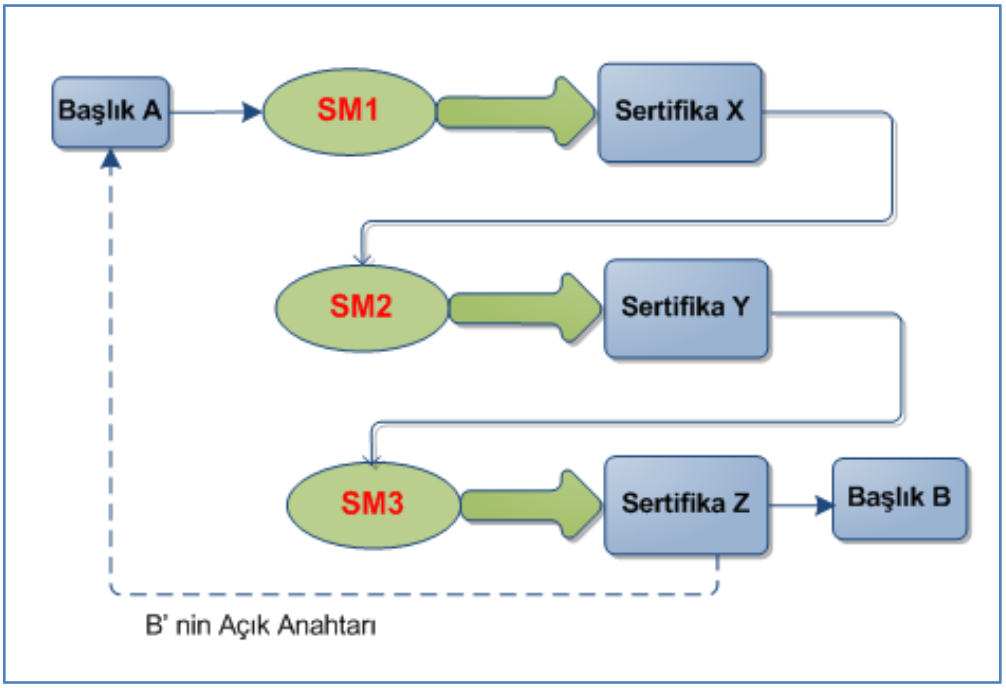
- Yanıtın sürümü
- Yanıt verenin adı
- İstek içinde yer alan tüm sertifikalar için ayrı ayrı yanıt bilgileri
- Seçime bağlı eklentiler
- İmzalama algoritması OID'i
- Yanıtın özetinden hesaplanan imza değeri

İstek içinde yer alan tüm sertifikaların yanıtlarının her birinde ise,

- Sorgulan sertifikaya ait ayırt edici bir özellik(Yayıncı-Seri No, Açık Anahtarın Özeti vb...)
- Sertifika durum bilgisi
- Yanıtın geçerli olduğu zaman dilimi
- Seçime bağlı eklentiler yer alır.

Farklı sertifikasyon makamlarından oluşan bir yapıda, sertifikasyon makamları birbirlerine güvenmelidir. Bu işleme sertifikasyon yolu doğrulanması denir.

Şekil 2.9'da, Başlık A'nın, Başlık B'nin açık anahtarına ihtiyacı vardır. Başlık A, direkt olarak SM3'ü doğrulayamadığı için SM3'ü doğrulayan SM2 ve SM2'yi doğrulayan SM1 vasıtasıyla doğrulama yapabilir. SM3, B'nin açık anahtarını bildiği için Başlık A'ya gönderebilir (Vacca, 2004).



Şekil 2.9. SM ile Sertifikasyon Yolu Doğrulanması

2.6. Açık Anahtar Altyapısı Bileşenleri

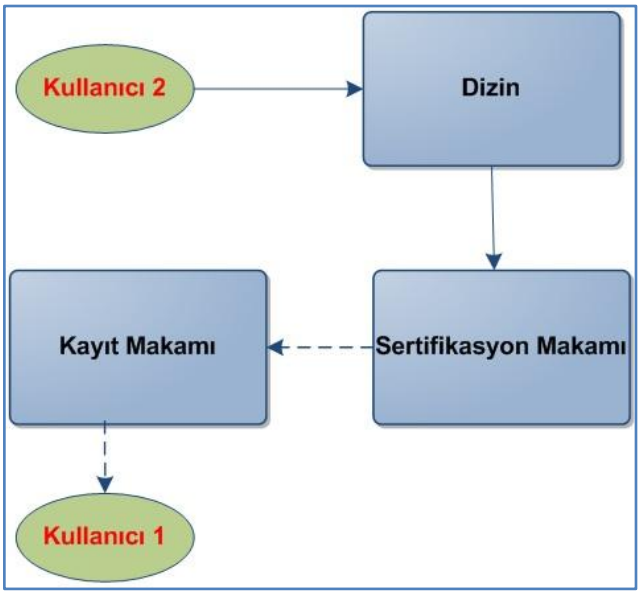
2.6.1. Sertifikasyon makamı

Sertifikasyon makamı, sertifikaları yayınlar ve iptal eder. Bir sertifikasyon makamı, tüm yaşam çevrimindeki Açık Anahtar Altyapısı fonksiyonlarını gerçekleştiren güvenilir bir makamdır. Sertifikasyon makamı; elektronik imzayla açık anahtarı bir kişi ya da kullanıcıya bağlayan sertifikaları yayınlar. Sertifikalar için bitiş tarihi kontrolünü yapar. Her sertifikanın SİL'lere göre iptal kontrolünü yapar.

Açık anahtar altyapısı implemente edilirken, kurumlar kendi sertifikasyon makamlarının, ticari sertifikasyon makamlarının ya da üçüncü parti sertifikasyon makamlarının sunduğu hizmetleri kullanabilir (Hunt, 2001).

Sertifikasyon makamları; sertifika ve SİL yayınlama gibi iki görevi yerine getirir. Aşağıda yer alan Şekil 2.10 sertifikasyon bileşenlerinin etkileşimini gösterir. Kullanıcı1 Kayıt Makamı aracılığıyla kimliğini doğrular. Kayıt makamı, sertifika isteğini açık anahtar ve diğer parametreleri güvenli iletişim yoluyla iletir. Sertifika isteği dijital olarak imzalanmış ve opsiyonel olarak şifrelenmiştir. Sertifikasyon makamı, isteğin güvenilir Kayıt makamında oluştuğunu ve sertifika isteğine bu Kayıt

makamının uygun olup olmadığını kontrol eder. İstek doğrulandıktan sonra Sertifikasyon makamı sertifika oluşturur ve imzalar (Kent, 1998).



Şekil 2.10. Sertifikasyon Bileşenleri Etkileşimleri

2.6.2. Kayıt makamı

Kayıt makamı, kayıt işlemi ve sertifika isteği kabul edilen kullanıcıların yetkilendirilmesiyle görevlidir. Bu işlemler için sertifika iptal kontrolü ve Açık anahtar Altyapısı ile etkileşimde bulunan hizmetleri sağlaması gereklidir.

Kayıt makamının işlevleri bir kişi tarafından manuel olarak gerçekleştirilebilir. Bunun için manuel prosedürlerin tanımlanması gereklidir. Yüksek güvenlikli ortamlar için manuel işletim gerekli olabilir (Nash ve diğ., 2001).

Kayıt Makamı'nın Sertifikasyon Makamı ile çalışması genelde iki şekilde olur (Gülaçtı, 2006a): KM sertifika isteği yapmadan önce gerekli bilgileri toplar ve doğruluğunu kontrol eder. Bu genelde Kayıt Makamı'na yapılan şahsi başvurularda kullanılan yoldur. Başvuran kişi kimlik ya da ehliyet gibi bir belge ile kim olduğunu Kayıt Makamı'na kanıtlar. Bu bilgilerle oluşturulan sertifika isteği Sertifika Makamı'na iletilir.

Kişi sertifika isteğini elektronik ortamda yapar. Örneğin bir e-posta adresinin kendisine ait olduğunu iddia eder ve sertifika ister. Sertifika Makamı bu isteği alır ve

istek içindeki bilgileri doğruluğunu kontrol etmesi için Kayıt Makamı'na gönderir. Kayıt Makamı onay verdikten sonra SM sertifika isteğini cevaplar.

2.6.3. AAA protokolleri

Açık anahtar altyapısı protokolleri AAA kullanıcıları ve yönetim nesneleri arasındaki etkileşimi sağlar. Mesela, yönetim protokolü sertifikasyon makamı ile anahtar çiftinin ilişkili olduğu istemci arasında kullanılır. Yönetim protokolü kullanıcı ya da istemci sistem kaydı bilgisini taşır veya sertifika iptali için isteği alır. Yönetim protokollerinin kullandığı mesaj formatı ve iletim şekilleri RFC 2510 standardında ve sertifika politikaları ve uygulama şekilleri RFC 2527 standardında anlatılmıştır (Hunt, 2001).

Yaygın olarak kullanılan yönetim protokolleri aşağıda listelenmiştir (Gutmann, 2003):

- PKCS #10 Sertifika Talep Standardı ve SSL
- PKCS #10 Sertifika Talep Standardı ve PKCS #7
- Sertifika Yönetim Protokolü (CMP)
- Certificate Management Using CMS
- Simple Certificate Enrollment Protocol (SCEP)

3. GELİŞTİRİLEN UYGULAMA

Elektronik imza uygulaması gerçekleştirilmesinde gerekli altyapı için Açık Anahtar Altyapısı, üçüncü parti güvenilir makamlar (Trusted Third Party) ve ESHS (Elektronik Sertifika Hizmet Sağlayıcı) gereklidir.

Ülkemizde elektronik imzalama işlemlerini yapan TÜBİTAK BİLGEM tarafından geliştirilmiş İmzager, E-Güven firması tarafından geliştirilmiş İmzala-Gönder ve E-Tuğra firması tarafından geliştirilmiş PDF İmzalama yazılımları bulunmaktadır.

TÜBİTAK BİLGEM tarafından geliştirilmiş İmzager uygulaması, Türkiye'de elektronik imzanın yaygınlaşması amacıyla üretilmiş bir yazılımdır. İmzager gelişmiş e-imza özelliklerini ücretsiz olarak kullanıma sunması açısından kendi alanında bir ilktir. İmzager'i iki türlü kullanmak mümkündür:

Güvenli Elektronik İmza oluşturma: Bu işlem sadece Türkiye'de faaliyet gösteren ESHS'lerin vermiş olduğu nitelikli elektronik sertifikalar kullanılarak yapılabilir.

Güvenli Elektronik İmza görüntüleme: Bu işlemi İmzager yazılımını bilgisayarına kuran herkes yapabilir (URL-8).

İmzala-Gönder uygulaması, imza gerektiren kurumsal dokümanların kontrol, onay ve imza yetkilileri arasında otomatik olarak dolaştırılarak elektronik veya mobil imza ile imzalanmasını ve başka sistemlere aktarılmasını sağlayan bir web uygulamasıdır (URL-9).

E-Tuğra PDF İmzalama Yazılımı, Web Üzerinde bir PDF formunu veya istemci üzerinde çalışan bir PDF dokümanı imzalayabilme özelliğine sahiptir (URL -10).

TÜBİTAK BİLGEM tarafından geliştirilmiş ESYA E-imza kütüphaneleri, güvenliği ve standartları belirlenmiş, kullanımı kolay arayüzleriyle, imzalama işlemlerinin hızlı ve güvenli bir şekilde yapılmasına imkan verir. Yazılımlara e-imza

entegrasyonu yapılabilmesi için Java ve .NET platformlarında yazılım kütüphaneleri geliştirilmiştir.

Temel imza (BES), zaman damgalı imza (ES-T), ilkeli imza (EPES), uzun dönemli imza (ES-X) ve arşiv imzası (ES-A) elektronik imza formatlarına destek vermektedir.

ESYA E-imza kütüphanesi ile belge üzerindeki tüm imzaların kontrol bilgileri ayrı ayrı tutularak imzaların bağımsız olarak işlem görmesi sağlanır. Böylece belge üzerindeki imzalardan biri ya da birkaçı geçersiz olduğunda bile geçerli imzalar üzerinden işlem yapılabilmesi mümkün olur. Bir belgeye birden çok seri/paralel imza ekleme imkanı sağlanır (URL-11).

Gerçekleştirilen çalışmada, Android işletim sistemi üzerinde Açık Anahtar Altyapısı kullanılarak elektronik imzalama uygulamasının geliştirilmesi anlatılmıştır. Android mobil işletim sistemi kullanımının giderek artması, elektronik ortamdaki işlemleri mobil cihaz ve cep telefonları üzerinde kolaylıkla yapılabilir hale getirmiştir. Android işletim sisteminin açık kaynak kodlu olması kullanım esnekliği sağlamaktadır. Geliştiriciler ihtiyaçlara göre Android işletim sistemi üzerinde uygulama geliştirerek kullanıma sunmaktadır.

3.1. Elektronik İmza Uygulaması Sistemi Altyapısı

Kriptografik işlemlerin güvenli bir ortamda yapılması amacıyla Pfx (Personal Information Exchange) dosyalarına ya da akıllı kartlara ihtiyaç duyulmaktadır. Pfx dosyaları ve akıllı kartlar özel anahtarın dışarıdan erişilmesine izin vermeyerek Açık Anahtar Altyapısı için gerekli güvenliği sağlarlar.

Pfx dosyaları ve akıllı kart içinde kullanıcının sertifikaları, özel anahtarları ve açık anahtarları bulunmaktadır. Her sertifikaya ait bir açık anahtar ve bir özel anahtar bulunmaktadır. Sertifikalar ve açık anahtarlar Pfx dosya içinden okunabilmektedir. Kriptografik işlemler, Pfx dosyasından özel anahtar çıkarılarak yapılır. Pfx dosyaları parola korumalı oldukları için işlemler güvenli bir şekilde yapılmış olur.

Pfx dosyaları, kriptografik öğeleri tek dosyada birleştiren arşiv formatıdır. İmzalama ve şifreleme işlemlerinde kullanılan anahtar ve sertifikaların saklanmasını sağlarlar.

Genellikle, X509 sertifika ile özel anahtarı ve dosya içinde bulunan tüm öğeler arasındaki güven ilişkisini tutar. Pfx uzantıları, P12 olarak da gösterilebilir (RSA Laboratories, 2012).

Uygulamada, özet alma işlemi SHA1 algoritması kullanılarak gerçekleştirilmiştir. SHA (The Secure Hash Algorithm), ulusal güvenlik ajansı NSA (National Security Agency) tarafından geliştirilmiş ve ilk olarak 1993 yılında Amerika'da standart olarak kabul edilmiştir. SHA özetleme algoritmasında bulunan önemli güvenlik açığı sebebiyle SHA-1 algoritması yayınlanmıştır. SHA-1 özetleme algoritması, Ronald L. Rivest'ın geliştirdiği MD4 ve MD5 mesaj özetleme algoritmalarının çalışma prensibini temel almaktadır (Bryson ve Gallagher, 2014).

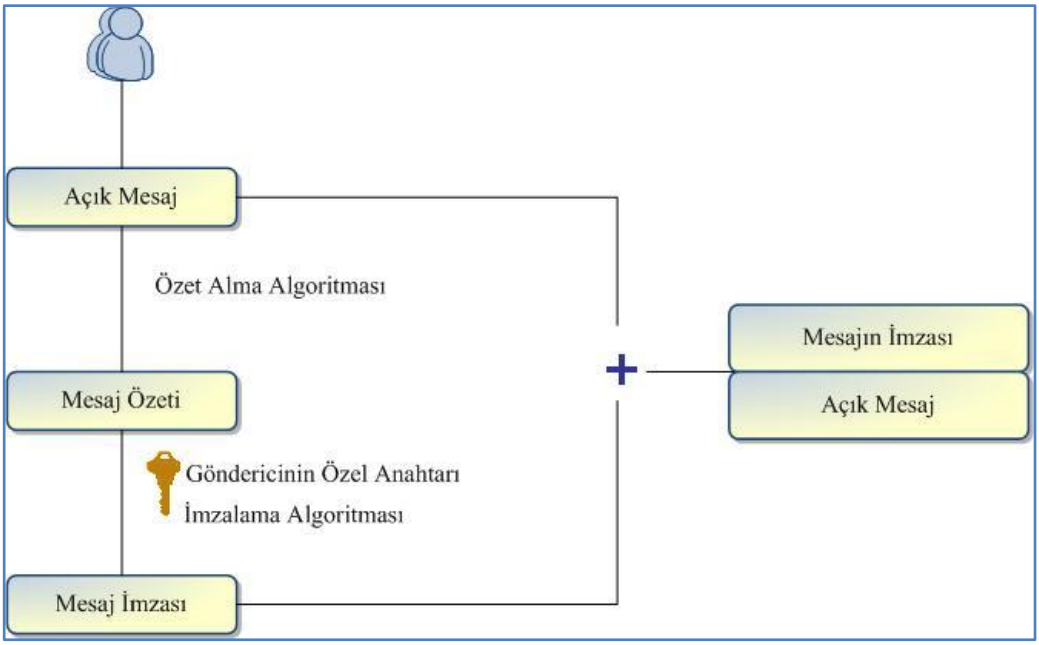
Uygulamada kullanılan SHA-1 özetleme algoritmasına en fazla 2^{64} uzunluğunda mesaj girdi olarak verilebilir ve çıktı olarak 160 bitlik mesaj özeti üretilir. Özet algoritmaları sabit çıkış uzunluğu üretirler. Özeti alınacak mesaj 512 bitlik bloklara ayrılır ve gerekirse son bloğun uzunluğunu 512 bite tamamlanır. Mesajdaki küçük değişiklikler bile özette büyük değişikliklere yol açabilir. SHA-1 algoritmasının iteratif bir yapısı vardır. Her iterasyonda bir sıkıştırma fonksiyonu kullanır. Bu fonksiyon, mesajın 512 bitlik bloğunu alır ve 16 bitlik kelimelere (m0, m1, ..., m15) çevirir. Daha sonra bu kelimeler, mi = (mi-3+ mi-8+ mi-14+ mi-16) <<1 fonksiyonu kullanılarak 2560 bite genişletilir ve her biri 20 matematiksel fonksiyon içeren 4 tur çalıştırılır ve 160 bitlik mesajın özeti elde edilir (Çalık ve diğ., 2006).

Özet alma fonksiyonları kriptografik tek yönlü fonksiyonlardır. Bu fonksiyonlar tek yönlü oldukları için veri bütünlüğü ve kimlik doğrulaması ile ilgili uygulamalarda temel haline gelmiştir. Bir mesajın özetini elde etmek çok kolaydır, bir özetten asıl mesajı çıkarmak ise çok zordur. Elektronik imza uygulamalarında hız büyük önem taşıdığı için önerilen algoritmaların çoğunda tüm mesaj yerine mesajın özeti imzalanır.

Uygulamada imzalama işlemi için RSA algoritması kullanılmıştır. RSA, en yaygın kullanılan asimetrik kriptografi algoritmasıdır. Adını, algoritmayı bulan R. Rivest, A. Shamir, L. Adleman isimlerinin ilk harflerinden alır. RSA algoritması, elektronik imzalama yöntemi olarak kullanılır. Çarpanlara ayırma problemini temel alır (Rivest ve diğ., 1978).

3.2. Elektronik İmza Uygulamasının Çalışması

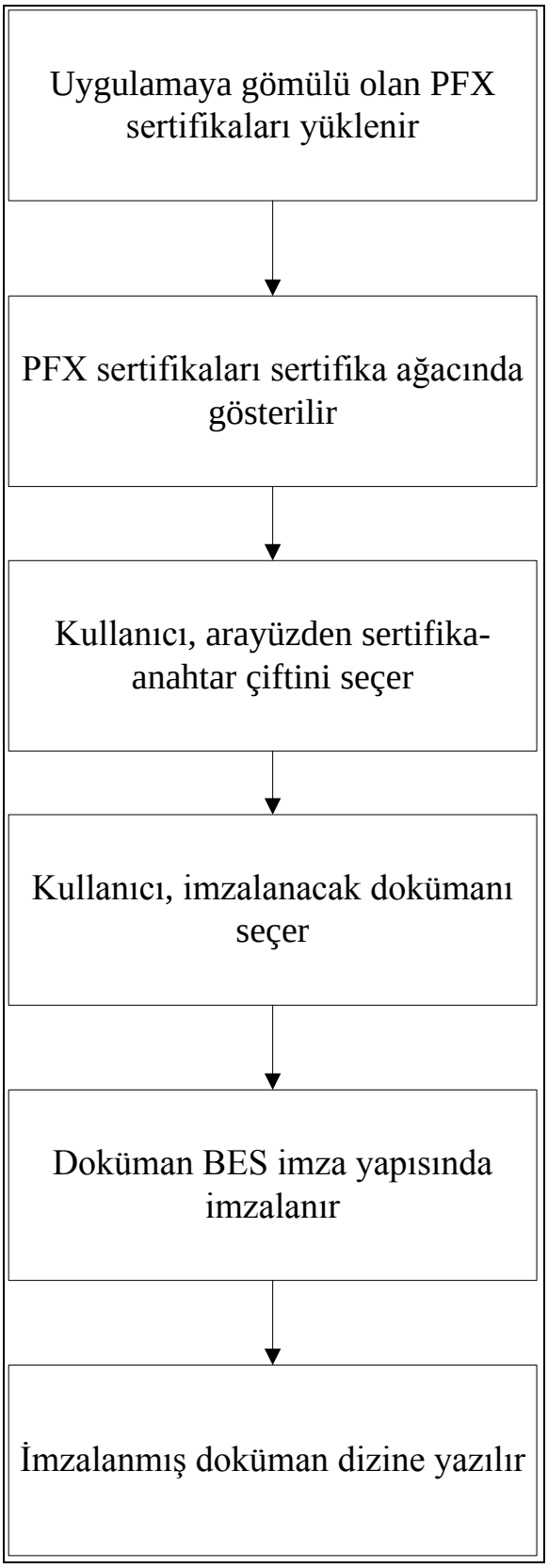
Geliştirilen uygulamada, uygulamaya gömülü olarak gelen sertifika dosyası ile kullanıcının seçtiği dosya elektronik olarak imzalanmaktadır. Uygulamanın çalışma mantığı Şekil 3.1'de görülmektedir. Uygulamanın çalışma adımları aşağıda yer almaktadır.



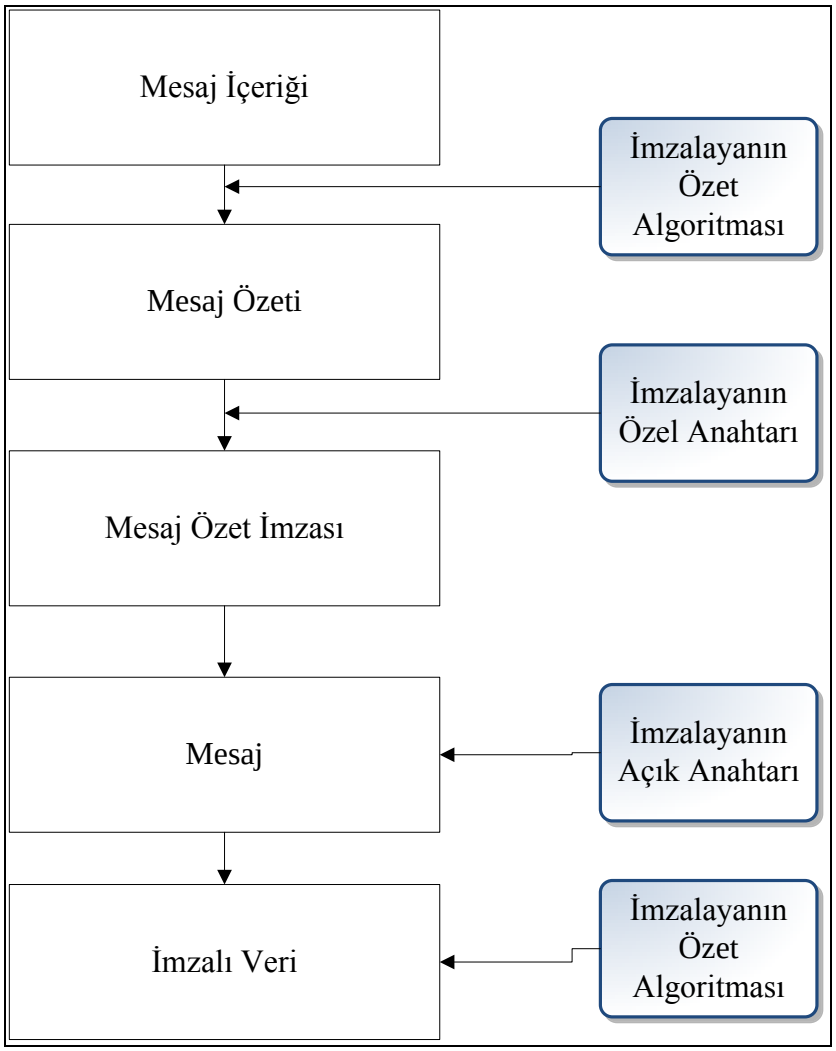
Şekil 3.1. Uygulamanın Çalışma Mantığı

Uygulama, main.xml dosyasının çalıştırılmasıyla başlar. Android uygulamasında Raw klasörü altında bulunan lisans dosyası yüklenmesi işlemi gerçekleştirilir. Uygulamaya gömülü olan Pfx anahtar dosyasındaki sertifikalar yüklenir. Pfx dosyaları, parola korumalı dosyalar oldukları için kriptografik işlemler güvenli bir şekilde gerçekleştirilebilir. Arayüzde kullanıcıya Pfx dosyası içinde yer alan sertifikalar sertifika ağacında gösterilir. Kullanıcı, buradan işlem yapacağı sertifikayı seçer. Ardından kullanıcı imzalama yapacağı dokümanı arayüzden seçer. Kullanıcının seçmiş olduğu sertifikayla ilişkili özel anahtar ile Doküman BES imza formatında imzalanır. BES imza formatı tüm elektronik imza formatlarına temel teşkil eder. İmza yapısının iskeletini oluşturur. İmzalanmış veri yapısı, RFC 3852 CMS(Cryptographic Message Syntax) (Housley, 2004) standardında açıklanmıştır. Elektronik imza ile imzalanmış doküman cep telefonu ya da tablet bilgisayar dizinine yazılır.

Uygulamanın çalışma adımları Şekil 3.2'de yer almaktadır.



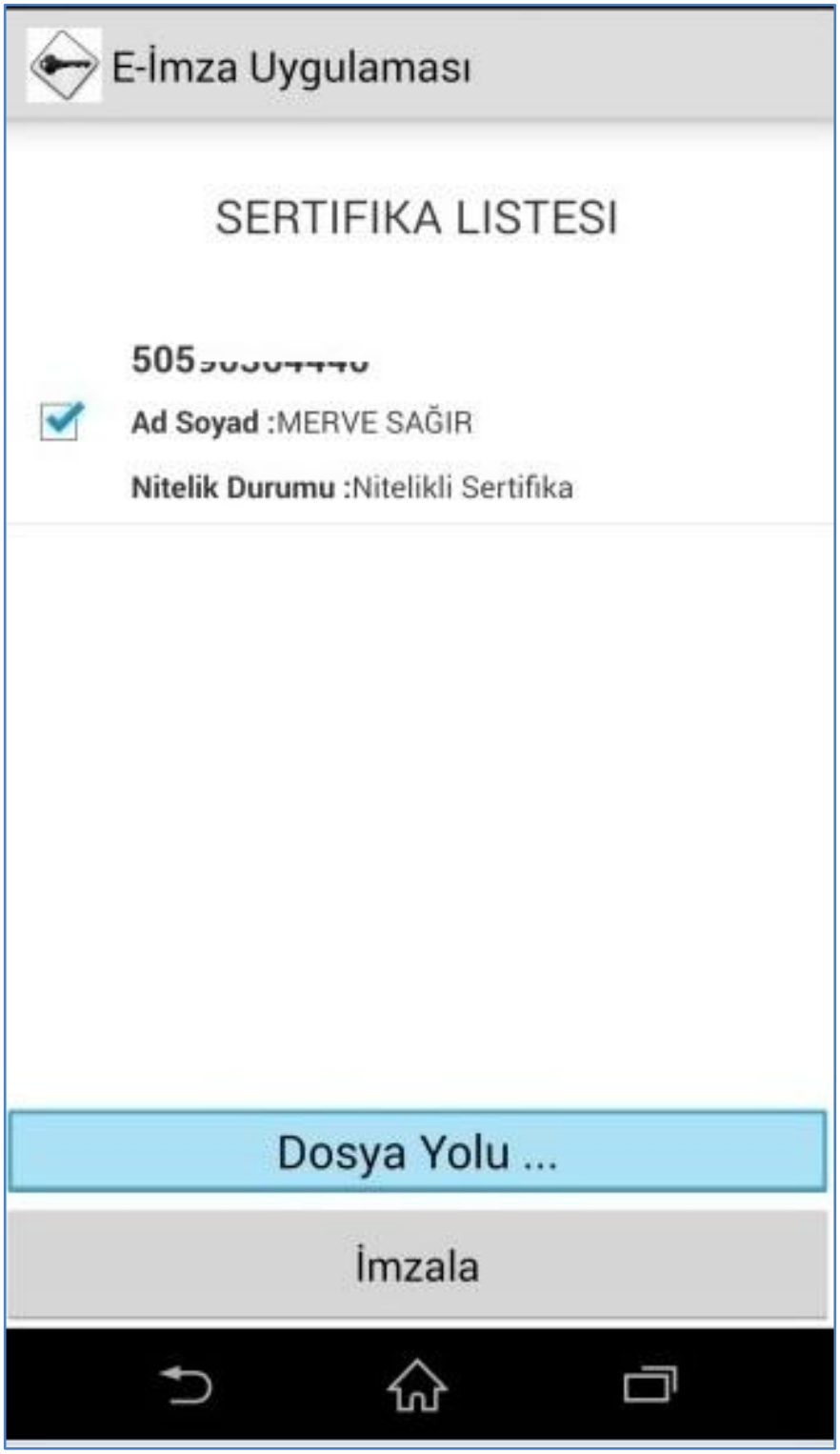
Şekil 3.2. Uygulamanın Çalışma Adımları



Şekil 3.3. Elektronik İmzanın Oluşturulması Akışı

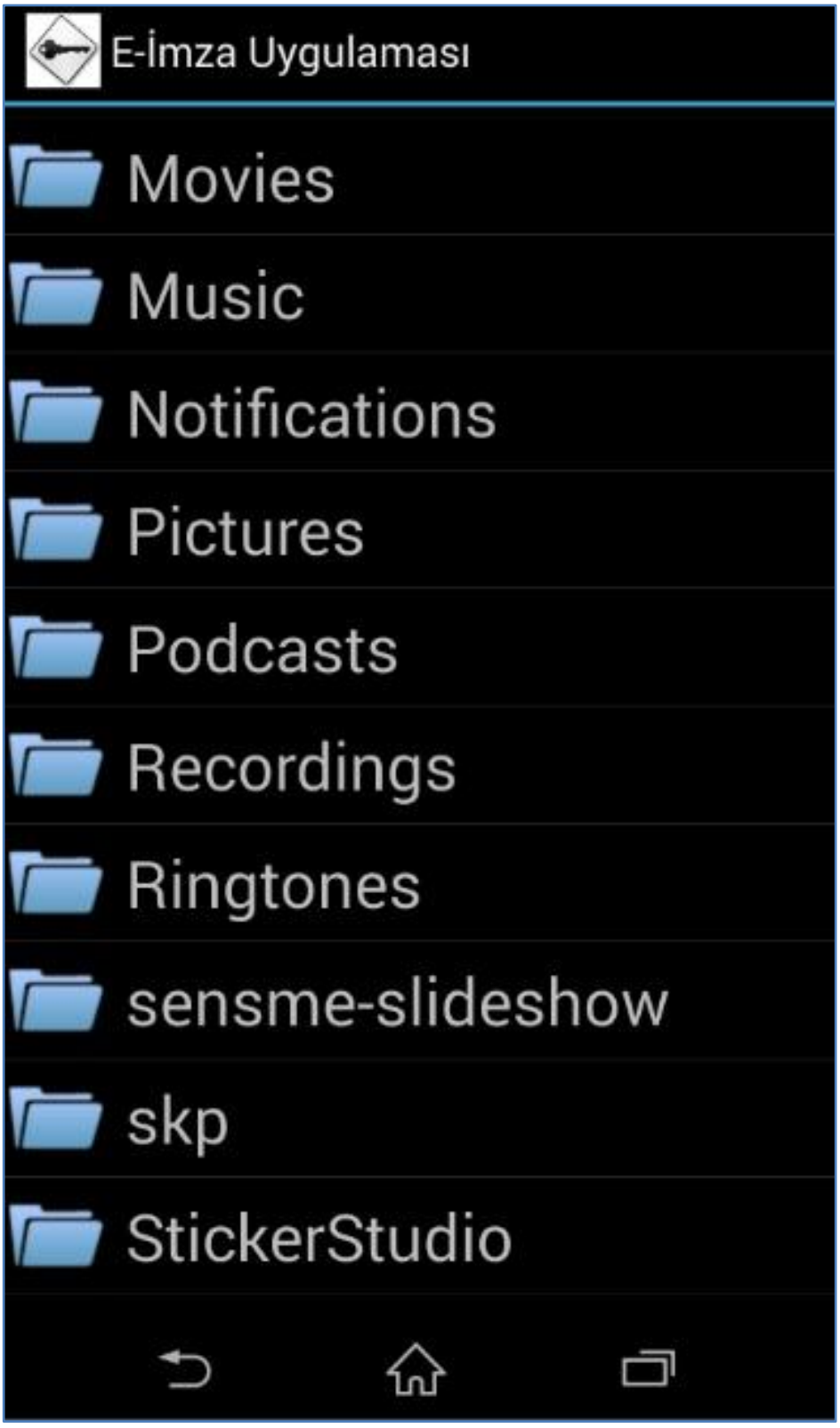
Elektronik imza oluşturulması akışı Şekil 3.3'te gösterilmiştir. Uygulamada, arayüzden seçilen sertifikaya ait imzalayıcı oluşturulur, imzalama algoritması set edilir ve imzalayıcıya bildirilir. İmzalanacak dosya içeriği alınır. İmzalama API(Application Programming Interface) sine imzalama zamanı, imza formatı, sertifika ile imzalayıcı bilgileri gönderilir. Kullanıcının seçmiş olduğu sertifikayla ilişkili özel anahtar ile doküman BES imza formatında imzalanır. BES imza formatı tüm elektronik imza formatlarına temel teşkil eder. İmza yapısının iskeletini oluşturur. İmzalanmış veri yapısı, RFC 3852 CMS (Cryptographic Message Syntax) (Housley, 2004) standardında açıklanmıştır.

Uygulamanın çalışmasını gösteren Şekil 3.4'te uygulamaya gömülü olan PFX dosyasına ait sertifika ekranda görülür. Sertifikanın ait olduğu kişinin TC numarası, adı ve soyadı ile sertifikanın nitelik durumu ekranda gösterilir.



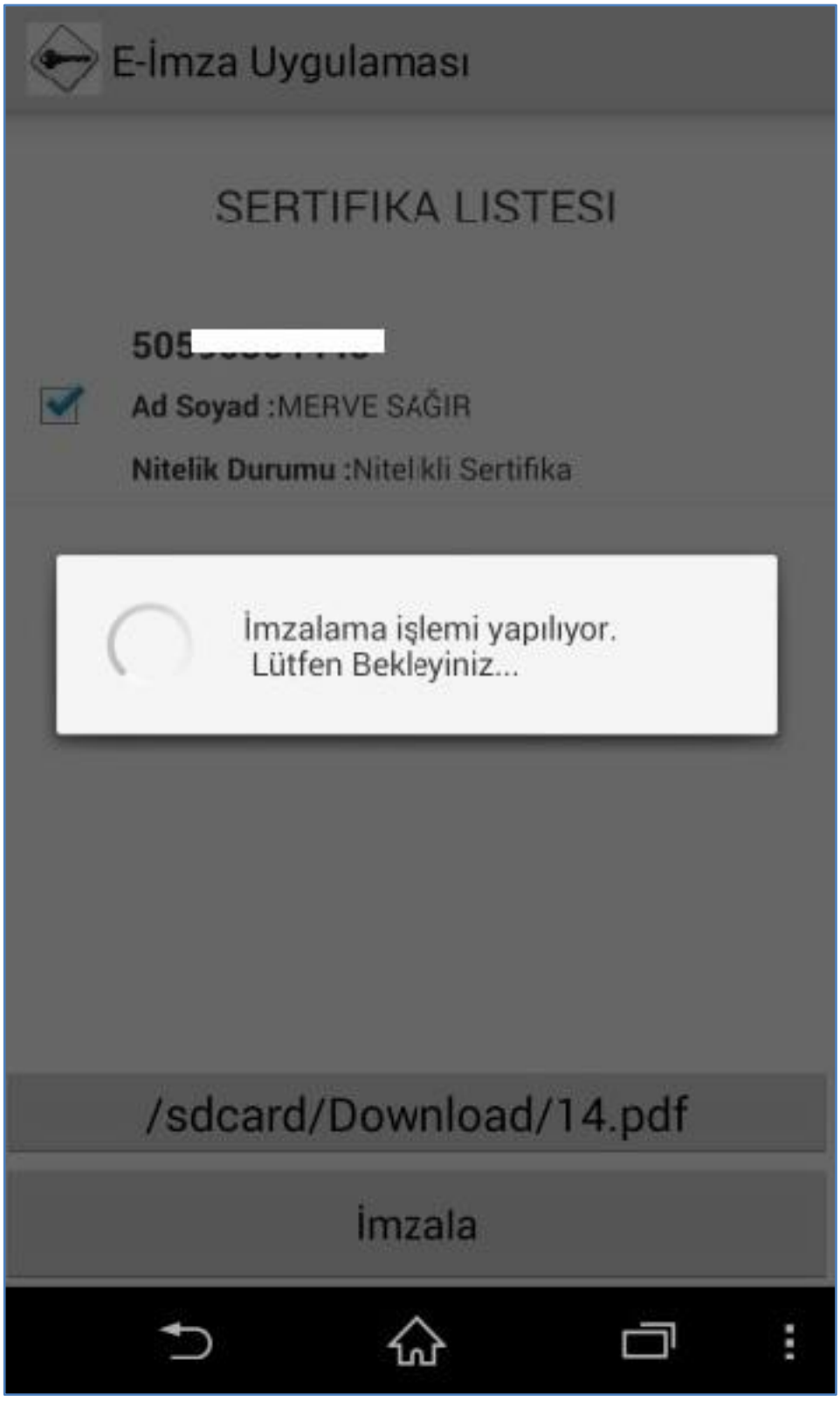
Şekil 3.4. Sertifikanın Gösterilmesi Ekranı

Uygulamanın koştuğu cep telefonu ya da tablet üzerinden "Dosya Yolu ..." butonuna basılarak Şekil 3.5'te görülen ekrandaki dosya dizin yapısı görüntülenir.



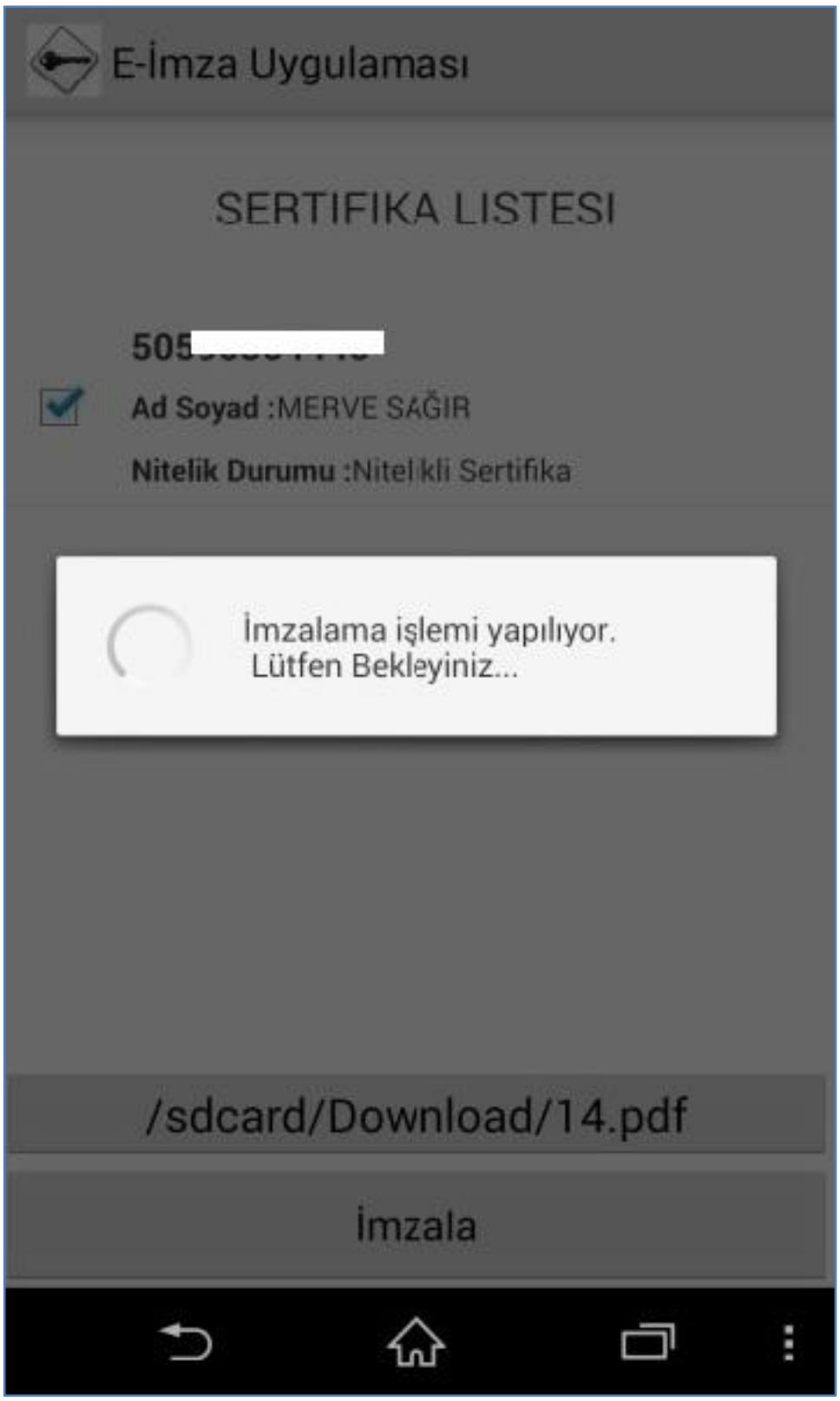
Şekil 3.5. Dosya Dizin Yapısının Gösterilmesi Ekranı

Ekrandaki "Dosya Yolu..." butonunda imzalanacak dosyasının yolu yazar. Şekil 3.6'da görüldüğü gibi "İmzala" butonuna basılır.



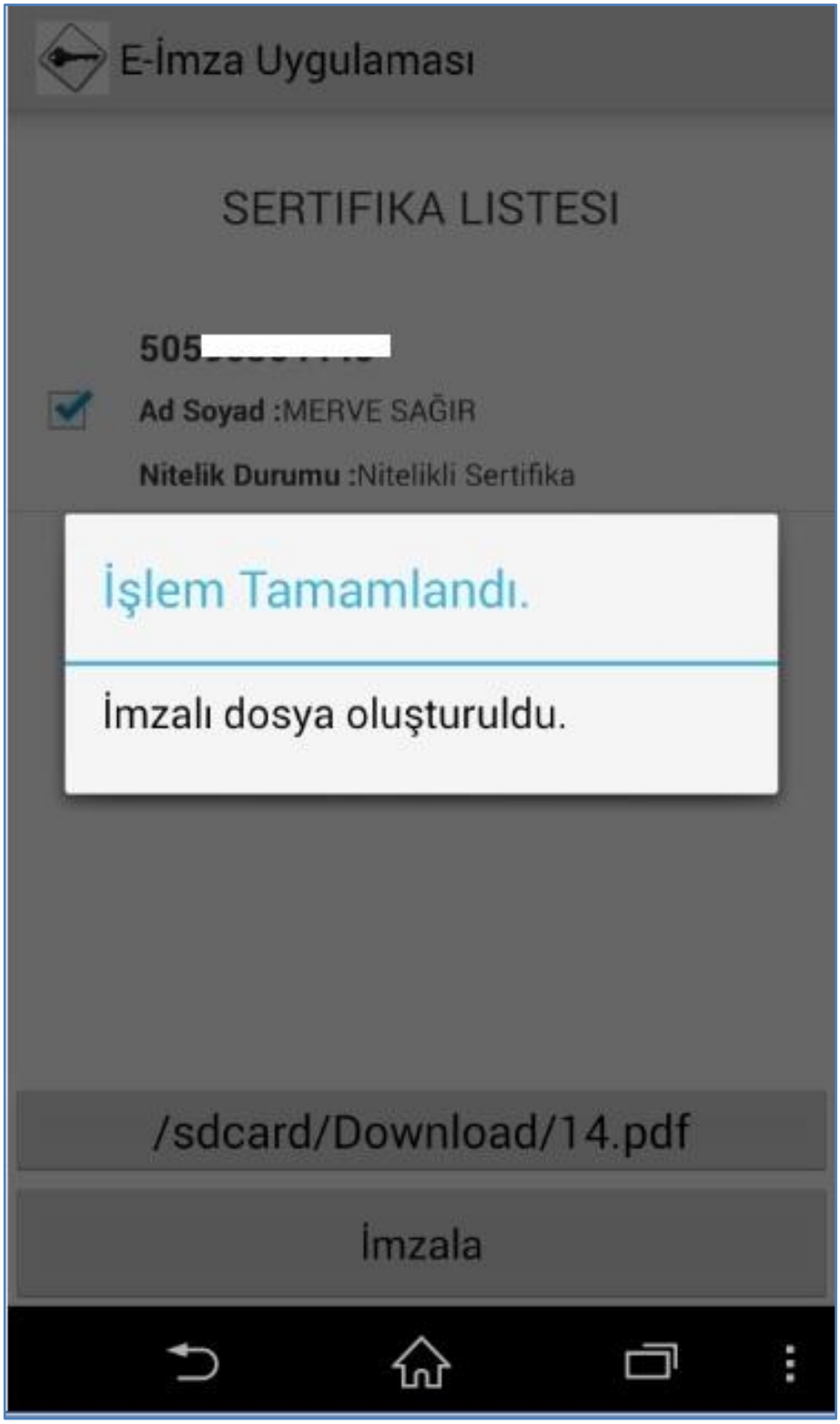
Şekil 3.6. İmzalanacak Dosyanın Gösterimi Ekranı

İmzalama işlemi yapılırken Şekil 3.7'de görüldüğü gibi kullanıcı bilgilendirilir.



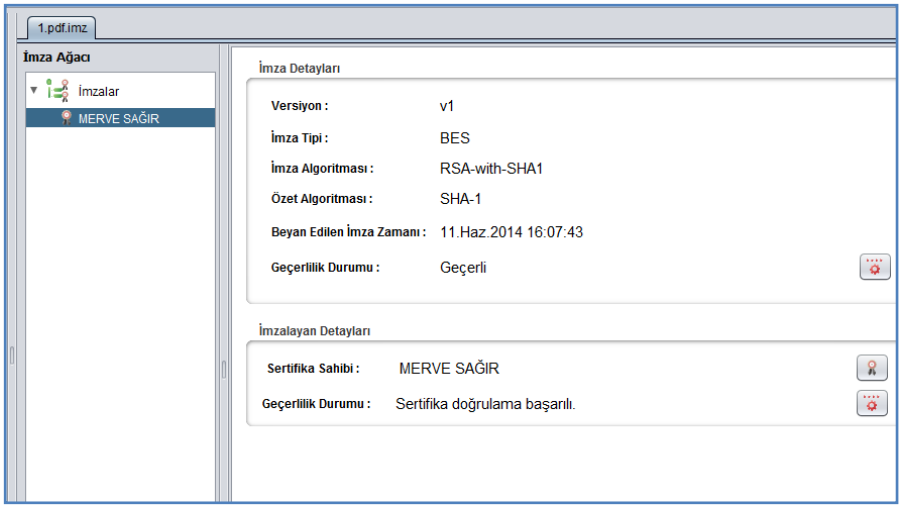
Şekil 3.7. İmzalama İşleminin Yapılması Ekranı

İmzalama işlemi tamamlandığında Şekil 3.8'deki gibi işlemin tamamlandığına dair mesaj verilir. İmzalanan dosya, imzalama işlemi için seçilen dosya ile aynı konumda ".imz" uzantılı olarak oluşturulur.



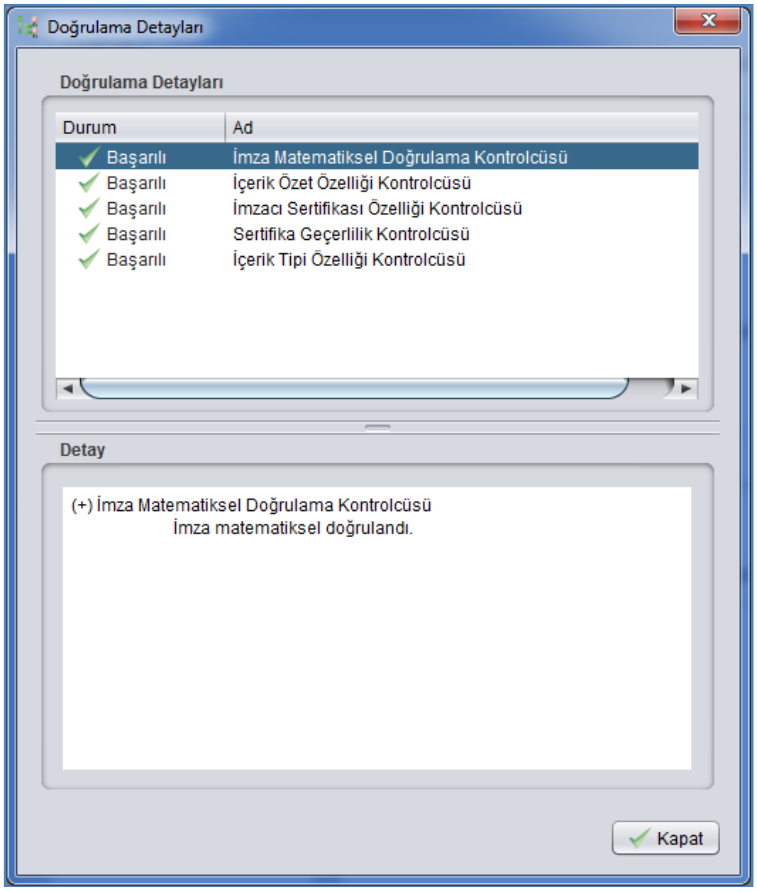
Şekil 3.8. Dosyanın İmzalanması Ekranı

İmzalanan ".imz" uzantılı dosya Windows işletim sistemi üzerinde çalışan İmzager uygulaması ile üzerine çift tıklanarak açılır. Dosyanın kim tarafından imzalandığı, imza detayları ve imzalayan detayları Şekil 3.9'da kullanıcıya gösterilir.



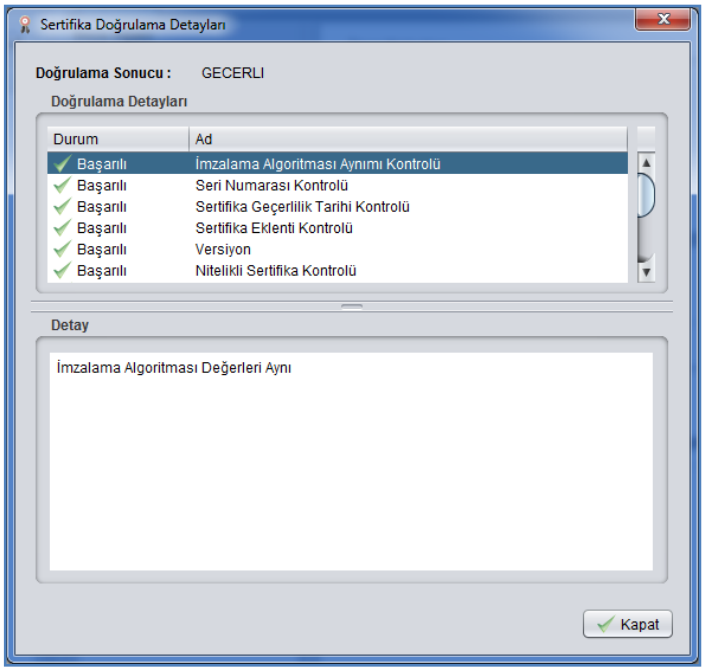
Şekil 3.9. İmzalanan Dosyanın İmzager Uygulaması ile Gösterimi

İmzalanan dosyanın doğrulama detayları Şekil 3.10'da gösterilir.



Şekil 3.10. İmzager Uygulaması ile Doğrulama Detayları

Elektronik imzalama işleminin gerçekleştirildiği imzalama sertifikasının doğrulama detayları Şekil 3.11'de görülür.



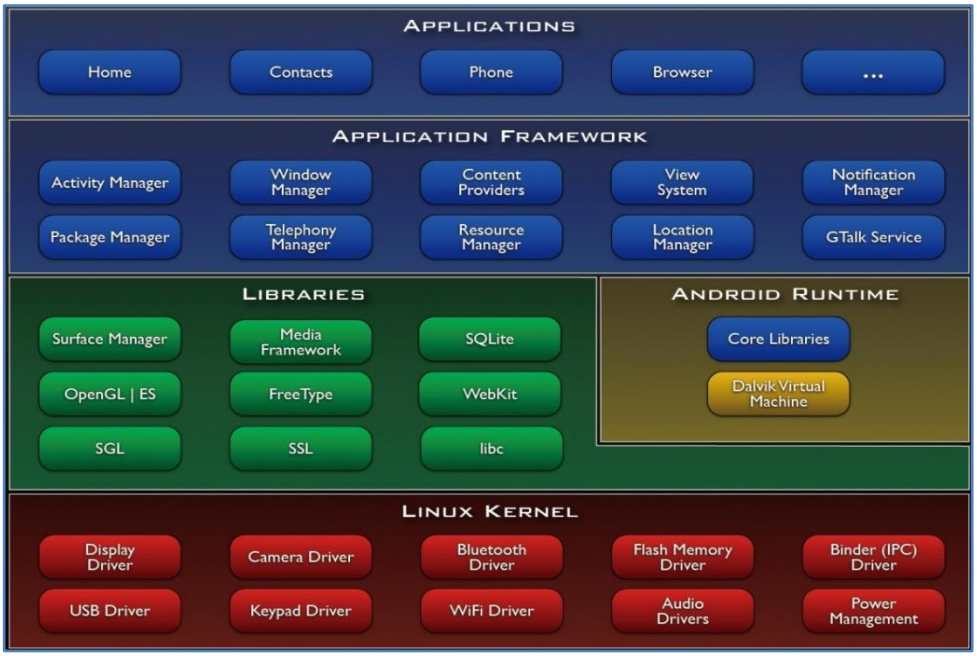
Şekil 3.11. Sertifika Doğrulama Detayları

3.3. Elektronik İmza Uygulaması Kullanılan Teknolojiler

Android Linux tabanlı özellikle dokunmatik telefon ve tablet bilgisayarlar için geliştirilmiş bir işletim sistemidir. Android dünya çapında en fazla market payına sahiptir. Geliştiriciler için özgür bir geliştirme ve dağıtım olanağı sağlar. Android işletim sistemi Assembly, Java, C++ ve C kullanılarak geliştirilmiştir. Google Playstore üzerinde 1 milyondan 200 binden fazla uygulama bulunmaktadır (URL-13).

Üzerinde Android koşan cihazlar en az 32MB RAM, 200 MHz 4 çekirdekli işlemci ve 2 GB RAM kapasitesine sahip olmalıdır. Donanım platformu ARM mimarisidir.

Aşağıda Şekil 3.12'de Android yapısı görülmektedir (URL-14).



Şekil 3.12. Android Yapısı (URL-14)

Android işletim sistemi beş kısımdan oluşur (Shu ve diğ., 2009) (URL-15).

Uygulamalar: Frameworkun en üstünde bulunan email istemcisi, SMS uygulaması, takvim harita uygulaması, web tarayıcı gibi uygulamalardır.

Uygulama Frameworku: Android işletim sisteminin geliştiriciye görünen kısmıdır. Geliştiricinin framework üzerindeki komponentleri kullanmasına uygun bir mekanizma vardır. Uygulamaya ait kaynakların yönetimi, uygulamalar arası veri paylaşımının yönetimi, uygulamaların yaşam döngülerinin yönetilmesi ve uygulamanın donanım isteklerinin yönetilmesi gibi işlemler bu framework üzerinde gerçekleştirilir.

Kütüphaneler: Android sisteminin çeşitli komponentleri tarafından kullanılan C/C++ kütüphanelerini içerir. Sistem kütüphaneleri, mp3, mpeg4vejpg gibi ses ve görüntü formatları için medya kütüphaneleri, veri tabanı için sqlite kütüphaneleri gibi temel kütüphaneleri bulundurur.

AndroidRuntime: Android; temel Java kütüphanelerinin hafıza yönetimi, donanım sürücüleri gibi alt seviye işleri yürütmek için sağladığı fonksiyonelliği sağlayan temel kütüphanelere sahiptir. Her Android uygulaması işletim sistemi tarafından verilen özel prosese sahiptir ve bu prosesin Dalvik sanal makinasında karşılığı vardır. Dalvik, Java uygulamaları çalıştırmak üzere Google tarafından geliştirilmiş bir sanal

makinedir. Dalvik sanal makinası, mobil aygıtlar gibi küçük cihazlarda daha performanslı çalışmak üzere tasarlanmıştır. Linux kerneli birden fazla Dalvik sanal makinasını çalıştırabilir.
Linux Kernel; Android OS, bellek yönetimi, işlemlerin yönetimi, güvenlik ve sürücü hizmetleri gibi temel işletim sistemi görevlerini Linux Kernel 2.6 ile gerçekleştirmektedir. Çekirdek ayrıca uygulamalar ve tüm donanımlar arasında donanım katmanı olarak davranır.

Android uygulamaları dört temel bileşenden oluşur. Temel bileşenler şunlardır; Activity, Intent Receiver, Service ve Content Provider. Tüm uygulamalar dördünü içermek zorunda değildir ancak bir uygulama bunların kombinasyonuyla yazılır. Geliştirici uygulama için hangi bileşenlere ihtiyaç olduğunu belirlediğinde, AndroidManifest.xml dosyasında tanımlamaları yapar. AndroidManifest.xml dosyasında uygulamadaki bileşenlerin kapasiteleri ve gerekleri tanımlanır (Haseman, 2008) (URL-16).

Android uygulamaları Java programlama diliyle yazılır ve Dalvik sanal makinesi üzerinde çalışır. Uygulama uzantısı .apk'dır.

Her Android uygulaması ayrı bir linux prosesi şeklinde çalışır. Bir uygulama çalışmak istediğinde Android yeni bir proses yaratır ve uygulama kapatılana kadar bu proses altında çalışır. Tabi sistem kaynakları yeterli olmadığı durumda Android uygulamalanın kapatılması beklenmeden prosesi sonlandırabilir.

Aktiviteler en çok kullanılan Android bileşenidir. Bir aktivite genellikle tek bir ekrandır. Her aktivite Activity temel sınıfını genişleten bir sınıf olarak tanımlanır. Sınıflar fonksiyonel işlemlerin sağlandığı bir kullanıcı arayüzü sunarlar. Mesela bir mesajlaşma uygulamasında, mesaj gönderilecek kişiyi seçmek için kişileri listeleyen bir aktivite sahip olabilir. Diğer aktivite seçilen kişiye mesaj yazmayı sağlarken bir diğer aktivite de gönderilen mesajları gösterebilir. Tüm aktiviteler birlikte çalışıyor gibi görünse de her aktivite birbirinden bağımsız çalışır ve Activity sınıfının alt sınıflarıdır.

Bir uygulamada bir aktiviteye ya da mesajlaşma uygulamasındaki gibi bir çok aktivite olabilir. Genelde uygulamalar ilk açıldıklarında aktif olan bir aktivite ve o aktivite aracılığı ile aktive edilen diğer aktiviteler şeklinde çalışır.

Android, Intent adı verilen ekrandan ekrana geçmeyi sağlayan bir özel sınıf kullanır. Intent üzerinde uygulamanın ne yapmasını istediğimizi belirtiriz. Intent veri yapısındaki iki önemli durum; aktivitenin ne olacağı ve verinin nasıl davranacağını belirtmemizdir.

Intent Receiverlar, dış olaya bağlı olarak uygulama kodunun ne yapması gerektiğini belirtmek için kullanılır. Mesela; batarya zayıf, sistem dili değiştirildi gibi sistem tarafından üretilen ya da diğer yazılımlar tarafından üretilen mesajları dinlemek ve yanıt vermek için kullanılırlar. Intent receiverların bir arayüzü yoktur, sadece kullanıcıyı bilgilendirirler. AndroidManifest.xml dosyasında tanımlanırlar, ayrıca Context.registerReceiver metodunu kullanarak da tanımlanabilirler. Kullanıcıyı bilgilendirme aşamasında yeni bir aktivite başlatabilirler ya da NotificationManager'i kullanarak kullanıcıyı bilgilendirebilirler. Bilgilendirme telefonu titreterek ya da bir ses çalarak olabilir.

Servislerin görsel bir arayüzü yoktur, servisler genellikle arka plan işlerini gerçekleştirmek için kullanılırlar. Bir müzik çaları düşünelim, listeden bir şarkıyı seçeriz dinlemeye başlarız. Fakat bu sırada başka işlerimizi de halletmek isteriz (tarayıcıda gezmek gibi) Bunun için müzik çaların ekranını kapatırız (aktiviteyi sonlandırırız). İşte servisler tam bu sırada işimize yarıyor. Aktivite kapatılırken arka planda müziğin çalması için servisleri kullanırız.

Servisler arayüz olmadan, uzun süre çalışan koddur. Müzik çalar uygulaması bunun için iyi bir örnektir. Uygulamadan listeden bir şarkı sürekli arka planda çalabilir. Kullanıcının bir şarkı seçip, çalması için bir ya da daha fazla aktivite çalışır. Servis arka planda çalan müziği Context.startService metoduyla başlatır. Sistem müzik bitene kadar servisi çalıştırır. Servise Context.bindService metoduyla bağlanılır. Servise bağlanınca arayüz aracılığıyla servisle iletişim kurulur. Böylece müziği durdur, geri sar gibi özellikleri kullanılır.

Uygulamalar, verilerini dosyalarda, SQLite veritabanında ya da işini kolaylaştıracak bir mekanizma üzerinde saklayabilir. Content provider, eğer uygulamanın verileri başka uygulamalarla paylaşılıyorsa yararlıdır. Content provider diğer uygulamaların erişmesi ve saklaması gereken veri setine erişmeleri için tanımlanan bir sınıftır.

Java uygulama geliştirmek için kullanılan bir dildir, ancak Dalvik sanal makinesi tarafından yorumlanabilmesi için Java olmayan byte koda çevrilir.

3.3.1. Uygulamanın geliştirme ortamı

Uygulama Eclipse IDE'si üzerinde Java diliyle geliştirilmiştir. Uygulamanın sanal olarak mobil cihaz üzerinde koşmasını emüle eden Android Emulator kullanılmıştır. Eclipse IDE'si üzerinde Android için uygulama geliştirmeyi sağlayan Android Development Tools Plugin kullanılmıştır. Bu pluginEclipse için güçlü eklentiler sağlar.

Android projeleri derlenerek ".apk" uzantılı paketler haline dönüştürülürler ve Android işletim sistemli cihazlara yüklenebilirler. Uygulama kaynak kodlarını ve kaynak dosyaları içerirler.

Bir Android projesi aşağıdaki dizin ve dosyaları barındırabilir (URL-17):

- Src:src/your/package/namespace altında Java kaynak kodlarını barındırır. ".java" veya ".aidl" uzantılı dosyalardır.
- bin: Derlenmiş çıktıların bulunduğu klasördür. Final ".apk" dosyası ve diğer derlenmiş kaynakları barındırır.
- jni: Android NDK ile geliştirilen yerel kaynak kodları barındırır.
- gen: ADT tarafından otomatik oluşturulan R.java ya da AIDL (AndroidInterface Definition Language (Android Ara Yüz Tanımlama Dili)) dosyalarını barındırır.
- assets: Bu dosya normalde boştur. Oyun dosyaları gibi isteğe bağlı dosyaları barındırır. AssetManager kullanılarak buradaki dosyalara erişilebilir.
- res: Uygulamada kullanılan resim, ekran düzenleri ve çoklu dil desteği için gerekli olan tanımlama dosyaları gibi uygulama kaynaklarını barındırır.
- anim: Animasyon nesnesine dönüştürülecek XML dosyalarını bulundurur.
- color:XML dosyalarındaki renkleri tanımlamak için kullanılır.

- drawable: PNG, JPEG, GIF, 9-Patch imaj dosyaları ve normal, basılmış ya da odaklanmış durumlarda gösterilecek olan resmedilebilir nesneleri içeren XML dosyaları bulunur.
- layout: Ekran düzenlerini içeren XML dosyaları bulunur.
- menu: Menü tanımlarını içeren XML dosyaları bulunur.
- raw: Assets klasörü gibi isteğe bağlı dosyalar bu klasörde saklanabilir. Assets klasöründe yer alan dosyalara erişim için AssetManager kullanmak gerekirken raw klasöründeki dosyalara R.raw.dosyaismi şeklinde erişebilmektedir.
- values: Çoklu dil desteği için gerekli olan tanımlamalar, dizi, stil veya tema tanımlamalarını içeren XML dosyalarını içerir. Raw ve layout klasörlerinin aksine içerdikleri dosya isimleri ile erişmek yerine içindeki tanımlamalara göre R.style.stiltanimi veya R.string.baslik gibi erişilebilir.
- xml: PreferenceScreen, AppWidgetProviderInfo ya da Searchability Metadata tanımlamalarını içeren XML dosyaları bulunur.
- libs: Java kütüphaneleri bulunur.
- AndroidManifest.xml: Bu dosya uygulama ve içerdiği bileşenler (aktiviteler, servisler, intent receiverlar ve content providerlar) hakkında bilgi sağlar. Uygulamanın gerektirdiği izinler, API seviyeleri, cihaz özelliklerin ve ihtiyaç duyulan harici kütüphaneler bu dosya içinde tanımlanır.
- project.properties: Bu dosya ADT tarafından otomatik oluşturulur ve projenin hangi Android sürümü için derleneceği gibi bazı ayarları içerir.
- local.properties: Proje derlenmesi için ANT kullanılıyorsa bilgisayara özgü ayarları içerir. Proje Eclipse ile geliştiriliyorsa bu dosyaya gerek yoktur.
- ant.properties: Özelleştirilebilen ANT seçenekleri bulunur. Proje Eclipse ile geliştiriliyorsa bu dosyaya gerek yoktur.
- build.xml: Ant için derleme ayarlarını içerir. Proje Eclipse ile geliştiriliyorsa bu dosyaya gerek yoktur.

4. SONUÇLAR VE ÖNERİLER

Hazırlanan tez, üç bölümden oluşmaktadır. Giriş bölümünde açık anahtarlı ve özel anahtarlı kripto sistemlerin gelişimini sağlayan algoritmalara kısaca değinilmiştir. Ülkemizde ve dünyada, Açık Anahtar Altyapısı uygulamaları ile elektronik imza uygulamalarına örnekler verilmiştir.

Tezin ilk bölümünde açık anahtarlı ve özel anahtarlı kripto sistemler anlatılmıştır. Açık anahtarlı ve özel anahtarlı kripto sistemlerde kullanılan algoritmaların çalışmalarına yer verilmiştir. Açık anahtarlı kripto sistemler ile özel anahtarlı kripto sistemlerin çalışması gizlilik, bütünlük, kimlik doğrulama, inkar edememezlik, performans ve güvenlik açısından karşılaştırılmıştır.

Tezin ikinci bölümünde; elektronik imza altyapısına değinilmiştir. Elektronik imzanın ne olduğu ve Windows işletim sistemi üzerinde uygulama geliştirmek için gerekli olan altyapı anlatılmıştır. Açık anahtar altyapısından ve açık anahtar altyapısını oluşturan bileşenlerden bahsedilmiştir. Açık anahtar altyapısının çalışması için kullanılan AAA protokolleri açıklanmıştır.

Tezin üçüncü bölümünde; Android işletim sistemi üzerinde Açık Anahtar Altyapısı kullanılarak elektronik imzalama uygulamasının geliştirilmesi anlatılmıştır. Uygulama geliştirilirken kullanılan sistem altyapısı, uygulamanın çalışması, uygulamada kullanılan teknolojiler ile uygulamanın geliştirme ortamı hakkında bilgi verilmiştir.

Uygulamada kullanıcının seçtiği dosya, uygulamaya gömülü olan sertifika yardımıyla elektronik imza ile imzalanmaktadır. Geliştirilen elektronik imzalama uygulamasında elektronik imzalama işlemi PFX dosyası yardımıyla yapılmaktadır. Tübitak Bilgem tarafından geliştirilmiş Milli ESYA Kütüphanesi kullanılarak uygulama geliştirilmiştir. Uygulama ile imzalanan dosya İmzager yazılımı ile Windows işletim sistemi üzerinde doğrulanabilmektedir. Dosya elektronik olarak BES (Basic Electronic Signature) formatında imzalanmaktadır.

Elektronik imza ile ıslak imzanın arasındaki temel farklardan biri kullanıcının ne imzaladığını görememesidir. Uygulama aracılığıyla özel anahtar kullanılarak mesaj özeti oluşturulur, imzalama algoritmasına sokularak mesaj özeti imzalanır. Bilgisayarın ya da mobil cihazın kullanımını ele geçiren kişi uygulamaya müdahale ederek imzalama işlemini engelleyebilir. İmza atan kullanıcı ekranda imzaladığı dokümanı görse bile arka planda başka dokümanı imzalamış olabilir. Böyle bir senaryoya karşı, imzalama uygulaması ile kullanıcının kullandığı uygulama arasında kimlik doğrulama mekanizması kurulmalıdır. İmzalama uygulaması ile kullanıcının kullandığı uygulama birbirinin bütünlüğünü doğrulamalıdır (URL-19).

İleride yapılacak çalışmalarda; farklı özetleme ve imzalama algoritmalarının kullanımı ile, farklı elektronik imza formatlarıyla ve İmzager uygulamasının desteklediği XML imza, PDF imzalama özelliklerinin kazandırılmasıyla uygulama daha kapsamlı bir hale getirilebilir. Mobil işletim sistemleri üzerinde çalışması için uygulamaya elektronik posta gönderme ve dokümana zaman damgası eklenmesi gibi işlemlerle işlevsellik arttırılabilir.

KAYNAKLAR

Berbecaru D., Lioy A., Marian M., Secure Digital Administration in Medical Environment, *IADIS International Conference WWW/Internet 2002*, Lisbon, Portugal, November 13-15 2002.

Boyacı U., Birinci F., Kiraz M.S., Elektronik Seçim: Norveç'in İnternet Üzerinden Oylama Sistemi ve Kriptografik Alt Yapısı, *7. Uluslararası Bilgi Güvenliği ve Kriptoloji Konferansı*, Ankara, Türkiye, 17-18 Mayıs 2012.

Bryson J., Gallagher P., FIPS PUB 180-4 Federal Information Processing Standards Publication Secure Hash Standard, http://csrc.nist.gov/publications/fips/fips180-4/fips-180-4.pdf (Ziyaret tarihi: 23 Mayıs 2014).

CGI Whitepaper, Public Key Encryption and Digital Signature: How do they work?, http://www.cgi.com/files/white-papers/cgi_whpr_35_pki_e.pdf (Ziyaret tarihi: 04 Aralık 2013).

Çağlar E., Açık Anahtarlı Kriptografi ve Güvenlik Uygulamaları, Yüksek Lisans Tezi, Çanakkale 18 Mart Üniversitesi, Fen Bilimleri Enstitüsü, Çanakkale, 2004, 184824.

Çalık Ç., Sönmez Turan M., Yüce Z., E-İmzada SHA-1 Özetleme Algoritmasının Kullanımı, *Ulusal Elektronik İmza Sempozyumu*, Ankara, Türkiye, 7-8 Aralık 2006.

Çelebi Başçı G., Sertifika Geçerlilik Kontrolündeki Sorunların Giderilmesi, *Ulusal Elektronik İmza Sempozyumu*, Ankara, Türkiye, 7-8 Aralık 2006.

Diffie W., Hellman M., New Directions in Cryptography, *IEEE Transactions On Information Theory*, 1976, **22**(6)**,** 644-645.

Erol H., Kurumsal Ağlarda Açık Anahtar Altyapısı Tabanlı Elektronik İmza Uygulaması, Yüksek Lisans Tezi, Gazi Üniversitesi, Fen Bilimleri Enstitüsü, Ankara, 2006, 184753.

Erol H., Sayısal (Elektronik) İmza ve Açık Anahtar Altyapısı, http://www.nvi.gov.tr/Files/File/Sayisalimza/HuseyinEROL_BBG.pdf, (Ziyaret tarihi: 08 Aralık 2013).

Federal Information Processing Standards Publication 197 Advanced Encryption Standard, http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf, (Ziyaret tarihi: 12 Mayıs 2014).

Garner W., Diffie-Hellman Key Exchange, University of California, http://math.ucsd.edu/~wgarner/research/pdf/diffie-hellman_key_exchange.pdf, (Ziyaret tarihi: 02 Aralık 2013).

Gutmann P., *Cryptographic Security Architecture: Design and Verification*, 1st ed., Springer-Verlag, New York, 14-303, 2003.

Gülaçtı E., Bir Açık Anahtar Altyapısı Nasıl Planlanmalı?, *Ulusal Elektronik İmza Sempozyumu*, Ankara, Türkiye, 7-8 Aralık 2006.

Gülaçtı E., *Açık Anahtar Altyapısı ve Elektronik İmza Uygulamaları Eğitim Kitapçığı*, 1. Basım, TÜBİTAK UEKAE, Gebze, 2006.

Haseman C., *Android Essentials*, 1st ed., Apress, New York, 2008.

Housley R., Cryptographic Message Syntax (CMS), http://tools.ietf.org/html/rfc5652 (Ziyaret tarihi: 10 Haziran 2014).

Hunt R., PKI and Digital Certification Infrastructure, *9th IEEE International Conference on Networks*, Helsinki, Finland, July 11-14 2001.

Karakoçak K., Saka O., Tüfekçi A., Çarkıt N., Hançer A., Ekiz M., Kandemir U., Öksüz B., Ayvalı A., Çamurdan Ç., E-imzanın Toplumsal Boyutu 2. Çalışma Grubu Raporu, *TBD Kamu-BİB Kamu Bilişim Platformu VII*, Antalya, Türkiye, 26-29 Mayıs 2005.

Kent S., Evaluating Certification Authority Security, *9th IEEE International Conference on Aerospace*, Colorado, USA, March 21-28 1998.

Kaya Bensghir T., Topcan F., *E-imza Türkiye'de Kamu Kurumlarında Uygulanması*, 2. Basım, Türkiye ve Orta Doğu Amme İdaresi Enstitüsü, Ankara, 2010.

Kırımlı M., Açık Anahtar Kriptografisi ile Sayısal İmza Tasarımı ve Uygulaması, Yüksek Lisans Tezi, Gazi Üniversitesi, Fen Bilimleri Enstitüsü, Ankara, 2007, 199883.

Mehuron W., Proposed Federal Information Processing Data Encryption Standard, http://csrc.nist.gov/publications/fips/fips46-3/fips46-3.pdf (Ziyaret tarihi: 23 Mayıs 2014).

Myers M., Ankney R., Galperin S., Adams C., X.509 Internet Public Key Infrastructure Online Certificate Status Protocol - OCSP, http://tools.ietf.org/html/rfc2560 (Ziyaret tarihi: 03 Haziran 2014).

Nash A., Duane W., Joseph C., Brink, D., *PKI Implementing and Managing E-Security*, 2nd ed., RSA Press, California, 2001.

Öğretmen B., Çevrimiçi Sertifika Durum Protokolü (OCSP), *Ulusal Elektronik İmza Sempozyumu*, Ankara, Türkiye, 7-8 Aralık 2006.

Özbey R., Akıllı Kart Teknolojileri, *Ulusal Elektronik İmza Sempozyumu*, Ankara, Türkiye, 7-8 Aralık 2006.

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

RSA Laboratories, PKCS #12 v1.1: Personal Information Exchange Syntax, http://www.emc.com/emc-plus/rsa-labs/pkcs/files/h11301-wp-pkcs-12v1-1-personal-information-exchange-syntax.pdf (Ziyaret tarihi: 08 Haziran 2014).

Sağır M., Becerikli Y., Android Üzerinde Elektronik İmza Uygulaması Geliştirilmesi, *7. Uluslararası İleri Teknolojiler Sempozyumu,* İstanbul, Türkiye, 30 Eylül-1 Kasım 2013.

Sağıroğlu Ş., Alkan M., *Her Yönüyle Elektronik İmza*, 1. Basım, Grafiker Yayınları, Ankara, 2005.

Shu X., Zhenjun D., Rong C., Research on mobile location service design based on Android, *5th IEEE International Conference on Wireless Communications, Networking and Mobile Computing*, Beijing, China, September 26-29 2009.

Toorani M., LPKI - A Lightweight Public Key Infrastructure for the Mobile Environments, *11th IEEE International Conference on Communication Systems*, Guangzhou, China, November 19-21 2008.

Tübitak Bilgem Kamu SM, Kamu Sertifikasyon Merkezi 2013 Yılı Faaliyet Raporu, Tübitak Bilgem KamuSm, http://www.kamusm.gov.tr/kurumsal/raporlar/KamuSM FaaliyetRaporu2013.pdf, (Ziyaret tarihi: 01 Haziran 2014).

URL-1: http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=54788, (Ziyaret tarihi: 24 Kasım 2013).

URL-2: http://cihangir.forgottenlance.com/presentations/aks_cihangir_tezcan_asimetrik_kriptografi.pdf, (Ziyaret tarihi: 24 Kasım 2013).

URL-3: http://www.bilgiguvenligi.gov.tr/gizlilik/rsa-algoritmasi.html, (Ziyaret tarihi: 16 Mayıs 2014).

URL-4: https://www.cs.columbia.edu/~smb/classes/f06/l03.pdf, (Ziyaret tarihi: 24 Kasım 2013).

URL-5: http://tr.wikipedia.org/wiki/DES, (Ziyaret tarihi: 03 Aralık 2013).

URL-6: http://www.tbmm.gov.tr/kanunlar/k5070.html, (Ziyaret tarihi: 04 Aralık 2013).

URL-7: http://en.wikipedia.org/wiki/Public_key_certificate, (Ziyaret tarihi: 08 Aralık 2013).

URL-8: https://yazilim.kamusm.gov.tr/?q=tr/imzager, (Ziyaret tarihi: 22 Aralık 2013).

URL-9: http://www.e-guven.com/Urunlerimiz.aspx?PageID=19, (Ziyaret tarihi: 22 Aralık 2013).

URL-10: http://techmeonline.com/most-used-smart-card-commands-apdu/, (Ziyaret tarihi: 19 Mayıs 2014).

URL-11: http://yazilim.kamusm.gov.tr/?q=/node/14, (Ziyaret tarihi: 22 Aralık 2013).

URL-12: http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=54550, (Ziyaret tarihi: 06 Ocak 2014).

URL-13: http://www.appbrain.com/stats/number-of-android-apps, (Ziyaret tarihi: 06 Mayıs 2014).

URL-14: http://www.android-app-market.com/android-architecture.html, (Ziyaret tarihi: 06 Mayıs 2014).

URL-15: http://tr.wikipedia.org/wiki/Android_(i%C5%9Fletim_sistemi), (Ziyaret tarihi: 07 Mayıs 2014).

URL-16: http://developer.android.com/guide/index.html, (Ziyaret tarihi: 08 Mayıs 2014).

URL-17: http://developer.android.com/tools/projects/index.html, (Ziyaret tarihi: 08 Mayıs 2014).

URL-18: http://tr.wikipedia.org/wiki/Totient, (Ziyaret tarihi: 08 Haziran 2014).

URL-19: http://en.wikipedia.org/wiki/Digital_signature, (Ziyaret tarihi: 20 Temmuz 2014).

Vacca J. R., *Public Key Infrastructre Building Trusted Applications and Web Services*, 1st ed., Auerbach Publications, Florida, 2004.

Yeşil S., Alkan M., Acarer T., E-imza Uygulamalarında AB ve Türkiye'de Mevcut Durum ve Öneriler, *Ulusal Elektronik İmza Sempozyumu*, Ankara, Türkiye, 7-8 Aralık 2006.

Yıldırım M., Çelikyılmaz S., Barbaros Ö. A., ve diğ., E-İmza Ulusal Koordinasyon Kurulu Altyapı Çalışma Grubu İlerleme Raporu, *Bilgi Teknolojileri ve İletişim Kurumu*, 2-5, 2009.

EKLER

Ek-A. Dünyada Elektronik İmza Yasaları Uygulama Yılları

Ülkeler	**Yasalar**	**Tarihler**
Malezya	Sayısal İmza Yasası	1998
Singapur	Elektronik İşlemler Yasası	1998
İspanya	Elektronik İmza Yasası	1999
İtalya	AAA Esasına Dayanan Sayısal İmza Kanunu	1999
Portekiz	Elektronik İmza Yasası	1999
ABD	Küresel ve Ulusal Ticarette E-İmzalar Yasası	2000
Bulgaristan	Elektronik Belgeler ve Elektronik İmza Yasası	2000
Çek Cum.	Elektronik İmza Yasası	2000
Danimarka	Elektronik İmza Yasası	2000
Estonya	Elektronik İmza Yasası	2000
Finlandiya	Elektronik Hizmet Yasası	2000
Hindistan	Bilgi Teknolojileri Yasası	2000
Hong Kong	Elektronik İşlemler Yönetmeliği	2000
İngiltere	Elektronik Haberleşme Yasası	2000

Ülkeler	**Yasalar**	**Tarihler**
İsrail	Elektronik İmza Yasası	2000
Litvanya	Dijital İmzalar Yasası	2000
Slovenya	Elektronik Ticaret ve E-İmza Yasası	2000
Almanya	Alman Elektronik İmza Yasası	2001
Arjantin	Sayısal İmza Kanunu	2001
Belçika	Sertifika Servisleri ve E-İmzaların Hukuki Çerçevesinin Esasları	2001
Fransa	Elektronik İmza ve Belgeleme Esasları	2001
İsveç	Nitelikli E-İmza Yasası	2001
İzlanda	Elektronik İmza Yasası	2001
Japonya	E-İmzalar ve Sertifika Hizmetler Yasası	2001
Kanada	Elektronik İşlemler Yasası	2001
Macaristan	E-İmza Yasası	2001
Norveç	Elekt. İmzaların Kullanımı ve Tanınması Yasası	2001
Polonya	E-İmza Yasası	2001
Romanya	Elektronik İmza Yasası	2001

KİŞİSEL YAYINLAR VE ESERLER

[1] **Sağır M.**, Becerikli Y., Android Üzerinde Elektronik İmza Uygulaması Geliştirilmesi, *7th International Advanced Technologies Symposium*, İstanbul, 30 Ekim-1 Kasım 2013.

[2] İlhan S., Duru N., Karagöz Ş., **Sağır M.**, Metin Madenciliği ile Soru Cevaplama Sistemi, *Elektrik-Elektronik ve Bilgisayar Mühendisliği Sempozyumu*, Bursa, 26-30 Kasım 2008.

ÖZGEÇMİŞ

Merve SAĞIR 1986 yılında İzmit'te doğdu. İlk, orta ve lise öğrenimini İzmit'te tamamladı. 2004 yılında girdiği Kocaeli Üniversitesi Mühendislik Fakültesi Bilgisayar Mühendisliği Bölümü'nden 2008 yılında Bilgisayar Mühendisi olarak mezun oldu. Eylül 2010'da Kocaeli Üniversitesi Fen Bilimleri Enstitüsü, Bilgisayar Mühendisliği Anabilim Dalı'nda Yüksek Lisans eğitimine başladı. 2010 yılı Ocak ayından itibaren TÜBİTAK BİLGEM'de araştırmacı olarak çalışmaktadır.